# Dynamical Modeling of Physiology

Kristian Soltesz

# Dynamical Modeling of Physiology

Kristian Soltesz


LUND
UNIVERSITY

Department of Automatic Control

Illustrations: Leif Andersson and Frida Heskebeck.

ISBN 978-91-8104-018-0 (web)
DOI:https://doi.org/10.37852/oblu.258

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden



# Preface

Human physiology is the study of how organisms, organ systems, and individual organs function under normal circumstances. As a consequence of its paramount importance to medicine, physiology has been studied since ancient times, but the models studied in this book follow the tradition of modern physiology pioneered by Claude Bernard in the mid 1800s.

The focus of the book lies on how *dynamical* mechanisms and phenomena in physiology can be understood and studied using mathematical modeling. Each chapter commences with the introduction of a well-established physiological model. Although these models are collected from seemingly disparate areas such as hemodynamics, pharmacology, and biomechanics, it quickly becomes apparent how theory and methods surrounding differential equations and dynamical systems constitutes a fabric that unites all the studied models.

Both nonlinear and linear models are treated. Simulation of dynamical models through numerical integration is presented, and for linear models it is shown how responses can be obtained both in the time and "frequency" (Laplace) domains.

The powerful concept of feedback is introduced, and the book serves as an excellent spring board for studying cyber-physical "closed-loop" systems in medicine.

The intended audience is primarily undergraduate biomedical engineering students.

Kristian Soltesz,
Lund, April 2024

## Acknowledgment

This book grew out of the need for a textbook that covers physiological modeling from the perspective of mathematical concepts and methods. I would like to thank Professor emeritus Rolf Johansson for the inspiration to write this book, and PhD students Henry Pigot and Ylva Wahlquist for valuable input and help along the way. Illustrations are mainly due to Leif Andersson, with additional contribution from Frida Heskebeck and Henry Pigot. Leif Andersson has also been very helpful in sharing his typesetting expertise. The work has been possible through encouragement from the inclusive and supporting research environments at the Lund University Department of Automatic Control, and the Igelösa Life Science community.

## How to use this book

The book was primarily written for biomedical engineering students at the second year BSc or MSc level. The first seven chapters provide suitable material for a 5 credits within the European Credit Transfer and Accumulation System (ECTS), while the full ten chapters should correspond roughly to a 7.5 ECTS course.

Each chapter focuses on a topic within modeling of dynamical systems. This is predominantly done by first providing a physiological application example, that is then used as a basis for exploring and elaborating on the topic. Before diving into the topic, there is also a brief historic note, providing a broader context.

When used as a course book, it is recommended to study chapters 1–7 in order, after which any of the remaining chapters 8–10 should be accessible.

To make the book accessible also to students and readers outside of the (biomedical) engineering curriculum, it has been written so that anyone with fundamental knowledge in calculus (derivatives, integrals) and linear algebra (vectors, matrices) should be able to learn from it.

Please report any errors or suggested improvements to
kristian.soltesz@control.lth.se,
referring to c20b754, being a unique identifier of the version in front of you.

# Contents

# 1

# Dynamical models

**Learning goals**
After reading this chapter you should (be able to)

- Know the difference between physiology and anatomy.
- Explain the difference between a static and dynamic relation.
- Mention a few dynamical systems in human physiology.
- Explain what is meant by a model; an animal model; a mathematical model.
- Distinguish between systems, signals, and states.
- Know the difference between a forced and an autonomous system.
- Know what it means for a dynamical system to be in equilibrium.
- explain what time-invariant and linear mean in the context of dynamics.

## 1.1 Dynamical systems in physiology

***Historic Note.*** While anatomy is the study of *structure* or *configuration* of the body and its systems, physiology is the study of their *function*.

Although studied to some extent already in ancient times—Hippocrates proposed a theory of four bodily fluids: black bile, yellow bile, phlegm, and blood, that needed to be in balance to maintain a healthy state. The term physiology itself was first introduced by French physician Jean Fernel (1497–1558). He combined the Greek words φύσις (physis, meaning *nature* or *origin*) with λογία (logia, meaning *study of*), thus coining the term physiology.

It is another Frenchman, Claude Bernard (1813–1878), depicted in Figure 1.1, who is regarded by many as the "father" of physiology. He promoted the use of devised experiments to investigate physiological phenomena with the purpose of rejecting hypotheses, which is the methodology applied still within modern medicine.

**Figure 1.1** Claude Bernard (1813–1878), the "father" of modern physiology, with his pupils.

Bernard paved the way for a very productive era. In fact, much of the science underlying the models in this book was conducted in the decade following Bernard's death.

In the early studies of organs and other physiological systems it turned out, similarly to what had happened around 100 years earlier in physics, that tools from calculus and (linear) algebra were well-suited

to describe, or *model* the behavior of the studied systems. The introduction of modeling paradigms such as the compartment model and electric circuit equivalent models—both studied in this book—followed.

In this age, it may feel a bit discouraging to consider the skewed gender balance in Figure 1.1, and indeed much—but not all—of the early work within psychological modeling was conducted by men, broadly reflecting the norm of the era. Fortunately, the research scene has since changed, exemplified by an increasing fraction of Nobel prizes for physiological discoveries going to women, although the imbalance is not yet eliminated.

### Life

As mentioned in the historic note, physiology is the study of function within a living organism, while anatomy is the study of its structure or configuration. However, there still today exists no unanimously accepted definition of what life itself is. Contemporary definitions are descriptive, and constitute lists of processes that a living organism must be capable of. An attempt of such definition is shown in Figure 1.2.

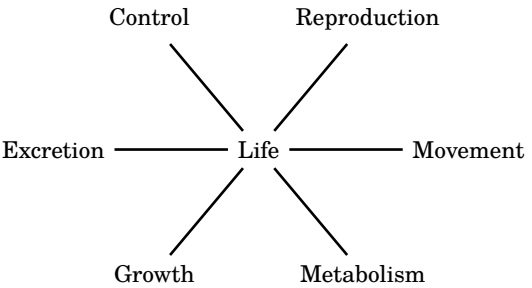


**Figure 1.2** Life processes. A common definition of life is that if each of these processes can take place within an organism, it is labeled as living.

Using the descriptive definition of Figure 1.2, we can note that it excludes viruses, since they for example do not maintain a metabolic process. However, many scientists would argue that viruses indeed constitute a form of life.

### Physiological complexity

The functions defining life are complex. It therefore helps to study processes within individual systems of the living organism. Here “system” is a quite vague word for a collection of things. In the context of physiological modeling we often define these systems through their relation to anatomy. For example, we can consider individual body organs as systems to be studied,

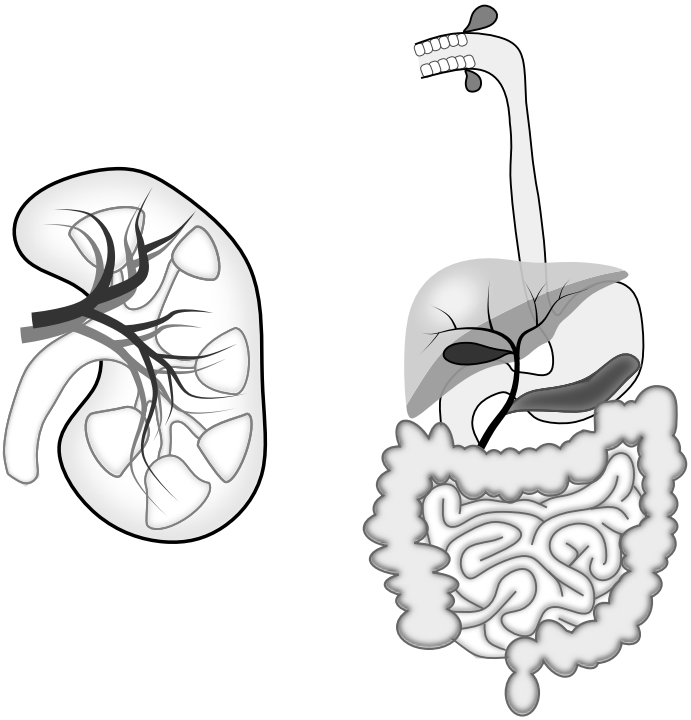

**Figure 1.3** For the study of physiology to be practically manageable, we typically delimit modeling to a system within the organism. This could be an anatomic delimitation, for example to study individual organs, as the kidney shown to the left. It could also be a functional delimitation, as illustrated by the metabolic system (where several organs are involved), to the right.

as illustrated in the left half of Figure 1.3. A physiological system can also be delimited to a particular function, as exemplified in the right half of Figure 1.3.

### Homeostasis

Homeostasis is a composition of Greek words όμοιος (homoios, meaning *similar*) and στάσις (stasis, meaning "standing still"), introduced in the 17$^{\text{th}}$ century into modern Latin (being the scientific language of the time) to describe a physiological state of "business as usual".

A homeostatic state is necessary for life processes to function optimally. For example, human metabolism relies on a core body temperature of around 37 °C (normothermia), and a blood plasma glucose concentra-

tion roughly in the range of 3–8 mmol/L (normoglycemia). Some variables, like body temperature, are tightly controlled to a constant set-point, while others, like blood plasma glucose concentration, may vary within some admissible range, often as a result of external factors.

Homeostasis is achieved by the organism through actively controlling variable entities to lie within admissible ranges. In this book we will mainly be concerned with the life processes necessary for homeostasis, rather than how they are controlled.

## 1.2 Models

### What is a model?

A model is an informative representation of something—the modeled object or system. With this definition, many things are models:

- a mental image;
- a map;
- a toy.

A trait of a good model is that it shares features of interest with the subject being modeled. Physiology often holds several options when it comes to modeling:

- *in silico*, meaning "in silicon", refers to modeling through simulation within a digital computer[1]. This will be our focus;
- *in vitro*, meaning "in glass", refers to isolating the system to be modeled *ex vivo*—outside of the organism—for the purpose of studying its behavior;
- *in vivo*, meaning "in life" refers to using the organism itself as a (surrogate) model. Animal models and clinical trials fall under this category.

Typically, the realism of the model increases from top-to-bottom of the bullet list above. But there remain several reasons to study *in silico* models. For example, they are cheap and do not come with the ethical concerns or medical risks associated with *in vitro* or *in vivo* models. However, to devise a realistic *in silico* model almost always relies on availability of informative experimental data.

[1] Integrated circuits that form the hardware of modern computers are made from silicon wafers.

### Mathematical models

A mathematical model is a construct that uses mathematical expressions to approximate the object or relation being modeled. Let us consider a very simple physiological model.

**Example 1.1: Body mass index.** The body mass index (BMI) is defined as the body mass in kilograms, divided by the squared body height in meters. The unit of BMI is therefore $\mathrm{kg/m^2}$. The body mass index is broadly used to model whether an individual is underweight, normal weight, overweight or, obese.

It is a very simple model. For extremes of low or high BMI it provides adequate classification that can provide helpful information to physicians, but in-between it is a blunt model.

For example, the BMI model does not take into account that individuals with a relatively large muscle mass may have a larger body mass for their height, without necessarily being overweight or obese, in the sense of having an excess of fat tissue. Conversely, tall persons can easily be mis-classified as being underweight under the BMI model.

The important take-away from Example 1.1 is that models are—often simplifying—approximations, useful within some range of validity, within which the error of the approximations are sufficiently small. Further insight into what this entails will be provided in Section 4.2, when studying local linearizations of nonlinear dynamics.

## 1.3 Modeling dynamics

The word *dynamic* or *dynamical* in the context of mathematical system modeling refers to a special property: the internal state of the model is—either explicitly or implicitly—a function of time. Below, we will introduce some of the core concepts and terminology from dynamical modeling. Understanding the presented concepts and terms will facilitate reading the remainder of the chapters in this book.

First, let us introduce the difference between static and a dynamic relations through an example.

**Example 1.2: Dynamic versus static relation.** The BMI model introduced in Example 1.1 is an example of a static model. Provided with inputs in form of a body mass and height, it produces a well-defined output, the BMI. Provided with the same mass and height values, it will always produce the same BMI. The BMI model provides a static relation between mass, height and BMI. We therefore call it a static model.

In contrast, the output of a dynamical model can additionally depend on time. This is often captured by an internal model state, and the model output depends both on the model inputs and the model state. Perhaps this sounds abstract? Let us provide an example: if you take a headache pill, its effect will first increase, then decay, as a function of how the concentration of active substance dissolved in the blood plasma varies. This is conceptually illustrated in Figure 1.4.

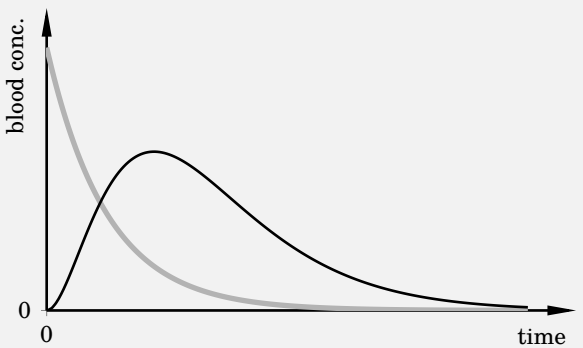


**Figure 1.4** Conceptual illustration of how the mass of undissolved drug (grey) and blood plasma concentration (black) vary following ingestion of a headache pill at time $t = 0$.

Regarding the administration of drug as input and the clinical effect as output, we can no longer relate the two with a static relation, as we could in the BMI example. Instead, we now also need to keep track of time, either explicitly, or through introducing states that capture the amount of yet undissolved drug in the stomach, and of dissolved drug in the blood plasma, respectively.

We will have reason to get back to the notions of inputs, states, and outputs of models at several occasions throughout the book. But for now, the take-away is that there is a distinction between static and

dynamic models, where the output of the latter depend not only on their input at the present time, but also their past input. The information about this past input can be encoded into what we call the state of the system.

In our headache pill example, the undissolved drug mass and blood plasma concentration together make up a state. This state enables us to predict the evolution of the clinical drug effect, if we have access to a dynamical model that relates the output, state and input. Much of this book will concern such models, and particularly how we can express and analyze them using mathematical tools.

Somewhat simplified—and to be clarified in the remainder of this chapter—a static relation can be modeled without involving derivatives with respect to time, as opposed to a dynamical relation, where you will find $d/dt$ in the system model.

### State

In Example 1.2, the notion of an internal system *state* was introduced, but not thoroughly explained. The state of a dynamical system (model) is a collection of variables that, together with possible external inputs, provide sufficient information to uniquely define the time-evolution of the system.

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}. \tag{1.1}$$

While $\boldsymbol{x}$ depends on time $t$, it is customary to write $\boldsymbol{x}$ instead of $\boldsymbol{x}(t)$, to simplify notation.

The number of state variables, $n$ in (1.1), is referred to as the *order* of the system. For the case of $n = 1$, we say that the system is scalar, or equivalently of first order. Throughout the book we denote scalar signals with standard typeface, such as $x$, and vector-valued signals with boldface, such as $\boldsymbol{x}$. We will also take a closer look at system states in Section 3.3, but let us begin with an illustrative example.

**Example 1.3: System state.** In this brief example, let us consider a system with state variables $x_1$ and $x_2$. They could for example be the undissolved drug mass and blood plasma concentration that together constituted the state in Example 1.2. For that example, the

choice of those particular state variables was quite natural as they represent physical entities that are, at least conceptually, possible to measure.

However, we could equivalently well describe the system state with the variables $\tilde{x}_1 = x_1 + x_2$ and $\tilde{x}_2 = x_1 - x_2$, since any pair $(\tilde{x}_1, \tilde{x}_2)$ uniquely defines a pair $(x_1, x_2)$, and vice versa. We could also let $\tilde{x}_1 = \alpha x_1 + x_2$ for any $\alpha \neq 0$ and be able to uniquely determine $(x_1, x_2)$ from $(\tilde{x}_1, \tilde{x}_2)$ and vice versa. This serves to show that the same dynamics can be represented using one of infinitely many possible state variable choices.

Often, the choice of state variables is motivated by their physiological or physical interpretations, as in Example 1.2. However, sometimes it might be advantageous to make other state variable choices, motivated for example by computational efficiency or numeric robustness. We will see also such examples in Section 4.3.

The take-away from Example 1.3 is that the state defines the configuration of the system at any given time. Provided that the state at some time is known, and that any possible *external* signals that may affect the system state are known for all future times, the state at any future time is well-defined and can be computed. How to practically do this will be covered in Chapter 2, and for the special case of linear dynamics in Section 4.3.

While there is always a minimum number of state variables—the *true* system order—required to represent the dynamics of a given system model, it is worthwhile noting that it is always possible to add state variables. For example, we can always add the temperature in Berlin as a state variable (component of $\boldsymbol{x}$), although for most physiological models this state will not have any dynamical coupling to other (relevant) state variables.

#### Forced versus autonomous systems

An autonomous system is one where there are no *external* signals that can affect the system state. The dynamics of such system can be described by

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t). \tag{1.2}$$

The dot notation is a convenient way to represent a derivative. For example

$$\dot{\boldsymbol{x}} = \frac{d\boldsymbol{x}}{dt} = \begin{bmatrix} \frac{dx_1}{dt} \\ \vdots \\ \frac{dx_n}{dt} \end{bmatrix}. \tag{1.3}$$

Higher-order derivatives can also be represented using dots. For example $\ddot{\boldsymbol{x}} = d^2\boldsymbol{x}/dt^2$, and so on.

**Example 1.4: Cooling body.** An example of an autonomous dynamical system, with relevance to forensic physiology, would be a cooling deceased body in a room with constant temperature and negligible air exchange. The core body temperature is uniquely defined by an initial state, being the temperature $x(0) = 37$ °C at the instance of death, $t = 0$. We can model the core body temperature using Newton's cooling law,

$$\dot{x} = -\alpha(x - x_e), \tag{1.4}$$

where $x_e$ is the temperature of the environment, and $\alpha > 0$ is a numeric constant defined by the heat transfer coefficient and surface area of the body.

We note that (1.4) is an autonomous (input free) system model, since it constitutes a special case of (1.2) with the vector $\boldsymbol{x}$ having just one component, and thus being a scalar $x$:

$$\boldsymbol{f}(\boldsymbol{x}, t) = f(x, t) = -\alpha(x - x_e). \tag{1.5}$$

In this case, the model of the dynamical system (the cooling body) has no explicit time dependence. There is an important distinction here between the model and the model state: although the model has no explicit time dependence, the model state, body temperature $x$, does change with time as it cools as described by the model $f(x, t)$.

The reason we chose a dead body to represent an autonomous system in Example 1.4 system was not pure coincidence. In fact, most systems we study in physiology are not autonomous. Instead, the time-evolution of their states are affectable by some external stimulation. We model this using one or several exogenous, or *input* signals, that are themselves functions of time. These inputs are assembled into the vector $\boldsymbol{u}$, and we have

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t). \tag{1.6}$$

As opposed to an autonomous system with no inputs, a system with input $\boldsymbol{u}$ is often referred to as a *forced* system. In the context of mechanical systems, as the ones we will consider in Section 4.1, the input is indeed often a force. However, the notion of forced systems has a much broader relevance, and we will encounter systems where $\boldsymbol{u}$—in our case often a scalar $u$—will

represent entities other than mechanic force. Before moving on, we can also note that autonomous systems are a special case of forced systems.

It might at first appear confusing that terms such as state, system and signal sometimes appear in singular, and other times in plural, often even within the same body of text. This is because sometimes the state vector $\boldsymbol{x}$ can be referred to as the state, which also holds for its individual elements $x_1, \dots, x_n$. In contrast, thinking of $\boldsymbol{x}$ as the collection of its individual elements, it is natural to refer to $\boldsymbol{x}$ as the states of the system. In practice this seldom causes ambiguity, and possibly ambiguities can be resolved by for example referring to $x_1, \dots, x_n$ as the state components, or state variables, of the system.

#### Time-varying versus time-invariant systems

An important special case of dynamical systems are ones where the dynamics do not change over time. This is equivalent to the absence of the argument $t$ of $\boldsymbol{f}$ in (1.6). A forced time-invariant system model can thus be written on the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}). \tag{1.7}$$

The corresponding special case of time-invariant autonomous systems have dynamics on the form $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$.

We have already seen an example of a time-invariant system—the cooling body in Example 1.4. It can be worthwhile to emphasize that time-invariant refers to the dynamics defined by the function $\boldsymbol{f}$ and not to the state $\boldsymbol{x}$: the state of a time-invariant system may change over time, but the function describing how it changes does not explicitly take time as an argument, as opposed to the case in (1.6). For example $\dot{x}(t) = -3x(t)$, with $f(x,t) = -3x$, is an example of a time-invariant system, while $\dot{x}(t) = -3x(t) + t$, with $f(x,t) = -3x + t$, is not.

Most dynamic systems encountered in physiology, and indeed in the real world, have dynamics that are, to some degree, time-varying. However, often the variation over the time scale relevant for the model is sufficiently small for the dynamics to be adequately approximated by a time-invariant model. One example would be modeling of neurological pathways, that do change as a function of the individual's age. However, while the modeled dynamics act on the millisecond time scale, the aging acts on a time scale of years, and can therefore typically be disregarded.

All systems considered henceforth in this book will be assumed to be time-invariant, unless otherwise stated.

#### Equilibrium

A system is in equilibrium when the state does not change over time. Therefore, an equilibrium is also referred to as the system being in "steady state".

Static systems are typically in equilibrium unless forced by some external signal. A dynamical system is in equilibrium if $\dot{\boldsymbol{x}} = 0$. Note that for a forced time-invariant system (1.7), any pair $(\boldsymbol{x}^0, \boldsymbol{u}^0)$ with $\boldsymbol{f}(\boldsymbol{x}^0, \boldsymbol{u}^0) = 0$ defines an equilibrium. If $\boldsymbol{u}^0$ is constant, it is referred to as a static equilibrium. It is not uncommon for a forced system to have infinitely many static equilibria.

**Example 1.5: Equilibrium heart rate.** Consider what happens to your heart rate $x$ [bpm] under physical work load $u$ [Watt]. When you increase physical work load, the heart rate dynamically increases to meet the increased gas exchange demands of your working muscles.

If the work load is held constant over time—as it would be for someone running on a treadmill at constant speed—the heart rate will reach an equilibrium matching the gas exchange requirement for the particular work load. If the work load is slightly increased, the heart rate will increase, and if it is slightly decreased it will decrease. Thus this is an example of a dynamical system with infinitely many static equilibria $(x^0, u^0)$, one for each possible work load level $u^0$.

It is again worthwhile reminding ourselves that we are dealing with models here, and that these models are simplifying abstractions of reality. It might for example happen that your heart rate does not stay exactly constant although you do not vary physical work load, or even that it increases slightly when you decrease work load and vice versa. But on the whole, the described relationship between work load and steady state heart rate is a fair approximation.

#### Measurement signals

In some cases, it is straightforward to introduce sensors that can measure the entire state—one measurement per state vector component. For example, the body temperature in Example 1.4 could simply be measured using a thermometer. In other cases only some combination of systems states can be measured. We typically denote the vector of corresponding measurement signals $\boldsymbol{y}$, and use an observation model (1.8b) to model them:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}), \tag{1.8a}$$

$$\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}). \tag{1.8b}$$

Quite often the input $\boldsymbol{u}$ only affects the measurement $\boldsymbol{y}$ through the state, and we have $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x})$. The measurement $\boldsymbol{y}$ is often referred to as output of the system model.

A dynamical model may have one or several inputs and one or several measurements, or outputs, and there are commonly used acronyms to describe the possible cases:

- SISO: single input, single output;
- SIMO: single input, multiple output;
- MISO: multiple input, single output;
- MIMO: multiple input, multiple output.

#### Linear systems

Another special case of dynamical systems are ones where the dynamics and observation functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are linear. For the herein considered case of time-invariant systems, this means that the system model can be expressed on state space form as

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}, \tag{1.9a}$$
$$\boldsymbol{y} = C\boldsymbol{x} + D\boldsymbol{u}, \tag{1.9b}$$

where $A$, $B$, $C$, and $D$ are constant matrices of appropriate dimensions. Models of the type (1.9), that are both linear and time-invariant, are referred to as LTI models for short. The representation (1.9) of LTI systems will be the topic of Section 3.3, and in Section 4.2 we will see how nonlinear models on the form (1.8) can be locally approximated with linear ones on the form (1.9).

#### Block diagrams, signals, and states

As we will see later, it can be convenient to graphically illustrate dynamical systems using blocks as the one in Figure 1.5, where $G$ denotes the system, defined for example through (1.9).

While technically the states are variables that change over time, they are typically not referred to as signals. The word signal is instead reserved

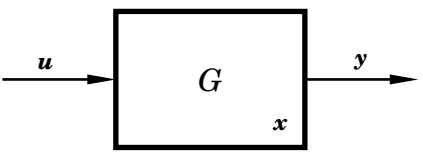


**Figure 1.5** Graphical block representation of a dynamical system $G$ with input signal $\boldsymbol{u}$, output or measurement signal $\boldsymbol{y}$, and internal state $\boldsymbol{x}$. The arrows indicate signal flow directions.

for variables that interface externally to the dynamical system, as illustrated through the arrows in Figure 1.5.

A similar floating semantic boundary is that between system model, and system. As mentioned in Section 1.1, "system" is a general term for a regularly interacting or interdependent group of items forming a unified whole. This holds also for a dynamical system in physiology, where—for "dynamical" signals—there is a time-dependence to the interaction. A dynamical system model is a mathematical description of such system. In this book we will use ordinary differential equations (ODEs) for such descriptions. It is important to keep in mind that the models we consider are *approximations*. The system of differential equations, defined through the elements of $\boldsymbol{f}$ in (1.8), should not be confused with the actual dynamical system that the equations represent.

### Uses of dynamical models

Now that we have established some notation, we are ready to discuss in what ways dynamical models can be useful. Principal uses of dynamical models include

1. parameter estimation;
2. simulation;
3. state estimation;
4. prediction.

***Parameter estimation.*** Estimation of parameters, also known as system identification or learning of dynamics, refers to using the combined recorded input and output $(\boldsymbol{u}, \boldsymbol{y})$ of a system to estimate parameters of the dynamics and observation functions $\boldsymbol{f}$ and $\boldsymbol{g}$. For example, if an LTI model on the state-space form (1.9) is used for a SISO system that can be adequately modeled using $n = 2$ state variables, the dynamics are fully parameterized by the elements of the $2 \times 2$ $A$ matrix, $2 \times 1$ $B$ matrix, $1 \times 2$ $C$ matrix and $1 \times 1$ $D$ matrix, where the $D$ matrix is thus a scalar. In this case, the system identification problem concerns determining the constant elements of these matrices from $(\boldsymbol{u}, \boldsymbol{y})$ time series data.

***Simulation.*** A common use of dynamical models is to investigate the behavior of the modeled system, without performing experiments on it. Here experiments typically refer to exposing the system to a known but varying input $\boldsymbol{u}$, and investigating the resulting evolution of the output $\boldsymbol{y}$.

In engineering, this surrogate use of models is often referred to as "digital twins", since the models are typically implemented in a digital computer. In medical applications, simulation models can (to a certain extent)

be used to replace animal models in development of new treatments and drugs. They also have a use in training personnel without exposing patients to risk.

For a simulation model to be useful, it typically needs to extrapolate well, meaning that if the modeled system is exposed to some input signal $\boldsymbol{u}$ that was not represented in the data from which the model was devised, it is desirable that the model responds similarly to the system it aims to represent. One example would be to construct a model for blood insulin based on bolus (shot) injections, and then use the same model to simulate blood insulin level resulting from steady rate delivery using an infusion pump. How well a model extrapolates, and the related topics of over-fitting and sensitivities, are essential but unfortunately easily forgotten, particularly in data-driven modeling.

***State estimation.*** State estimation concerns partial or full reconstruction of the state vector $\boldsymbol{x}$ form input–output recordings $(\boldsymbol{u}, \boldsymbol{y})$. State estimation can be used to estimate state variables that are not—and sometimes cannot be—directly measured. Such estimators are referred to as "soft sensors" since they estimate (rather than measure) a state using software (rather than direct sensor hardware). A trivial example of a soft sensor would be the use of a flow sensor in a spirograph to estimate the volume of inhaled gas. The state variable—inhaled volume—is hard to measure directly (unless using some expensive imaging modality such as MR), but can easily be estimated as the time-integral of the flow sensor signal.

A more subtle use of state reconstruction is in the estimation of state variables that are directly measurable, but where the measurements are corrupted by noise. In such case, measurements that are coupled to the state variable(s) of interest through the system dynamics can be used to obtain a better estimate of the measured signal than available through the noisy estimate. A good example of this is inertial measurement units (IMUs), used in tracking of body movements. The IMU is a combined sensor that measures 3D velocities and accelerations. In this case, the dynamic relation is *exactly* known, since the accelerations constitute the time derivatives of the velocities. Utilizing this, and taking into account that the acceleration measurements from the accelerometer is corrupted by high-frequency measurement noise, while the velocity measurements of the gyroscope are corrupted by low-frequency drift, it is possible to obtain a better velocity estimator than the gyroscope alone.

***Prediction.*** Prediction is closely related to simulation, and indeed simulations can be used to provide predictions. As opposed to the simulations described further above, that have as their main purpose to replicate the modeled dynamics, predictions are used to investigate (within the model!) how the state and output would evolve as a result of certain actions. This

can be used to compare the anticipated outcome of several input signal candidates, to choose the one that is optimal in some pre-determined sense. It could for instance inform how long before a painful procedure a local anesthetic should be administered.

***Modeling for control.*** Dynamic models also have a central role in the design of control systems. There, the dynamic models are used to design the control law, typically implemented as a computer algorithm.

A simple example of a feedback controller is one that measures temperature $y$ of blood circulating within an extra-corporal membrane oxygenator (ECMO) and applies heating $u > 0$ or cooling ($u < 0$) to keep the blood temperature at a desired set-point, or reference, $r$. One way to do this is to let the control signal be $u = K(r - y)$, where the controller gain $K$ is a design parameter, in this case a real number. Having access to a dynamical model relating the control signal $u$ to the resulting blood temperature $y$ makes it possible to make an informed choice of $K$. The system described above is schematically illustrated in Figure 1.6, from which it is also easy to see why this configuration is referred to as a feedback or closed-loop control system.

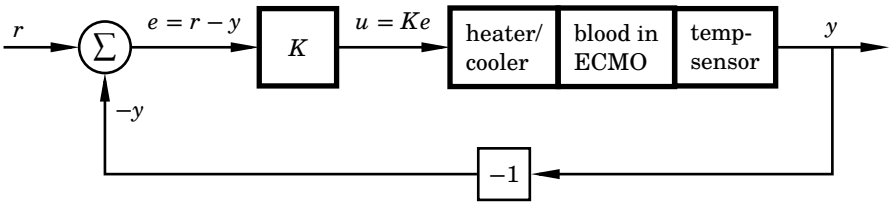


**Figure 1.6** Closed-loop or feedback control system, where the current control signal $u$ is determined by the (current and possibly past) measurement $y$, using a control law, with the objective to track a reference or set-point signal $r$. In this example, the control law is simply a proportional gain $K$, by which the control error $e = r - y$ is multiplied.

In the example above, a dynamical model is used offline to determine a suitable $K$. Dynamical models can also be used online within control systems. The most common example is referred to as model predictive control (MPC). In MPC, a prediction model is used to find the optimal $u$ over some time horizon. Then this control signal is applied, and the procedure is repeated periodically with a period that is typically much smaller than the prediction horizon. A schematic illustration of MPC is provided in Figure 1.7. For an MPC solution to be successful, the model needs to be of sufficient accuracy for the optimization across the horizon to make sense. It is for example uncommon (within physiology) to have models that provide reliable predictions on a time scale that is much larger than that of

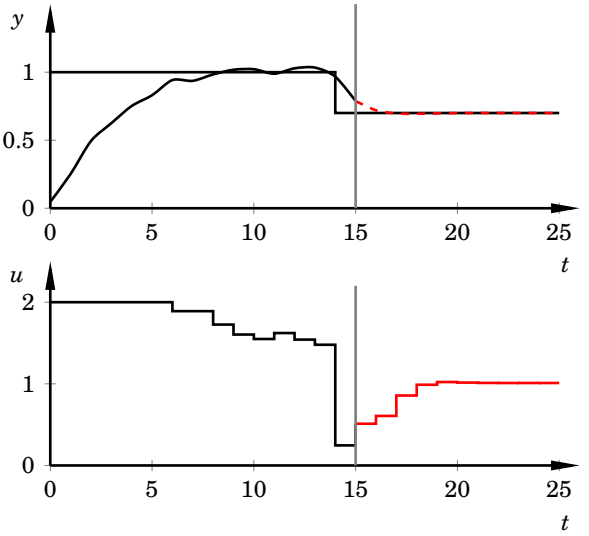


**Figure 1.7** Conceptual illustration of model predictive control, MPC. In each execution of the MPC algorithm an optimization problem is solved. The result of this optimization at $t = 15$, an optimal control signal $u$, is shown in red in the lower pane. Only the first sample of this input is applied to the process input. At the next iteration, here $t = 16$, the optimization is repeated, and so on. This enables the MPC controller to cope with model errors and external disturbances.

the modeled dynamics. And, of course, the optimization criterion by which candidate control signals are compared needs to be carefully chosen with the particular application in mind.

This book focuses on models that can be used for control, rather than control itself.

***Purpose-based model choice.*** In general, it is important to have a clear idea of what the model will be used for before modeling a dynamical system. For example, to obtain a *mechanistic* understanding of oxygen uptake, you might need to model how individual alveolae within the lungs work, but if your modeling objective is to study how exercise affects oxygen uptake over time you could–and should–use a coarser model. Settling for the right detail level when modeling is indeed a challenge, as what is right depends on for example what you will use the model for, and what data is available to determine the parameters that define the behavior of the model. Pickinga model that is too simple might result in it not providing a sufficiently accurate description to be useful to you. Picking too complex a model might result in the model being too sensitive to provide reliable results.

***Further reading.*** If you are, or become, interested in dynamical modeling, there exist a rich literature on the topic. While relatively few works focus specifically on physiological models, the concepts, theory and methods typically translate across disciplinary boundaries. For a general introduction we would recommend [Ljung et al., 2021]. For a book on mathematical models in physiology, we can recommend [Claudio Cobelli and Carson, 2019]. If you are interested in a historic review of physiology, [Rothschuh, 1973] can provide some inspirational examples that can be better understood with access to dynamic modeling methods.

Ljung, L., T. Glad, and A. Hansson (2021). *Modeling and identification of dynamic systems*. Studentlitteratur, Lund, Sweden. ISBN: 9789144153452.

Claudio Cobelli, C. and E. Carson (2019). *Introduction to modelling in physiology and medicine*. Elsevier, Amsterdam, Netherlands. ISBN: 9780128157565.

Rothschuh, K. E. (1973). *History of physiology*. Krieger, Malabar, FL. ISBN: 9780882750699.

# 2

# Simulating differential equation models

**Learning goals**
After reading this chapter you should (be able to)

- Explain the difference between anabolism and catabolism.
- Describe the steps of an enzymatic reaction, and the role of the enzyme.
- Represent an enzymatic reaction as a system of differential equations based on a reaction formula.
- Derive the Michaelis-Menten model and understand under what approximation it holds.
- Approximately solve a differential equation using finite-difference approximations.
- Know the difference between the explicit and implicit Euler methods.

***Historic Note.*** In this chapter we will study enzyme kinetics, and particularly a dynamical model thereof, introduced by the German biochemist Leonor Michaelis (1875–1949) and the Canadian biomedical researcher Maud Menten (1879–1960), seen in Figure 2.1.

**Figure 2.1** Maud Menten (1879–1960), Canadian chemist and physician most famous for formulating the Michaelis-Menten model of enzyme kinetics.

Maud Menten was one of the first women to earn a medical doctorate in Canada, and moved to Germany in 1912 since it was not possible to conduct medical research as a woman in Canada at the time.

In Berlin, Maud Menten worked at the lab of Leonor Michaelis. Despite the lab being a relatively simple establishment, they together managed to formulate the dynamical model that constitutes Michaelis-Menten enzyme kinetics. This model has since been used to describe a large number of enzymatic reactions across the discipline of biochemistry.

## 2.1 Enzyme kinetics

### Metabolism

The human body, like any organism or other system in nature, obeys the laws of thermodynamics. Without some active—meaning energy consuming—action, the body will reach a thermal equilibrium with the environment, associated with death, as already brought up in Example 1.4. In contrast, energy needs to be added to maintain homeostasis. In plants,

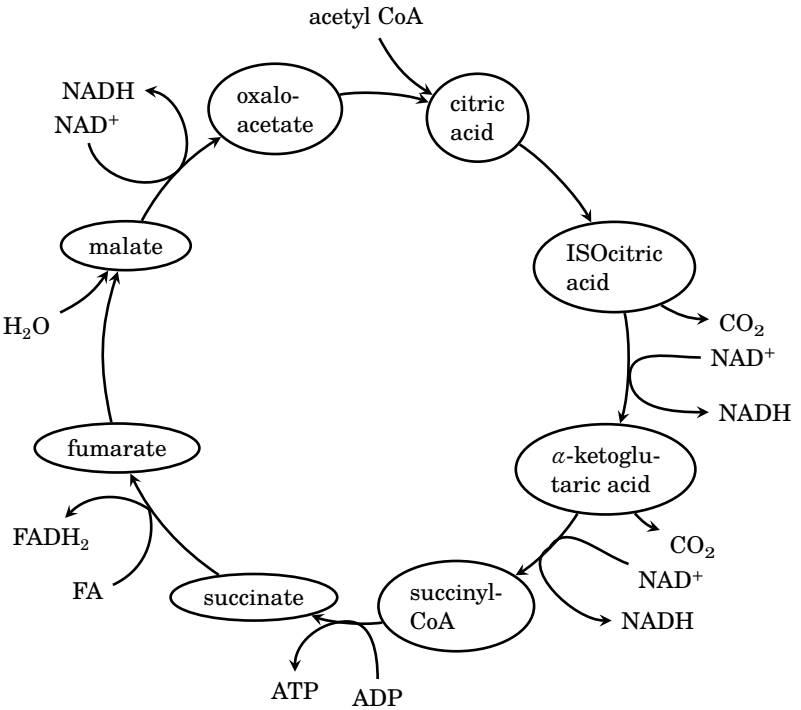


**Figure 2.2** Schematic illustration of the citric acid cycle, being a well-studied catabolic pathway.

this is achieved through photosynthesis, while animals ingest nutrients[1].

Metabolism is the collective term for the life-sustaining processes needed to maintain homeostasis, and thereby life. Human metabolism can be divided into

- *Anabolism*— Process of creating bigger molecules out of smaller ones.
- *Catabolism*—Processes breaking down molecules into smaller ones that are oxidized to release energy or used in anabolic reactions.

Both anabolic and catabolic processes are arranged into pathways. Perhaps the most well-known catabolic pathway is the citric acid cycle which generates energy ATP, NADH, and $FADH_2$—also known as the Krebs cycle—schematically depicted in Figure 2.2.

Human metabolism comprises of a large number of metabolic pathways.

[1] There are some notable exceptions, including sea slugs that can steal chloroplasts from ingested algae and use them to maintain their own photosynthesis.

### Enzymatic reactions

The reactions within metabolic pathways are catalyzed by biological catalysts, called enzymes. Let us exemplify this with the catabolic reaction that converts the sugar maltose into two glucose sugars, as depicted in Figure 2.3.

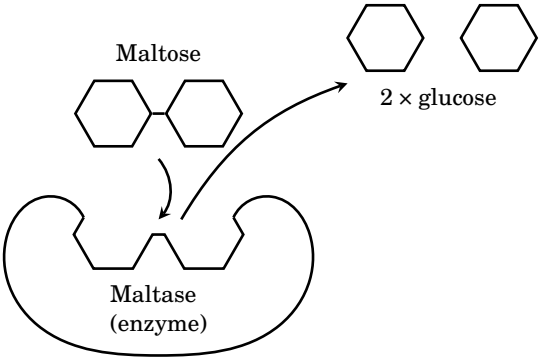


**Figure 2.3** Illustration of the catabolic conversion of maltose into glucose, facilitated by the enzyme maltase.

Enzymes function through an active site to which the substrate—in our case maltose—can bind. While bound, the per-time probability of splitting into two maltose molecules is importantly increased, as opposed to when the maltose molecule is not bound to the enzyme.

Enzymes play a crucial role in all of biology, and enzymology is the field of biochemistry devoted to the study of enzymatic reactions, being reactions catalyzed by enzymes.

### Modeling enzyme dynamics

An enzymological model of the process depicted in Figure 2.3 can be expressed as

$$E + S \rightleftharpoons X_{ES} \rightharpoonup 2P + E. \tag{2.1}$$

This should be read as "one unit of enzyme $E$ combines with one unit of substrate $S$ to form one unit of enzyme-substrate compound, or complex, $X_{ES}$; one unit of $X_{ES}$ can be decomposed back into $E$ and $S$, or it can produce (in this case) two units of product $P$ and free up the enzyme."

Instead of units, it is often convenient to handle concentrations instead. In (2.2), being an enzymatic reaction in which one unit of substrate can produce one unit of product, $[E]$ denotes the concentration of $E$ within the

reaction system, and so on:

$$[E] + [S] \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} [X_{ES}] \xrightarrow{k_2} [P] + [E]. \tag{2.2}$$

The SI base unit for chemical concentration is $\mathrm{mol/m^3}$, but the more commonly used unit is molarity ($\mathrm{mol\,L^{-1}}$ or equivalently $\mathrm{mol/dm^3}$). The numbers $k_1$, $k_{-1}$, and $k_2$ are called rate constants. They determine at what rates the reactions indicated by arrows occur as functions of reactant concentration. For example the production rate $dP/dt$ is $k_2[X_{ES}]$, and we see that the rate constants $k_{-1}$ and $k_2$ have the SI base unit of $\mathrm{s^{-1}}$, while $k_1$ has the unit of$\mathrm{mol\,m^{-3}\,s^{-1}}$. Keeping track of units can save a modeler much work, and we will return to this topic in Section 5.3.

Here we will assume that the enzymatic reaction (2.2) takes place within a volume that is constant over time. Translating between quantity ($E$, $S$, $X_{ES}$, $P$) and concentration ($[E]$, $[S]$, $[X_{ES}]$, $[P]$) then comes down to multiplication with a positive constant. It is therefore often convenient to drop the square bracket notation, and this is often (somewhat sloppily) done when concentrations are considered.

The law of mass action is a model that states that the rate of a chemical reaction of well-mixed reactants (under some additional assumptions) is directly proportional to the product of their concentrations. Applied to the enzyme binding reaction, it enables us to express (2.2) as a system of ordinary differential equations

$$\dot{E} = -k_1 ES + (k_{-1} + k_2) X_{ES}, \tag{2.3a}$$

$$\dot{S} = -k_1 ES + k_{-1} X_{ES}, \tag{2.3b}$$

$$\dot{X}_{ES} = k_1 ES - (k_{-1} + k_2) X_{ES}, \tag{2.3c}$$

$$\dot{P} = k_2 X_{ES}. \tag{2.3d}$$

Before continuing, let us verify that (2.3) indeed describes a dynamical system on one the standard forms introduced in Chapter 1.

**Example 2.1: ODE on standard form.** Provided we consider a reaction system that is closed in the sense that no enzyme, substrate or complex are externally added or removed, we can consider the dynamics (2.3) to constitute an autonomous system, since there are then no external signals acting on it once the reaction has started. We can also note that (2.3) describes a time-invariant system, since there is no explicit time-dependence of the right-hand-side. Thus, we

can re-write (2.3) onto the standard form of a an autonomous time-invariant differential equation, $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$. To do this, we need to introduce state variables. In Section 3.3 the choice of state variables will be discussed further, but often it is possible to choose states with some natural physiological interpretation. This is indeed the case for (2.3), where a natural state choice is constituted by

$$\boldsymbol{x} = \begin{bmatrix} E & S & X_{ES} & P \end{bmatrix}^\top = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^\top. \tag{2.4}$$

With this choice, we see that $\boldsymbol{f}$ has four components, one per equation in (2.3):

$$\dot{x}_1 = f_1(\boldsymbol{x}) = f_1(x_1, x_2, x_3, x_4) = -k_1 x_1 x_2 + (k_{-1} + k_2) x_3, \tag{2.5a}$$
$$\dot{x}_2 = f_2(\boldsymbol{x}) = f_1(x_1, x_2, x_3, x_4) = -k_1 x_1 x_2 + k_{-1} x_3, \tag{2.5b}$$
$$\dot{x}_3 = f_3(\boldsymbol{x}) = f_1(x_1, x_2, x_3, x_4) = k_1 x_1 x_2 - (k_{-1} + k_2) x_3, \tag{2.5c}$$
$$\dot{x}_4 = f_4(\boldsymbol{x}) = f_1(x_1, x_2, x_3, x_4) = k_2 x_3. \tag{2.5d}$$

We end the example by noting that we regard $f_3$ as a function of $x_1, x_2, x_3, x_4$ despite not explicitly being dependent on $x_4$. This will turn out convenient, for example when we consider local linear approximations of $\boldsymbol{f}$ in Section 4.2.

Knowing the dynamic (2.3), equivalently expressed as (2.5), is not sufficient to simulate the model. To do that, we additionally need boundary conditions for the differential equation system to have a unique solution. One such set of boundary conditions is to specify the state at time $t = 0$ through $\boldsymbol{x}(0) = \boldsymbol{x}_0$, where

$$\boldsymbol{x}_0 = [E_0 \; S_0 \; 0 \; 0]^\top. \tag{2.6}$$

In Sections 2.2 and 2.3 we will look into how we can simulate the resulting initial value problem using numeric software algorithms.

#### Michaelis-Menten kinetics

In the often physiologically relevant situation of substrate excess, the production rate is limited by the concentration of enzyme within the reaction system. Assuming the system reaches a dynamic equilibrium in the sense that the concentration of enzyme-substrate complex $X_{ES}$ is constant, we can combine the equivalent condition $\dot{X}_{ES} = 0$ with (2.3c) to obtain

$0 = k_1 ES - (k_{-1} + k_2) X_{ES}$, that we can then rewrite as

$$ES = \underbrace{\frac{k_{-1} + k_2}{k_1}}_{K_m} X_{ES}. \tag{2.7}$$

Assuming an initial condition where all enzyme is unbound, $X_{ES}(0) = 0$, $E(0) = E_0$, conservation of mass within the reaction system gives us $E(t) + X_{ES}(t) = E_0$ for all times, $t$, and we can insert $E_0 - X_{ES}$ in place of $E$ in (2.7) to obtain $ES = (E_0 - X_{ES})S = K_m X_{ES}$, that we can rewrite as

$$X_{ES} = \frac{E_0 S}{K_m + S}. \tag{2.8}$$

Finally, combining (2.8) with (2.3d), we obtain the Michaelis-Menten kinetics[2]

$$\dot{P} = v = \frac{k_2 E_0 S}{K_m + S} = \frac{v_{\max} S}{K_m + S}, \tag{2.9}$$

where $K_m$ is the Michaelis-Menten constant, and $v = \dot{P}$ is the production rate. Assuming an abundance of substrate, such that $S \to \infty$ and $S \gg E_0$ we have that

$$v_{\max} = \lim_{S \to \infty} v = \lim_{S \to \infty} \frac{k_2 E_0 S}{K_m + S} = k_2 E_0. \tag{2.10}$$

Thus $v_{\max} = k_2 E_0$ constitutes an upper limit for the production rate $v$, and this upper limit is approached when there is an abundance of substrate.

**Example 2.2: Linear production rate.** Assume an enzymatic reaction system of the form (2.3), that has reached an equilibrium at which $\dot{X}_{ES} = 0$, and that there is an abundance of substrate so that (2.10) is a good approximation. Then, the production rate can be expressed

$$P(t) = \int_0^t \dot{P}(\tau) d\tau = P(0) + v_{\max} t. \tag{2.11}$$

So in this case, with an abundance of substrate, plotting the amount of formed product against time would form a straight line with inclination $v_{\max}$, and passing through $(t, P) = (0, P(0))$.

[2] Kinetics and dynamics can be used interchangeably here. Kinetics is the study of how something (here enzyme) moves, and thus a somewhat more precise term than dynamics.

### The Lineweaver-Burk plot

In its standard form (2.9), the Michaelis-Menten kinetics relate substrate concentration $S$ to production rate $v$. If we introduce the variable transformation (reversible change of variables)

$$S' = 1/S, \tag{2.12a}$$

$$v' = 1/v, \tag{2.12b}$$

we can rewrite (2.9) as

$$v' = \frac{1}{v} = \frac{K_m + S}{v_{\max} S} = \frac{K_m + 1/S'}{v_{\max}/S'} = \frac{K_m S' + 1}{v_{\max}} = \frac{K_m}{v_{\max}} S' + \frac{1}{v_{\max}}. \tag{2.13}$$

The plot of $S' = 1/S$ against $v' = 1/v$ is called the Lineweaver-Burk diagram or plot. For the Michaelis-Menten kinetics it constitutes a straight line, as shown below in Example 2.3.

**Example 2.3: Lineweaver-Burk diagram.** Let us assume that we have are studying an enzymatic reaction with $K_m = 2$ and $v_{\max} = 1/3$ (in some unit system). In Figure 2.4 we have plotted $v$ against $S$, using (2.9). From this plot, it is not straightforward to determine the Michaelis-Menten constant $K_m$.

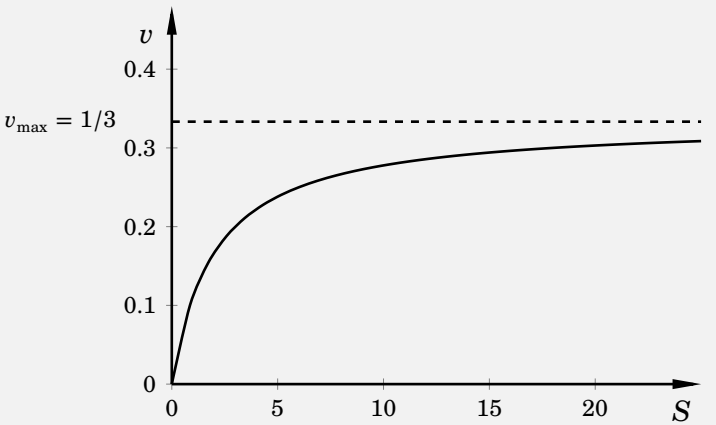


**Figure 2.4** Production rate $v$ plotted against substrate concentration $S$ for the Michaelis-Menten dynamics of the example.

If we instead plot $v' = 1/v$ against $S' = 1/S$ we obtain Figure 2.5, showing the relation (2.13). Indeed, we can directly see from (2.13)

that the plot should be a straight line with slope $k = K_m/v_{\max} = 6$ that intersects the $v'$-axis at $m = 1/v_{\max} = 3$.

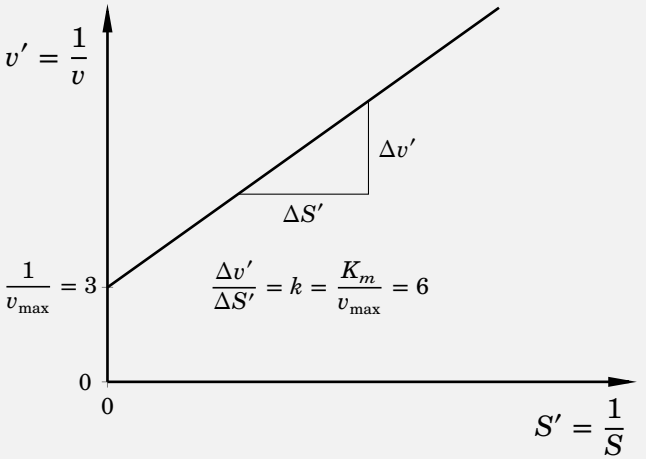


**Figure 2.5** Transformed variables $v' = 1/v$ and $S' = 1/S$ plotted against each other for the same Michaelis-Menten dynamic as shown in Figure 2.4.

Reading out the slope $k$ and $v'$-axis intersection $m$ from a Lineweaver-Burk plot allows us to retrieve

$$K_m = \frac{k}{m}, \tag{2.14a}$$

$$v_{\max} = \frac{1}{m}. \tag{2.14b}$$

## 2.2 Finite-difference approximations

In Section 1.3 we established fundamental concepts and nomenclature for dynamical system modeling, and in Section 2.1 we just saw a concrete example of how ordinary differential equation (ODE) models can be used to describe physiological dynamics. But what if we want to simulate the enzyme kinetics of (2.3), or more generally an ODE on the form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \tag{2.15a}$$

$$\boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2.15b}$$

as outlined in Section 1.3?

First of all, to simulate or solve—the words can be used interchangeably here—the system (2.15) refers to determining $\boldsymbol{x}(t)$. Since (2.15a) only determines the rate of change of the state $\boldsymbol{x}$ resulting from some input $\boldsymbol{u}$, the model also needs to be "grounded" by some boundary condition, as illustrated in Example 2.4 below.

**Example 2.4: Initial conditions.** Assume we have a system with a scalar state $x$ and autonomous dynamics described by

$$\dot{x} = f(x) = -\alpha x, \tag{2.16}$$

for some real $\alpha > 0$. Then we can verify that

$$x(t) = e^{-\alpha t} + c \tag{2.17}$$

solves (2.16), since

$$\dot{x} = \frac{d}{dt}\left(e^{-\alpha t} + c\right) = -\alpha e^{-\alpha t} = -\alpha x. \tag{2.18}$$

But to determine the real number $c$, we need some boundary condition. If $x(t_0) = x_0$ is known, we can use (2.17) to determine $c$:

$$x(t_0) = e^{-\alpha t} + c = x_0 \Rightarrow c = x_0 - e^{-\alpha t_0}. \tag{2.19}$$

It is conventional to choose $t_0 = 0$. In fact, this is not much of a limitation, since if we have $t_0 \neq 0$, we can introduce the shifted time variable $\tau = t - t_0$, and say that our "new $t$" is $\tau$.

As in Example 2.4, we will assume throughout the book that the state $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ is known at some initial time $t_0$. Together with (2.15a) this constitutes what is called an initial value problem (IVP)[3].

For some special $\boldsymbol{f}$ it is therefore possible to solve (2.15) analytically . This means that the solution is a symbolic expression for $\boldsymbol{x}$ like the one in (2.17), that can be directly evaluated without numeric approximation of the expression. Due to this property, such expression are referred to as being of "closed form". Notably, and further investigated in Section 4.3, linear time-invariant (LTI) systems—that is, systems where $\boldsymbol{f}$ is a linear function of $\boldsymbol{x}$ and $\boldsymbol{u}$—have analytic solutions. But for most $\boldsymbol{f}$, we need to rely on numerical methods to solve (2.15). Such methods are referred to as numeric integration methods, since they are concerned with numerically approximating the

[3] Other possibilities also exist. For example, reconstructing the historic state using (2.15a) together with a known end-time state is referred to as an final value problem (FVP).

integral

$$\boldsymbol{x}(t) = \int_{t_0}^{t} \dot{\boldsymbol{x}}(\tau)\, d\tau = \int_{t_0}^{t} \boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), \tau)\, d\tau. \tag{2.20}$$

The arguably simplest way to approximate a derivative

$$\frac{dx}{dt} = \lim_{h \to 0^+} \frac{x(t+h) - x(t)}{h}, \tag{2.21}$$

is to not take the limit, but instead evaluate the right-hand-side of (2.21) for some small, but finite, $h$, resulting in the approximation

$$\dot{x} \approx \frac{x(t+h) - x(t)}{h}. \tag{2.22}$$

The right-hand-side of (2.22) is referred to as a finite difference approximation, since $h$ is a finite difference, as opposed to in (2.21), where it is instead inifinitesimal.

***Forward Euler.*** Going back to (2.15), assuming we know $\boldsymbol{x}(t)$ and $\boldsymbol{u}(t)$, we can evaluate $\dot{\boldsymbol{x}}(t)$ and use it in (2.22) to approximate $\boldsymbol{x}(t+h)$ as

$$\boldsymbol{x}(t+h) \approx \boldsymbol{x}(t) + h\dot{\boldsymbol{x}}(t). \tag{2.23}$$

Iteratively applying (2.23) we can thus obtain approximations of $\boldsymbol{x}(t+h)$, $\boldsymbol{x}(t+2h)$, $\boldsymbol{x}(t+3h)$, and so on. Using a known initial state $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$ to ground the solution, it is possible to approximate $\boldsymbol{x}$ at $t_0 + kh$ for all integers $k > 0$.

The described method of numerically solving, or rather approximating the solution of, an ODE is referred to as forward, or explicit, Euler. It is named after $18^{\text{th}}$ century mathematician Leonard Euler, who made the method popular. Forward, because it relies on $\boldsymbol{x}$ at the current time $t$ to approximate $\boldsymbol{x}$ at the future time $t+h$. Explicit, because the approximation of $\boldsymbol{x}(t+h)$ is an explicit (direct) function of the known $\boldsymbol{x}(t)$, $\boldsymbol{u}(t)$, and $t$. The method is simple, but relies on an adequate choice of step length, as illustrated in Example 2.5 below.

**Example 2.5: Forward Euler approximation errors.** Let us return to the initial value problem of Example 2.4, with $a = 1$, $t_0 = 0$, and $x_0 = 1$. The IVP we want to find an approximate solution to is

$$\dot{x} = -x, \tag{2.24a}$$
$$x(0) = 1. \tag{2.24b}$$

We already know from Example 2.4 that an exact solution is $x(t) = e^{-t}$, shown in Figure 2.6.

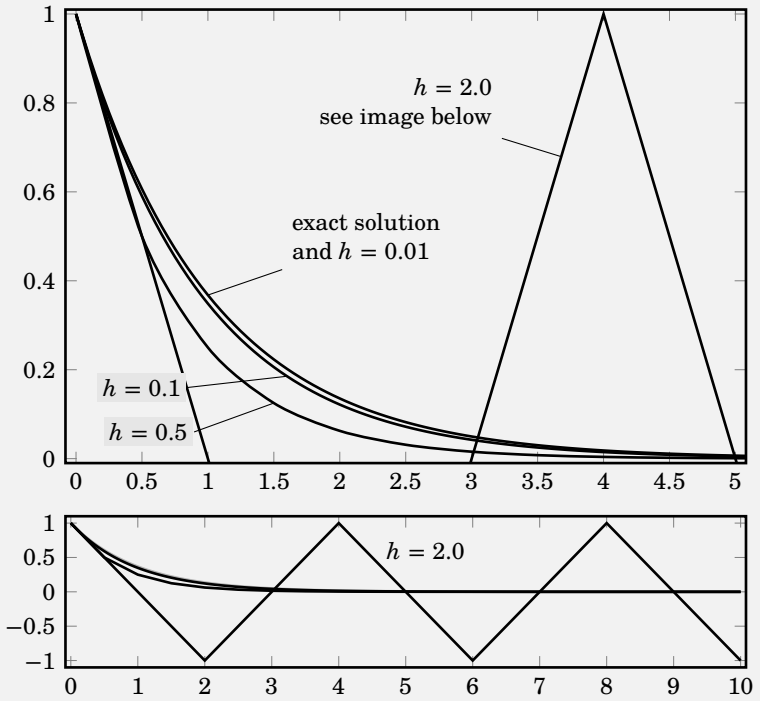


**Figure 2.6** Exact solution of (2.24) together with approximation based on $h = 0.01$, $h = 0.5$, and $h = 2$.

For a given step length $h$, the forward Euler method gives us

$$\hat{x}(t+h) = \hat{x}(t) + h\dot{x}(t) = \hat{x}(t) + hf(x) = (1-h)\hat{x}(t), \qquad (2.25)$$

where the hat over $x$ signifies that $\hat{x}$ is an approximation of $x$. Recursive application of the method gives us the forward Euler approximate solution

$$\hat{x}(kh) = (1-h)^k x_0 = (1-h)^k. \qquad (2.26)$$

We have plotted the forward Euler approximation of the solution $x$ for some values of $h$ together with a plot of the actual solution in Figure 2.6. Particularly, we can note that for $h > 1$, the approximate

solution maintains an undamped oscillation as $t \to \infty$, while the true solution tends to zero.

Choosing a sufficiently small value of the step length $h$ is therefore required. There are, however, two problems associated with this. Firstly, it becomes more computationally expensive with decreasing the step size, since the approximation of the solution will then be evaluated at a larger number of time points. Secondly, real numbers are represented using finite resolution floating point numbers inside digital computers. Thus, the machine precision $\epsilon$ constitutes a lower feasibility bound for $h$, since (2.23) would result in $x(t+h) = x(t)$ for any positive $h < \epsilon$, corresponding to the constant solution $x(t) \equiv x_0$.

***Backward Euler.*** An alternative to forward, or explicit, Euler, is constituted by the reverse, or implicit, Euler method. In it, $\dot{\boldsymbol{x}}(t+h) \approx (x(t+h) - x(t))/h$ is approximated instead of $\dot{\boldsymbol{x}}(t)$, resulting in

$$\boldsymbol{x}(t) \approx \boldsymbol{x}(t+h) - h\dot{\boldsymbol{x}}(t+h) = \boldsymbol{x}(t+h) - h\boldsymbol{f}(\boldsymbol{x}(t+h), \boldsymbol{u}(t+h), t). \quad (2.27)$$

Denoting $\boldsymbol{x}(t)$ with $\boldsymbol{x}$, $t+h$ with $t^+$, $\boldsymbol{x}(t^+)$ with $\boldsymbol{x}^+$, and $\boldsymbol{u}(t^+)$ with $\boldsymbol{u}^+$, we can rewrite (2.27) as

$$\boldsymbol{x} \approx \boldsymbol{x}^+ - h\boldsymbol{f}(\boldsymbol{x}^+, \boldsymbol{u}^+, t^+). \quad (2.28)$$

Replacing the approximate equality with a true one, we can solve (numerically) for $\boldsymbol{x}^+$. This explains why the backward Euler method is also called implicit, as the approximation only provides an implicit solution in the form of (2.28), that then needs to be solved in each time step. Below in Example 2.6, the explicit and implicit Euler methods are illustrated on a scalar example.

**Example 2.6: Implicit Euler.** Let us again consider the IVP of Example 2.5. The backward Euler method (2.27) provides the following update equation:

$$\hat{x}(t) = \hat{x}(t+h) - h\dot{\hat{x}}(t+h) = (1+h)\hat{x}(t+h). \quad (2.29)$$

Solving (2.29) for $\hat{x}(t+h)$ we obtain

$$\hat{x}(t+h) = (1+h)^{-1}\hat{x}(t), \quad (2.30)$$

and through recursion

$$\hat{x}(kh) = (1+h)^{-k}x_0 = (1+h)^{-k}. \quad (2.31)$$

Figure 2.7 shows the true solution alongside explicit Euler for $h = 0.5$ in dashed, and implicit Euler for the same $h$ in dotted.

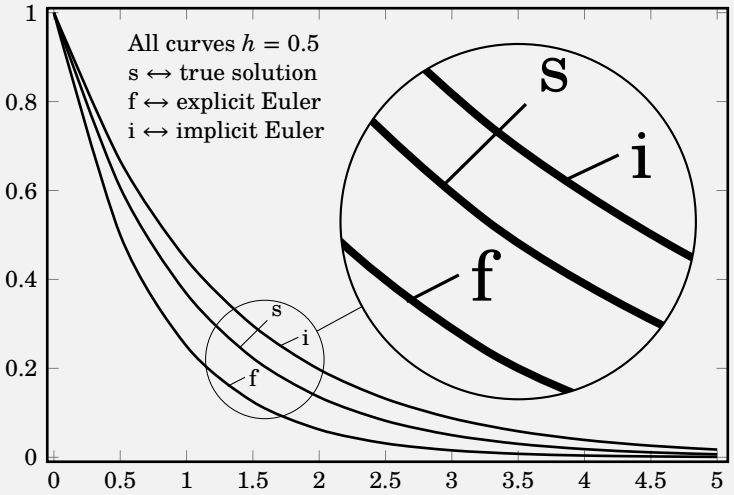


**Figure 2.7** The true solution of (2.24), together with implicit and explicit Euler for $h = 0.5$.

As can be seen in Figure 2.7, the implicit Euler method provides better approximations, which is commonly the case.

The finding of Example 2.6 generalizes, and implicit methods often outperform explicit ones, when it comes to approximation error for a given step length. However, they require solving an equation in each iteration. In Example 2.6 that equation, (2.28), had a simple analytic solution, but for more complicated $\boldsymbol{f}$ it typically does not. This means that while explicit methods generally have faster execution times per time step, they require shorter time steps to perform equally well in terms of approximation error.

### Difference equations

Finite difference approximations turn differential equations into difference equations. If the infinitesimal $dt$ is approximated with a (small) positive real number $h$, an expression containing the differential operator $d/dt$ is turned into an ordinary function of $h$. Let us first illustrate this with a simple example

**Example 2.7: Difference equation approximation.** Let us consider the enzyme kinetics ODE (2.3). Applying the forward Euler approximation (2.23) to its first equation (2.3a), we obtain

$$E(t+h) = E(t) + h(-k_1 E(t)S(t) + (k_{-1} + k_2)X_{ES}(t)), \quad (2.32)$$

Just like the differential operator $d/dt$ acts on whatever stands to the right of it, we can introduce a time-shift operator $q$ that shifts any function of time by $h$, so that $qf(t) = f(t+h)$. This helps us simplify the notation, and we can write (2.32) as

$$qE = E - hk_1 ES + h(k_{-1} + k_2)X_{ES}. \quad (2.33)$$

We can do the same for the remaining equations of (2.3), thus turning the differential equation into a system of difference equations.

Turning an ordinary differential equation (one in $d/dt$) into a difference equation (one in $q$) is referred to as discretizing the ODE. Relatedly, while ODE system models are referred to as being continuous-time models, difference equation system models are referred to as a discrete time ones. The focus of this book is on continuous-time models. For readers that have a background in digital signal processing—a subject that deals with discrete-time systems—it should be worthwhile to build an understanding of how the two relate. If you do not, there is no need to worry, and you can view the time shift operator $q$ as a side note.

## 2.3 Higher-order and adaptive step length methods

Both the explicit and implicit Euler methods are first-order methods. This property refers to the function $\boldsymbol{f}$ being approximated through a truncated Taylor series, where terms of order exceeding one are discarded. In case of explicit Euler approximation (2.23), the associated Taylor series is

$$\boldsymbol{x}(t+h) = \boldsymbol{x}(t) + h\dot{\boldsymbol{x}}(t) + \mathcal{O}(h^2), \quad (2.34)$$

where $\mathcal{O}(h^2)$ denotes the approximation error and informs that it consists of quadratic and higher order terms in $h$. Since the first error term is quadratic (of order two), the Euler methods are referred to as first-order methods.

The Runge-Kutta (RK) methods, named after German mathematicians Runge and Kutta, are generalizations of the Euler's method, in that they also consider inclusion of higher-order terms in the truncated Taylor series

used to represent $\boldsymbol{f}$. The original Runge-Kutta method, RK4, includes terms up to order four, and thus has an approximation error of $\mathcal{O}(h^5)$.

Just like we saw in Example 2.5 that the approximation error could be decreased by decreasing the step size $h$, it can also be decreased by increasing the method order. But, particularly for implicit methods, using high-order approximation of $\boldsymbol{f}$ becomes computationally expensive.

In general, a larger step size can be allowed for times $t$, where $\boldsymbol{x}$ changes slowly, than for other $t$, where $\boldsymbol{x}$ changes more abruptly. This inspires the use of adaptive step lengths. But since the studied problem is to find an approximation of $\boldsymbol{x}$, we do not beforehand know at which times we can get away with taking longer integration steps, and at which we should instead decrease step size to maintain an acceptably small approximation error $\boldsymbol{x} - \hat{\boldsymbol{x}}$. A common method for step length adaptation is to run two RK methods in parallel. The step length $h$ for the considered step is then adaptively decreased until the difference between the RK4 and RK5 methods for the considered step is smaller than some admissible threshold.

In this book, all simulation examples were generated with an explicit RK45 solver, unless otherwise stated. In fact, it is often a sufficient solver for ODEs that are not stiff, meaning that the derivatives of $\boldsymbol{f}$ vary sufficiently slowly.

#### Numerical solver software

Many programming languages have packages or libraries that provide unified interfaces to suites of numeric ODE integrators, for example the common implicit and explicit Runge-Kutta methods. To solve an initial value problem, the user typically needs to define the right-hand-side function $\boldsymbol{f}$ of the ODE as a function within the language, typically with a signature like

```
def xdot=f(t,x)
    u=getu(t,x)
    xdot=...
return
```

where `getu` fetches the input from a memory area or a function that computes the input based on the current time, and possibly the current state. The user then passes the name of the function, in our example `f` to the ODE integrator, together with an initial value. This could look like

```
x,t=solve(f,tstart,tstop,x0,method)
```

where `tstart` and `tstop` specify the time range of interest, `x0` is a vector containing the initial value, and the solution states `x` are returned as a matrix, where each column (or row) corresponds to one time instance, stored in the

vector `t`. In our example `method` indicates which of the available solver algorithms will be used. An important observation is that `f` is never invoked by the user, only by the solver.

Many ODE integrators also have additional features, like the possibility to obtain solutions when some user-defined function of `x` or `t` crosses zero. This is often implemented by registering a callback function, and it enables for example obtaining solutions at user-specified times within the simulation model (like every second) or when a particular variable attains a value of interest. The latter is particularly interesting for hybrid models, that have some dynamics for certain values of `x` and other dynamics for others. The classic example is a bouncing ball, that has two distinct regimens: free in space and ground contact. In order to simulate such system, the ODE integrator will need to provide a solution exactly for the time when the ball hits the ground, since the "free in space" dynamics do not apply anymore from that time on, until the ball leaves the ground again.

***Further reading.*** If you are interested to learn more about the Michaelis-Menten model, [Johnson and Goody, 2011] is a good resource.
Linear differential equations play a central role in dynamical modeling and control systems design. For a thorough background, we recommend [Åström and Wittenmark, 1984].
More generally, numerical analysis studies, among other things, the accuracy of numerical solutions to differential equations. Due to the importance of this topic within essentially all disciplines of engineering, there exist a rich literature within the area. One example, particularly targeting initial value problems (which have been our focus in this chapter) is [Lambert, 1991].

Johnson, K. A. and R. S. Goody (2011). "The original Michaelis constant: translation of the 1913 Michaelis–Menten paper". *Biochemistry* **39**, pp. 8264–8269. DOI: 10.1021/bi201284u.

Åström, K. J. and B. Wittenmark (1984). *Computer-controlled systems: theory and design*. Prentice Hall, Hoboken, NJ. ISBN: 978-0486486130.

Lambert, J. D. (1991). *Numerical methods for ordinary differential equation systems: the initial value problem*. Wiley, New York. ISBN: 978-0471929901.

# 3

# Compartment models

**Learning goals**
After reading this chapter you should (be able to)

- Explain what a compartment model is.
- Distinguish between common compartment model topologies.
- Know what rate constants are.
- Determine if a particular LTI state space model represents a compartmental system, and if so draw its topology.
- Derive differential equation system representations for linear compartment models.
- Describe what is meant by state space and represent an LTI system of differential equations on state-space form.
- Be able to perform state transformations, for example moving between compartment models where the states represent masses and concentrations, respectively.

## 3.1 Pharmacokinetics

***Historic Note.*** Pharmacokinetics, PK for short, is the study of how drug is taken up by, distributed within, and eliminated from the body. The field was pioneered by Swedish researchers Erik Widmark and Torsten Teorell. Widmark studied ethanol kinetics within the body, and in 1922 he published a paper with an equation corresponding to a one-compartment model. The proper introduction of modern pharmacokinetics took place in 1937, when Torell published a two-part paper titled "Kinetics of distribution of substances administered to the body", in which compartment models were used to describe drug PK. The first part considered extravehicular drug administration, while the second part considered intravascular. A schematic illustration from Teorell's work shown in Figure 3.1 summarizes the basic idea: the body tissues are divided into compartments, between which drug (or another substance of interest) flow.

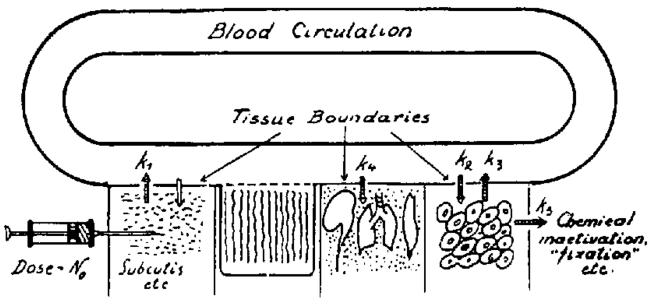


**Figure 3.1** Schematic illustration of pharmacokinetic compartment model by Torsten Teorell.

### Compartments

The human body can be partitioned into fluid compartments, as schematically illustrated in Figure 3.2. In pharmacology, compartment models are used to describe the kinetics of substances as they move between compartments. For intravenously administered drugs, the blood plasma constitutes a natural choice of compartment, and if it holds a given mass $m$ of substance the concentration within the plasma compartment is given by $\rho = m/v$, where $v$ is the blood plasma volume, also referred to as the volume of distribution (of the plasma compartment). At least conceptually, the volume of distribution can be determined experimentally as outlined in Example 3.1

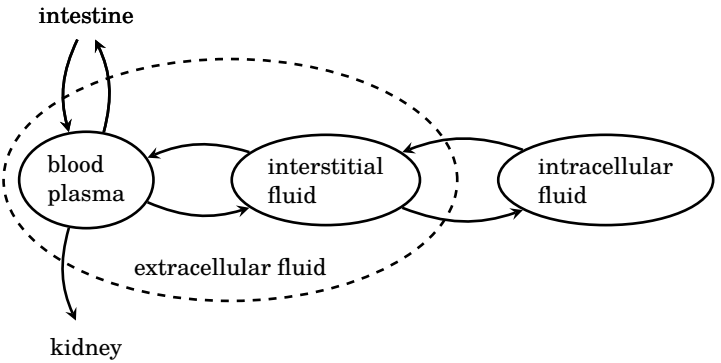


**Figure 3.2** Schematic illustration of fluid compartments of the human body.

**Example 3.1: Volume of distribution.** The following procedure can be used to determine the volume of distribution of a compartment:

1. Inject mass $m^+$ of the considered substance into the compartment of interest, and ensure no additional substance enters, or is synthesized within, the compartment.
2. Monitor the concentration of the substance within the compartment and await equilibrium. Denote the equilibrium concentration $\rho$.
3. Estimate mass $m^-$ of substance that has left the compartment, for example through metabolism or through transport to a communicating compartment. (More on communication between compartments will follow shortly.)

The volume of distribution $v$ can now be determined as

$$v = \frac{m^+ - m^-}{\rho}. \tag{3.1}$$

While the procedure is conceptually simple, it has its practical limitations. Particularly, estimating $m^-$ can be practically infeasible. Parameters of compartment models are therefore instead often estimated from the dynamic evolution of concentration following injection of a substance, rather than from the arising steady state levels.

A compartment can also be an abstract construct. For example, some drugs have dynamics that are well-explained by a model with three compartments, but where there is no (obvious) mapping between these compartments and anatomy. In fact, there are numerous examples, where a compartment model best fits the data when a compartment has a volume exceeding that of the human body. It is therefore often more meaningful to think of compartment models as an abstraction that replicate input—output behavior, such as the blood ethanol level after drinking a beer, than a verbatim anatomical model were each compartment has a clear interpretation. What this all means will hopefully become a bit clearer, once we reach Example 3.5.

#### Transport between compartments

Drugs and other substances can move between body fluid compartments, by one of several mechanisms:

- *Diffusion* is the movement of a substance from a compartment with higher concentration to one with lower.
- *Osmosis* is the movement of a substance across a selective membrane.
- *Filtration* is the movement of a fluid (containing the substance) from a compartment with higher hydrostatic pressure to one with lower.
- *Active transport processes* are ones where energy is continuously added to drive transport against gradients.

Both diffusion and osmosis are passive transport processes striving toward concentration equilibrium. In this context, passive means that no energy addition is needed to drive the process (although at some point energy has been added to achieve the gradient driving the process.) Filtration is a third example of passive transport process, where the gradient is one in hydrostatic pressure.

In contrast, active transport mechanisms rely on exogenously added energy to drive transport against (concentration, pressure, or other) gradients.

#### Compartment topologies

Compartment models can be of several structures or topologies. A few commonly encountered such topologies have special names:

- *Catenary* refers to a topology where compartments only communicate with their nearest neighbor, forming a chain.

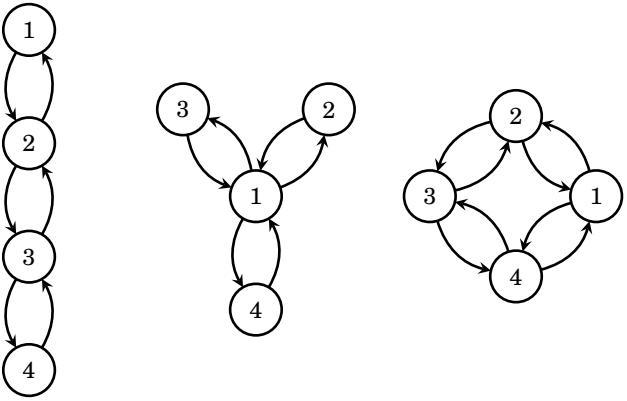


**Figure 3.3** Common compartment model topologies: catenary (left), mammilary (center), and cyclic (right).

- *Mamillary* (as in mammal) refers to a topology where all peripheral communicate only with a single central compartment.
- *Cyclic* refers to a catenary model where the two end compartments communicate with each other, thus forming a cycle.

Examples of the model types above are illustrated in Figure 3.3. Note that there exist some overlap. For example, a model with only two compartments that communicate with each other is catenary, mammillary and cyclic at the same time.

**Example 3.2: Adjacency matrix.** Element $E_{ij}$ of the adjacency matrix $E$ of a graph equals 1 if there is an edge from node $i$ to node $j$, and 0 otherwise. Regarding compartments as nodes in a graph, adjacency matrices of the models in Figure 3.3 can be obtained after numbering the compartments. If we number them from left to right for each of the models, we get:

- Catenary:

$$E = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{3.2a}$$

- Mammillary:

$$E = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \tag{3.2b}$$

- Cyclic:

$$E = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \tag{3.2c}$$

We can note that if we number in another way, for example replacing $2 \leftrightarrow 3$, this corresponds to swapping rows $2 \leftrightarrow 3$ and columns $2 \leftrightarrow 3$ in the adjacency matrix.

Another note worth taking is that we are not interested in the diagonal elements of the adjacency matrix here. It makes no difference to the dynamics whether a compartment has a direct flow-path to itself. Therefore, we have simply chosen to let all diagonal elements of $E$ equal 0.

In Figure 3.3 all compartment-compartment communication is bidirectional. How would (3.2) change if the edge $2 \to 3$ was removed?

All compartment models have a special "compartment" that is not explicitly drawn in schematic pictures like the ones in Figure 3.3. This compartment is the environment, modeling "everything else". The environment can be thought of as a compartment with infinite volume, that takes up all space in the schematic pictures that is consumed by the other compartments. We distinguish between

- *Closed* models, in which there is no communication between the environment and the other compartments.
- *Open* models, in which at least one compartment communicates with the environment.

In a schematic drawing of an open model, communication with the environ-

ment is depicted as either an arrow pointing towards a compartment or out from a compartment, as we will see in Example 3.5. It is common to number the compartments of the model using the natural numbers $1, \ldots$ and reserve 0 to denote the environment. Flow from the environment is regarded as an input to the system. A closed model therefore constitutes an autonomous system, since it has no inputs.

Apart from possible communication with the environment, we will assume that communication between compartments is bi-directional, as illustrated in Figure 3.3. It is of course possible to devise compartment models with directed (uni-directional) communication between some compartments, and this typically signals that the communication between them is the result of an active transport process.

As a final note, and as the careful reader might already have observed, the compartment "extracellular fluid" in Figure 3.2 is itself the union of other compartments. Defining new compartments as unions of others fits the formalism well, as we can write down mass balances by considering the connection going into or out of the group. Models where this is done are referred to as hierarchical or nested compartment models.

## 3.2 Compartmental dynamics

### State variable selection

In Section 3.1 we have already seen a few examples of compartment models. In this section we will look closer at how we can describe the kinetics (dynamics) of these models using ODE systems.

As we have already touched upon in Example 1.3, there are typically infinitely many equivalent state space representations for any given dynamical system. When modeling, it is often beneficial to select state variables that have a physiological (or more generally physical) interpretation. For a compartment models with $n$ compartments (not counting the environment), there are two such natural choices: the compartment substance masses

$$\boldsymbol{x} = \boldsymbol{m} = \begin{bmatrix} m_1 & \ldots & m_n \end{bmatrix}^\top, \tag{3.3a}$$

or the compartment substrate concentrations

$$\boldsymbol{x} = \boldsymbol{\rho} = \begin{bmatrix} \rho_1 & \ldots & \rho_n \end{bmatrix}^\top. \tag{3.3b}$$

Note that the environment, "compartment 0" is excluded from the state vector, since it has infinite volume. In the literature, the letter $\boldsymbol{q}$ for quantity is often used in place of the $\boldsymbol{m}$ in our notation.

Letting $V$ be a matrix with diagonal elements $v_1, \dots, v_n$, and all off-diagonal elements 0, we then have that

$$V\boldsymbol{\rho} = \boldsymbol{m}. \tag{3.4a}$$

Under the reasonable assumption that all compartment volumes are strictly positive and finite—remember, the infinite-volume environment with index $k = 0$ is excluded from the state vector—the inverse of $V$ is well-defined since $0 < \det V = v_1 \cdots v_n < \infty$, since the determinant of $V$ is in this case simply the product $v_1 \cdots v_n$ We can rewrite (3.4a) as

$$\underbrace{\begin{bmatrix} \rho_1 \\ \vdots \\ \rho_n \end{bmatrix}}_{\boldsymbol{\rho}} = \underbrace{\begin{bmatrix} \frac{1}{v_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{v_n} \end{bmatrix}}_{V^{-1}} \underbrace{\begin{bmatrix} m_1 \\ \vdots \\ m_n \end{bmatrix}}_{\boldsymbol{m}}. \tag{3.4b}$$

The matrices $V$ and $V^{-1}$ can therefore be viewed as transforms taking us back and forth between the state representations $\boldsymbol{x} = \boldsymbol{m}$ and $\boldsymbol{x} = \boldsymbol{\rho}$, and we will take a closer look at such transformations in Section 3.3.

A third possible state representation would be to use the molarity [mol $\mathrm{L}^{-1}$], expressing the number of substance particles per volume, as state variables. Yet another possibility would be to represent the state in some compartments as their mass content, in others as the concentration, and in yet others as molarity. While this would be possible, it is rarely practical. So which of all possible state representations should one choose? Often it comes down to what one wants to track. If we are interested in concentrations, it makes sense to use them also to represent the system state. In some occasions it might, however, be that certain state variable choices come with computational advantages, and we will get back to this in Sections 3.3 and 4.3.

The compartment models that we focus on in this chapter describe the transport of substance between compartments. This focus is motivated by the usefulness of compartment models in pharmacological modeling. However, compartment models have a broader use. For example, we can utilize the compartment model paradigm to model the flow of (heat) energy or electric charge. In fact, the variable does not have to be a physiological, or even a physical entity.

### Balance equations

Regardless of how transport between compartments take place, compartment model dynamics are governed by (mass) balance equations. In our

case, the rate of change $\dot{m}$ within the compartment may have the following contributors:

+ exogenous inflow;

+ endogenous production;

- endogenous consumption;

- exogenous outflow.

Let $\varphi_{ij}$ [kg s$^{-1}$] be the flow of substance mass from compartment $i = 1, \dots, n$ to compartment $j = 0, \dots, n$, where, again, compartment 0 is the environment.

Here we choose to model consumption within compartments as outflow to the environment, since the two are equivalent from the perspective of individual compartments: mass is removed. This explains why we include the destination index $j = 0$, but not the source index $i = 0$. Furthermore, it is nonsensical to model flow from a compartment to itself, so we set $\varphi_{ii} = 0$ for all compartment indices $i$.

Denoting the exogenous substance mass inflow $\varphi_{0i}$ from the environment to compartment $i = 1, \dots, n$ by $u_i$, we can now write down a mass balance equation for our compartment model:

$$\dot{m}_i = \sum_{j=1}^{n} \left( \varphi_{ji} - \varphi_{ij} \right) - \varphi_{i0} + u_i. \tag{3.5}$$

Here $u$ represents the sum of exogenous inflow and endogenous production.

While formulation (3.5) is general and can express any compartment model, it is not very useful without further elaboration, since it does not explicitly address the kinetics (dynamics) governing the flows $\varphi_{ij}$. If we assume that the flows are explicit functions of the state, and possibly also the input and time, then we have that $\varphi_{ij}(\boldsymbol{m}, \boldsymbol{u}, t)$, where $\boldsymbol{m} = \boldsymbol{x}$. This means that we can express the ODE on the standard form (1.6), with $\boldsymbol{f} = [f_1, \dots, f_n]^\top$, where

$$f_i(\boldsymbol{m}, \boldsymbol{u}, t) = \sum_{j=1}^{n} \left( \varphi_{ji}(\boldsymbol{m}, \boldsymbol{u}, t) - \varphi_{ij}(\boldsymbol{m}, \boldsymbol{u}, t) \right) - \varphi_{i0} + u_i \tag{3.6}$$

for $i = 1, \dots, n$.

Using (3.4) we can replace $\boldsymbol{m}$ with $V^{-1}\boldsymbol{\rho}$ if we prefer to model the compartment substrate concentration, rather than the masses.

We still have a very general compartment model. For example, we can use it to model scenarios where $V$ is time-varying. A physiological situation

where this occurs is when one of the volumes model the blood plasma of a patient that bleeds as a result of trauma (volume decrease) or receives intravenous fluid (volume increase).

Before moving on, let us consider a few types of kinetics that the individual components of $\Phi$ can model:

1. *Michaelis-Menten* There are situations where elimination from a particular compartment follows the Michaelis-Menten kinetics introduces in Section 2.1. For example, assume that the substrate $S$ is a drug that is converted into a product substance $P$ though an enzymatic reaction that follows the Michaelis-Menten kinetics (2.9), and that this reaction takes place within the blood plasma of the patient. Denoting the plasma drug concentration by $\rho$, there is an elimination that contributes to the change rate in plasma drug concentration with $v_{\max}\rho/(K_m + \rho)$.

2. *Zero-order elimination* This type of kinetics arises when substance is removed from a compartment with constant rate $k$ so that $\dot{\rho} = -k$. For example, zero-order kinetics with $k = v_{\max}$ is a good approximation of Michaelis-Menten kinetics when $\rho \gg K_m$, which leads to $v_{\max}\rho/(K_m + \rho) \approx v_{\max}$.

### LTI compartment models

While we have listed a few examples of nonlinear compartment kinetics and provided an example where time-varying volumes are motivated within the model, the class of linear time-invariant compartment models can be used to model many physiological systems. Particularly, they are useful in modeling the uptake, distribution and elimination of many drugs within the body, as we will demonstrate shortly in Example 3.5. These dynamics are referred to as the pharmacokinetics (PK) of the considered drug.

In an LTI compartment model, the individual compartment volumes are constant, so $V$ of (3.4) is constant. Furthermore, the kinetics are linear functions of the state, meaning that we can express the general compartment model as

$$\dot{\boldsymbol{m}} = \boldsymbol{f}(\boldsymbol{m}, \boldsymbol{u}, t) = \underbrace{(\Phi - \Phi_0)}_{A}\boldsymbol{m} + \boldsymbol{u}. \tag{3.7}$$

This looks like (1.9a), with the $A$ matrix broken up into two terms. In (3.7), $\Phi$ models the inter-compartment kinetics, and row $i$ of $\Phi\boldsymbol{m}$ corresponds to the sum in (3.6). That is, element $\phi_{ij}$ of $\Phi$ relates to the flows $\varphi_{ij}$ of (3.6) through

$$\begin{bmatrix} \phi_{i1} & \dots & \phi_{in} \end{bmatrix} \boldsymbol{m} = \sum_{j=1}^{n} (\varphi_{ji} - \varphi_{ij}). \tag{3.8}$$

The matrix $\Phi_0$ models elimination and row $i$ of $\Phi_0 \boldsymbol{m}$ corresponds to $\phi_{i0}$ in (3.6).

Row $i$ of $\boldsymbol{u}$, models an exogenous inflow of substance into compartment $i$, corresponding to $u_i$ in (3.6). Thus, the matrix $B$ determines into which compartment(s) substance is administered.

Let us consider a concrete example:

**Example 3.3: Closed LTI compartment system.** We consider a mammillary compartment model—*cf.* Figure 3.3—with three compartments, numbered so that compartment 1 is the central compartment. Let

$$\boldsymbol{m} = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \tag{3.9}$$

denote the compartment masses of a substance of interest, and let

$$\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{3.10}$$

be the compartment volumes.

We now assume that the dynamics governing mass transport are such that the amount of mass that flows from compartment $i$ to compartment $j$ per unit time is proportional to the concentration gradient $\rho_j - \rho_i$, ie. the larger the concentration in compartment $i$ relative to compartment $j$ the more rapid the flow of mass from $i$ to $j$. We call the proportionality constant that governs transport between compartments 1 and 2 $\alpha$, and let one that governs transport between 1 and 3 $\beta$. This gives us the dynamics

$$\dot{m}_1 = \alpha(\rho_2 - \rho_1) + \beta(\rho_3 - \rho_1), \tag{3.11a}$$
$$\dot{m}_2 = \alpha(\rho_1 - \rho_2), \tag{3.11b}$$
$$\dot{m}_3 = \beta(\rho_1 - \rho_3). \tag{3.11c}$$

Assuming we want to use the masses as state variables, we have

that

$$\dot{m}_1 = -\frac{\alpha+\beta}{v_1}m_1 + \frac{\alpha}{v_2}m_2 + \frac{\beta}{v_3}m_3 \tag{3.12a}$$

$$\dot{m}_2 = \frac{\alpha}{v_1}m_1 - \frac{\alpha}{v_2}m_2, \tag{3.12b}$$

$$\dot{m}_3 = \frac{\beta}{v_1}m_1 - \frac{\beta}{v_3}m_3. \tag{3.12c}$$

We can express (3.12) as

$$\dot{\boldsymbol{m}} = \Phi\boldsymbol{m}, \tag{3.13}$$

where

$$\Phi = \begin{bmatrix} -\frac{\alpha+\beta}{v_1} & \frac{\alpha}{v_2} & \frac{\beta}{v_3} \\ \frac{\alpha}{v_1} & -\frac{\alpha}{v_2} & 0 \\ \frac{\beta}{v_1} & 0 & -\frac{\beta}{v_3} \end{bmatrix}. \tag{3.14}$$

The matrix $\Phi_0$ in (3.7) models elimination of substance. Its element $ii$ defines the elimination rate constant of compartment $i$, and must thus be zero or larger. If we consider the most common case, where the elimination rate of compartment $i$ is proportional to the mass—and since $V$ is constant also the concentration—in the same compartment, all off-diagonal elements of $\Phi_0$ need to be zero. Otherwise, the requirement is that $\Phi_0$ is a positive semi-definite matrix.

So are all systems in the form of (3.7) compartment models? To be a compartment model, the mass balance (3.5) needs to be fulfilled. Let us assume that we are given a diagonal (or positive semi-definite) elimination matrix $\Phi_0$ and a matrix $\Phi$ governing the inter-compartment kinetics. Note that the closed compartment model is a special case that we can arrive at by replacing $\Phi_0$ with $0 \cdot \Phi_0$ and $B$ with $0 \cdot B$, resulting in

$$\dot{\boldsymbol{m}} = \Phi\boldsymbol{m}. \tag{3.15}$$

The total mass within the system is $M = m_1 + m_2 + \cdots + m_n$, and since that mass needs to be constant in time for our closed system we have that

$$\dot{M} = 0 = \dot{m}_1 + \dot{m}_2 + \cdots + \dot{m}_n = 1^\top\dot{\boldsymbol{m}}, \tag{3.16}$$

where $1^\top$ is an $n \times 1$ vector where each element is 1, making $1^\top\dot{\boldsymbol{m}}$ equal to the sum of elements of $\dot{\boldsymbol{m}}$.

Combining (3.15) and (3.16) we thus have

$$1^\top \dot{\boldsymbol{m}} = \underbrace{1^\top \Phi}_{\boldsymbol{\phi}} \boldsymbol{m} = \boldsymbol{\phi}\boldsymbol{m} = 0 \tag{3.17}$$

must hold for any substrate mass distribution $\boldsymbol{m}$. This means that any linear combination of the elements of $\boldsymbol{\phi} = 1^\top \Phi$ (with non-negative weights) must be zero. Assume that $\phi_i$ is non-zero and that $\boldsymbol{\phi}\boldsymbol{m} = 0$ for some $\boldsymbol{m}$. Changing $m_i$ will then make $\boldsymbol{\phi}\boldsymbol{m} \neq 0$. This is true for $i = 1, \ldots, n$, and means that we require $\boldsymbol{\phi} = 0$ must hold. So to test whether $\Phi$ is a compartmental matrix—that is a matrix representing the (closed) inter-compartment kinetics of a LTI compartment model—we can simply check whether

$$1^\top \Phi = 0 \tag{3.18}$$

which is the same as

$$0 = \sum_{i=1}^{n} \phi_{ij} \tag{3.19}$$

holding for $j = 1, \ldots, n$, as demonstrated in Example 3.4 below. In words: $\Phi$ holds the rate constants describing inter-compartment kinetics within a compartment model if and only if each column of $\Phi$ sums to 0.

The matrix $\Phi$ also reveals the topology of the compartment model. Setting the diagonal elements to zero and replacing all non-zero off-diagonal elements by ones, we can obtain the adjacency matrix (see Example 3.2) from the rate coefficient matrix $\Phi$.

**Example 3.4: Compartmental matrices.** Assume that we are presented the with the system

$$\dot{\boldsymbol{x}} = \underbrace{\begin{bmatrix} 1 & 2 & 3 \\ 4 & -2 & 0 \\ -5 & 0 & -3 \end{bmatrix}}_{A} \boldsymbol{x} + \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}_{B} u \tag{3.20}$$

and want to know if it represents a compartmental model without outflow to the environment. We can determine this by investigating whether $A$ in (3.20) is a compartmental matrix:

$$1^\top A = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 4 & -2 & 0 \\ -5 & 0 & -3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}. \tag{3.21}$$

The test revealed that $A$ is a compartmental matrix. The corresponding adjacency matrix is

$$E = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \tag{3.22}$$

### Linear diffusion

A common way to model transport between connected compartments is by assigning a mass flow rate $D_{ij}(\rho_j - \rho_i)$ from compartment $i$ to $j \neq i$. This corresponds to linear diffusion, where $D_{ij} = D_{ji}$. In the closed (no sources or sinks) case, mass balance for compartment $i$ gives

$$v_i \rho_i = \sum_{i \neq j} D_{ij} \rho_j - \rho_i \sum_{j \neq i} D_{ij}. \tag{3.23}$$

Defining

$$D_{ii} = \sum_{j \neq i} D_{ij}, \tag{3.24}$$

the system can be expressed

$$\dot{\boldsymbol{\rho}} = V^{-1} D \boldsymbol{\rho} \tag{3.25}$$

Let us close this section with a real-world modeling example, where we transition between masses and concentrations as state variables in a model involving linear diffusion.

**Example 3.5: Propofol pharmacokinetics.** The anesthetic drug propofol is administered through intravenous infusion. Let $u(t)$ [mg s$^{-1}$] denote the drug mass that is pumped into the blood plasma of the patient per unit time.

The pharmacokinetics (PK) of propofol model uptake, redistribution and elimination of drug within the body. It is natural to consider the blood plasma as one compartment, with volume $v_1$ [L]. The input, $u$ feeds into this compartment. Many drugs, including propofol, are eliminated from the blood plasma through renal (kidney) or heptic (liver) metabolism.

When modeling the kinetics of a new intravenous drug it is custom-

ary to perform experiments comprising of infusion step changes and boluses and recording the resulting changes in blood plasma concentration and then see how well compartment models with different numbers of compartments and topologies match the data. Often this process is aided by physiological insight. We will return to such experiments.But for now it is sufficient to mention that it was concluded through early clinical trials that a three-compartment mammillary model approximates the PK of propofol sufficiently well. Such model is schematically illustrated in Figure 3.4.

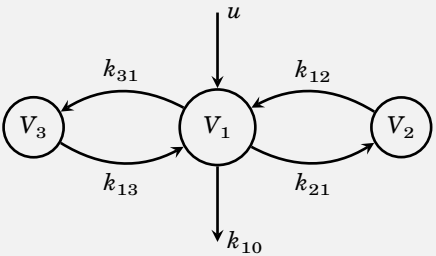


**Figure 3.4** Schematic illustration of three-compartment PK model for propofol. The rate constants $k$, central volume $v_1$ and infusion rate $u$ are defined in the text of the example.

Let us now introduce as state variables the substance mass in each compartment:

$$\boldsymbol{x} = \begin{bmatrix} m_1 & m_2 & m_3 \end{bmatrix}^\top. \tag{3.26}$$

Using Figure 3.4 we can thus write down the state space dynamics

$$\dot{\boldsymbol{m}} = \underbrace{\begin{bmatrix} -(k_{10}+k_{12}+k_{13}) & k_{21} & k_{31} \\ k_{12} & -k_{21} & 0 \\ k_{13} & 0 & -k_{31} \end{bmatrix}}_{\Phi - \Phi_0} \boldsymbol{m} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u. \tag{3.27}$$

Since the matrix $\Phi_0$ governs elimination, we see from (3.27) that it has $-k_{10}$ as its top left element, and all other elements zero. We can then also easily verify that $\Phi$ is indeed a compartmental matrix, since $1^\top \Phi = 0$.

Let us now express the dynamics (3.27) with the compartment concentrations (rather than masses) as state variables.

The dynamics for the first state then become

$$v_1\dot{\rho}_1 = -(k_{10} + k_{12} + k_{13})v_1\rho_1 + k_{21}v_2\rho_2 + k_{31}v_3\rho_3 + u. \quad (3.28)$$

Dividing by $v_1$ we obtain

$$\dot{\rho}_1 = -(k_{10} + k_{12} + k_{13})\rho_1 + k_{21}\frac{v_2}{v_1}\rho_2 + k_{31}\frac{v_3}{v_1}\rho_3 + \frac{1}{v_1}u. \quad (3.29)$$

To proceed, let us consider what happens if there is an equilibrium so that $\rho_1 = \rho_2$, or equivalently

$$\frac{m_1}{v_1} = \frac{m_2}{v_2}. \quad (3.30)$$

Since we are dealing with linear diffusion, there will be no net flow between compartments 1 and 2 at such equilibrium, meaning that

$$k_{21}m_2 = k_{12}m_1. \quad (3.31)$$

Combining (3.30) and (3.31) we arrive at

$$\frac{v_2}{v_1}k_{12} = k_{21}. \quad (3.32)$$

Carrying out the corresponding computations imposed by equilibrium conditions between other combinations of compartments, we can eliminate $v_2$ and $v_3$ from (3.29) to obtain

$$\dot{\rho}_1 = -(k_{10} + k_{12} + k_{13})\rho_1 + k_{21}\rho_2 + k_{31}\rho_3 + \frac{1}{v_1}u. \quad (3.33)$$

Repeating for the state equations for $\rho_2$ and $\rho_3$ we thus arrive at

$$\dot{\boldsymbol{\rho}} = \begin{bmatrix} -(k_{10} + k_{12} + k_{13}) & k_{21} & k_{31} \\ k_{12} & -k_{21} & 0 \\ k_{13} & 0 & -k_{31} \end{bmatrix} \boldsymbol{\rho} + \frac{1}{v_1}\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u. \quad (3.34)$$

We can note that the system matrix for the system with concentrations as states is the transpose of the corresponding system matrix with masses as states.

## 3.3 Linear state space models

We have already encountered state space models. In the time-invariant case any state space model can be written as (1.7). The state is "stored" in the vector $\boldsymbol{x}$ and it constitutes a sufficient boundary condition for the differential equation system to have a unique solution. That is, if we know $\boldsymbol{x}(t)$ the future evolution of $\boldsymbol{x}$ is uniquely defined by the input $\boldsymbol{u}$.

A concrete example is the mass drug in a compartment. If we know the mass $x$ at time $t$, we can (at least numerically) compute the mass at a future time $\tau > t$, as long as we know the rates at which drug is added and removed between instances $t$ and $\tau$. But if we do not know the mass at time $t$, then information about the addition and subtraction rates is not sufficient to compute $\boldsymbol{x}(\tau)$. You can also think of the state as something that grounds the solution of the system.

A particularly important special case arises when the function $\boldsymbol{f}$ is (jointly) linear in $\boldsymbol{x}$ and $\boldsymbol{u}$. This means that we can express the dynamics as

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + B\boldsymbol{u}, \tag{3.35a}$$

$$\boldsymbol{y} = C\boldsymbol{x} + D\boldsymbol{u}. \tag{3.35b}$$

where $A$, $B$, $C$, and $D$ are matrices.

For obvious reasons $B$ is referred to as the input matrix, while $C$ is referred to as the output matrix. The matrix $A$ is called the system matrix. It defines how the state would evolve, for the constant zero input $\boldsymbol{u} = 0$. The matrix $D$ is referred to as the direct term (matrix). It provides a direct link between input and output. If we only have one state component, the $A$ "matrix" will only have one element, represented a scalar number. However, we still refer to $A$ as a matrix for consistency.

At this point, it might not be obvious why (3.35) is an *important* special case of (1.7). We will see why this is the case as we build up methodology for handling linear time-invariant systems throughout the chapters to follow.

### State variable transformations

As we have already seen in Example 3.6, the choice of state variables is associated with some freedom. In fact, there are infinitely many valid state choices. To convince yourself of this, consider a system with state $\boldsymbol{x}$. We might as well represent it with the alternative state $\boldsymbol{x}' = \alpha\boldsymbol{x}$ for some scalar $\alpha \neq 0$.

In the linear and time-invariant (LTI) case (1.7), it is straightforward to characterize all valid state representations. To see this, let $n$ be the number of elements in $\boldsymbol{x}$ of (3.35) and $T$ be an invertible (same as full rank) $n \times n$

matrix. We can then introduce the new state $\boldsymbol{x}'$ that relates to $\boldsymbol{x}$ through

$$\boldsymbol{x}' = T\boldsymbol{x}, \tag{3.36a}$$
$$\boldsymbol{x} = T^{-1}\boldsymbol{x}'. \tag{3.36b}$$

Inserting (3.36b) into (3.35) we get

$$T^{-1}\dot{\boldsymbol{x}}' = AT^{-1}\boldsymbol{x}' + B\boldsymbol{u}, \tag{3.37a}$$
$$\boldsymbol{y} = CT^{-1}\boldsymbol{x}' + D\boldsymbol{u}. \tag{3.37b}$$

Left-multiplying (3.37a) with $T$ we have

$$\dot{\boldsymbol{x}}' = TAT^{-1}\boldsymbol{x}' + TB\boldsymbol{u}, \tag{3.38a}$$
$$\boldsymbol{y} = CT^{-1}\boldsymbol{x}' + D\boldsymbol{u}, \tag{3.38b}$$

that we can write as

$$\dot{\boldsymbol{x}}' = A'\boldsymbol{x}' + B'\boldsymbol{u}, \tag{3.39a}$$
$$\boldsymbol{y} = C'\boldsymbol{x}' + D'\boldsymbol{u}, \tag{3.39b}$$

where

$$A' = TAT^{-1}, \tag{3.40a}$$
$$B' = TB, \tag{3.40b}$$
$$C' = CT^{-1}, \tag{3.40c}$$
$$D' = D, \tag{3.40d}$$

and conversely

$$A = T^{-1}A'T, \tag{3.41a}$$
$$B = T^{-1}B', \tag{3.41b}$$
$$C = C'T, \tag{3.41c}$$
$$D = D'. \tag{3.41d}$$

Using (3.40) and (3.41) we can thus move back and forth between the state space representations (3.35) and (3.39). Because there exist infinitely many invertible matrices $T$—for example if $T$ is invertible, so is $\alpha T$ for any scalar $\alpha \neq 0$—it directly follows that there exist infinitely many state space realizations of the LTI system (3.35).

What if $T$ is not invertible? Then, we will lose information going from $\boldsymbol{x}$ to $\boldsymbol{x}'$, and so $\boldsymbol{x}'$ will in general not be a valid state representation for the original dynamics. What then if our state transformation is not a linear mapping, as signified by the $T$ being a matrix? We could of course also

consider non-linear state transformations. However, applying them would in general result in the dynamics becoming nonlinear in the transformed state $\boldsymbol{x}'$, which means they cannot be written on the form (3.35).

In example Example 3.6 we see a concrete example of when a state transformation of the form (3.40) can be useful.

**Example 3.6: Compartment concentrations as state.** Assume we have an LTI compartment model on the form (3.7) and want to express the same dynamics with the compartment concentrations as state variables. Left-multiplying (3.7) with $V^{-1}$ we obtain

$$V^{-1}\dot{\boldsymbol{m}} = V^{-1}K\boldsymbol{m} - V^{-1}K_0\boldsymbol{m} + V^{-1}Bu. \tag{3.42}$$

Since $V^{-1}V = I$ we can left-multiply $\boldsymbol{m}$ with that to obtain

$$V^{-1}\dot{\boldsymbol{m}} = V^{-1}KVV^{-1}\boldsymbol{m} - V^{-1}K_0VV^{-1}\boldsymbol{m} + V^{-1}Bu, \tag{3.43}$$

which we can write as

$$\dot{\boldsymbol{\rho}} = K'\boldsymbol{\rho} - K_0'\boldsymbol{\rho} + B'u, \tag{3.44}$$

where

$$K' = V^{-1}KV, \tag{3.45a}$$
$$K_0' = V^{-1}K_0V, \tag{3.45b}$$
$$B' = V^{-1}B. \tag{3.45c}$$

Conversely, if we have a model with compartment substance concentrations as state variables, we can convert it into one with compartment substance masses as state variables, using the inverse transform of (3.45):

$$K = VK'V^{-1}, \tag{3.46a}$$
$$K_0' = VK_0'V^{-1}, \tag{3.46b}$$
$$B = VB'. \tag{3.46c}$$

Another practically motivated case, illustrated in Example 3.7, is when $T$ can be chosen so that $A' = TAT^{-1}$ is a diagonal matrix. We will return to when a diagonal state space realization is useful in Section 4.3.

**Example 3.7: Diagonalizing transform.** Assume we have a state space realization with

$$A = \begin{bmatrix} * & * \\ * & * \end{bmatrix}, \tag{3.47}$$

where each $*$ represents a number, and want to find a $2 \times 2$ matrix $T$ so that $A' = TAT^{-1}$ is diagonal:

$$A' = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}. \tag{3.48}$$

Denoting by $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ the eigenvectors of $A$, and by $\lambda_1$ and $\lambda_2$ the corresponding eigenvalues, we have from the definition of eigenvalue that

$$A\boldsymbol{v}_1 = \lambda_1 \boldsymbol{v}_1, \tag{3.49a}$$
$$A\boldsymbol{v}_2 = \lambda_2 \boldsymbol{v}_2. \tag{3.49b}$$
$$\tag{3.49c}$$

Packing the (column) eigenvectors into matrices we can equivalently write (3.49) as

$$A = \underbrace{\begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 \end{bmatrix}}_{S} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 \end{bmatrix} \underbrace{\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}}_{\Lambda}. \tag{3.50}$$

If $\Lambda$ is invertible, we thus have that

$$S^{-1}AS = \Lambda. \tag{3.51}$$

Combining (3.41) with (3.51), we see that the diagonalizing transform is given by

$$T = S^{-1} = \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 \end{bmatrix}^{-1} \tag{3.52}$$

Note that this example extends to cases with $n > 2$ states. We can also note that the diagonalizing transform exists when $S$ is diagonalizable. This happens exactly when the eigenvectors of $A$ are linearly independent.

In Chapter 4 we continue to work with linear models, and start to get a sense of why they are so popular. We will also see how we can approximate nonlinear models with linear ones.

***Further reading.*** Both (pharmacological) compartment models and linear systems enjoy a rich literature. An overview of the history of pharmacokinetics is found in [Wagner, 1981]. As we will see, linear systems will be a central theme throughout the remainder of this book. For the interested reader, a thorough mathematical reference is provided in [Kaliath, 1980]. The book [Rugh, 1995] covers essentially the same content, but at a somewhat more accessible level.

Wagner, J. G. (1981). "History of pharmacokinetics". *Pharmacology & therapeutics* **12**, pp. 537–562. DOI: 10.1016/0163-7258(81)90097-8.

Kaliath, T. (1980). *Linear systems*. Prentics Hall, Hoboken, NJ. ISBN: 978-0135369616.

Rugh, W. J. (1995). *Linear systems theory*. Pearson, London, England. ISBN: 978-0134412054.

# 4

# Linear models

**Learning goals**
After reading this chapter you should (be able to)

- Know what a constitutive model is, and how it differs from a balance equation.
- Write down a system of differential equations based on a component diagram of a material model.
- Define linearity and exemplify why it is a desirable property when it comes to system modeling.
- Identify stationary points of a nonlinear system.
- Perform local (Jacobian) linearization around a stationary point.
- verify and use the formula for the general solution to a linear system.
- Know what is meant by zero-order-hold sampling.

## 4.1 Biomechanics

***Historic Note.*** Biomechanics concerns mechanical properties (anatomy) and movement (physiology) of tissue. Fluid dynamics that govern blood and breathing gas flows are also included under the umbrella of biomechanics.

Like many other branches of basic physiology, biomechanics has a long history. Aristotle (384–322 B.C.) opened up the discipline of kinesiology—a biomechanical discipline that fuses kinetics and physiology—in his book “Movement of animals”. Another notable con-

tributor to biomechanics was Leonardo da Vinci (1452–1519). Within biology and medicine, da Vinci is most famous for his contributions to anatomy, including detailed drawings like the one in Figure 4.1. However, he went beyond anatomy and contributed to basic physiology by also describing how muscles, bones and tendons interact.

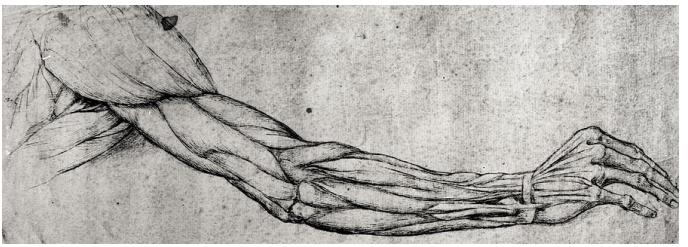

**Figure 4.1** Study of the human arm by Leonardo da Vinci.

Another milestone along the journey to modern biomechanics was "Physiologie des mouvements" by G. B. A. Duchenne (1806–1875). Based on knowledge from electrophysiological experiments, the book sets out to describe how individual muscles contribute to animal movement. These and similar contributions have since become cornerstones of orthopedics and sports medicine.

Diving deeper into physiological aspects later opened up the study of tissue biomechanics, to describe dynamic properties of tissues. Its close connection to material science, combined with the advent of modern tomography techniques including X-ray, synchrotron, and neutron scattering have led to a 22$^{\text{nd}}$ century renaissance of tissue biomechanics.

### Constitutive material models

As evident from the historic note, biomechanics is a broad term, and encompasses phenomenons from macroscopic down to microscopic scales. Here, we will delimit ourselves to study a class of models that can be used to describe properties of muscles and other tissues. Particularly, we will focus on the linearity property of these models.

So far, our modeling foundation has been balance equations. For example, in Section 3.1 a mass balance equation was used to derive the dynamics of compartment models. While balance equations are important, they are not the only building block of physical models. Consider a spring obeying Hooke's law

$$F = kx, \tag{4.1}$$

where $F$ [N $=$ kg m s$^{-2}$] is net pulling force, $x$ [meter] is elongation (displacement), and $k$ is a proportionality constant—the spring constant.

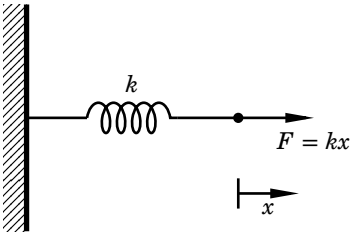


**Figure 4.2** A spring obeying Hooke's law. The convention is that $k \geq 0$, and that $x$ denotes the elongation of the spring. The externally applied force required to maintain the static equilibrium at displacement $x$ is $F = kx$. The vertical line with diagonal dashes to the left in the figure is the symbol of a fixed wall.

Hooke's law (4.1) is known as a constitutive equation since it relates two *different* physical entities; a force and a displacement. For the units to match, the spring constant must have the unit of $\mathrm{N\,m^{-1}} = \mathrm{kg\,s^{-2}}$. Conducting correct unit analyses is a fundamental part of good modeling practice, and we will return to it in Section 5.3.

A schematic illustration of a spring obeying Hooke's law is shown in Figure 4.2. The illustration shows a static equilibrium, in which the externally applied force $F$ is balanced by the contracting spring force at the displacement $x$ fulfilling $F = kx$. Note that this displacement is in relation to the equilibrium length $x = 0$ of the spring, corresponding to no applied force $F = k0 = 0$. A common source of mistakes is to confuse the internal spring force to the equal but opposite externally applied pulling force. We can also note that a net pushing force simply translates into $F < 0$.

Plotting the force $F$ against the elongation $x$, we obtain a straight line that passes through the origin of our coordinate system, and that has a slope of $k$, as shown in Figure 4.3. We can see in Figure 4.3 that the linear relation between force and displacement holds best for small displacements and forces. If we pull harder on the spring than was done in Figure 4.3 it will plasticise, meaning that it will permanently deform and not return to its original length if the external force is removed. Pulling even harder, the spring will eventually snap off. The linear model that Hooke's law defines thus has a *local* validity region. This is true for most (if not all) physiological models, and important to keep in mind when using the model. Particularly, if extrapolating to data outside those used when arriving at the model, it is not certain that the model will provide an adequate or useful approximation. In Section 4.2 we will see how we can obtain local linear approximations of dynamics from a nonlinear model.

Another fundamental element in constitutive material modeling is the viscous damper, schematically depicted in Figure 4.4. As with the spring,

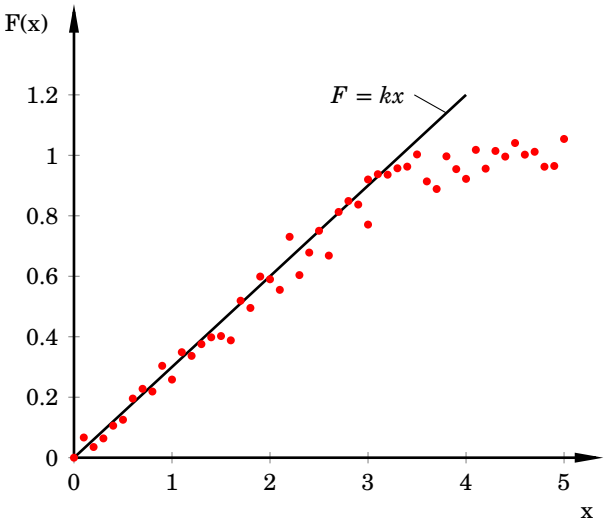


**Figure 4.3** Experimental data points (elongation, pulling force)= $(x, F)$ and the line $kx$ that best fits the data in the least-squares sense. Only data points with $x \leq 3$cm were used when fitting $k$, since the linear relation starts to deteriorate for $x > 3$ cm in our example.

the viscous damper defines a linear constitutive relation

$$F = \eta \dot{x}. \tag{4.2}$$

Here, the externally applied force needed to maintain a steady (stationary) velocity $\dot{x}$ of the damper element is $F = \eta\dot{x}$. In the spring, zero force is needed to maintain the elongation $x = 0$; in the damper zero force is needed to maintain the velocity $\dot{x} = 0$. As with the spring, the sign convention is that the damping coefficient $\eta$ is greater than zero for a damper. There are

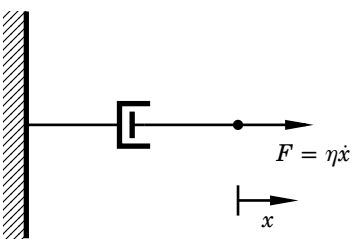


**Figure 4.4** Schematic illustration of a viscous damper. While a spring contributes with a force $kx$ proportional to its elongation $x$, a viscous damper contributes with a force $-\eta\dot{x}$ proportional to its rate of change of elongation, $\dot{x}$.

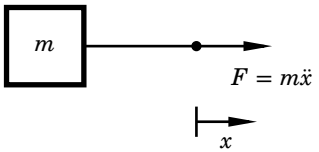


**Figure 4.5** Schematic illustration of a mass element. When a net external force $F$ is applies, the mass $m$ undergoes an acceleration $\ddot{x}$ governed by Newton's second law (4.3).

situations that can be modeled with $\eta < 0$, and we will return to these in Section 6.2 when discussing stability.

A third constitutive element is the mass. The spring (4.1) relates force to elongation $x$, and the damper (4.2) relates force to the time derivative of elongation, $\dot{x}$. A mass element instead relates the net applied force to acceleration $\ddot{x}$ through Newton's second law:

$$F = m\ddot{x}, \tag{4.3}$$

as schematically illustrated in Figure 4.5.

Next, we demonstrate how the constitutive elements can be combined into a constitutive material model, and give two examples of such models: two commonly used visco-elastic material models due to Voigt and Maxwell, respectively. "Viscoelastic" signals that they have both viscous damping and elastic properties, thus containing both a damper and a spring. They each also contain a mass element, giving them an inertial property, since the mass needs to be accelerated for elongation to occur. The combination of constitutive elements that make up each model can be expressed visually as a component diagram, from which we can infer the model's dynamic behavior.

**Example 4.1: The Voigt model.** The Voigt model is shown in Figure 4.6. It can be used to describe how a specimen of tissue responds when pulled on by an external force $F$.

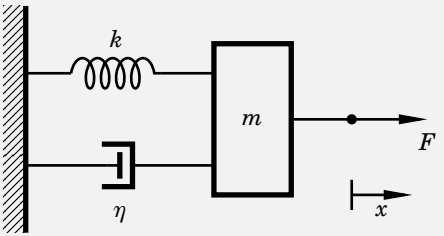


**Figure 4.6** Component diagram of the Voigt model, where the spring and damper are connected in parallel to the mass element.

The displacement $x$ of the damper equals the elongation of the spring. The force pulling the mass to the left (negative direction) is thus $-kx - \eta\dot{x}$, and the net force on the mass is the sum of this force and the externally applied force $F$. Applying Newton's second law (4.3) to this force, we obtain

$$m\ddot{x} = -kx - \eta\dot{x} + F, \tag{4.4}$$

being a differential equation in $x$, driven by $F$. Introducing a state vector $\boldsymbol{x} = [x\ \dot{x}]^\top$, input $u = F$, and output $y = x_1$, we could write (4.4) on state space form (1.9a):

$$\dot{x}_1 = x_2, \tag{4.5a}$$

$$\dot{x}_2 = -\frac{k}{m}x_1 - \frac{\eta}{m}x_2 + \frac{1}{m}F, \tag{4.5b}$$

$$y = x_1. \tag{4.5c}$$

Alternatively we can provide the equations of (4.5) in matrix form

$$\dot{\boldsymbol{x}} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\eta}{m} \end{bmatrix}}_{A}\boldsymbol{x} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}}_{B}u, \tag{4.6a}$$

$$y = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C}, \tag{4.6b}$$

(with $D$=0).

Moving on to the Maxwell model, let us derive the differential equation in a systematic way, that will work also for constitutive material models of

higher complexity (more elements). This is done by introducing temporary variables to denote the elongations or displacements of individual elements, alongside forces acting on the individual elements. We will use two fundamental properties when combining (groups of) elements:

- If connected in parallel, their total elongations or displacement must be equal.
- If connected in series, the elongation or displacement of the interconnection is the sum of elongations and displacements of the connected parts.

For the spring and damper, the force pulling to the left on the left side of the element is equal in magnitude to the force pulling to the right at the right side of the element. Note that for the mass, this is generally not true. Finally, forces are continuous across connections between elements: the sum of forces at each interconnection must equal zero.

**Example 4.2: The Maxwell model.** The Maxwell model is schematically illustrated in Figure 4.7, with the positive direction being to the right.

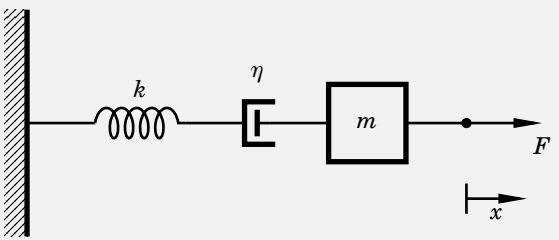


**Figure 4.7** Component diagram of the Maxwell constitutive material model.

Let us denote by $x_s$ the elongation of the spring and by $F_s$ the force pulling to the right on the right side of the spring . From (4.1) we have that

$$F_s = kx_s. \tag{4.7a}$$

Denoting by $x_d$ the elongation of the damper, and by $F_d$ the force pulling to the right on the right side of the damper, (4.2) gives

$$F_d = \eta \dot{x}_d. \tag{4.7b}$$

The force pulling to the right on the right side of the spring is $F_s$. The force pulling to the right on the right side of the damper is $F_d$. The force pulling to the left on the left side of the damper is thus $-F_d$. For force to be continuous across the intersection between the spring and the damper, it is required that

$$F_s - F_d = 0, \tag{4.7c}$$

which we can interpret as the sum of the force pulling to the right on the right side of the spring being equal in magnitude to the force pulling to the left on the left side of the damper.

The total elongation of the series interconnection of the spring and the damper is

$$x = x_s + x_d, \tag{4.7d}$$

where $x$ is also the displacement of the mass. The net force on the mass,

$$F - F_d = m\ddot{x}, \tag{4.7e}$$

obeys Newton's second law (4.3). The force terms in (4.7e) are the force $F$ pulling to the right on the right side of the mass, and $-F_d$, pulling to the left on the left side of the mass.

Combining the equations of (4.7) we obtain the system of equations

$$\begin{aligned} F_s &= -kx_s, && \text{(4.8a)}\\ F_d &= -\eta\dot{x}_d, && \text{(4.8b)}\\ F_s &= F_d, && \text{(4.8c)}\\ x &= x_s + x_d, && \text{(4.8d)}\\ F &= F_d + m\ddot{x} && \text{(4.8e)} \end{aligned}$$

and are left with eliminating the temporary variables $x_s, F_s, x_d$, and $F_d$, to end up with a differential equation parameterized in $k$, $\eta$, and $m$, that relates $F$ to $x$.

We can write the time-derivative of (4.8d) as

$$\begin{aligned} \dot{x} = \dot{x}_s + \dot{x}_d &= -\frac{d}{dt}\left(\frac{F - m\ddot{x}}{k}\right) - \frac{F - m\ddot{x}}{\eta} \\ &= \frac{m}{k}\dddot{x} + \frac{m}{\eta}\ddot{x} - \frac{1}{k}\dot{F} - \frac{1}{k}F. \end{aligned} \tag{4.9}$$

Unlike the previous differential equations we have studied, (4.9) also

involves differentiation of the input—in our case the driving force $F$. Looking at (4.9) it is therefore not directly obvious that it can be written on the standard state space form (3.35) of an LTI system. In Section 6.1 we will introduce the Laplace transform that will both serve to show how this is done, and as a tool that enables us to establish the relation between $F$ and $x$ in a more straightforward way than the one in Example 4.2.

A related, and important, thing to note about the constitutive elements is that it is not obvious which of the related physical entities should be regarded as an input (action), and which should be regarded as an output (reaction). For example, we could view applied force as an input to the spring element, and deformation as the resulting output. But we could also view deformation as the input, and corresponding force as an output. In contrast, when dealing with pharmacokinetic compartment models in Section 3.1, it was natural to view drug administration as input and arising (central) compartment drug concentration as output.

#### The viscoelastic Hill muscle model

Let us next attempt to model the contractile dynamics of a skeletal muscle. The contractile force is generated by contractile elements comprising of the proteins myosin and actin. These elements are arranged in individual muscle cells, also known as muscle fibers. A motor neuron connects to a number of muscle fibers, forming a motor unit. The motor unit is electrically activated by the motor neuron in a process involving the neurotransmitter acetylcholine.

The properties of an isolated contractile element can be studied through electrophysiological experiments, in which a specimen of muscle fiber is suspended in a setup where contractile force resulting from applying an electric excitation potential across the specimen can be measured. Assume we have thus obtained a good model for the contractile element itself, and are interested in modeling how its contractile force contributes to contraction of the muscle. The spring plays a natural role here as it models elasticity of the muscle, and in Example 4.3 we derive the dynamics of the elastic two-element Hill model.

**Example 4.3: Hill muscle model.** The Hill model, named after physiologist and Nobel price laureate Archibald Hill (1886–1977), uses constitutive elements (springs) to model the dynamics of a muscle. The structure of the two-element elastic Hill model is shown in Figure 4.8. The contributions from all contractile elements are lumped together and contribute with a total contractile force $F_c$, that can be regarded as the input of our system.

In this example we will consider isometric contraction, meaning that the muscle is constrained so that it cannot change length. This is for example what you would have if you attempt to lift an object that is too heavy for you to lift. The externally applied pulling force, denoted $F$ in Figure 4.8, can then be regarded as the output of our system, and we want to establish a relationship between $F_c$ and $F$.

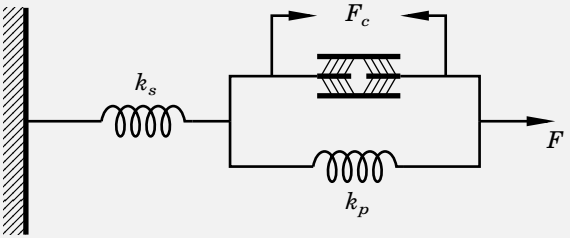


**Figure 4.8** Component diagram of the two-element elastic Hill model, with contractile force $F_c$ and externally applied pulling force $F$.

The series connected spring with spring constant $k_s$ models the elasticity of the tendon to which the muscle attaches, as well as the muscle fibres themselves, while the spring connected in parallel, with spring constant $k_p$, models forces from connecting tissues.

Let $x_s$ denote elongation of the serial element, and $x_p$ elongation of the parallel element. Furthermore, let $F_s$ and $F_p$ denote the corresponding contractile forces within each element. Force balance, geometric constraints, and the property of the constitutive elements (in this case two springs) give

$$F = F_s + F_p, \tag{4.10a}$$
$$F_s = F_c, \tag{4.10b}$$
$$x_s + x_p = 0, \tag{4.10c}$$
$$x_s = -k_s F_s, \tag{4.10d}$$
$$x_p = -k_p F_p. \tag{4.10e}$$

Solving for $F$ we obtain

$$F = \frac{k_p - k_s}{k_p} F_c. \tag{4.11}$$

Note that (4.11) is a static model, as $F$ can be written as an ordinary function of $F_c$. But if we would replace one of the springs with a damper, its force contribution would instead be $F = -\eta\dot{x}$, and we would arrive at a dynamic relation between $F_c$ and $F$.

## 4.2 Linearity and linearization

### Linearity

We have seen a few examples of linear dynamics, such as the linear compartment models in Chapter 3 and the linear viscoelastic models in this chapter. Before moving on, let us establish an understanding of what is meant by linearity, and why it is a property of interest in dynamic system modeling.

We begin with the definition of a linear function. A function $f$ is linear if and only if

$$f(\alpha v) = \alpha f(v), \tag{4.12a}$$
$$f(v + w) = f(v) + f(w) \tag{4.12b}$$

holds for all scalar coefficients $\alpha$ and possible combinations of arguments $v$ and $w$. Depending on $f$, the arguments could be either scalar or vector-valued. And the same definition is used if the value set of $f$ is vector-valued, in which case we denote the function $\boldsymbol{f}$ instead.

We can use the *definition* (4.12) of a linear function to determine whether a particular function is linear, as illustrated below in Example 4.4.

**Example 4.4: Nonlinear function.** Let us consider the function $f(x) = \sqrt{x}$. We have that

$$f(\alpha x) = \sqrt{\alpha x} = \sqrt{\alpha}\sqrt{x} = \sqrt{\alpha} f(x). \tag{4.13}$$

The right-hand-side is not equal to $\alpha f(x)$ unless $\alpha = 0$ or $\alpha = 1$, and therefore $f$ is nonlinear (the same as not linear) according to the definition (4.12).

Now that we have seen an example of a nonlinear function, let us follow up on the previously made claim that $\boldsymbol{f}$ defines the dynamics of the LTI system (3.35) is in fact linear.

**Example 4.5: Linear function.** The function we are considering is defined by

$$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) = A\boldsymbol{x} + B\boldsymbol{u}, \tag{4.14}$$

where the arguments $\boldsymbol{x}$ and $\boldsymbol{u}$ are column vectors, while $A$ and $B$ are constant "coefficient" matrices. Since $\boldsymbol{f}$ has two arguments, we cannot directly check whether (4.14) fulfills the linearity definition involving (4.12). But just as we can regard a function of the two scalar arguments $x_1$ and $x_2$ as a function of a single vector argument $\boldsymbol{x} = [x_1\ x_2]^\top$, we can introduce $\boldsymbol{z} = [\boldsymbol{x}^\top\ \boldsymbol{u}^\top]^\top$. Letting $n$ and $m$ denote the number of elements of $\boldsymbol{x}$ and $\boldsymbol{u}$, respectively, we have that

$$\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u}) = A\boldsymbol{x} + B\boldsymbol{u} = \tilde{\boldsymbol{f}}(\boldsymbol{z}) = \underbrace{\begin{bmatrix} A & 0_{n\times m} \\ 0_{n\times n} & B \end{bmatrix}}_{M}\boldsymbol{z}, \tag{4.15}$$

where the subscripts indicate the sizes (rows × columns) of matrices, while $\tilde{f}$ represents the same function as $f$, but with with a single input $\boldsymbol{z} = [\boldsymbol{x}^\top\ \boldsymbol{u}^\top]^\top$. It is thus sufficient to test for linearity of $\tilde{\boldsymbol{f}}$.

We have that

$$\tilde{\boldsymbol{f}}(\alpha\boldsymbol{v}) = M\alpha\boldsymbol{v} = \alpha\tilde{\boldsymbol{f}}(\boldsymbol{v}), \tag{4.16}$$

which fulfills (4.12a) for all scalar $\alpha$. We also have (from the behavior of vector–vector addition that

$$\tilde{\boldsymbol{f}}(\boldsymbol{v}+\boldsymbol{w}) = M(\boldsymbol{v}+\boldsymbol{w}) = M\boldsymbol{v} + M\boldsymbol{w} = \tilde{\boldsymbol{f}}(\boldsymbol{v}) + \tilde{\boldsymbol{f}}(\boldsymbol{w}). \tag{4.17}$$

We have therefore shown that $\tilde{\boldsymbol{f}}$, and consequently $\boldsymbol{f}$, are linear functions. The same applies to the output or measurement equation of (3.35), with $C$ and $D$ replacing $A$ and $B$.

To get a glimpse of why linearity is a useful property, assume we have an LTI system on the form (3.35), and that $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$ is known at some fixed initial time $t_0$. Let us apply an input signal $\boldsymbol{u}(t) = \boldsymbol{v}(t)$ to the system and record the corresponding output $\boldsymbol{y}(t) = \boldsymbol{y}_v(t)$. Then we take the system back to the state $\boldsymbol{x}_0$ and repeat the experiment for another input $\boldsymbol{u}(t) = \boldsymbol{w}(t)$ and record the resulting measurement $\boldsymbol{y}(t) = \boldsymbol{y}_w(t)$. Knowing that the system is linear, the resulting signals $\boldsymbol{y}_v$ and $\boldsymbol{y}_w$ contain complete information of what the resulting output for any linear combination of inputs $\alpha\boldsymbol{v}(t) + \beta\boldsymbol{w}(t)$ would be, namely $\boldsymbol{y}(t) = \alpha\boldsymbol{y}_v(t) + \beta\boldsymbol{y}_w(t)$. A graphical illustration of this concept is shown in Figure 4.9

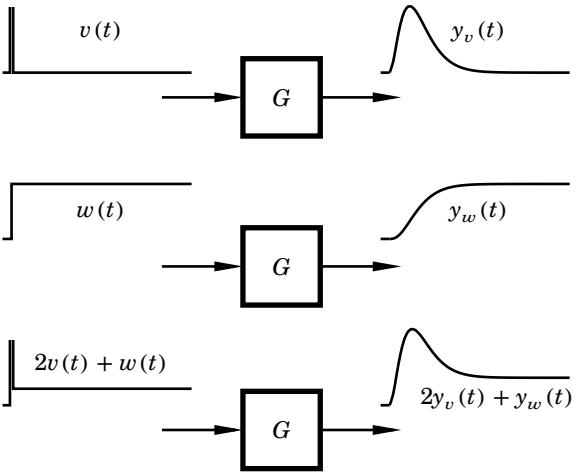


**Figure 4.9** Superposition allows us to compute the output of a linear system when subjected to the input $\alpha v(t) + \beta w(t)$ if we know the outputs resulting from the inputs $v(t)$ and $w(t)$.

In other words, if we can decompose the input $\boldsymbol{u}$ into a linear combination of components for which we each know the system output, then we can compute the system output directly, without having to simulate the system or run an experiment. The important property that a linear combination $\alpha \boldsymbol{v}(t) + \beta \boldsymbol{w}(t)$ of inputs result in the corresponding linear combination $\alpha \boldsymbol{y}_v(t) + \beta \boldsymbol{y}_w(t)$ (as described above), is referred to as superposition. In Section 6.1 we will use the superposition property to completely characterize scalar-input LTI systems in terms of their responses to inputs on the form $u(t) = \sin(\omega t)$. When considering systems of several inputs, the superposition property enables us to consider one input at the time, as their combined effect on the system state and output is simply the sum of their individual contributions.

Another consequence of superposition is that the LTI property is preserved under various compositions of linear systems, as we will look more closely at in Section 6.2. For example, if two systems are connected in series, so that the input $\boldsymbol{u}$ of the second system is the output $\boldsymbol{y}$ of the first, then the combined system will also be LTI.

Finally, there exist a formula for the solution of linear differential equations, that becomes particularly useful in certain cases, as we will see in Section 4.3.

### Linearization

We have now seen some initial motivation as to why the superposition principle, being valid for LTI systems, is a desirable property. The true power of this seemingly subtle property will become more clearly apparent when we consider interconnections between, and responses of, LTI systems in Section 6.2 and Chapter 7, respectively.

Yet, most systems you will encounter in nature are to some extent nonlinear, meaning that superposition does not hold. Here we will show how we can approximate a nonlinear system with a linear one, around a considered *operating point*.

We define the operating point as some point $\boldsymbol{x}_0$ in the state space of the system, in vicinity of which we want our model to approximate the system dynamics well. You can for example think of the spring that obeys Hooke's law, as long as elongation is small, as was illustrated in Figure 4.3. Here we consider only stationary operating points, being points $(\boldsymbol{x}, \boldsymbol{u}) = (\boldsymbol{x}_0, \boldsymbol{u}_0)$ for which

$$0 = \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{u}_0) \tag{4.18}$$

holds. In words: a stationary operating point $(\boldsymbol{x}_0, \boldsymbol{u}_0)$ is such that there for a constant input $\boldsymbol{u}_0$ the system remains in $\boldsymbol{x}_0$, since $\dot{\boldsymbol{x}} = 0$. Let us illustrate this with a simple scalar example.

**Example 4.6: Stationary operating point.** Consider a constitutive modeling element that behaves somewhat like a spring, but with the nonlinear constitutive relation

$$F_s = kx^2 \tag{4.19}$$

between spring force $F_s$ and elongation $x$.

We connect this element in series with a mass element according to Figure 4.10 and apply an external pulling force $F = u$.

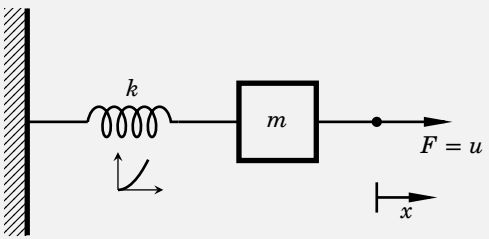


**Figure 4.10** Constitutive model consisting of mass element and nonlinear spring element.

The relation between force $u$ and displacement $x$ is thus

$$(u - kx^2) = m\ddot{x} \tag{4.20}$$

Introducing state variables $x_1 = x$ and $x_2 = \dot{x}$ we have that

$$\dot{x}_1 = f_1(\boldsymbol{x}, u) = x_2, \tag{4.21a}$$

$$\dot{x}_2 = f_2(\boldsymbol{x}, u) = -\frac{k}{m}x_1^2 + \frac{1}{m}u, \tag{4.21b}$$

where $\boldsymbol{f} = [f_1\, f_2]^\top$ and $\boldsymbol{x} = [x_1\, x_2]^\top$.

Possible stationary points are those $(\boldsymbol{x}, u)$ for which $\boldsymbol{f}(\boldsymbol{x}, u) = 0$, which we can expand to

$$0 = x_2, \tag{4.22a}$$

$$0 = -\frac{k}{m}x_1^2 + \frac{1}{m}u. \tag{4.22b}$$

The first equation (4.22a) gives us that $x_2 = 0$, while the second equation (4.22b) gives us that $-kx_1^2 + u = 0$. There are thus infinitely many stationary points for this system. Parameterizing the solution with a parameter $x_1 = p$, we can write all stationary points $(\boldsymbol{x}_0, u_0)$ as $([p\ 0]^\top, kp^2)$. In words, this means that for any elongation $x$, there is a corresponding pulling force $F = u$ that keeps the system stationary at $x$.

Assume we have identified a stationary point $(\boldsymbol{x}_0, \boldsymbol{u}_0)$ of interest to us. What we can then do is to introduce new state variables $\Delta\boldsymbol{x}$ and inputs $\Delta\boldsymbol{u}$, that define the deviation from the stationary point:

$$\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_0 \tag{4.23a}$$

$$\Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_0. \tag{4.23b}$$

Writing out the first terms of the Taylor series of $\boldsymbol{f}$ around that stationary point, we obtain

$$\begin{aligned}\Delta\dot{\boldsymbol{x}} &= \dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_0 = \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \\ &\approx \underbrace{\boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{u}_0)}_{0} + \underbrace{\left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right)(\boldsymbol{x}_0, \boldsymbol{u}_0)}_{A}\underbrace{(\boldsymbol{x} - \boldsymbol{x}_0)}_{\Delta\boldsymbol{x}} + \underbrace{\left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right)(\boldsymbol{x}_0, \boldsymbol{u}_0)}_{B}\underbrace{(\boldsymbol{u} - \boldsymbol{u}_0)}_{\Delta\boldsymbol{u}},\end{aligned} \tag{4.24}$$

where the approximation error consists of us having removed the higher-order terms from the Taylor series expansion of $\boldsymbol{f}$.

The matrix

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}. \tag{4.25}$$

is known as the Jacobian of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$. The type of linearization we are looking into is sometimes referred to as Jacobian linearization, in order not to mix it up with feedback linearization (that we will not deal with here). The Jacobian with respect to $\boldsymbol{u}$ is definied similarily:

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}} = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}. \tag{4.26}$$

As in (1.1), $n$ denotes the number of state components and $m$ the number of input components. The parenthesis around the Jacobians in (4.24) signify that the Jacobians should first be determined, then evaluated at the stationary point (as opposed to evaluating $\boldsymbol{f}$ at the stationary point and then determining the Jacobian of the resulting expression). An output equation $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})$ is approximated in the same way with

$$C = \left( \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} \right) (\boldsymbol{x}_0, \boldsymbol{u}_0), \tag{4.27a}$$

$$D = \left( \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}} \right) (\boldsymbol{x}_0, \boldsymbol{u}_0). \tag{4.27b}$$

Thus evaluating the Jacobians and using the short-hand notation introduced in (4.24), we have now arrived at the LTI system dynamics

$$\Delta \dot{\boldsymbol{x}} = A \Delta \boldsymbol{x} + B \Delta \boldsymbol{u}, \tag{4.28a}$$

$$\Delta \boldsymbol{y} = C \Delta \boldsymbol{x} + D \Delta \boldsymbol{y}, \tag{4.28b}$$

where

$$\Delta \boldsymbol{y} = \boldsymbol{y} - \boldsymbol{y}_0, \tag{4.29}$$

and

$$\boldsymbol{y}_0 = \boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{u}_0). \tag{4.30}$$

Let us give an illustrative example in the scalar case.

**Example 4.7: Linearization of a scalar system.** We consider a time-invariant system with dynamics given by

$$\dot{x} = -x(x+2) + u, \tag{4.31}$$

Stationary points for this system are the solutions of

$$0 = -x(x+2) + u. \tag{4.32}$$

Assuming we are at a stationry point defined through $u = u_0$, the corresponding stationary $x = x_0$ is thus given by

$$x_0 = -1 \pm \sqrt{1+u_0}. \tag{4.33}$$

This means that for any $u_0 > -1$ we have two stationary points, for $u_0 = -1$ there is a unique stationary point, and for $u_0 < -1$ the system lacks stationary points.

Let us linearize the system around a the stationary point $(x_0, u_0) = (1, 3)$. (In a real scenario, the choice of stationary point is often easy, since we would typically know ahead of time around what operating point we are interested in modeling the system dynamics.)

To visualize the situation, we can fix $u = u_0 = 3$ and plot $f(x,u) = -x(x+2)+u$ as a function of $x$ alone, as shown in figure Figure 4.11.

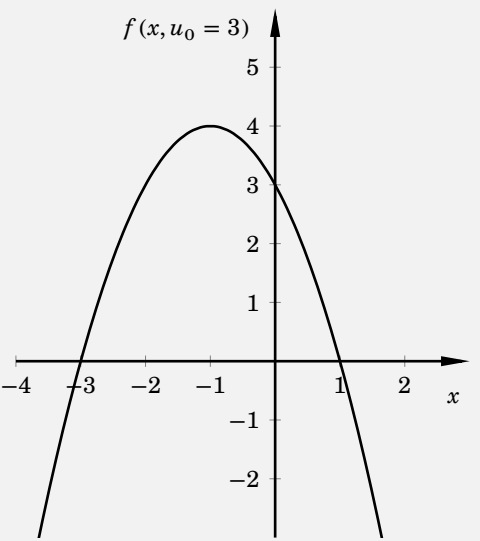


**Figure 4.11** Dynamics function $f(x, u_0) = -x(x+2) + 3$ plotted as a function of $x$.

We could of course draw a similar plot as that in Figure 4.11, showing a function of $u$, where we instead fix $x = x_0$.

We are interested in constructing a linear approximation around the stationary point $(x_0, u_0) = (1, 3)$ and introduce the offset variables

$$\Delta x = x - x_0 = x - 1, \tag{4.34a}$$
$$\Delta u = u - u_0 = u - 3. \tag{4.34b}$$

In Figure 4.11 the offset variables correspond to moving the stationary point to the origin, as illustrated in Figure 4.12.

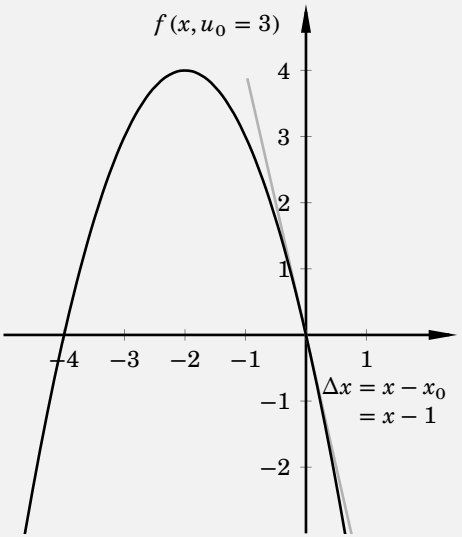


**Figure 4.12** Dynamics function $f(x, u_0 = 3)$ plotted as a function of $\Delta_x = x - x_0 = x - 1$ in black, together with linear approximation in gray. The inclination of the linear approximation is $\partial f(x, u)/\partial x$, evaluated at $(x, u) = (x_0, u_0) = (1, 3)$.

Going back to (4.24), we have that

$$\Delta\dot{x} \approx \left(\frac{\partial f}{\partial x}\right)(x_0, u_0)\,\Delta x + \left(\frac{\partial f}{\partial u}\right)(x_0, u_0)\,\Delta u, \tag{4.35}$$

where the first term constitutes the straight line in Figure 4.12. We see that when $|\Delta x|$ is small, the difference between $f$ an the linear approximation is relatively small, but also that the difference increases

for larger $|\Delta x|$. Again, we could make a corresponding plot to analyze the approximation error in $u$ with $x = x_0$ fixed. In this particular example, there will be no such approximation error, since $f$ is already linear in $u$.

The situation in Example 4.7 generalizes to vector valued $\boldsymbol{x}$, $\boldsymbol{u}$ and $\boldsymbol{f}$. In the example, the linear approximation of $f(x, u)$ is a plane in three-dimensional space, and when we fixed $u = u_0$ we cut through this plane to obtain the line in Figure 4.12. In the general case, the linear approximation is the generalization of a plane, a hyperplane, living in a space of dimension $n + m + 1$, where $n$ is the number of states, and $m$ the number of scalar inputs. We cannot draw this hyperplane, but we utilize the intuition we gained from Example 4.7. To summarize, linearization follows the following scheme:

1. Find a stationary point $(\boldsymbol{x}_0, \boldsymbol{u}_0)$ of interest that fulfils $0 = \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{u}_0)$, at (or around) which to approximate the dynamics with a linear systems.

2. Introduce the offset variables $\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_0$ and $\Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_0$, that translate the stationary point to the origin of our coordinate system.

3. Approximate $\boldsymbol{f}$ with a linear function defined by the first terms of its Taylor series expansion. Since we have translated the stationary point to the origin, this corresponds to replacing $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$ with two hyperplanes that are tangent to $\boldsymbol{f}$ at the origin. These hyperplanes are defined by the Jacobians $\partial\boldsymbol{f}/\partial\boldsymbol{x}$ and $\partial\boldsymbol{f}/\partial\boldsymbol{u}$, and model deviations in $\boldsymbol{x}$ and $\boldsymbol{u}$ away from the stationary point.

4. If there is an output equation $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})$, introduce the offset variable $\Delta\boldsymbol{y} = \boldsymbol{y} - \boldsymbol{y}_0$, where $\boldsymbol{y}_0 = \boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{u}_0)$, and repeat the above points to approximate it with two hyperplanes in the same way as $\boldsymbol{f}$ was approximated.

As a special case, let us see what happens if the function that we are linearizing is already linear.

**Example 4.8: Linearization of a linear function.** Let us return to Example 4.7, and assume that there is also an output equation,

$$y = 3x - 2u. \tag{4.36}$$

At the considered stationary point $(x_0, u_0) = (1, 3)$ we have that

$$y_0 = g(x_0, u_0) = -3, \tag{4.37}$$

and

$$\Delta y = y + 3. \tag{4.38}$$

Furthermore we have that

$$\frac{\partial g}{\partial x} = 3, \tag{4.39a}$$

$$\frac{\partial g}{\partial u} = -2. \tag{4.39b}$$

The linearized output equation is thus

$$\Delta y = 3\Delta x - 2\Delta u = g(\Delta x, \Delta u). \tag{4.40}$$

In other words, the linearized output equations is the same as the original, already linear, output approximation. This should not come as a surprise: if a function that is itself defined by a (hyper)plane is approximated with a tangenting (hyper)plane, then the approximation will be the function itself. While we have considered the output equation here, the same result of course also holds for the function $f$ or $\boldsymbol{f}$ defining the dynamics.

Now we know everything there is to know about Jacobian linearization. Let us finish off with an example involving a system with vector-valued state, and scalar valued output.

**Example 4.9: Linearization.** In this example we perform a linearization of the dynamics introduced in Example 4.6. The state dynamics are given by (4.21), and we introduce a measurement according to (4.41c):

$$\dot{x}_1 = x_2, \tag{4.41a}$$

$$\dot{x}_2 = -\frac{k}{m}x_1^2 + \frac{1}{m}u, \tag{4.41b}$$

$$y = \log(x_1 + 1). \tag{4.41c}$$

Assume that we want to model how the system behaves when $u \approx 4$. From Example 4.6 we see that this corresponds to stationary $\boldsymbol{x}$

candidates $\boldsymbol{x}_0 = [\pm 2/\sqrt{k}\ 0]^\top$. Let us assume that we are interested in $\boldsymbol{x}_0 = [2/\sqrt{k}\ 0]^\top$. We can then introduce the offset variables

$$\Delta x_1 = x_1 - \frac{2}{\sqrt{k}}, \tag{4.42a}$$

$$\Delta x_2 = x_2. \tag{4.42b}$$

We also have that

$$y_0 = \log(1) = 0, \tag{4.43a}$$

$$\Delta y = 1. \tag{4.43b}$$

Next, we compute the Jacobians:

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{f_1}{x_1} & \frac{f_1}{x_2} \\ \frac{f_2}{x_1} & \frac{f_2}{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{-2k}{m} x_1 & 0 \end{bmatrix}, \tag{4.44a}$$

$$\frac{\partial \boldsymbol{f}}{\partial u} = \begin{bmatrix} \frac{f_1}{u} \\ \frac{f_2}{u} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}, \tag{4.44b}$$

$$\frac{\partial g}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial g}{\partial x_1} & \frac{\partial g}{\partial x_2} \end{bmatrix} = \begin{bmatrix} \frac{1}{x_1 + 1} & 0, \end{bmatrix}, \tag{4.44c}$$

$$\frac{\partial g}{\partial u} = 0. \tag{4.44d}$$

Evaluating the Jacobians at the stationary point, we obtain

$$A = \begin{bmatrix} 0 & 1 \\ \frac{-4}{m\sqrt{k}} & 0 \end{bmatrix}, \tag{4.45a}$$

$$B = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}, \tag{4.45b}$$

$$C = \begin{bmatrix} \frac{\sqrt{k}}{2 + \sqrt{k}} & 0 \end{bmatrix}, \tag{4.45c}$$

$$D = 0, \tag{4.45d}$$

that define our linearized system

$$\Delta \dot{\boldsymbol{x}} = A\Delta \boldsymbol{x} + B\Delta u, \tag{4.46a}$$
$$\Delta y = C\Delta \boldsymbol{x} + D\Delta u. \tag{4.46b}$$

## 4.3 Solving linear differential equations

In this section we study the *exact* solution of LTI systems. In Chapter 2 we used numeric integration techniques (such as forward and backward Euler) to *approximate* the solution of ordinary differential equations on the form (2.3). It turns out that for LTI systems, we can instead obtain the solution using an explicit formula. Before presenting and verifying this formula, let us start with a special-case example: a scalar autonomous LTI system no output.

**Example 4.10: Solution of scalar (first order) LTI system.**
We consider an autonomous LTI system with scalar state:

$$\dot{x}(t) = ax(t). \tag{4.47}$$

We know from fundamental calculus that

$$\frac{d\log x(t)}{dt} = \frac{1}{x(t)}\frac{dx(t)}{dt} = \frac{\dot{x}(t)}{x(t)}, \tag{4.48}$$

where the last factor in the intermediate expression is the inner derivative resulting from the chain rule. Dividing (4.47) by $x(t)$ and integrating with respect to time, we thus obtain

$$\int \frac{\dot{x}(t)}{x(t)}\,dt = \log x(t) + c_1 = \int a\,dt = at + c_2, \tag{4.49}$$

for any real constants $c_1$ and $c_2$. Letting $c = c_2 - c_1$ we can write

$$\log x(t) = at + c. \tag{4.50}$$

Assuming the initial condition

$$x(0) = x_0 \tag{4.51}$$

is known, we can write (4.50) as

$$\log x_0 = c. \tag{4.52}$$

Inserting this into (4.50) we have

$$\log x(t) - \log x_0 = at. \tag{4.53}$$

Computing the exponent of both sides and re-arranging we finally arrive at

$$x(t) = x_0 e^{at}. \tag{4.54}$$

To verify that (4.54) is indeed a solution of (4.47), we differentiate (4.54) with respect to $t$:

$$\dot{x}(t) = a x_0 e^{at} = a x(t). \tag{4.55}$$

We will not make a derivation of the general-case formula here. Instead, we will present it, and then verify that it indeed constitutes a solution. For a system on state space form (1.9), with $\boldsymbol{x}(0) = \boldsymbol{x}_0$, the solution is given by

$$\boldsymbol{y}(t) = Ce^{At}\boldsymbol{x}_0 + C\int_0^t e^{A(t-\tau)}B\boldsymbol{u}(\tau)\, d\tau + D\boldsymbol{u}(t). \tag{4.56}$$

Before verifying that (4.56) is indeed a solution to (1.9), we need to understand what is meant by $e^A$, where $A$ is a square matrix.

Any function $f$ that has a convergent Taylor series expansion is called analytic. For analytic functions it therefore holds that

$$f(A) = \sum_{k=0}^{\infty} \frac{1}{k!}\left(\frac{\partial^k f}{\partial A^k}\right)(0)\, A^k. \tag{4.57}$$

That is, $f(A)$ is a weighted sum of terms $A^k$. The weights are proportional to the $k^{\text{th}}$ derivative of $f$ evaluated at 0, with proportionality constant $1/k! = 1/(1 \cdot \ldots \cdot k)$.

Since multiplication between square matrices is defined, we can compute $A^k$, itself being a square matrix, by repeatedly multiplying $A$ by itself $k-1$ times. For the exponential function, we can compute

$$\frac{\partial f}{\partial A} = \frac{\partial}{\partial A}e^A = Ae^A. \tag{4.58}$$

Applying the chain rule to (4.58), we can thus also compute the higher-order derivatives needed in the evaluation of (3.3a). Furthermore, since multiplication between a square matrix and a scalar is defined, we can compute any term of (4.57). These terms are also square matrix, and thus can be added to form the sum defining $f(A)$ through (4.57).

The above provides a definition of functions of a square matrix, such as $e^A$ or $\cos(A)$. You might object and say that it is impractical to evaluate the infinite sum of (4.57). It is, but this problem also shows up when trying to evaluate its scalar counterparts, such as $e^3$. By including sufficiently many terms of the Taylor series expansion, one can obtain an approximation of $e^A$ or $e^3$ of sufficient accuracy. Luckily, we do not have to do these hand calculations, as modern computer tools enable us to type things like `exp(A)` or `exp(3)` to get numeric approximations that are valid down to machine precision. It can also be mentioned that there exist more efficient related methods to compute exponential matrices than straight-off truncation of (4.57).

Anyway, we are now ready to verify that (4.56) is indeed a solution to (1.9). To do this, we see that in order for (4.56) to be a solution, $\boldsymbol{y} = C\boldsymbol{x} + D\boldsymbol{y}$ requires

$$\boldsymbol{x}(t) = e^{At}\boldsymbol{x}(0) + \int_0^t e^{A(t-\tau)}B\boldsymbol{u}(\tau)\, d\tau \tag{4.59}$$

to hold. We can differentiate the second right-hand-side term of (4.59) with respect to $t$ using the Leibniz integral rule, taught in basic calculus courses. It states that

$$\frac{\partial}{\partial t}\left(\int_0^t f(t,\tau)d\tau\right) = f(t,t) + \int_0^t \frac{\partial}{\partial t} f(t,\tau)\, d\tau. \tag{4.60}$$

Substituting

$$f(t,\tau) = e^{A(t-\tau)}B\boldsymbol{u}(\tau) \tag{4.61}$$

into (4.60) we obtain

$$\frac{\partial}{\partial t}\int_0^t e^{A(t-\tau)}B\boldsymbol{u}(\tau)\, d\tau = B\boldsymbol{u}(t) + \int_0^t Ae^{A(t-\tau)}B\boldsymbol{u}(\tau)\, d\tau. \tag{4.62}$$

Adding the derivative of the first right-hand-side term of (4.59) with respect to $t$ to the right-hand-side of (4.62) we obtain

$$\boldsymbol{x}(t) = Ae^{At}\boldsymbol{x}_0 + B\boldsymbol{u}(t) + \int_0^t Ae^{A(t-\tau)}B\boldsymbol{u}(\tau)\, d\tau, \tag{4.63}$$

which we can finally re-write as $A\boldsymbol{x}(t) + B\boldsymbol{u}(t)$ using (4.59). It is thus clear that (4.59) indeed constitutes the solution to (1.9).

We close this chapter with an example of high practical importance, that shows the usefulness of the solution formula (4.59)

**Example 4.11: Zero-order-hold sampling.** Let us consider a smart insulin pump. Once per second it reads off a blood glucose sensor, decides on new insulin infusion rate, and ensures that the pump delivers this rate until the next sample (sensor reading) arrives. This is called zero-order-hold, since the control signal is held constant (a function that can be described by a Taylor series of order zero) between samples.

If we assume a simple LTI model that relates infusion rate $u$ to blood glucose concentration $y$, we have that

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + Bu, \tag{4.64a}$$
$$y = C\boldsymbol{x} + Du. \tag{4.64b}$$

Let us say that the system starts at $t = 0$ in the state $\boldsymbol{x} = \boldsymbol{x}_0$, and we have a sampling period of $h$. We can then use the solution formula (4.56) to compute the system state after one sampling period

$$\boldsymbol{x}(h) = e^{Ah}\boldsymbol{x}_0 + \int_0^h e^{A(t-\tau)}\, d\tau B u(0). \tag{4.65}$$

Note that since $u$ is constant between $t = 0$ and $t = h$, we can write $Bu(0)$ outside the integral. If we introduce the *constant* matrices

$$\Phi = e^{Ah}, \tag{4.66a}$$
$$\Gamma = \int_0^h e^{A\tau}\, d\tau B, \tag{4.66b}$$

we can write $\boldsymbol{x}(h) = \Phi\boldsymbol{x}(0) + \Gamma u(0)$, and more generally for all sampling times

$$\boldsymbol{x}(kh + h) = \Phi\boldsymbol{x}(kh) + \Gamma u(kh). \tag{4.67}$$

To evaluate the integral in (4.66b) we can use a little trick originating in the chain rule. Letting

$$M = \begin{bmatrix} \Phi & \Gamma \\ 0 & I \end{bmatrix}, \tag{4.68a}$$
$$E = \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \tag{4.68b}$$

we see from (4.66) that $dM/dh = ME$, which means we can obtain $M$ by evaluating the matrix exponential $M = \exp(Eh)$.

Figure 4.13 shows the measurement $y$ resulting from applying a zero-order-hold input $u$ to continuous dynamics of the form (4.64).

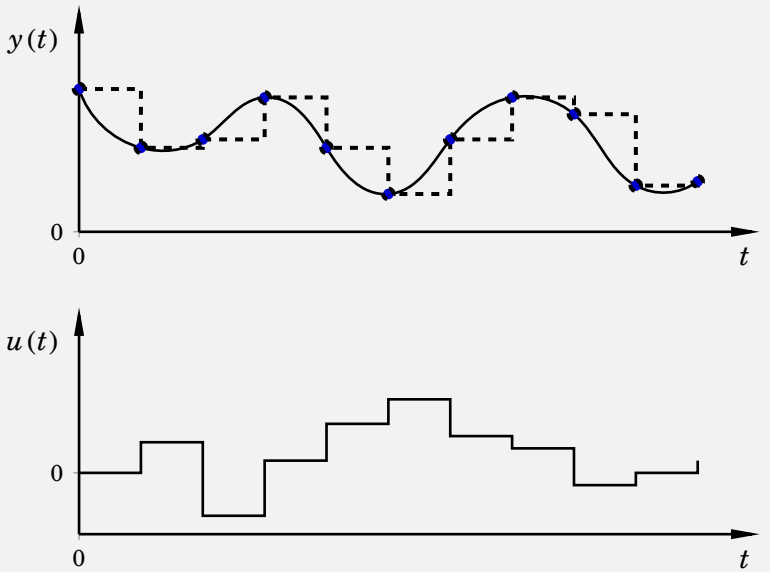


**Figure 4.13** Input $u$ and corresponding output $y$ from an LTI system on the form (4.64) in solid. The dashed line shows the corresponding output of the zero-order-hold sampled system (4.67).

The output from the corresponding ZOH system (4.67) is shown in dashed. In particular, note that there is no approximation error in the ZOH output at the sampling instances. This means that as long as we do not care about the solution between sampling points, we can compute an exact result without the need of using numeric solver schemes as those introduced in Chapter 2. Note that this is possible only because the dynamics we are considering are LTI.

***Further reading.*** The books referenced from the corresponding section of Chapter 3 constitute excellent further reading also for the theory content of this chapter. For those who are interested in

biomechanics—both constitutive models and more generally—the book [Nordin and Frankel, 2001] can be recommended.

Nordin, M. and B. H. Frankel (2001). *Basic biomechanics of the musculoskeletal system*. 3rd ed. Lippincott Williams & Wilkins, Philadelpha,PA. ISBN: 0-683-30247-7.

# 5

# Circuit models

**Learning goals**
After reading this chapter you should (be able to)

- Provide a physiological interpretation of the Windkessel model elements.
- Interpret a circuit diagram as a muscle model, a cardiac afterload model, or a respiratory model.
- Transition between circuit diagram and state space representation of a dynamical system.
- Know what is meant by over-fitting, and generalization.
- Be able to perform a correct dimension (or unit) analysis.

## 5.1 Vascular dynamics

***Historic Note.*** Hemodynamics is the study of the dynamics of the circulatory system, with hema being the latinized form of the Greek word for blood. Some of the mathematical foundation of hemodynamics was laid by German physiologist Otto-Frank (1865–1944), and his English colleague Ernest Henry Starling (1866–1927). Together they are arguably most famous for the Frank-Starling mechanism, also referred to as "the law of the heart". Somewhat simplified, this "law" states that the inotropy (contracting strength) of the heart is regulated through the amount of venous filling of the atria. In other words: the heart adjusts its pumping capacity to accommodate the amount of venous blood it receives.

Otto Frank is also famous for his work on the family of Windkessel vascular models. When the heart pumps blood through the systemic and

pulmonary arterial trees, their flow resistances are not constant, but instead vary dynamically. There is of course a static resistive component, but also some compliance due to flex in the arterial walls, and inertance, due to the mass of the blood that is accelerated through the system in each cardiac cycle. The family of Windkessel models formalizes these properties, as we will see in this chapter.

The name "Windkessel" comes from the German word for a pressure chamber, used for example to maintain water flow in old, manually pumped, fire fighting equipment, as illustrated in Figure 5.1.

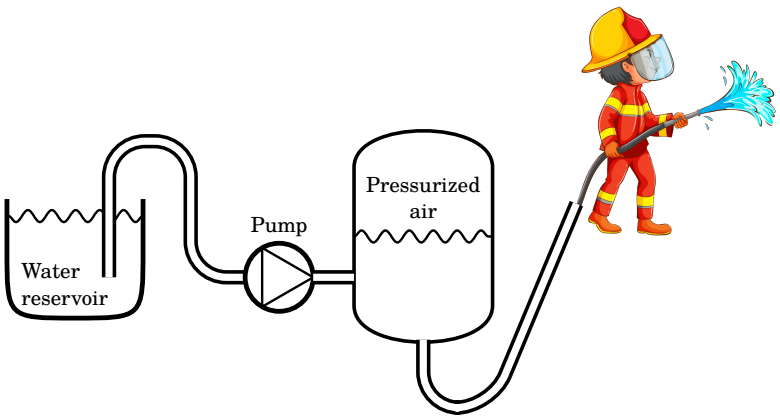


**Figure 5.1** The Windkessel is a pressurized air vessel, used to maintain a steady flow of water. The compressible air provides compliance to the system. It is this property that explains why the arterial dynamics model bears the same name.

Let us take a look at a simple Windkessel model, consisting of a compliance and a resistance. Since it has two constitutive components, it is commonly referred to as the two-element Windkessel model.

In general, the Windkessel models are built by combining constitutive components that model compliance, resistance and intertance. These are properties of the vascular system that relate blood pressure to flow. As we will look closer at in Section 5.2, there is analogy between these constitutive components and those that we have used to describe tissue dynamics in Section 4.1, and similar models from other domains: they can all be expressed as electric circuits.

In the Windkessel case, the analogy is (with SI units)

- volumetric flow↔electric current (A),
- hydrostatic pressure↔electric potential (V).

Within this analogy, the component analogies are

- volumetric flow resistance↔electric resistance (Ω),
- flow compliance↔electric capacitance (C),
- flow inertance↔electric inductance (H).

Letting $\varphi$ denote the flow through the component, and $p$ the pressure gradient across it, we have

$$p = \varphi R, \tag{5.1a}$$
$$\varphi = C\dot{p}, \tag{5.1b}$$
$$p = L\dot{\varphi}, \tag{5.1c}$$

for the three respective components, where $R$, $C$, and $L$ are positive parameters resistance, compliance, and inertance.

You are likely already familiar with Ohm's law (5.1a), the definition of capacitance $C$ (5.1b), and inductance $L$ (5.1c). To build some intuition around (5.1) we can note that flow is a per-time change in mass (or volume). The electric analogy of mass (or volume) is charge, commonly denoted $Q$. The capacitor equation (5.1b) can thus be written $\varphi = C\dot{Q}$. Integrating this both sides with respect to time thus results in $Q = Cp$. In terms of the electric analogy this states that the charge stored in the capacitor equals the product of the voltage across it and its capacitance.

Using Kirchhoff's current law—the net current into (or out from) a connection point is zero—we can obtain a system of linear differential equations describing the combined dynamics, as will be illustrated in Example 5.1 and Example 5.2.

In the literature, the analogies are used sloppily, and it is not uncommon that, for example, a flow compliance is referred to as a capacitance. In practice, this seldom poses any problem, as there is a one-to-one correspondence.

The particular task of the Windkessel model is to describe the dynamics between pressure and flow in the aorta. The heart is modelled as a time-varying flow source, and this flow passes through a circuit of constitutive components.

**Example 5.1: The two-element Windkessel model.** A concrete example is shown in Figure 5.2, illustrating a two-element Windkessel model.

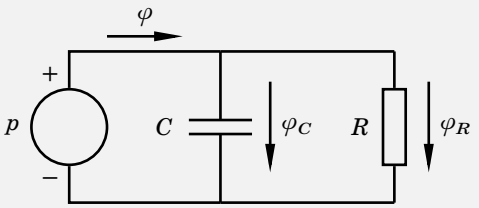


**Figure 5.2** Two-element Windkessel model comprising of resistance $R$ and compliance $C$. The heart is modeled as a flow source (circle). Aortic pressure and flow are denoted $p$ and $\varphi$, respectively. We denote the flow through the resistor and capacitor $\varphi_R$ and $\varphi_C$, respectively.

The physiological interpretation of the model is that the resistor models the total resistance of the systemic arteries, while the capacitor models their compliance (elastic vessel walls).

The pressure gradient $p$ across the resistor equals the pressure gradient across that capacitor. Each of Kirchhoff's current law, the definition of the resistor (5.1a), and the definition of the capacitor (5.1b) gives us (in that order) the three equations

$$\varphi = \varphi_R + \varphi_C, \tag{5.2a}$$
$$p = R\varphi_R, \tag{5.2b}$$
$$\varphi_C = C\dot{p}. \tag{5.2c}$$

We can eliminate the flows through the individual components to obtain

$$\varphi = \frac{1}{R}p + C\dot{p}, \tag{5.3}$$

being an LTI ODE relating aortic pressure $p$ to aortic flow $\varphi$.

Using recordings of pressure and flow, as those in Figure 5.3, one can search for combinations of numeric values for $R$ and $C$ that (assuming the initial condition is known) result in as good a data fit as possible. In addition to recorded human data, Figure 5.3 shows simulations of two Windkessel models. The blue curve corresponds to a two-element model with parameters chosen to minimize the root-mean-square (RMS) error. As can be seen there remains quite a bit of model mismatch. This is because the model is overly simplistic. One way to fix this is to increase the model complexity by adding more constitutive elements. This increases the parameter count,

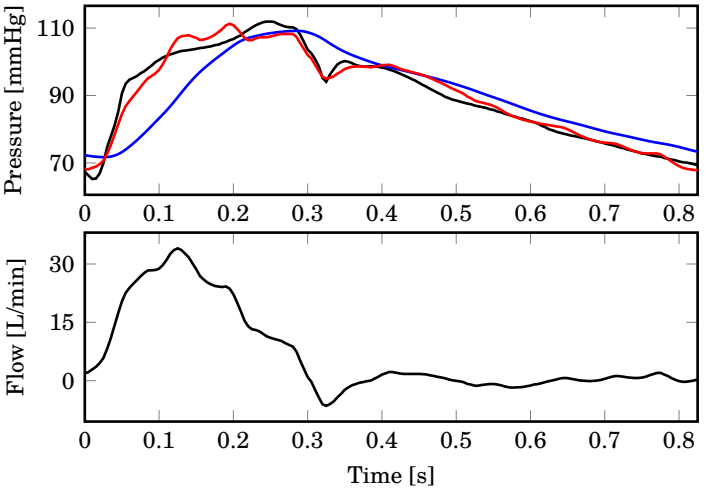


**Figure 5.3** Representative human aortic pressure (top, black) and flow (bottom) waveforms. The blue line shows the pressure waveform arising from a 2-element Windkessel model driven with the flow waveform in the bottom plot; the red line corresponds to a 4-element Windkessel model. The parameters of both models were chosen to minimize the root-mean-square (RMS) error between simulated and actual (black) pressure.

making it possible to replicate a broader range of dynamics. However, it also leads to an increase risk of over-fitting. An over-fitted model replicates the behavior of the modeled system well for the particular data that it was obtained for (also called trained on). However, it does not generalize well. This means that when the same model is presented with a slightly differently shaped input signal, its output will differ more from the output of the true system, subjected to the same input.

## 5.2 Circuit equivalents

In Example 5.2 we will consider a 4-element Windkessel model. Particularly, we will see how we can transtion from a circuit diagram description to state space description.

**Example 5.2: The four-element Windkessel model.** The four-element Windkessel model consists of a resistor $R_c$ modeling the central (aortic) flow resistance, another resistor $R_p$ modeling the periph-

eral (arterial) flow resistance, a capacitor $C$ modeling the combined compliance of the system, and an inductor $L$ modeling the inertance of the blood within the system. A circuit diagram of such model is shown in Figure 5.4.

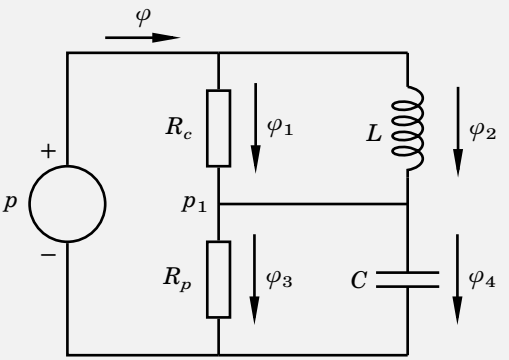


**Figure 5.4** Four-element Windkessel model comprising of central resistance $R_c$, peripheral resistance $R_p$, compliance $C$, and inertance $L$. The heart is modeled as a flow source (circle). Aortic pressure and flow are denoted $p$ and $\varphi$, respectively. We have introduced named variables for all flows and pressures within the model.

As in Example 5.1 we introduce help variables, that we then eliminate to arrive at an LTI ODE that relates $p$ and $\varphi$. A good way to do this is to name pressures at connection points. In our case, let the pressure across $R_p$ be $p_1$. We can also explicitly denote the flows through $R_c$, $L$, $R_p$ and $C$ as $\varphi_1$, $\varphi_2$, $\varphi_3$, and $\varphi_4$, respectively. Using Kirchhoff's current law and the definition of the constitutive elements as in Example 5.1 we thus arrive at

$$\varphi = \varphi_1 + \varphi_2, \tag{5.4a}$$
$$\varphi_1 + \varphi_2 = \varphi_3 + \varphi_4, \tag{5.4b}$$
$$p - p_1 = R_c \varphi_1, \tag{5.4c}$$
$$p_1 = R_p \varphi_3, \tag{5.4d}$$
$$p - p_1 = L \dot{\varphi}_2, \tag{5.4e}$$
$$\varphi_4 = C \dot{p}_1. \tag{5.4f}$$

We see that the time derivatives of $p_1$ and $\varphi_2$ occur, and therefor introduce the following two states

$$x_1 = p_1, \tag{5.5a}$$
$$x_2 = \varphi_2. \tag{5.5b}$$

Note that we can consider either of $\varphi$ and $p$ as outputs of the model. In one case, we think of the heart as a pressure source, and model the arising flow, in the other case we think of the heart as a flow source and model the arising pressure.

Let us here consider the output to be $p$. Using (5.4c), then the definition of $x_1$, and finally (5.4a), we arrive at

$$\begin{aligned} p &= p_1 + R_c\varphi_1 = x_1 + R_c\varphi_1 = x_1 + R_c(\varphi - \varphi_2) \\ &= x_1 - R_c x_2 + R_c\varphi = \underbrace{\begin{bmatrix} 1 & -R_c \end{bmatrix}}_{C}\boldsymbol{x} + \underbrace{R_c}_{D}u. \end{aligned} \tag{5.6}$$

Note that the matrix $C$ is not the same as the capacitance $C$. We could have used different symbols here, but the risk of confusing the two should be small in this case.

Next, we write down the dynamics for $x_1 = p_1$ using (5.4f), then (5.4b) and (5.4a), and finally (5.4c):

$$\dot{x}_1 = \frac{1}{C}\varphi_4 = \frac{1}{C}(\varphi - \varphi_3) = \frac{1}{C}\varphi - \frac{1}{CR_p}x_1. \tag{5.7}$$

Using (5.4e) together with (5.6) we have that

$$\dot{x}_2 = \frac{1}{L}(p - p_1) = \frac{1}{L}(x_1 - R_c x_2 + R_c\varphi - x_1) = -\frac{R_c}{L}x_2 + \frac{R_c}{L}\varphi. \tag{5.8}$$

Combining (5.7) and (5.8) we thus have

$$\dot{\boldsymbol{x}} = \underbrace{\begin{bmatrix} -\frac{1}{CR_p} & 0 \\ 0 & -\frac{R_c}{L} \end{bmatrix}}_{A}\boldsymbol{x} + \underbrace{\begin{bmatrix} \frac{1}{C} \\ \frac{R_c}{L} \end{bmatrix}}_{B}u \tag{5.9}$$

We have thus computed the $A$, $B$, $C$, and $D$ matrices of a state space realization describing the model of Figure 5.4.

**Table 5.1** Examples of common electric circuit equivalents. The thermal equivalents of inductance is seldom observed in physiological or physical systems, and therefore omitted here.

| | Electric | Mechanic | Hydraulic | Thermal |
|---|---|---|---|---|
| potential | voltage | force | pressure | temperature |
| flux | current | stress | volumetric flow rate | heat flux |
| $R$ | resistor | damper | flow resistance | thermal resistance |
| $C$ | capacitor | spring | compliance | thermal mass |
| $L$ | inductor | mass | inertance | - |
| Q | charge | mass | volume | heat |

The methodology introduced in Example 5.2 can be applied to obtain state space representations of any constitutive model consisting of resistive, capacitive and inductive elements. This representation is not unique, as there are several ways to introduce the state variables. However, letting each capacitor and inductor be associated with a state as in Example 5.2 will always work.

In Section 4.1 we described mechanical material models using circuit diagrams, and just above we used circuit diagrams to describe a hydrodynamic system. In fact, electrical circuits can be used to describe any system that has (LTI) dynamics governed by a potential and a flux. In the electric case, the potential is electrostatic voltage and the flux is electric current. In the Windkessel case, the potential is hydrostatic pressure, and the flow is volumetric flow. There is thus an analogy that enables us to analyze a variety of seemingly different systems within the electric circuit framework. Some examples of this analogy are shown in Table 5.1.

To give another concrete example of this analogy, let us consider the so-called RIC respiratory model.

**Example 5.3: RIC respiratory model.** Much like the Windkessel model relates aortic pressure and flow, one can use circuit models to describe the relation between tracheal pressure and flow. One such model is the RIC model, where the letters stand for Resis-

tance, Inductance, and Capacitance. The lungs are primarily modeled as a compliance (due to their flexible nature). The resistor models the combined flow resistance of the airways, while the inductance models the inertance of the gas residing within the airways.

A schematic illustration of the RIC model is shown in Figure 5.5.

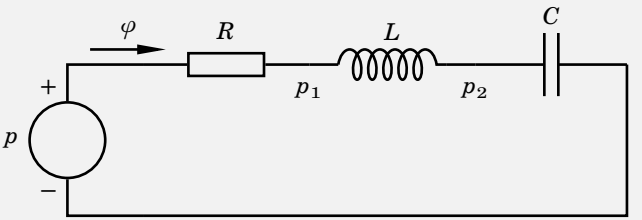


**Figure 5.5** RIC pulmonary model comprising of resistance $R$, inertance $L$, and compliance $C$. The circle models a pressure source, for example a mechanical ventilator to support gas exchange during intensive care. The pressure applied at the airway is $p$, and the associated flow $\varphi$.

Following the methodology of (5.4) we can introduced named pressures $p_1$ between $R$ and $L$ and $p_2$ between $L$ and $C$. This, together with the definition (5.1) give us

$$p - p_1 = R\varphi, \tag{5.10a}$$
$$p_1 - p_2 = L\dot{\varphi}, \tag{5.10b}$$
$$\varphi = C\dot{p}_2. \tag{5.10c}$$

Again, introducing states representing differentiated variables

$$x_1 = \varphi, \tag{5.11a}$$
$$x_2 = p_2, \tag{5.11b}$$

and eliminating $p_1$, we can write the dynamics in the standard state space form

$$\dot{x}_1 = -\frac{R}{L}x_1 - \frac{1}{L}x_2 + \frac{1}{L}p, \tag{5.12a}$$
$$\dot{x}_2 = \frac{1}{C}x_1. \tag{5.12b}$$

By performing experiments and identifying parameters of the RIC model that fit the response, it is possible to gauge for example the

flow resistance and compliance parameters, that are both known to change in individuals with pulmonary diseases such as chronic obstructive pulmonary disease (COPD).

## 5.3 Dimension analysis

When dealing with systems that have signals and states spanning across different domains, as exemplified by the entries of Table 5.1, it is easy to get confused if some signals are reported in one set of units, while others are reported in another. It is therefore important to perform thorough dimension analyses when working with dynamical systems. This is particularly true for medical systems, where a unit-miss can even be lethal.

The key to dimension analysis is to realize that units have the same algebraic properties as numbers. This means that when two entities are multiplied, both their quantities and units are multiplied. For example

$$3\ \mathrm{m} \cdot 4\ \mathrm{kg\,s^{-1}} = 12\ \mathrm{kg\,m\,s^{-1}}. \tag{5.13}$$

Division works similarly, for example

$$3\ \mathrm{m}/4\ \mathrm{kg\,s^{-1}} = \frac{3}{4}\ \mathrm{m\,s\,kg^{-1}}. \tag{5.14}$$

Dimension-wise the addition of two entities with different units does not make sense. For example, if an expression like $2\ \mathrm{m} + 7\ \mathrm{kg}$ shows up in your computations, that is a sign of an error having occurred somewhere upstream.

Some units are just scaled versions of each other. In the standardized international systems of units, SI, prefixes are used to indicate scaling. For example the prefix k as in kg is read “kilo” and amounts to a multiplication by a factor $10^3$. Some of the more common SI prefixes are enlisted in Table 5.2.

One can view prefixes as a form of unit conversion. For example the unit m equals $10^{-3}$ times the unit km:

$$1\ \mathrm{m} = \underbrace{10^{-3}\mathrm{k}}_{=1}\mathrm{m}. \tag{5.15}$$

There also exists another type of unit conversion, namely that between unit systems. For example 1 inch = 2.54 cm, where inch is the base unit for length in the imperial unit system.

Finally, we have composite units, that can also be thought of as a type of conversion. For example, the SI unit for force is the Newton, N, and $1\ \mathrm{N} = 1\ \mathrm{kg} \cdot \mathrm{m/s^2}$.

**Table 5.2** Common SI unit prefixes.

| Quantity | Symbol | name |
|---|---|---|
| femto | f | $10^{-15}$ |
| pico | p | $10^{-12}$ |
| nano | n | $10^{-9}$ |
| micro | $\mu$ | $10^{-6}$ |
| milli | m | $10^{-3}$ |
| centi | c | $10^{-2}$ |
| deci | d | $10^{-1}$ |
| deca | da | $10^{1}$ |
| hecto | h | $10^{2}$ |
| kilo | k | $10^{3}$ |
| mega | M | $10^{6}$ |
| tera | T | $10^{12}$ |
| giga | G | $10^{9}$ |
| peta | P | $10^{15}$ |

Dimension analysis is not hard, but it is important, and it is easily forgotten. At the same time it is an excellent way to debug computations, as unit-mismatches are a clear sign that something is not right. We leave the subject with a realistic example.

**Example 5.4: Drug infusion units.** The anesthetic drug propofol is given intravenously. It is common to have a drug solution of 10 $\mathrm{mg\,mL^{-1}}$. The delivery rate on the infusion pump is set in $\mathrm{mL\,h^{-1}}$, and the patient should get 50 $\mathrm{\mu g\,min^{-1}}$ per kilogram of body weight. What infusion rate should be set for a 75 kg patient?

The patient should receive

$$50\ \mathrm{\mu g\,min^{-1}\,kg^{-1}} \cdot 75\ \mathrm{kg} = 3.75 \cdot 10^{3}\ \mathrm{\mu g\,min^{-1}}. \tag{5.16}$$

Note that we have written the intermediate step in scientific notation: a number between 0 and 10, times an integer power of 10. It is handy to stick to this format, particularly when dealing with very large or very small numbers.

Breaking the computation down in small steps, we can continue with

$$3.75 \cdot 10^3\ \mu g\,min^{-1} \cdot 10^{-3}\ mg\,\mu g^{-1} = 3.75\ mg\,min^{-1}, \quad (5.17a)$$

$$3.75\ mg\,min^{-1} \cdot 6 \cdot 10^1\ min\,h^{-1} = 2.25 \cdot 10^2\ mg\,h^{-1}, \quad (5.17b)$$

$$2.25 \cdot 10^2\ mg\,h^{-1} \cdot 1.0 \cdot 10^{-1}\ mL\,mg^{-1} =, 2.25 \cdot 10^1\ mL\,h^{-1}. \quad (5.17c)$$

Taking good care, we thus arrive at the correct answer of 22.5 $mL\,h^{-1}$. Imagine doing this computation without explicitly keeping track of the units. It would be possible, but very fragile.

***Further reading.*** A comprehensive work on Windkessel dynamics is [Westerhof et al., 2019]. To get more confident with circuit analysis, it can be worthwhile to take on a course book in basic circuit theory, such as [Hayt et al., 2011]. If you want to delve deeper into unit analysis, there even exist entire books on the topic, one example being [Lemons, 2017].

Westerhof, N., N. Stergiopulos, M. I. Noble, and B. E. Westerhof (2019). *Snapshots of Hemodynamics*. 3rd ed. Springer International Publishing, Cham, Switzerland. ISBN: 978-3-319-91931-7.

Hayt, W., J. Kemmerly, and S. Durbin (2011). *Engineering circuit analysis*. 8th ed. McGraw-Hill Education, New York. ISBN: 978-0073529578.

Lemons, D. S. (2017). *A Student's Guide to Dimensional Analysis*. Cambridge University Press, Cambridge, England. ISBN: 978-1316613818.

# 6

# Laplace domain representation

**Learning goals**
After reading this chapter you should (be able to)

- Understand what is meant by a transform.
- Be able to show basic properties of the Laplace transform, using its definition.
- Describe what a transfer function is.
- Move between time domain differential equation and Laplace domain transfer function representations of linear dynamical systems.
- Be able to represent dynamical systems, and their interconnections, using block diagrams.
- Combine systems in series, parallel and feedback interconnections, and compute the corresponding transfer functions.
- Correctly characterize stability of an LTI system on state space form, or represented by a transfer function.
- Know what is meant by static gain and how to compute it.

***Historic Note.*** If you have used a camera with manual settings, you have probably adjusted the aperture to get an image that is neither under- nor over-exposed. Many cameras have automatic aperture adjustments. This is achieved by measuring the intensity of light hitting the camera, and adjusting accordingly.

This functionality in the camera is an almost verbatim implementation of what your eyes do all the time. When exposed to increased light intensity, the pupil shrinks, and when light intensity decreases, the pupil grows, as schematically illustrated in Figure 6.1.

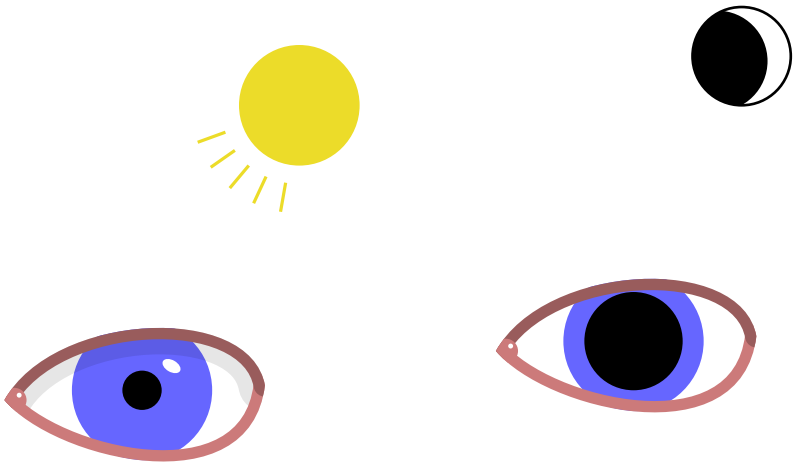

**Figure 6.1** Schematic illustration of the pupillary light reflex: the pupil shrinks when the intensity of incoming light increases, and *vice versa*.

The dynamics of the neuro-muscular feedback loop that make up the pupillary light reflex were studied by Stark and Sherman in the 1950s. By sending in light of sinusoidially varying intensity and recording the pupillary reaction, they were able to determine an LTI model of the dynamics.

Their experiments constitute a classic example of how dynamical system theory can be useful in basic physiology. In this chapter, we will study the Laplace transform, and build a theoretic foundation that we will then use in Chapter 7 to gain an understanding of why varying the light intensity as a sinusoid was a particularly clever choice, and what such an experiment reveals about the dynamics of the pupillary reflex.

## 6.1 The Laplace transform

A transform is a mapping that can take us back and forth between two equivalent representations. We have already seen a transform in for instance Example 3.6, where the linear mapping and its inverse take us back and forth between two state-space representations of the same system.

Let us consider a linear transform that we denote by $\mathcal{L}$, and call it the Laplace transform, after French mathematician Pierre-Simon Laplace (1749–1827). Applied to a function of time—a signal—$f(t)$, it produces a function $F(s)$, where $s$ is a complex variable:

$$f(t) \xrightarrow{\mathcal{L}} \int_0^\infty e^{-st} f(t)\, dt = F(s). \tag{6.1}$$

Note that once we integrate the right-hand-side with respect to $t$, the resulting function $F$ does no longer depends on $t$, but only on $s$.

It is conventional to denote signals in the time domain with small letters ($f$), and their Laplace domain counterparts with capital letters ($F$). We will stick to this convention most of the time, and when there are exceptions, it should be obvious from the context if we are dealing with an object in the time domain or in the Laplace domain.

While not visible at first glance, the transform (6.1) has very much in common with the linear transform of Example 3.6. It will be possible to use the Laplace transform for practical purposes without seeing this. However, your understanding of what the Laplace transform is (other than an opaque abstract formula) could be much aided by this analogy, that we explore further in Example 6.1.

**Example 6.1: Transforms as projections.** Assume you have a vector $\boldsymbol{x}$ and left-multiply by a matrix $A$ to obtain

$$\boldsymbol{z} = A\boldsymbol{x}. \tag{6.2}$$

We can think of a vector $\boldsymbol{x}$ with $n$ elements as a point in $\mathbb{R}^n$. Each elements is then a coordinate in some basis with basis vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n$, where $\boldsymbol{e}_k$ is a column vector of $n$ elements, all zero, except element $k$, which is 1. This means that

$$\boldsymbol{x} = \underbrace{\begin{bmatrix} \boldsymbol{e}_1^\top \\ \vdots \\ \boldsymbol{e}_n^\top \end{bmatrix}}_{I} \boldsymbol{x} \tag{6.3}$$

We can thus view the coordinates in $\boldsymbol{x}$ as the orthogonal projections of $\boldsymbol{x}$ onto the basis vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n$, as illustrated for an example with $n = 2$ in Figure 6.2.

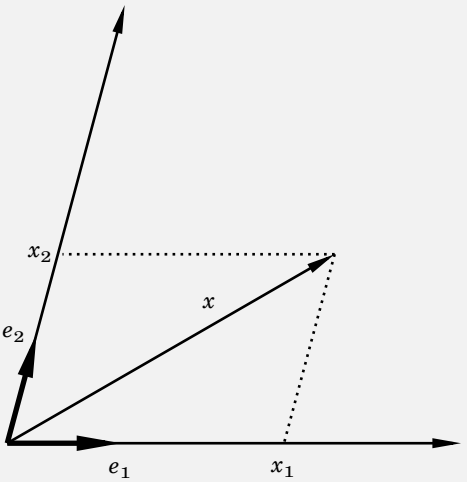


**Figure 6.2** The vector $\boldsymbol{x}$ with $n = 2$ elements, interpreted as projections onto basis vectors in $\mathbb{R}^n$.

We could just as well project onto another basis, defined by vectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ If these vectors are linearly independent, then we can express any $\boldsymbol{x}$ in this new coordinate system with the coordinates defined by $\boldsymbol{z}$, where

$$\boldsymbol{x} = \boldsymbol{a}_1^\top \boldsymbol{z} + \ldots + \boldsymbol{a}_1^\top \boldsymbol{z}. \tag{6.4}$$

We can equivalently write this as

$$\boldsymbol{x} = \underbrace{\begin{bmatrix} \boldsymbol{a}_1^\top \\ \vdots \\ \boldsymbol{a}_n^\top \end{bmatrix}}_{A} \boldsymbol{z} \tag{6.5}$$

The new basis vectors being linearly independent is equivalent to the $n \times n$ matrix $A$ having full rank, which in terms is equivalent to $\det(A) \neq 0$. Since $\det(A) \neq 0$ determines whether $A$ is invertible, we have that $A^{-1}$ exists only if the new basis vectors are linearly

independent. In this case we have that

$$\boldsymbol{z} = A^{-1}\boldsymbol{x}. \tag{6.6}$$

We can thus view $A$ as defining a transform between a coordinate system with axes in the $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n$ directions, to another coordinate system with axes in the $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ directions.

We can repeat the same reasoning where the basis vectors are scalar functions, instead of scalar numbers, moving the scene from the $n$ dimensional space $\mathbb{R}^n$, to an infinite-dimensional *function* space. In $\mathbb{R}^n$ we use the inner product (also known as the scalar product) for projections, and relatedly, we use it to test if two vectors are parallel. The inner product between vector $\boldsymbol{e}$ and $\boldsymbol{x}$ is defined by

$$< \boldsymbol{e}, \boldsymbol{x} >= \boldsymbol{e}^\top \boldsymbol{x} = \sum_{k=1}^{n} e_k x_k. \tag{6.7}$$

You can think of a function $f(t)$ as an infinitely long vector, where each $t$ corresponds to one entry. Thinking of the integral as an infinite sum of vertical slices, the following *definition* of the inner product between the functions $e(t)$ and $x(t)$, defined for $t > 0$, becomes natural:

$$< e, x >= \int_0^\infty e(t)x(t)\, dt. \tag{6.8}$$

Just like we can define a transform as the projection of a real vector $\boldsymbol{x}$ of dimension $n$ onto $n$ basis vectors, we can—using the definition of the inner product—project a real-valued function $f(t)$ onto infinitely many basis functions.

This is exactly what was done in (6.1), with the basis functions being

$$a(t) = e^{st}, \tag{6.9}$$

where $s$ is a complex number, and each possible value of $s$ corresponds to one of infinitely many basis functions.

If the above example serves to build your intuition, that is great. If not, do not worry about it too much. You will still be able to use the Laplace transform, but without fully understanding how it works.

A key property that makes the Laplace transform of particular interest

to us is what happens if we instead of $f(t)$ transform its time-derivative:

$$\frac{d}{dt}f(t) \xrightarrow{\mathcal{L}} \int_0^\infty e^{-st}\frac{d}{dt}f(t)\,dt = \left[e^{-st}f(t)\right]_{t=0}^{\infty} - \int_0^\infty (-s)e^{-st}f(t)\,dt$$
$$= sF(s) - f(0), \tag{6.10}$$

where we have used the Leibniz integration rule (4.60) to arrive at the right-hand-side expression.

Let us consider LTI systems that start at rest, such that all signals including the internal state variables are zero at time $t = 0$. While this might seem limiting, it is in practice often not. If $t = 0$ is designated to some instance far back in time, then the influence of the state at $t = 0$ at the current state will be negligible as long as the system has stable dynamics. In Section 6.3 we will provide formal definition of what is meant by stability in this context, but for now it can be assuring to know that all the systems we have considered so far have been stable ones. What this allows us to do is to write

$$\frac{d}{dt}f(t) \xrightarrow{\mathcal{L}} sF(s). \tag{6.11}$$

In words, this means that the Laplace transform of a derivative of a function is $s$ times the Laplace transform of the function itself. So if $F(s)$ is the transform of $f(t)$ then $sF(s)$ is the transform of $df/dt$.

Consequently it also holds that

$$\int_0^t f(\tau)\,d\tau \xrightarrow{\mathcal{L}} \frac{1}{s}F(s). \tag{6.12}$$

In words, (6.12) means that if $F(s)$ is the Laplace transform of $f(t)$, then $F(s)/s$ is the Laplace transform of its time integral $\int_0^t f(\tau)d\tau$.

We can sum up the properties (6.11) and (6.12):

- Differentiation with respect to time in the time domain corresponds to multiplication by $s$ in the Laplace domain.
- Integration with respect to time in the time domain corresponds to division by $s$ in the Laplace domain.

The Laplace transform has several other properties that are of interest to use.

- Since integration is a linear operator, it follows that the Laplace transform is linear: $\alpha f(t) + \beta g(t) \xrightarrow{\mathcal{L}} \alpha F(s) + \beta G(s)$.

- A delay $d$ in the time domain corresponds to multiplication with the complex exponential $\exp(-ds)$ in the Laplace domain: $f(t-d) \xrightarrow{\mathcal{L}} F(s)e^{-ds}$.

We will encounter yet another property of interest in Chapter 7 when studying frequency responses. But first, let us see how the Laplace transform can be of practical use to us in modeling physiological systems and signals.

## 6.2 Transfer functions and block diagrams

We are now ready to introduce a representation of LTI systems in the Laplace domain called transfer functions. Letting $u(t)$ be the input signal to our LTI system and $y(t)$ the resulting output, the transfer function $G(s)$ from $U(s)$ to $Y(s)$ is defined as

$$G(s) = \frac{Y(s)}{U(s)}. \tag{6.13}$$

We can use the transfer function to compute the response $y(t)$ of an LTI system represented by a transfer function $G(s)$ to an input $u(t)$. This is done by computing the Laplace transform $U(s)$, multiplying with $G(s)$ to obtain $Y(s) = G(s)U(s)$ and then applying the *inverse* Laplace transform to $Y(s)$ to obtain $y(t)$. In principle we can already compute $U(s)$ using the definition (6.1). But how do we determine the transfer function $G(s)$ of an LTI system, and how do we apply the inverse transform to obtain $y(t)$ from $Y(s)$? Let us begin with how to determine $G(s)$ for an LTI system. In Example 6.2 we show how it is done if we have a state space representation of the system.

**Example 6.2: State space to transfer function.** Assume we have a SISO (scalar input, scalar output) LTI system in standard state-space form (1.9). Since the Laplace transform is linear, we can transform the state-space representation to obtain

$$s\mathbf{X}(s) = A\mathbf{X}(s) + BU(s), \tag{6.14a}$$
$$Y(s) = C\mathbf{X}(s) + DU(s), \tag{6.14b}$$

where we have used the property (6.11). Since $s\mathbf{X}(s) = sI\mathbf{X}(s)$, where $I$ is the identity matrix, we can re-write (6.14a) as

$$sI\mathbf{X}(s) = A\mathbf{X}(s) + BU(s), \tag{6.15}$$

and solve for

$$\mathbf{X}(s) = (sI - A)^{-1}BU(s). \tag{6.16}$$

Combining (6.16) with (6.14b), we this have that

$$Y(s) = (C(sI - A)^{-1}B + D)U(s). \tag{6.17}$$

But we know from the definition (6.13) of the transfer function $G(s)$, that $Y(s) = G(s)U(s)$, which means that

$$G(s) = C(sI - A)^{-1}B + D. \tag{6.18}$$

We now know how to obtain the transfer function from the state space representation of an LTI system.

The inverse of a matrix is the adjoint matrix divided by the determinant. (Remember that the determinant is a scalar.) This means that we can write

$$G(s) = \frac{C\,\mathrm{adj}(sI - A)B}{\det(sI - A)} + D, \tag{6.19}$$

where both $C\,\mathrm{adj}(sI-A)B$ and $\det(sI-A)$ come out as polynomials in $s$. The transfer function of an LTI system with state-space representation (1.9) is thus a fraction of two polynomials in $s$. In Example 6.2 we have seen how the polynomials can be determined from the state space matrices $A$, $B$, $C$, $D$.

In the transformed domain, we can thus obtain $Y(s)$ through multiplying $U(s)$ with the function $G(s)$. This function is referred to as the transfer function from $U$ to $Y$. Note that in the Laplace domain both signals and LTI systems share the representation of functions in the complex variable $s$. If $U(s)$ is the input to a system $G(s)$, then the output of the system is simply $G(s)U(s)$. This comes in handy when combining several LTI systems into a bigger one, as we will do shortly.

To make things a bit more concrete, we first see how we can compute the transfer function for a model that we have already worked with, the four-element Windkessel model (5.9) in Example 6.3.

**Example 6.3: Transfer function of Windkessel model.** Let us return to the four-element Windkessel model of Example 5.2. Using

the state-space matrices

$$A = \begin{bmatrix} -\frac{1}{R_p C} & 0 \\ 0 & -\frac{R_c}{L} \end{bmatrix}, \tag{6.20a}$$

$$B = \begin{bmatrix} \frac{1}{C} \\ \frac{R_c}{L} \end{bmatrix}, \tag{6.20b}$$

$$C = \begin{bmatrix} 1 & -R_c \end{bmatrix}, \tag{6.20c}$$

$$D = \begin{bmatrix} R_c \end{bmatrix}, \tag{6.20d}$$

derived in the example, we can compute the transfer function from flow to pressure using (6.19).

First we compute

$$sI - A = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} -\frac{1}{R_p C} & 0 \\ 0 & -\frac{R_c}{L} \end{bmatrix} = \begin{bmatrix} s + \frac{1}{R_p C} & 0 \\ 0 & s + \frac{R_c}{L} \end{bmatrix}, \tag{6.21}$$

then

$$\det(sI - A) = \left(s + \frac{1}{R_p C}\right)\left(s + \frac{R_c}{L}\right), \tag{6.22}$$

and

$$\operatorname{adj}(sI - A) = \begin{bmatrix} s + \frac{R_c}{L} & 0 \\ 0 & s + \frac{1}{R_p C} \end{bmatrix}. \tag{6.23}$$

We then use the outcome of these computations to evaluate

$$G(S) = \frac{P(s)}{\Phi(s)} = \frac{1}{\det(sI - A)} C \operatorname{adj}(sI - A) B + D$$

$$= \frac{1}{\left(s + \frac{1}{R_p C}\right)\left(s + \frac{R_c}{L}\right)} \begin{bmatrix} 1 & -R_c \end{bmatrix} \begin{bmatrix} s + \frac{R_c}{L} & 0 \\ 0 & s + \frac{1}{R_p C} \end{bmatrix} \begin{bmatrix} \frac{1}{C} \\ \frac{R_c}{L} \end{bmatrix} + R_c$$

$$= \frac{\frac{1}{C}s + \frac{R_c}{LC} - \frac{R_c^2 s}{L} - \frac{R_c^2}{R_p C L}}{\left(s + \frac{1}{R_p C}\right)\left(s + \frac{R_c}{L}\right)} + R_c. \tag{6.24}$$

We end this example here, but you can further simplify the transfer function expression by for example collecting equal powers of $s$ in the denominator of the first term, or by writing the whole expression on a common denominator, or by performing a partial fraction decomposition.

We could just as well have replaced all occurrences of $d/dt$ in (5.4) with $s$ and computed the transfer function from the original equation system. This is a good exercise, and will likely be faster to perform, than computing the transfer function using the formula $C(sI - A)^{-1}B + D$ as in this example.

As hinted at the end of Example 6.3, we do not necessarily need to set up a state-space representation first, to obtain the transfer function of a system. To see this, let us again consider the Voigt constitutive material model of Example 4.1.

**Example 6.4: Transfer function of the Voigt model.**

We can take the Laplace transform of the constitutive equations of which the model is assembled. Assuming that for each element the displacement is $X(s)$, and that the corresponding force is $F(s)$, the

spring, damper, and mass come out as:

$$F_s(s) = kX(s), \tag{6.25a}$$
$$F_d(s) = \eta s X(s), \tag{6.25b}$$
$$F_m(s) = ms^2 X(s). \tag{6.25c}$$

(Here we have used subscripts to avoid the possible confusion that could arise from using the same symbol $F$ to mean the force on the spring, on the mass, and on the damper.)

Using the same reasoning as in Example 4.1, we can immediately write out

$$ms^2 X(s) = -kX(s) - \eta s X(s) + F(s). \tag{6.26}$$

Assuming $F$ is the input and $X$ the output, we can solve (6.26) for

$$G(s) = \frac{X(s)}{F(s)} = \frac{1}{ms^2 + \eta s + k}. \tag{6.27}$$

So far we have only dealt with transfer functions of SISO (single-input, single-output) systems. Expanding to the MIMO case is possible by organizing several SISO transfer functions in a matrix so that entry $ij$ denotes the transfer function from input $j$ to output $i$. Having mentioned this, we now stick to SISO transfer functions for the remainder of this book.

Graphically we can represent transfer functions as a block, like the one shown in Figure 6.3.

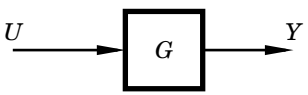


**Figure 6.3** Block diagram schematic of a transfer function $G$ from input $U$ to output $Y$.

If we connect two blocks in series, as in Figure 6.4, we have that the output of the first block is $G_1(s)U(s)$, making the output of the second block $G_2(s)G_1(s)U(s)$.

If, instead, the two blocks are connected in parallel, as depicted in Figure 6.5, the outputs $G_1(s)U(s)$ and $G_2(s)U(s)$ are added to produce $(G_1(s) + G_2(s))U(s)$.

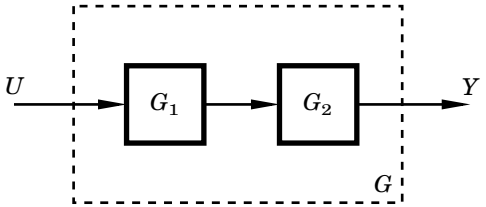


**Figure 6.4** Series interconnection of two LTI systems with transfer functions $G_1$ and $G_2$, resulting in the system $G = G_2G_1$.

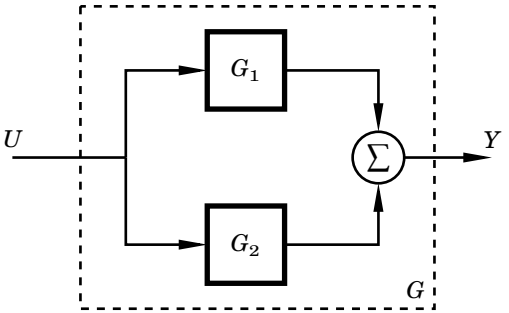


**Figure 6.5** Parallel interconnection of two LTI systems with transfer functions $G_1$ and $G_2$, resulting in the system $G = G_1 + G_2$.

The conclusion is that

- Series connection of (SISO) systems with transfer functions $G_1$ and $G_2$ results in a system with transfer function $G = G_2G_1$.
- Parallel connection of (SISO) systems with transfer functions $G_1$ and $G_2$ result in a system with transfer function $G_1 + G_2$.

Another frequently occurring interconnection is the feedback interconnection, depicted in Figure 6.6. In the feedback interconnection, the input of $G_1$ depends on the output of $G_2$, which in terms depends on the output of $G_1$. To break up this signal loop, we can introduce a named signal $E$ at the output of the summation, so that

$$E(S) = U(s) + G_2(s)Y(s). \tag{6.28}$$

We thus have an equation system comprising of

$$E(s) = U(s) + G_2(s)Y(s), \tag{6.29a}$$
$$Y(s) = G_1(s)E(s). \tag{6.29b}$$

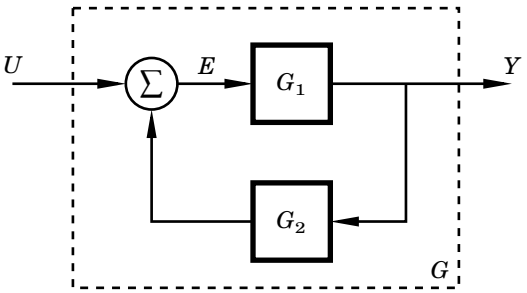


**Figure 6.6** Feedback interconnection between $G_1$ and $G_2$, resulting in the system $G = G_1/(1 - G_2G_1)$.

Eliminating $E$ leaves us with the desired relation

$$Y(s) = \frac{G_1(s)}{1 - G_2(s)G_1(s)}U(s). \tag{6.30}$$

In general, the approach is to name signals at the output of summations, setting up an equation system, eliminating the same signals from the system, and solving for the output signal. We illustrate it below in a somewhat more complex example.

**Example 6.5**
Consider the system in Figure 6.7, where each letter denotes a transfer function. How can we express the transfer function from $U$ to $Y$ using these transfer functions?

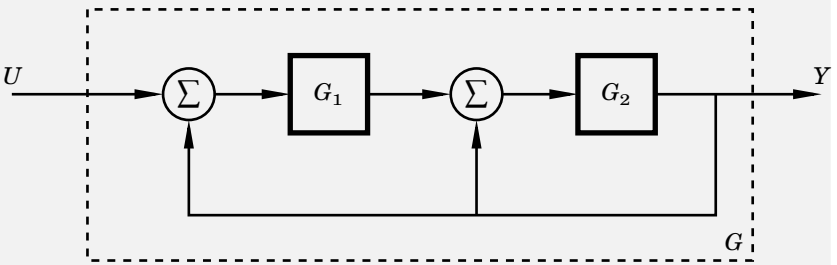


**Figure 6.7** Block diagram with two blocks.

Denoting by $E_1$ the output of the leftmost sum, and by $E_2$ the output

of the rightmost, we have that

$$E_1 = U + Y, \tag{6.31a}$$
$$E_2 = G_1 E_1 + Y. \tag{6.31b}$$
$$Y = G_2 E_2. \tag{6.31c}$$

Inserting the expression for $E_1$ from the first equation into the second, and then the resulting expression for $E_2$ into the third equation gives

$$Y = G_2(G_1(U + Y) + Y). \tag{6.32}$$

Solving for $Y$ we obtain the transfer function

$$G = \frac{G_2 G_1}{1 - G_2 G_1 - G_2} \tag{6.33}$$

from $U$ to $Y$.

We have seen that the transfer function allows us to work with block diagrams and build complex systems from less complex components, which was more difficult when working with state-space representations. We have also seen how to transition from state-space to transfer functions. But about moving back? For systems in the form

$$G(s) = \frac{B(s)}{A(s)}, \tag{6.34}$$

where $B$ and $A$ are polynomials, this is quite simple. If the system input is $U$ and the output is $Y$ we have that

$$A(s)Y(s) = B(s)U(s). \tag{6.35}$$

As a concrete example, if $B(s) = 2$ and $A(s) = 4s^2 + 1$ we have

$$4\ddot{y} + y = 2u. \tag{6.36}$$

Applying the inverse Laplace transform by replacing $s$ with $d/dt$ we thus end up with an ordinary differential equation, and then transition to state space as described in previous chapters.

A related and useful thing, that becomes evident when considering block diagrams, is that we can treat signal paths as separate thanks to linearity of the systems. For example, consider the system in Figure 6.8. Thanks to the superposition principle that follows directly from the linearity of all system components we have that

$$Y(s) = G_2(s)U(s) + G_2(s)G_1(s)V(s). \tag{6.37}$$

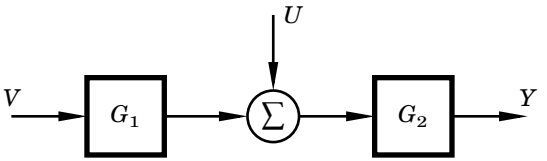


**Figure 6.8** A MISO system with two incoming signal paths $U$ and $V$, and one output $Y$.

If we set $u(t)$ to identically zero, it follows from (6.1) that $U(s)$ also becomes identically zero. In this case $Y(s) = G_2(s)G_1(s)V(s)$. Similarly, if we set $V$ to zero, we have that $Y(s) = G_2(s)U(s)$. We thus say that $G_2(s)G_1(s)$ is the transfer function from $V$ to $Y$, while $G_2$ is the transfer function from $U$ to $Y$. This means that as long as the components $G_1$ and $G_2$ are LTI systems, we can consider these two transfer functions entirely independent of each other.

## 6.3 Stability and static gain

A central concept in dynamical systems, applicable both within and outside of physiology, is stability. Stability tells us what will happen to the system state if we start in an initial state, set all inputs to zero, and wait. For an LTI system one of three things could happen:

- All system states tend to zero. This happening is equivalent to the system being asymptotically stable.
- All system states remain bounded, but at least one does not go to zero, referred to as being marginally stable.
- At least one state goes to (plus or minus) infinity, referred to as being unstable.

If we were dealing with a scalar system ($\boldsymbol{x} = x$ and $A = a$), the exact solution formula (4.56) with $u = 0$ would give us

$$x(t) = e^{at}x_0. \tag{6.38}$$

Any $a < 0$ would thus result in $x \to 0$ as $t \to \infty$. Conversely, $a > 0$ would result in $x$ growing unbounded. Finally, $a = 0$ would result in $x = x_0$ indefinitely. Here the stability classifications are decided by whether $a$ is strictly negative, strictly positive, or zero.

Since we are concerned with the evolution of the state while the input is identifically zero, (4.56) provides us with

$$\boldsymbol{x}(t) = e^{At}\boldsymbol{x}_0, \tag{6.39}$$

meaning that the stability property has to be determined by the $A$ matrix also when the state is vector-valued.

Let us first consider the special case where $A$ can be diagonalized so that $A = D\Lambda D^{-1}$, with $\Lambda$ being a diagonal matrix with real diagonal elements $\lambda_1, \ldots, \lambda_n$. Then we have that

$$A^2 = D\Lambda D^{-1} D\Lambda D^{-1} = D\Lambda^2 D^{-1}. \tag{6.40}$$

Since $\exp(At)$ can be expressed as a Taylor series in $A$, we can thus write

$$e^{At} = De^{\Lambda t} D^{-1}, \tag{6.41}$$

where $\exp(\Lambda t)$ is a diagonal matrix with elements $\exp(\lambda_1 t), \ldots, \exp(\lambda_n t)$. If all the $\lambda$:s are negative, $\exp(\Lambda t)$ will tend to zero, and thus also $\exp(At)$ will tend to zero. For this particular case, stability is determined by the sign of the $\lambda$:s, which are also the eigenvalues of $A$.

What about the general case? Let us start with an example.

**Example 6.6: Voigt model stability.** The Voigt model is shown in Figure 6.9.

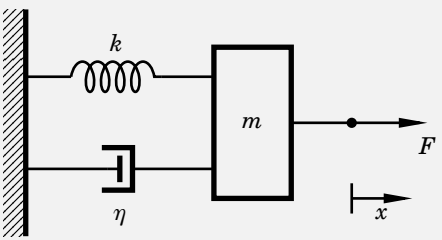


**Figure 6.9** Schematic drawing of the Voigt model, where the spring and damper are connected in parallel to the mass element.

It is governed by LTI dynamics Alternatively we can provide the equations of (4.5) in matrix form

$$\dot{\boldsymbol{x}} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{\eta}{m} \end{bmatrix}}_{A} \boldsymbol{x} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}}_{B} u, \tag{6.42a}$$

$$y = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C}, \tag{6.42b}$$

as previously derived in Example 4.1.

Assuming we start in a state $\boldsymbol{x}_0$, and apply no input ($u = 0$), we can use (4.56) to obtain the exact solution

$$\boldsymbol{x}(t) = e^{At}\boldsymbol{x}_0. \tag{6.43}$$

Fixing $m = 1$ and $k = 1$ we have that

$$A = \begin{bmatrix} 0 & 1 \\ -1 & -\eta \end{bmatrix} \tag{6.44}$$

Simulating the system numerically for different damping $\eta$ we would find that $\eta = 0$ corresponds to a sustained sinusoidal oscillation. This is not surprising as the system is now a mass-spring system without damping. For $\eta > 0$, kinetic energy is removed by the damping, and we have that $\boldsymbol{x} \to 0$ as $t \to \infty$. If, instead, the damping is negative, $\eta < 0$, kinetic energy is pumped into the system by the damper, and $\boldsymbol{x}$ will grow unbounded.

Computing the eigenvalues of $A$ for our special case by solving the characteristic $\det(sI - A) = s^2 + \eta s + 1 = 0$, we find that they are

$$\lambda = -\frac{\eta}{2} \pm \sqrt{\frac{\eta^2}{4} - 1}. \tag{6.45}$$

From our insights into the dynamics, we know that the system is asymptotically stable for $\eta > 0$, marginally stable for $\eta = 0$, and unstable for $\eta < 0$. We can relate this to the real part of the eigenvalues. When the eigenvalues have strictly negative real part, the system is asymptotically stable, and when at least one eigenvalue has positive real part, the system is unstable. Marginal stability arises when the real part of the eigenvalues are zero.

While mathematically somewhat outside the scope of this introductory book, it can be shown that it is indeed the real part of the eigenvalues of the system matrix $A$ that determine stability of the system. If they all have strictly negative real part, the system is asymptotically stable. If at least one has strictly positive real part, the is unstable. Testing for marginal stability in the vector-valued case is a bit trickier, and we will not go into it here, since for most cases of physiologic relevance, it is sufficient to distinguish between asymptotic stability and instability.

According to (6.22), $\det(sI - A)$ is also the denominator of the transfer function from system input to output. This is referred to as the characteristic polynomial. We can therefore equivalently determine stability of a system by equating the transfer function denominator to zero. The solutions, being equal to the eigenvalues of the $A$ matrix, are in this context referred to as the poles of the system.

In an asymptotically stable system, the state goes to zero in the absence of input. But what happens if there is a constant input? To answer this, let us consider a SISO system with $u(t) = 1$ for all positive times. According to (4.56) the output at time $t$ is

$$y(t) = C\boldsymbol{x}(t) + Du(t) = Ce^{At}\boldsymbol{x}_0 + C\int_0^t e^{A(t-\tau)}Bu(\tau)\,d\tau + Du(t). \quad (6.46)$$

After long time the term $C\exp(At)\boldsymbol{x}_0$ will be small as a consequence of the system being asymptotically stable. In particular

$$\lim_{t\to\infty} y(t) = C\underbrace{\lim_{t\to\infty}\int_0^t e^{A(t-\tau)}\,d\tau}_{-A^{-1}}B + D = -CA^{-1}B + D. \quad (6.47)$$

Here we have used that $u(t) = 1$, and that

$$\lim_{t\to\infty}\int e^{A(t-\tau)}\,d\tau = \lim_{t\to\infty}\left[-A^{-1}e^{A(t-\tau)}\right]_{\tau=0}^{t} = -A^{-1} + A^{-1}\lim_{t\to\infty} e^{At}, \quad (6.48)$$

where the left term goes to zero of $A$ is the system matrix of an asymptotically stable system.

When enough time has passed for the initial condition transient term $C\exp(At)\boldsymbol{x}_0$ to have faded away, we can thus obtain the associated stationary output

$$y_\infty = -CA^{-1}B + D. \quad (6.49)$$

Since the system is linear, letting $u(t) = \alpha\cdot 1$ instead of $u(t) = 1$, results in a corresponding scaling of $y$ by a factor $\alpha$. Thus the input $u(t) = \alpha$ results in the stationary output $y_\infty\alpha$. The fraction between output and input of a system is called the gain of the system, and the static gain is therefore $y_\infty$ in this case.

You may already have noticed the resemblance between (6.49) and (6.18). Indeed, the former is what you get from the latter by setting $s = 0$. For an asymptotically stable system we can thus obtain the static gain $y_\infty$ directly from the transfer function as

$$y_\infty = G(0). \quad (6.50)$$

Note, however, that $G(0)$ is only the static gain if the system is asymptotically stable. Otherwise, the system never reaches stationarity, and although $G(0)$ might still be finite, it no longer has the interpretation of being the static gain of the system. Let us illustrate how to compute the static gain using a by now familiar example.

**Example 6.7: Static gain of the Voigt model.** In Example 6.4 we derived the transfer function from force $F$ to elongation $X$ of the Voigt constitutive material model:

$$G(s) = \frac{1}{ms^2 + \eta s + k}. \tag{6.51}$$

Assume we have a tissue that we describe with this model, and that we want to know what the elongation becomes if we keep pulling with a steady force $F_0$.

First we check that the system is stable by solving the characteristic polynomial

$$ms^2 + \eta s + k = 0. \tag{6.52}$$

Assuming that $m$, $\eta$ and $k$ are all strictly positive, the poles are then

$$p = -\frac{\eta}{2} \pm \sqrt{\frac{\eta^2}{4m^2} - \frac{k}{m}}. \tag{6.53}$$

If the expression under the root is negative, the real part of either pole is $-\eta/(2m)$, and thus negative. If the expression under the root is positive, it is always larger than $\eta/(2m)$, and therefore the poles will have negative real parts also in this case. The system is thus asymptotically stable. This should not come as a great surprise: if you stop pulling on the tissue, it will return to its resting state.

We thus know that the stationary elongation corresponding to $F = 1$ is

$$G(0) = \frac{1}{k}, \tag{6.54}$$

end hence that the stationary elongation corresponding to the constant force $F_0$ is $x = F_0/k$. This looks like Hooke's law. Indeed, in stationarity, the damper has no effect since nothing is moving, and there is no acceleration force of the mass, for the very same reason.

We end with an important remark. What does it mean that a state goes unbounded (to infinity or negative infinity)? In fact, this typically does not

occur. Instead, what happens in practice is that a state grows very rapidly until it exits the region in state space where the model constitutes a good approximation of reality. This could be because of some subtle nonlinearity, or it could simply be because the real-world system breaks in one way or another. A physiological analogy here could be a cancer. If the state that we are tracking is the number of cells in a tissue, a cancer could be modeled as an unstable, exponential growth. If not mitigated by the immune system or otherwise treated, our simple model of exponential growth would break down when the combination of size and growth rate would require more metabolic components than can be delivered by the blood.

***Further reading.*** An introduction to mathematical transforms used throughout engineering is provided in [Özhan, 2022]. The book [Janisse, 1974] comprises a collection of scientific texts covering pupil dynamics. Block diagrams are used extensively throughout control system engineering, and to some extent covered in most control system textbooks, including [Åström and Murray, 2020].

Özhan, O. (2022). *Basic transforms for electrical engineering*. Springer, Berlin, Germany. ISBN: 978-3-030-98845-6.

Janisse, M., (Ed.) (1974). *Pupillary dynamics and behavior*. Springer, New York, NY. ISBN: 978-1-4757-1642-9.

Åström, K. J. and R. Murray (2020). *Feedback systems*. 2nd ed. Princeton University press, Princeton, NJ. ISBN: 978-0691135762.

# 7

# Linear model responses

**Learning goals**
After reading this chapter you should (be able to)

- Compute the time domain step and impulse response of a linear system using tabulated Laplace transforms.
- Account for different methods by which the time response of a linear system can be obtained.
- Be able to characterize step responses of first and second-order systems from their step responses in terms of static gain, time constant, and damping.
- Understand that the response to a sinusoidal input signal is the sum of a sinusoid and an initial state transient.
- Know what magnitude (gain) and argument (phase) mean.
- Be able to account for what the Bode diagrams show.

***Historic Note.*** The oral glucose tolerance test (OGTT) is clinically employed to diagnose a variety of conditions, most notably diabetes type 2. The standard test comprises of dissolving 75 mL of glucose into 250-300 mL of water. The patient drinks this in one sweep, in effect making it a bolus dose of glucose delivered to the stomach. The stomach can be modeled as one compartment that communicated (directly or indirectly) with the blood plasma (modeled as another compartment). In order to ensure comparability between test results, it is common that the test dictates a fasting episode of 10 hour or more prior to ingesting the glucose solution. In dynamic modeling terms this can be understood as bringing the subjects to a shared initial state prior to delivering the bolus.

By taking one or several blood samples, typically at the finger tip, and analyzing them for glucose concentration, it is then possible to identify parameters of a simple compartment model that models glucose metabolism. Diabetes 2, and some other conditions, are indicated by the dominating time constant of the identified dynamics differing significantly from that expected in a fully healthy individual.

While the test itself is simple, interpretation of its outcome relies heavily on dynamical modeling concepts. In particular, it relates strongly to the concept of impulse response, which we will study more in-depth in this chapter.

## 7.1 Step and impulse responses

### The step response

In Chapter 6 we have seen that both signals and systems can be modeled in the Laplace domain, as functions of a complex variable $s$. When modeling physiological systems, and a wide range of other dynamical systems as well, two special input signals are of particular interest: the unit step and the unit impulse.

The unit step signal, shown in Figure 7.1 is zero for all negative times, then one for all non-negative times:

$$\theta(t) = \begin{cases} 0, & t < 0, \\ 1, & t \geq 0. \end{cases} \tag{7.1}$$

Applying the Laplace transform (6.1) to the unit step, we get

$$\theta(s) = \int_0^\infty e^{-st}\theta(t)\, dt = \int_0^\infty e^{-st}\, dt = \left[-\frac{1}{s}e^{-st}\right]_{t=0}^\infty = \frac{1}{s}. \tag{7.2}$$

The Laplace transform of the unit step function is thus $s^{-1}$, and if we apply a unit step to the input of an LTI system with transfer function $G$, the

resulting output—the unit step response—becomes

$$Y(s) = G(s)\frac{1}{s}. \tag{7.3}$$

Let us consider the step response of the first-order system

$$G(s) = \frac{k}{sT+1}. \tag{7.4}$$

We can obtain the time-domain expression by inverse-transforming

$$Y(s) = G(s)\frac{1}{s}, \tag{7.5}$$

using a Laplace transform table, as exemplified further below in Example 7.3. Alternatively, we can re-arrange (7.5) with $U(s) = s^{-1}$ as follows

$$Y(s)(sT+1) = kU(s), \tag{7.6a}$$

$$sY(s) = -\frac{1}{T}Y(s) + \frac{k}{T}U(s), \tag{7.6b}$$

and apply the inverse Laplace transform to obtain the time-domain relation

$$\dot{y} = -\frac{1}{T}y + \frac{k}{T}u. \tag{7.7}$$

Introducing the state $x = y$, we can write (7.7) on state-space form (1.9), with

$$\{A,B,C,D\} = \left\{-\frac{1}{T}, \frac{k}{T}, 1, 0\right\}. \tag{7.8}$$

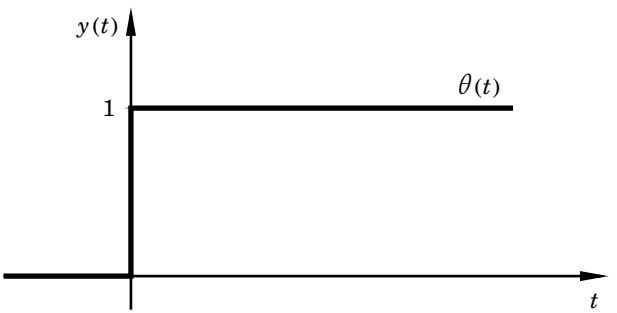


**Figure 7.1** The unit step function $\theta(t)$, sometimes referred to as the Heaviside step function after Oliver Heaviside (1850–1925).

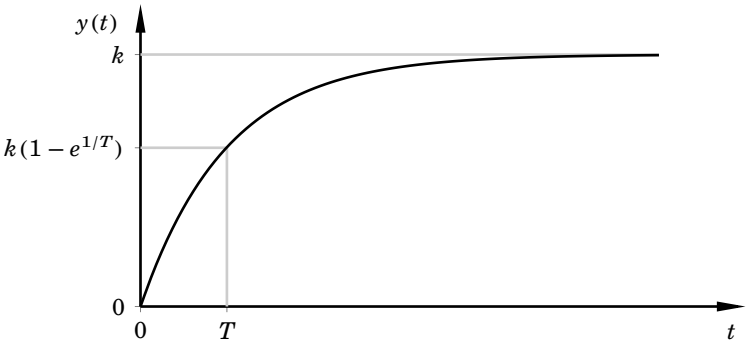


**Figure 7.2** Step response of the first-order system (7.4), with static gain $k$ and time constant $T$.

Inserting the state-space matrices into the explicit solution formula (4.56) we obtain

$$\begin{aligned} y(t) &= e^{-t/T}x(0) + \int_0^t e^{-\frac{1}{T}(t-\tau)}\frac{k}{T}\,d\tau = e^{-t/T}x(0) + k\left[e^{(t-\tau)/T}\right]_{\tau=0}^{t} \\ &= e^{-t/T}x(0) + k(1-e^{-t/T}). \end{aligned} \tag{7.9}$$

The unit step response, often simply referred to as the step response, is defined as thew response to the unit step, with the responding system starting in its rest state $\boldsymbol{x} = 0$. For (7.4), we thus have from (7.9), with $x(0) = 0$, that the step response is

$$y(t) = k(1-e^{-1/T}). \tag{7.10}$$

The step response (7.10) is shown in Figure 7.2.

We see from (7.10) that if $T > 0$, the step response $y(t) \to k$ as $t \to \infty$. But we know from Chapter 6 that $T > 0$ is also the asymptotic stability condition for (7.4), since $A = -1/T$ has the (real) eigenvalue $-1/T$, which is smaller than zero exactly when $T > 0$. In the case $T > 0$ we thus have that

$$G(0) = \frac{k}{0T+1} = k. \tag{7.11}$$

This holds true in general: the unit step response of an asymptotically stable LTI system converges to the static gain of the system.

Since the system (7.4) has one pole (zero of the denominator polynomial), in $s = -1/T$, we refer to it as a first-order system. The number $T$, with unit of time, is referred to the time constant of the first-order system. When $t = T$ we have from (7.10) that

$$y(T) = k(1-e^{-1}) \approx 0.63k. \tag{7.12}$$

The time constant is therefore a measure of how fast the step response converges to the static gain of the system, as we investigate closer in Example 7.1

**Example 7.1: Time constant of first-order system.** Assume we have a system with transfer function

$$G(s) = \frac{b}{s+a}e^{-sL} \tag{7.13}$$

that models the nervous system responding to a stimulus. We issue the stimulus at time $t = 0$, and obtain the response shown in Figure 7.3

In order to obtain a useful model of the system, we want to estimate the parameters $b$, $a$, and $L$ of the model (7.13). In Section 6.1 we found that multiplication with $\exp(-Ls)$ in the Laplace domain corresponds to delaying by time $L$ in the time domain. From (7.13) we thus see that $L = 2$ time units is a good approximation of the delay.

We also see from the figure that the static gain is $k = 3$, and that the time $T$ it takes the response to reach $y(T) = ke(1 - \exp(-T/T)) = k(1 - \exp(-1)) \approx 0.63k$ is roughly $T = 5$. Note that this time excludes the delay. That is, the time constant of a system is defined so that it remains un-changed if the delay of the system is changed.

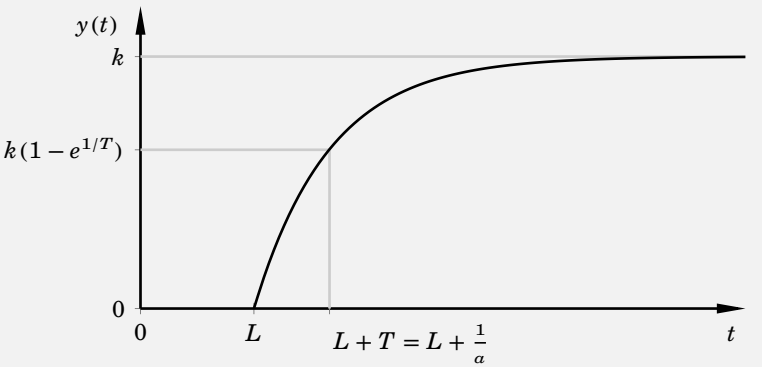


**Figure 7.3** Step response of a system with transfer function (7.13), with delay $L = 2$, time constant $T = 5$, and static gain $k = 3$.

Next, we re-write the un-delayed part of (7.13) as

$$\frac{b}{s+a} = \frac{b/a}{\frac{1}{a}s+1}, \tag{7.14}$$

which reveals to us that $b/a = k = 3$ and $1/a = T = 5$. This is an equation system with two equations and two unknowns, that we easily solve for $a = 1/5$ and $b = 3/5$.

Next, let us study second-order systems. One way of obtaining a second-order system is to simply connect two first-order systems in series, as we do in the following example.

**Example 7.2: Series interconnection of first-order systems.** Many physiological systems respond to a step change in input with a corresponding stop change in output, that is smoothed and scaled. Such systems can often be adequately modeled using the first-order transfer function

$$G(s) = \frac{b}{s+a} = \frac{k}{sT+1}, \tag{7.15}$$

where the static gain is $k = b/a$, while the time constant is $T = 1/a$. It makes sense to talk about static gain and time constant as long as the system is asymptotically stable, which is equivalent to $T > 0$ or $a > 0$.

If we connect the two first-order systems

$$G_1(s) = \frac{b_1}{s+a_1}, \tag{7.16a}$$

$$G_2(s) = \frac{b_2}{s+a_2}, \tag{7.16b}$$

in series, we obtain the system

$$G(s) = G_2(s)G_1(s) = \frac{b_1 b_2}{(s+a_1)(s+a_2)}. \tag{7.17}$$

If we are interested in the step response of $G(s)$, we could use a Laplace transform table and hope to find it there. But we could also

perform a partial-fraction decomposition by asserting

$$G(s) = \frac{b_1 b_2}{(s+a_1)(s+a_2)} = \frac{c_1}{s+a_1} + \frac{c_2}{s+a_2}. \tag{7.18}$$

Putting the right-hand-side on a common denominator, and matching equal powers of $s$ results in the following system of equations:

$$b_1 b_2 = a_2 c_1 + a_1 c_2, \tag{7.19a}$$
$$0 = c_1 + c_2, \tag{7.19b}$$

with solution

$$c_1 = \frac{b_1 b_2}{a_2 - a_1}, \tag{7.20a}$$
$$c_2 = \frac{b_1 b_2}{a_1 - a_2}, \tag{7.20b}$$

and thus

$$G(s) = \underbrace{\frac{b_1 b_2/(a_2 - a_1)}{s+a_1}}_{G_1'} + \underbrace{\frac{b_1 b_2/(a_1 - a_2)}{s+a_2}}_{G_2'}. \tag{7.21}$$

This means that (in this particular case) we can interpret the series interconnection of $G_1$ and $G_2$ as a parallel interconnection of $G_1'$ and $G_2'$. The step response of $G$ is this the sum of the step responses of $G_1'$ and $G_2'$.

There are also second-order systems that cannot be viewed as the series (or parallel) interconnections of two first-order systems. This is the case when the poles are complex. We have already seen example of such system when studying constitutive tissue models, such as the Voigt model in Example 6.4, with transfer function

$$G(s) = \frac{1}{ms^2 + \eta s + k}, \tag{7.22}$$

that we can write in the form

$$G(s) = \frac{b}{s^2 + a_1 s + a_0}, \tag{7.23}$$

with $b = 1/m$, $a_1 = \eta/m$, and $a_0 = k/m$.

We see that if the system is asymptotically stable, its static gain will be

$$G(0) = \frac{b}{a_0} = \frac{1}{k}. \tag{7.24}$$

The condition for asymptotic stability is that the poles of the system, being the zeros of the denominator polynomial, both have negative real part. The poles of (7.23) are the solutions to

$$s^2 + a_1 s + a_0 = 0, \tag{7.25}$$

namely

$$s = -\frac{a_1}{2} \pm \sqrt{\frac{a_1^2}{4} - a_0}. \tag{7.26}$$

The real part of $s$ is strictly negative if and only if $a_1 > 0$, and the square root is real and smaller than $a_1/2$. The latter is the case whenever $a_0 > 0$. The condition for asymptotic stability of (7.23) is thus that both $a_0$ and $a_1$ be strictly positive. In our example this corresponds to $\eta/m$ and $k/m$ both being positive. If we assume that mass is positive, $m > 0$, the condition for asymptotic stability is that both $k > 0$ and $\eta > 0$ hold.

In the following example we investigate the step response of second-order systems, and the Voigt model in particular

**Example 7.3: Step response of second order system.** Using a Laplace transform table we find that

$$Y(s) = \frac{G(s)}{s} = \alpha \frac{\omega_0^2}{(s^2 + 2\zeta\omega_0 s + \omega_0^2)s} \xrightarrow{\mathcal{L}^{-1}} y(t) \tag{7.27}$$

with

$$y(t) = \alpha \left( 1 - \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_0 t} \sin\left( \omega_0 \sqrt{1-\zeta^2}\, t + \phi \right) \right), \tag{7.28a}$$

where

$$\phi = \cos^{-1} \zeta \tag{7.28b}$$

for the case $0 < \zeta < 1$, corresponding to complex poles. The response is illustrated in Figure 7.4.

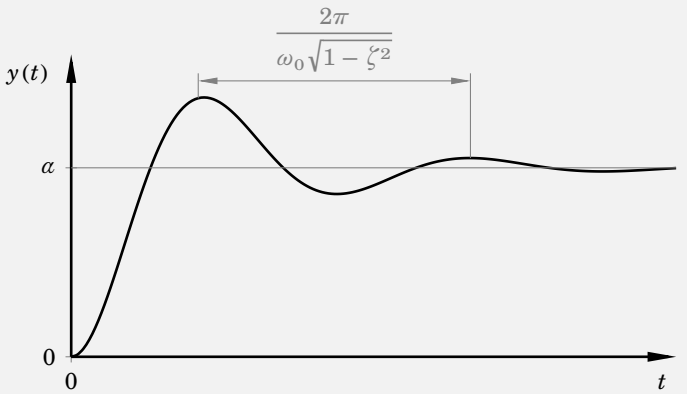


**Figure 7.4** Step response (7.28a) of a second-order system (7.27) with complex poles.

We see from (7.22) that

$$\omega_0 = \sqrt{\frac{k}{m}}, \tag{7.29a}$$

$$\zeta = \frac{\eta}{2\sqrt{km}}, \tag{7.29b}$$

are required for the denominator polynomials to match, and that this imposes

$$\alpha = \frac{1}{\omega_0^2} = \sqrt{\frac{m}{k}}. \tag{7.29c}$$

For a second-order system on the form (7.28a), the value $0 \leq \zeta \leq 1$ is referred to as the relative damping of the system.

In the extreme case $\zeta = 1$, (7.27) turns into

$$G(s) = \frac{\alpha\omega_0^2}{(s+\omega_0)^2}, \tag{7.30}$$

which is a special case of the system studied in Example 7.2. Since the response is the sum of two first-order system responses, it lacks oscillations, and thus systems on the form (7.27) with $\zeta = 1$ are referred to as fully damped.

For the other extreme, $\zeta = 0$, (7.27) turns into

$$G(s) = \frac{a\omega_0^2}{s^2 + \omega_0^2}. \tag{7.31}$$

In this case, we can look up in a Laplace transform table that the step response $y(t) = \mathcal{L}^{-1}(G(s)/s)$ is the undamped offset sinusoid

$$y(t) = a(1 - \cos(\omega_0 t)). \tag{7.32}$$

Thus a system on the form (7.27) with $\zeta = 0$ is referred to as (fully) undamped.

Returning to the Voigt model, we see from (7.29b) that it approaches a fully undamped system when $\zeta \to 0$ (i.e. $\eta \ll k$) and is fully damped when $\eta^2 = 4km$.

We have now seen examples of second-order system step responses. Of course a similar analysis can be conducted for higher-order systems. However, a lot of dynamic behaviors in physiology can be modeled using low-order models. This explains why we have put this emphasis on first and second-order systems.

Before moving on, we present a simple relation between pole location and step response characteristics for second-order systems. For the system

$$G(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0 + \omega_0^2} \tag{7.33}$$

the poles will be complex-conjugated and located as shown on the left in Figure 7.5. It is straightforward to show that the magnitude of either pole is $\omega_0$, and that they form an angle $\phi$ to the negative real axis, that relates to the relative damping through $\cos\zeta = \phi$.

The right side of Figure 7.5 shows how changing pole locations affects the step response of (7.33): the smaller the distance $\omega_0$ to the origin from the poles, the slower the response, and the larger angle $\phi$ to the negative real axis, the less damped the system becomes. When the poles lie on the imaginary axis, the system is fully undamped, and when the poles lie on the negative real axis the system is fully damped, as already explained in Example 7.3.

#### The impulse response

Imagine you inject one unit of drug evenly between time $t = 0$ and $t = \tau$. The injection rate is then $1/\tau$. As we let $\tau \to 0$, the injection rate will go

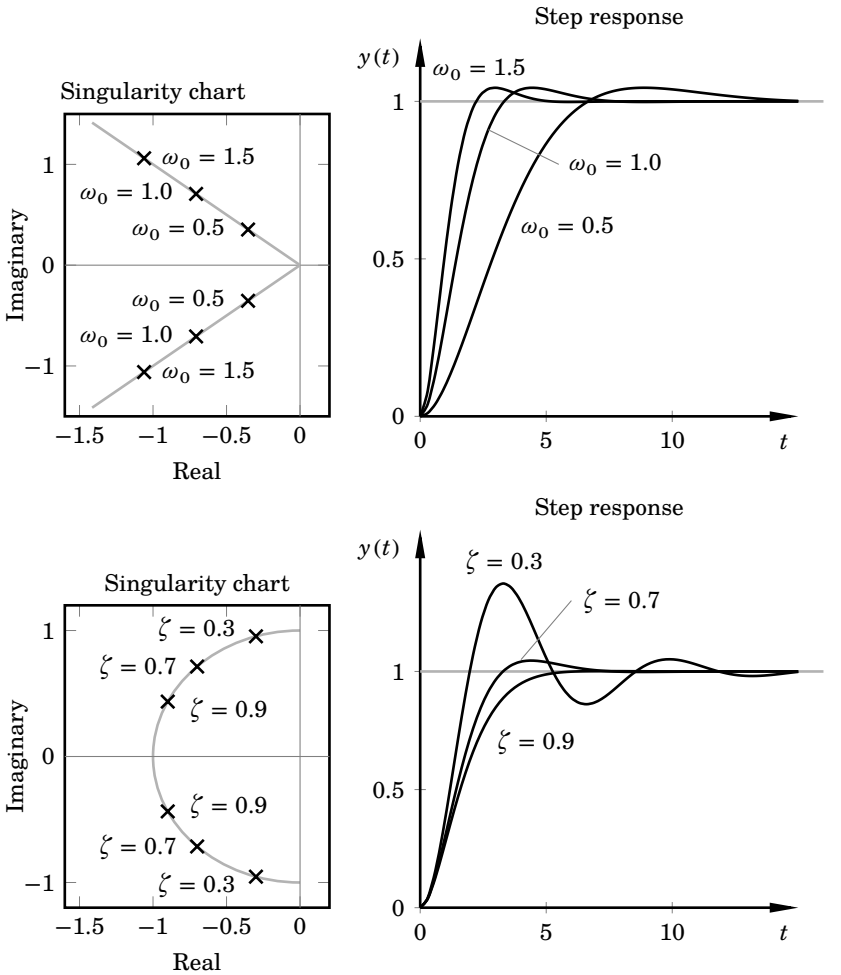


**Figure 7.5** Pole locations (left) for (7.33) and corresponding step responses (right). The bottom plots show what happens when the relative damping $\zeta$ is varied (for a fixed natural frequency $\omega_0$), while the top plots shows what happens when the natural frequency $\omega_0$ is varied (for a fixed relative damping).

to infinity, while the total injected mass remains 1. In the limit, we obtain the signal that we refer to as the impulse, and denote by $\delta(t)$, where the argument denotes that the impulse takes place at time $t = 0$. It has the defining properties

$$\delta(t) = 0, \; t \neq 0, \tag{7.34a}$$

$$\int_0^\infty \delta(t) \, dt = 1. \tag{7.34b}$$

If the injection is very fast, you can model this by all drug being injected at $t = 0$, using the input signal $u(t) = \delta(t)$. If there would be a second injection at $t = 5$ with three times as much drug mass, we would model that as $3\delta(t - 5)$, and the combined input signal would then be $u(t) = \delta(t) + 3\delta(t - 5)$.

Using the definition (6.1) of the Laplace transform, we find that the unit impulse transforms to

$$\delta(t) \xrightarrow{\mathcal{L}} \int_0^\infty e^{-st}\delta(t) \, dt = 1. \tag{7.35}$$

The unit impulse response of a system with transfer function $G(s)$ is thus

$$Y(s) = G(s)1 = G(s). \tag{7.36}$$

Remembering from (7.5) that the step response of the same system is $G(s)/s$, and that multiplication by $s$ in the Laplace domain corresponds to differentiating with respect to time in the time domain, we arrive at the following conclusion: If a system has step response $y(t)$, the corresponding impulse response is

$$h(t) = \dot{y}(t). \tag{7.37}$$

The impulse response is thus the derivative of the step response. Consequently, since the step response of a system with transfer function $G(s)$ is $G(s)/s$ in the Laplace domain, and since division by $s$ in the Laplace domain corresponds to integration with respect to time in the time domain, we have that

$$h(t) \xrightarrow{\mathcal{L}} G(s). \tag{7.38}$$

In words: The transfer function is the Laplace transform of the impulse response.

The impulse response also has an interpretation as a weighing function in the time-solution of an LTI system starting at rest. According to (4.56) the impulse response is

$$h(t) = C\int_0^t e^{A(t-\tau)}B\delta(\tau) \, d\tau + D\delta(t) = Ce^{At}B + D\delta(t), \tag{7.39}$$

and the general solution to the system can be written as

$$y(t) = Ce^{At}\boldsymbol{x}(0) + C\int_0^t h(t-\tau)u(\tau)\,d\tau, \tag{7.40}$$

which we can easily verify by combining (7.39) and (4.56). The impulse response thus serves as a weight by which the input is multiplied at each time point. The relation (7.40) can be intuitively understood if we think of the input $u(t)$ as the infinite sum (thus linear combination) of impulses issued at times $\tau$, and spaced $d\tau$ from each other. In the limit $d\tau \to 0$ This sum becomes

$$u(t) = \int_0^\infty u(\tau)\,\delta(t-\tau)\,d\tau. \tag{7.41}$$

## 7.2 The frequency response

Let us investigate the special case when the input to an LTI system is a sinusoid

$$u(t) = \sin(\omega t), \tag{7.42}$$

with angular frequency $\omega$. With $h(t)$ being the impulse response of our system, which we assume to be asymptotically stable, let us see what happens to the response after the initial-state transient has faded away. In absence of the initial state transient, the response to the input (7.42) is

$$\begin{aligned} y(t) &= \int_0^t h(t-\tau)u(\tau)\,d\tau = [r = t-\tau] \\ &= \int_0^t h(r)u(t-r)\,dr \\ &= \int_0^t h(r)\sin(\omega(t-r))\,dr \\ &= \operatorname{Im} e^{i\omega t}\int_0^t h(r)e^{-i\omega r}\,dr \end{aligned} \tag{7.43}$$

where we have made a change of integration variable and used Euler's formula

$$e^{i\omega} = \cos(\omega) + i\sin(\omega), \tag{7.44}$$

that enable us to write

$$\sin(\omega) = \operatorname{Im} e^{i\omega}. \tag{7.45}$$

In the limit $t \to \infty$, the last integral of (7.43) defines that Laplace transform of $h$ at $s = i\omega$. And since we have shown in Section 7.1 that the Laplace transform of the impulse response is the transfer function, we have that the response to $u(t) = \sin(\omega t)$ converges to

$$y(t) = \operatorname{Im} G(i\omega) e^{i\omega t}, \tag{7.46}$$

which, using the polar form, we can re-write as

$$\begin{aligned} y(t) &= |G(i\omega)| \operatorname{Im}\left(\exp(i\omega t) \exp(i \arg G(i\omega))\right) \\ &= |G(i\omega)| \sin(\omega t + \arg G(i\omega)). \end{aligned} \tag{7.47}$$

While the derivation was quite long, the result is simple and states that if the input signal to an asymptotically stable system is $\sin(\omega t)$, the resulting output will converge to $|G(i\omega)| \sin(\omega t + \arg G(i\omega))$ once initial state transient has faded away, where $|G(i\omega)|$ and $\arg G(i\omega)$ are the magnitude (gain) and argument (phase) of the transfer function at input frequency $\omega$. Let us illustrate how this can be used, using a practical example.

**Example 7.4: Frequency response.** Let

$$G(s) = \frac{3}{(s+1)(s+2)} \tag{7.48}$$

be a transfer function describing the dynamics between light falling onto the pupil, and the pupil's resulting aperture deviation. If light is varied according to

$$u(t) = 4 \sin(5t), \tag{7.49}$$

what will the response of the system be after initial transients have died out?

First we conclude that the system is asymptotically stable, since both poles $s = -1$ and $s = -2$ have strictly negative real part.

The transfer function gain (magnitude) at $s = i\omega = 5i$ is

$$\begin{aligned} |G(i\omega)| &= \frac{3}{|5i+1||5i+2|} = \frac{3}{\sqrt{5^2+1^2}\sqrt{5^2+2^2}} \\ &= \frac{3}{\sqrt{26 \cdot 29}} = \frac{3}{\sqrt{754}} \approx 0.11, \end{aligned} \tag{7.50}$$

and the corresponding argument, also known as phase angle, is

$$\begin{aligned} \arg G(i\omega) &= \arg 3 - \arg(5i+1) - \arg(5i+2) \\ &= 0 - \tan^{-1}(5) - \tan^{-1}(5/2) \approx 10.5^\circ. \end{aligned} \tag{7.51}$$

Since the input was not $\sin(5t)$ but $4\sin(5t)$, we need to multiply the output with the scale factor of 4, resulting in

$$y(t) \approx 0.44\sin(5t + 0.18), \tag{7.52}$$

where we have used $10.5^\circ \approx 0.18$ rad.

In words: after a possible initial-state transient fades, the response of the system (7.48) to the input (7.49) is given by (7.52).

## 7.3 The Bode diagram

Using the Fourier series, closely related to the Laplace transform, it is possible to decompose most practically meaningful physiological signals into (infinite) sums of sinusoidal components. This means that we can, at least in principle use (7.47) to simulate the response of a system, by first computing the input signal Fourier transform, feeding the individual components through, and recomposing the output. It also means that the magnitude $|G(i\omega)|$ and the phase $\arg G(i\omega)$ carry all information about the system that we need in practice.

A compact graphical representation of a transfer function $G$ therefore consists of plotting magnitude and phase as functions of $\omega$. In the Bode diagram, also referred to as the Bode plot, and named after Hendrik Bode (1905–1982), magnitude is plotted in log-log scale, while the phase is plotted in log-lin scale, as illustrated in Figure 7.6, showing the Bode plot of

$$G(s) = \frac{3}{s+2}. \tag{7.53}$$

For the transfer function (7.53), the logarithm of the magnitude is given by

$$|G(i\omega)| = \log\left(\frac{3}{|i\omega + 2|}\right) = \log(3) - \log\sqrt{\omega^2 + 4}, \tag{7.54}$$

and the phase is given by

$$\arg G(i\omega) = 0 - \tan^{-1}(\omega/2) = -\tan^{-1}(\omega/2). \tag{7.55}$$

By reading off the magnitude and phase plots for a particular angular frequency $\omega$ in the Bode diagram of a transfer function $G$, one recovers $|G(i\omega)|$ and $\arg G(i\omega)$. Using (7.47) it is thus possible to answer what the response (output) to a sinusoidal input $\sin(\omega t)$ will be after the initial state transient has died out, as long as $G$ is asymptotically stable.

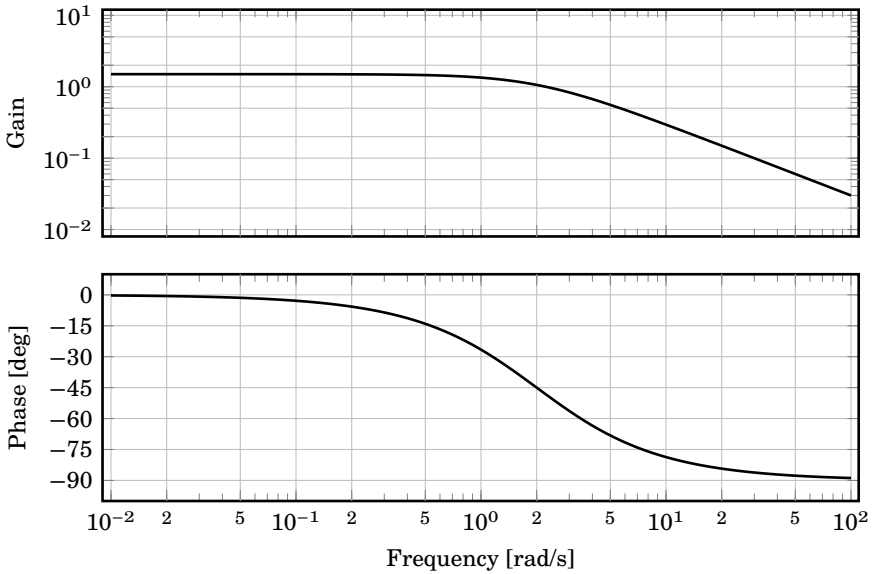


**Figure 7.6** Bode plot of the the transfer function $G(s)k/(sT+1) = 3/(s+2)$..

By now you have gained a rich set of tools from dynamical modeling, that can be applied to a vast set of applications in physiology, and also outside of physiology. We have tried to look at the same thing, or at least similar things, from many angles. To most this is at first confusing and it takes some re-reading, doing exercises, thinking, and discussing, to finally realize how all the topics fit together. You can certainly use the methods introduced in the book without ever reaching that point, but grasping the underlying principles and mechanisms will make things much easier and put less requirements on you to remember things by heart.

***Further reading.*** As a closing suggestion, we recommend [Ljung et al., 2021], a book that provides additional perspectives on dynamic system modeling. We also include a reference to computer-control systems [Åström and Wittenmark, 1984]. Grasping these additional topics will enable you both to model physiological dynamics from data, and construct cyber-physical control systems within the context of medical devices.

Ljung, L., T. Glad, and A. Hansson (2021). *Modeling and identification of dynamic systems*. Studentlitteratur, Lund, Sweden. ISBN: 9789144153452.

Åström, K. J. and B. Wittenmark (1984). *Computer-controlled systems: theory and design*. Prentice Hall, Hoboken, NJ. ISBN: 978-0486486130.

# Index