Universität Bremen*

Modeling and System Identification

— Lecture Notes —

Jürgen Pannek

Dynamics in Logistics
Fachbereich 04: Produktionstechnik
Universität Bremen

*EXZELLENT.

Foreword

This script originates from a correspondent lecture held during the summer term 2018 at the University of Bremen. The central aims of the lecture are the introduction of modeling and system identification techniques for (dynamical) systems. In particular, we model differential equation systems to design

- Deterministic Processes as well as
- Stochastic Processes.

To deal with these kind of systems properly, we give a short introduction/repetition to differential equations. Based on these basic models, we then identify „the real“ system, i.e. we fit data to model. To this end, we introduce basic stochastic definitions and discuss

- Least Square Estimaton and
- Kalman Filtering.

At the end of the lecture, students should understand the concepts, know basic formulas, be able to comprehend and interpret input and output of the methods and to make a suitable choice between the presented methods.

Parts of the scripts are based on script the of Prof. Grüne [4] and the books [2, 5, 10] regarding the modeling part, and the script of Prof. Schoukens [8] as well as the books [6, 9] served as a basis for the identification part.

Contents

Chapter 1

Introduction

Within this chapter, we give a brief introduction to modeling and identification. To this end, we use a simple example to model and to illustrate pitfalls associated with a model built from noisy measurements. Additionally, we give a recap of terms from differential equations and probability theory, which we will require throughout the lecture.

1.1 What is a "Model"?

Intuitively, we all know what a model is. We have identified it by learning to control our actions using predictions of the effect of these actions. These predictions are based on a model, and form a model of reality in our mind. There are simple connection, e.g. "I push a ball, then it rolls". Yet, we may also accumulate very complicated systems such as cars, supply chains or weather forecast. Hence, the model is something deterministic, without uncertainty and predictable for all times.

Unfortunately, as we all experienced, models do not represent reality one–to–one. So if we use a model, there may be deviations between model prediction and reality, especially if long time horizons are considered. The reason for that is due to the following: For a model, we always focus on those aspects we are interested in, and do not try to describe all of reality. Hence, the problem is split into two parts,

- the model, which describes what we are interested in, and
- the environment, which contains everything else.

Since we cannot tell anything about the environment (as it is not modeled), interactions between model and environment can only be interpreted as disturbances.

During the modeling process, six principles need to be met:

1. Principle of Correctness: A model needs to present the facts correctly regarding structure and dynamics (semantics). Specific notation rules have to be considered (syntax).
2. Principle of Relevance: All relevant items have to be modeled. Non-relevant items have to be left out, i.e. the value of the model doesn't decline if these items are removed.
3. Principle of Cost vs. Benefit: The amount of effort to gather the data and produce the model must be balanced against the expected benefit.
4. Principle of Clarity: The model must be understandable and usable. The required knowledge for understanding the model should be as low as possible.

5. Principle of Comparability: A common approach to modeling ensures future comparability of different models that have been created independently from each other.

6. Principle of Systematic Structure: Models produced in different views should be capable of integration. Interfaces need to be designed to ensure interoperability.

Here, an interesting point arises: Since typically modeler and model user are different entities, which exhibit different perspectives on the process, a good model for the modeler may be very different from a good model for the model user. For example, a detailed model may reflect reality very well, but it may be too complex to evaluate in a real-time setting and hence not usable for feedback control. Hence, modeling must be in line with the usage, and the quality of a model is determined by the degree it meets the requirements of the model user ("fitness for use").

Within this lecture, we focus on the quantitative description of a model, i.e. qualitative results such as "a ball will roll downhill" are not the kind of model properties we are looking for. Instead, we utilize laws, which may, e.g., be given by physics or econometrics, and describe at least some part of our impression of reality.

In particular, we consider models satisfying the so called nonlinear continuous time control systems form

$$\dot{x}(t) = f(x(t), u(t)), \tag{1.1}$$

where x represents the internal state of the system, u the external force on the system, f the law or dynamics of the system, and t the continuous time respectively. We want to develop models for a number of applications, to analyze respective techniques and to discuss assumptions made regarding the model. Additionally we will recall some aspects of the qualitative analysis of differential equations, which are required to investigate properties of the models. Since this will be a simple recall, no details on proofs will be given, but instead we refer to [2,5,10] for details.

We will particularly focus on applications from

- Biological Processes,
- Mechanical Processes, as well as
- Financial Processes.

Each of these topics is that large, that we cannot cover them entirely. Therefore, we restrict ourselves to certain aspects from these topics. Regarding biological processes, we consider the growth model, which can be used to describe the growth of a market for a product or of the population of a species, or for several competing ones. Regarding mechanical processes, we will use the laws of motion from classical mechanics to develop modular models of mechanical processes. And last, in financial applications we focus on the assessment of options using the Black–Scholes equation.

The various sub-areas provide the opportunity to learn about different techniques and possibilities, but also about limitations of modeling. In biology, we will see that — although the individual elements of the models can be quite mathematically rigorously justified — the significance of the overall models is greatly influenced by many external influences, which cannot be taken into account if the mathematical model shall remain tractable. As a consequence, one has to introduce model assumptions, for example, that a product- or eco-system is closed and not influenced from the outside, or that the effects of changing seasons are neglected. In mechanics, we also have to integrate model assumptions, yet the neglected effects can be estimated

much better. An example of this is the static friction, which is often neglected. If, however, a mechanical system is sufficiently fast, then this effect does not even appear. In contrast to the laws that can be used in mechanics, financial processes are based on differential equations, which model the stock development in a mainly phenomenological manner. Indeed, equations are applied which restate real stock developments, yet without underlying principles.

1.2 What is "Identification"?

Let us now assume that we have a model or a process at hand. Then our next task is to match the behavior of the model to the one of the real process, which is also called fitting. To this end, we not only need the model itself, we also require data from the process and a possibility to simulate the model. Matching simulated to real data is then qualified by an optimization criterion, which is defined, as described before, by the degree of the model meeting the requirements of the model user. This criterion allows us to actually fit the model. Last, the model should always be validated, i.e. tested for failure or unfruitful results. Hence, each identification process consists of a series of basic steps:

1. Collect information on the system
2. Select a model to represent the system
3. Choose an optimization criterion
4. Fit the model parameters to the measurements accordingly
5. Validate the computed model

Note that some of the steps may be hidden from the user or selected without being aware of a choice, which may result in suboptimal or even poor performance. Unfortunately, fitting laws or models to observations creates new problems:

- For one, we consider noisy measurements. In this context, noisy means that if we take a measurement, e.g. length, weight, time etc., then errors occur since the instruments we use are not perfect.
- And secondly, our laws and models are imperfect as reality is far more complex than the rules we apply. They also show a stochastic behavior, which makes it impossible to predict exactly their output.

To still identify the system, we split the model into a deterministic and a stochastic part. The deterministic aspects are captured by the mathematical system model. These are complemented by the stochastic behavior, which are modeled as a noise distortion. Hence, the aim of identification theory is the following:

> Identification theory provides a systematic approach to fit the mathematical model to the deterministic part as well as possible, and to eliminate the noise distortions as much as possible.

Within this lecture, we particularly focus on the techniques of the

- Least Square Estimator,
- Maximum Likelihood and Bayes Estimator, and of the

- Kalman Filter.

Note that the terms estimator and filter are similar, yet an estimator refers to a static problem and a filter to a dynamic one. Still, estimators can be applied to dynamical problem, but are not ideally suited.

Before defining the required terms, we motivate and illustrate many of the aspects and problems in identification theory by a simple example.

1.3 A simple example

Using two electric circuits as shown in Figure 1.1, we pass a constant but unknown current through the resistor. The voltage u across the resistor and the current i through it are measured using a voltmeter and an ampere meter, where the input impedance of the voltmeter is chosen large compared with the unknown resistor to ensure that all the measured current passes through the resistor.

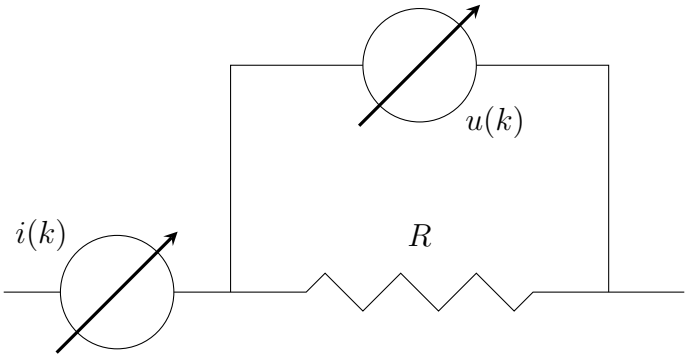


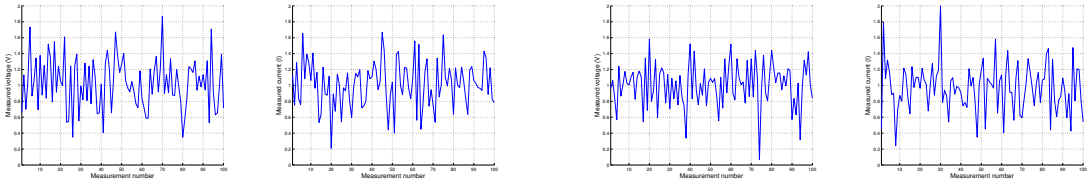
Figure 1.1: Measurement of a resistor.

The resistor model is given by Ohm's law

$$R = U/I, \tag{1.2}$$

and our aim is not to identify the resistance given some measurements, i.e. to fit the model.

Here, we suppose that two sets of measurements $u(k)$, $i(k)$ with $k = 1, 2, \ldots, N$ are taken and called group A and B, cf. Figure 1.2, and the resistances $r(k)$, $k = 1, 2, \ldots, N$ are computed via $r(k) = u(k)/i(k)$, see Figure 1.3 for respective results.



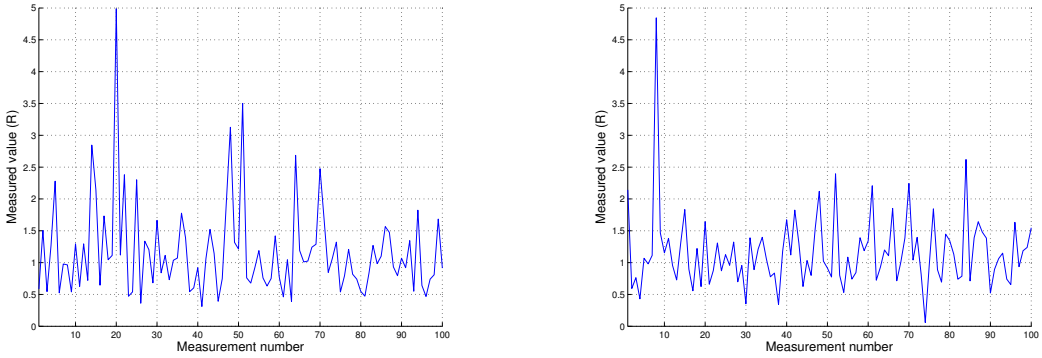
(a) Group A of measurements

(b) Group B of measurements

Figure 1.2: Measurement values for two groups

Since the measurements are very noisy, we apply different estimators to analyze the resistor:

$$\hat{R}_{\mathrm{SA}}(N) = \frac{1}{N} \sum_{k=1}^{N} \frac{u(k)}{i(k)} \tag{1.3}$$



(a) Computed resistance from group A (b) Computed resistance from group b

Figure 1.3: Computed resistances from measurement groups

$$\hat{R}_{\mathrm{EV}}(N) = \frac{\frac{1}{N}\sum_{k=1}^{N} u(k)}{\frac{1}{N}\sum_{k=1}^{N} i(k)} \tag{1.4}$$

$$\hat{R}_{\mathrm{LS}}(N) = \operatorname*{argmin}_{R\in\mathbb{R}} \sum_{k=1}^{N} (R\cdot i(k) - u(k))^2 \tag{1.5}$$

Within these estimators, N indicates the number of used measurements. Note that the three estimators result in the same estimate on noiseless data. The first estimator, the **S**imple **A**pproach, averages the quotients of voltage and current measurements. The second estimator, the **E**rror-in-**V**ariable approach, averages the voltages and currents first and then computes the quotient. And the last estimator, the **L**east **S**quare approach, computes the minimal distance linear function between voltage and current pairs in the 2-norm.

To compute the latter, we set $f(R) := \sum_{k=1}^{N} (R\cdot i(k) - u(k))^2$ and minimize it, i.e., we are looking for a value R such that

$$\frac{\partial f(R)}{\partial R} = 0.$$

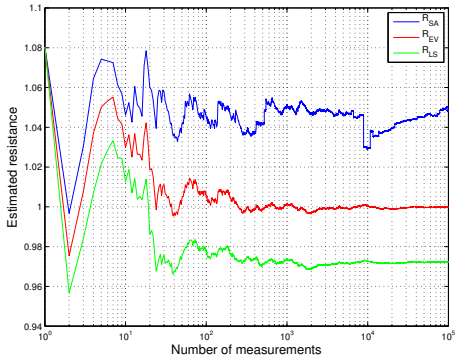
From the definition of $f(R)$, we obtain

$$\begin{aligned}\frac{\partial f(R)}{\partial R} &= \sum_{k=1}^{N} 2\cdot(R\cdot i(k) - u(k))\cdot i(k)\\ &= 2\cdot R\cdot\sum_{k=1}^{N} i(k)^2 - 2\cdot\sum_{k=1}^{N} u(k)\cdot i(k).\end{aligned}$$

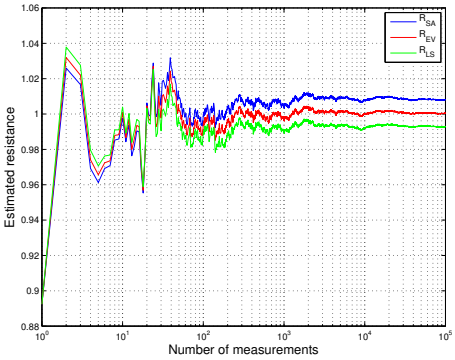
Hence, we have

$$\hat{R}_{\mathrm{LS}}(N) = \frac{\sum_{k=1}^{N} u(k)\cdot i(k)}{\sum_{k=1}^{N} i(k)^2}. \tag{1.6}$$

Utilizing these estimation formulas, we can compute the estimated resistances as displayed in Figure 1.3. From this figure, we can make several observations:



(a) Estimated resistance from group A



(b) Estimated resistance from group B

Figure 1.4: Estimated resistances from measurement groups with $\hat{R}_{\mathrm{SA}}$ in blue, $\hat{R}_{\mathrm{EV}}$ in red and $\hat{R}_{\mathrm{LS}}$ in green.

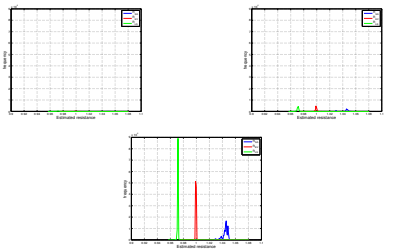
1. All estimators have large variations for small values of N, and – except for $\hat{R}_{\mathrm{SA}}$ from group A – seem to converge to an asymptotic value for large values of N. This corresponds to the intuitively expected behavior: if a large number of data points are processed, then we should be able to eliminate the noise influence due to the averaging effect.

2. The asymptotic values of the estimators depend on the kind of averaging technique that is used. This shows that there is a serious problem: at least two out of the three methods converge to a wrong value. It is not even certain that any one of the estimators is doing well. This is quite catastrophic: even an infinite amount of measurements does not guarantee that the exact value is found.

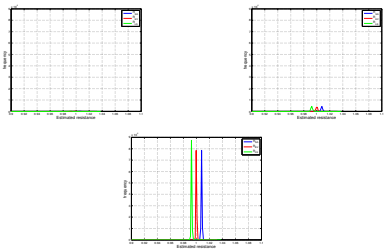
3. The $\hat{R}_{\mathrm{SA}}$ from group A behaves very strangely. Instead of converging to a fixed value, it jumps irregularly up and down.

These observations clearly indicate that a good theory is needed to explain and understand the behavior of candidate estimators. This will allow us to make a sound selection out of many possibilities and to indicate in advance if a method is prone to serious shortcomings before running expensive experiments.

To gain more insight, we can plot approximations of the probability density functions based on the data, cf. Figure 1.5. From this figure, we observe the following:



(a) Observed probability density functions for group A



(b) Observed probability density functions for group B

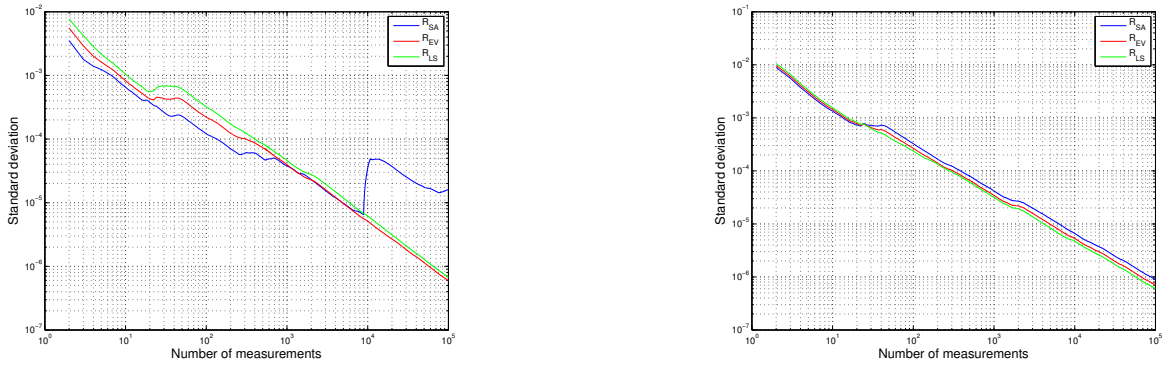
Figure 1.5: Observed probability density functions for groups. From left to right $N = 1000$, $N = 10000$ and $N = 100000$ with $\hat{R}_{\mathrm{SA}}$ in blue, $\hat{R}_{\mathrm{EV}}$ in red and $\hat{R}_{\mathrm{LS}}$ in green.

1. For small values of N, the estimates are widely scattered. As the number of processed measurements increases, the probability density function becomes more concentrated.

2. The estimates $\hat{R}_{\mathrm{LS}}$ are less scattered than $\hat{R}_{\mathrm{EV}}$ and $\hat{R}_{\mathrm{SA}}$, and odd behavior for $\hat{R}_{\mathrm{SA}}$ in group A appears again. The distribution of this estimate does not contract to a single value for growing values of N for group A, while it does for group B.

3. It is clearly visible that the distributions are concentrated around different values.

The choice of the estimator is not yet clear. And additionally, there seems to be a major problem with the measurements of group A, which was observed via $\hat{R}_{\mathrm{SA}}$. In order to quantify the scattering of the estimates, in particular of $\hat{R}_{\mathrm{SA}}$, the standard deviation can be calculated, cf. Figure 1.6. Here, we observe that the standard deviation decreases monotonically with N –

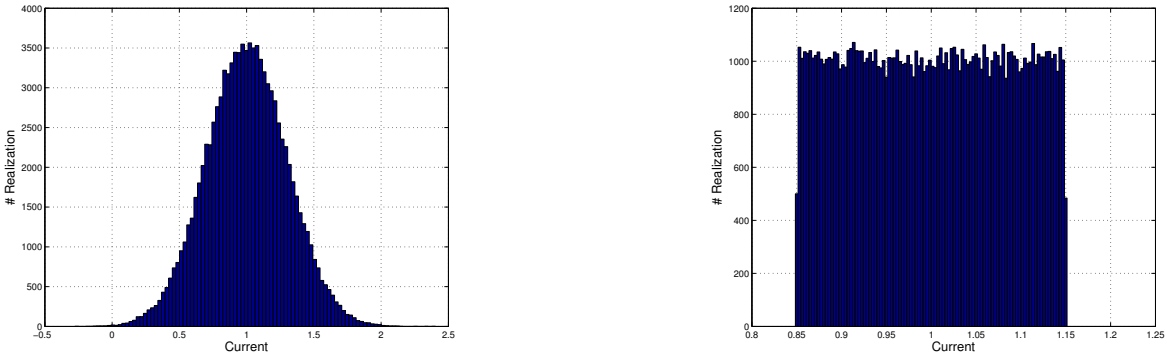


(a) Observed standard deviation for group A (b) Observed standard deviation for group B

Figure 1.6: Observed standard deviation for groups. From left to right $N = 1000$, $N = 10000$ and $N = 100000$ with $\hat{R}_{\mathrm{SA}}$ in blue, $\hat{R}_{\mathrm{EV}}$ in red and $\hat{R}_{\mathrm{LS}}$ in green.

except for $\hat{R}_{\mathrm{SA}}$ of group A. Moreover, the decrease is proportional to $1/\sqrt{N}$, which is the rule of thumb for the uncertainty on an averaged quantity obtained from independent measurements. Additionally, the uncertainty depends on the estimator.

Regarding the strange behavior of $\hat{R}_{\mathrm{SA}}$ of group A, we reconsider the measurement data displayed in Figure 1.2 and compute respective histograms, cf. Figure 1.7. Due to possibly



(a) Histogram for $i(\cdot)$ for group A (b) Histogram for $i(\cdot)$ for group B

Figure 1.7: Comparison of histograms for the current $i(\cdot)$

occuring zero values for the current in group A, we obtain a drastic increase in the estimation

using the simple approach. This is due to a division by (almost) zero. In group B, such a case does not exist.

This example shows that there is a clear need for methods, which can generate and select between different estimators. We also like to note that although the noise on the measurements in group A is completely different distributed, the resulting estimation, e.g. by $\hat{R}_{\mathrm{EV}}$ and $\hat{R}_{\mathrm{LS}}$, seem to be the same as in group B.

Before coming to a structured approach of identifying a process in Chapter 5, we first need to introduce some notation and make us familiar with basic definitions.

1.4 Basic Terms

Similar to the denomination of the lecture itself, also the terms which we are going to use are from different fields. Regarding the modeling part, we will restrict ourselves to differential equations, and for the largest portion of the lectures, we will narrow it down to ordinary differential equations. This will give us the deterministic part. Regarding the stochastic part, we require tools from stochastic analysis, mainly regarding probability theory but also regarding certain concepts of convergence. The identification part of the lecture will make use of both the deterministic model, and the stochastic part of the process.

1.4.1 Recall from Differential Equations

An ordinary differential equation relates the derivative of a function $x : \mathbb{R} \to \mathbb{R}^{n_x}$ with its onedimensional argument and the function itself. More formally:

Definition 1.1 (Ordinary Differential Equation)
An ordinary differential equation in $\mathbb{R}^{n_x}$, $n_x \in \mathbb{N}$, is given by

$$\frac{d}{dt}x(t) = f(t, x(t)) \tag{1.7}$$

where $f : \mathbb{D} \to \mathbb{R}^{n_x}$ is a continuous function and $\mathbb{D}$ is an open subset of $\mathbb{R} \times \mathbb{R}^{n_x}$.

The solution of (1.7) is a continuously differentiable function $x : \mathbb{R} \to \mathbb{R}^{n_x}$, which satisfies (1.7). In general, we will use the following denomination throughout the script:

- The independent variable t is referred to as time, although other interpretations are possible.
- Instead of $\frac{d}{dt}x(t)$ we will often use the abbreviation $\dot{x}(t)$.
- The function $x(t)$ is called solution or trajectory.
- If the function f is independent of t, i.e. $\dot{x}(t) = f(x(t))$, then the differential equation is called autonomous.

An ordinary differential equation typically possesses infinitely many solutions. Examplarily, we consider the differential equation

$$\dot{x}(t) = x(t)$$

with $x(t) \in \mathbb{R}$. Moreover, suppose $x(t) = C \exp^t$ with $C \in \mathbb{R}$ arbitrary but fixed. Hence, we have

$$\dot{x}(t) = C \exp^t = x(t),$$

which holds for each $C \in \mathbb{R}$, i.e. the differential equation has infinitely many solutions.

To obtain a unique solution, we have to introduce a constraint, the so called initial value constraint. Combined with the differential equation (1.7), this reveals the so called initial value problem:

Definition 1.2 (Initial Value Problem)
Consider values t_0 and $x_0 \in \mathbb{R}^{n_x}$ to be given. Then the initial value problem is to find the solution satisfying the differential equation

$$\dot{x}(t) = f(t, x(t)) \tag{1.7}$$

and the initial value condition

$$x(t_0) = x_0. \tag{1.8}$$

Here, the time $t_0 \in \mathbb{R}$ is called initial time and the value $x_0 \in \mathbb{R}^{n_x}$ is called initial value. Both the pair (t_0, x_0) and equation (1.8) are called initial condition.

Remark 1.3
A continuously differentiable function $x : \mathbb{D} \to \mathbb{R}^{n_x}$ solves the initial value problem (1.7), (1.8) *for some $t_0 \in \mathbb{D}$ and $x_0 \in \mathbb{R}^{n_x}$ if and only if for each $t \in \mathbb{D}$ the integral equation*

$$x(t) = x_0 + \int_{t_0}^{t_f} f(\tau, x(\tau))\, d\tau \tag{1.9}$$

holds. This follows directly by integration of (1.7) *with respect to t or via differentiation of* (1.9) *with respect to t using the central theorem of differentiation and integration. Note that each continuous function $x(t)$ satisfying* (1.9) *is automatically continuously differentiable since continuity of $x(t)$ on the right hand side of* (1.9) *implies continuous differentiability of the right hand side, and hence of $x(t)$ itself.*

Under certain conditions, existence and uniqueness of a solution to the problem from Definition 1.2 can be shown. This is the so called Lipschitz condition

Definition 1.4 (Lipschitz Condition)
Consider a function $f : \mathbb{D} \to \mathbb{R}^{n_x}$ with $\mathbb{D} \subset \mathbb{R} \times \mathbb{R}^{n_x}$. Then f is called Lipschitz in its second argument, if for each compact set $K \subset \mathbb{D}$ there exists a constant $L > 0$ and

$$\|f(t, x) - f(t, y)\| \leq L\|x - y\| \tag{1.10}$$

holds for all $t \in \mathbb{R}$ and all $x, y \in \mathbb{R}^{n_x}$ with $(t, x), (t, y) \in K$.

Using this property, we can show the following:

Theorem 1.5 (Existence and Uniqueness)
Consider a differential equation (1.7) *with* $f : \mathbb{D} \to \mathbb{R}^{n_x}$ *and* $\mathbb{D} \subset \mathbb{R} \times \mathbb{R}^{n_x}$. *Moreover,* f *is considered to be continuous and Lipschitz continuous in the second argument. Then for each initial condition* $(t_0, x_0) \in \mathbb{D}$, *there exists a unique solution* $x(t; t_0, x_0)$ *of the initial value problem* (1.7), (1.8). *This solution is defined for all* t *from an open maximal interval of existence* I_{t_0,x_0} *with* $t_0 \in I_{t_0,x_0}$.

Proof. We show the results in three steps:

1. We show that for each initial condition $(t_0, x_0) \in \mathbb{D}$ there exists a closed interval J around t_0 such that the solution exists and is unique.
2. Next we show uniqueness of the solution on an arbitrary large interval I.
3. Last we show existence of a maximal interval of existence.

Part 1: We choose a bounded closed interval I around t_0 and $\varepsilon > 0$, such that the compact neighborhood $U = I \times B_\varepsilon(x_0)$ of (t_0, x_0) lies in $\mathbb{D}$. Since $\mathbb{D}$ is an open set, this is always possible. Since f is continuous and U is compact, there exists a constant M such that $\|f(t, x)\| \leq M$ for all $(t, x) \in U$. Now we choose $J = [t_0 - \delta, t_0 + \delta]$ with $\delta > 0$ such that $J \in I$ and $L\delta < 1$ and $M\delta < \varepsilon$. Now we apply Banach's Fixpoint Theorem on the Banach space $\mathcal{C}(J, \mathbb{R}^{n_x})$ with norm $\|x\|_\infty := \sup_{t\in J} \|x(t)\|$. On $\mathcal{C}(J, \mathbb{R}^{n_x})$ we define the map

$$T : \mathcal{C}(J, \mathbb{R}^{n_x}) \to \mathcal{C}(J, \mathbb{R}^{n_x}), \quad T(x)(t) := x_0 + \int_{t_0}^{t} f(\tau, x(\tau))\, d\tau.$$

Note that for each $t \in J$ and each $x \in B := \mathcal{C}(J, B_\varepsilon(x_0))$ the inequality

$$\begin{aligned} \|T(x)(t) - x_0\| = \left\| \int_{t_0}^{t} f(\tau, x(\tau))\, d\tau \right\| &\leq \left| \int_{t_0}^{t} \|f(\tau, x(\tau))\|\, d\tau \right| \\ &\leq \delta M \leq \varepsilon \end{aligned}$$

holds, and hence T maps the set B on itself. To apply Banach's Fixpoint Theorem to this set, we have to show that $T : B \to B$ is a contraction, i.e.

$$\|T(x) - T(y)\|_\infty \leq k\|x - y\|_\infty$$

holds for all $x, y \in B$ and a constant $k < 1$. For $k = L\delta < 1$, this follows from

$$\begin{aligned} \|T(x) - T(y)\|_\infty &= \sup_{t\in J} \left\| \int_{t_0}^{t} f(\tau, x(\tau))d\tau - \int_{t_0}^{t} f(\tau, y(\tau))d\tau \right\| \\ &\leq \sup_{t\in J} \left| \int_{t_0}^{t} \|f(\tau, x(\tau))d\tau - f(\tau, y(\tau))\|\, d\tau \right| \\ &\leq \sup_{t\in J} |t - t_0| L\|x - y\|_\infty = \delta L\|x - y\|_\infty. \end{aligned}$$

Hence, the assumption of Banach's Fixpoint Theorem are satisfied and T exhibits a fixed point. Since the iteration satisfies the integral equation (1.9) by construction, the resulting solution $x(t)$ is a continuously differentiable function.
It remains to show that $x(t)$ is also unique. From Banach's Fixpoint Theorem we know that

in $B = \mathcal{C}(J, B_\varepsilon(x_0))$ there exist no other fixed point of T. Hence, we only have to show that there exists no fixed point outside B. Suppose there exists such a fixed point $y \notin B$ of T, i.e. we have $\|y(t) - x_0\| > \varepsilon$ for some $t \in J$, where we assume $t > t_0$ without loss of generality. By continuity, there exists a $t^\star \in J$ such that $\|y(t^\star) - x_0\| = \varepsilon$ and $y(s) \in B_\varepsilon(x_0)$ for $s \in [t_0, t^\star]$. Hence, we have

$$\begin{aligned}\varepsilon = \|y(t^\star) - x_0\| &= \left\| \int_{t_0}^{t^\star} f(\tau, y(\tau)) d\tau \right\| \leq \int_{t_0}^{t^\star} \|f(\tau, y(\tau))\| d\tau \\ &\leq (t^\star - t_0) M < \delta M.\end{aligned}$$

By $\delta M \leq \varepsilon$, this results in a contradiction. Hence, uniqueness of $x(t)$ follows.
Part 2: Suppose x, y are two solutions of the initial value problem, which are defined on the interval I. Now suppose that there exists a $t \in I$ such that $x(t) \neq y(t)$. Without loss of generality, we assume that $t > t_0$. Since both solutions coincide and are continuous (Part 1), there exist $t_2 > t_1 > t_0$ such that

$$x(t_1) = y(t_1) \quad \text{and} \quad x(t) \neq y(t) \text{ for all } t \in (t_1, t_2).$$

Both solutions solve the initial value problem with initial condition $(t_1, x(t_1)) \in \mathbb{D}$. From Part 1 we can conclude that there exists a unique solution of the initial value problem on an interval $\tilde{J}$ around t_1, i.e. $x(t) = y(t)$ for all $t \in \tilde{J}$. Since $\tilde{J}$ as an interval around t_1 contains at least one point t with $t_1 < t < t_2$, this is a contradiction and x and y must coincide on the entire interval I.
Part 3: For J from Part 1 we define

$$\begin{aligned}t^+ &:= \sup\{s > t_0 \mid \text{there exists a solution on } J \cup [t_0, s)\} \\ t^- &:= \inf\{s < t_0 \mid \text{there exists a solution on } J \cup (s, t_0]\}\end{aligned}$$

and set $I_{t_0,x_0} = (t^-, t^+)$. Since the set for generating supremum and infimum are nonempty due to containing $s \in J$, both t^- and t^+ exist. By definition of t^- and t^+ no larger interval $I \supset I_{t_0,x_0}$ exists showing the assertion. □

Note that at the boundary of the interval of existence I_{t_0,x_0} the solution ceases to exist. If the interval is bounded, then there are two possible reasons for that: For one, the solution may diverge, or secondly the solution converges to a boundary point of $\mathbb{D}$. In the remainder of this script, we will always assume that the assumptions of Theorem 1.5 are met without explicitly stating it.

Remark 1.6
1. A simple consequence of Theorem 1.5 is the so called cocycle property. This property states that for $(t_0, x_0) \in \mathbb{D}$ and two time instances $t_1, t \in \mathbb{R}$, we have

$$x(t; t_0, x_0) = x(t; t_1, x_1) \tag{1.11}$$

with $x_1 = x(t_1; t_0, x_0)$ given that all terms are defined according to Theorem 1.5.
2. Another consequence is that two solutions cannot intersect, as they would have to coincide for all times.
3. Some ordinary differential equations can be solved analytically via various methods. In general, this is not true and numerical methods must be used for this purpose. Yet, one typically not only applies numerical methods, but tries to show certain properties of the solution analytically.

1.4.2 Recall from Stochastics

In order to analyze estimators, we first need to classify them. As we have seen in the previous sections, estimators are obtained as functions of a finite number of noisy measurements. Hence, they are stochastic variables, just as the noisy measurements are. To characterize a stochastic variable completely, we require the respective probability density function. In practice, however, it is very hard to derive that function. Yet, the behavior of the estimates can be described by a few numbers, i.e. the mean value and the covariance, which may be seen as the location and dispersion of the estimate.

To formally introduce these numbers, we first require the notion of a probability space:

Definition 1.7 (Probability space)
Consider a set Ω, a set of subsets $\mathcal{F} \subseteq 2^{\Omega}$ and a function $\mathrm{P} : \mathcal{F} \to [0,1]$. Then, we call the triple $(\Omega, \mathcal{F}, \mathrm{P})$ a probability space if

- the sample space Ω is a non–empty set,
- the σ–algebra $\mathcal{F}$ of events satisfies
  - $\mathcal{F}$ contains the empty set, i.e.
$$\emptyset \in \mathcal{F},$$
  - $\mathcal{F}$ is closed under complements, i.e.
$$A \in \mathcal{F} \implies \Omega \setminus A \in \mathcal{F},$$
  - $\mathcal{F}$ is closed under countable unions, i.e.
$$A_i \in \mathcal{F}\ \forall i \in \{1, 2, \ldots, k\}, k < \infty \implies \bigcup_{i \in \{1,2,\ldots,k\}} A_i \in \mathcal{F}$$
- the probability measure P satisfies
  - P is countably additive, i.e.
$$A_i \in \mathcal{F}\ \forall i \in \{1, 2, \ldots, k\}, k < \infty \text{ with } A_i \cap A_j = \emptyset\ \forall i, j \in \{1, 2, \ldots, k\}, i \neq j$$
$$\implies \mathrm{P}\left(\bigcup_{i \in \{1,2,\ldots,k\}} A_i \right) = \sum_{i \in \{1,2,\ldots,k\}} \mathrm{P}\left(A_i\right),$$
  - the measure of the sample space Ω is one, i.e.
$$\mathrm{P}\left(\Omega\right) = 1.$$

In short, a probability space is a measure space, but with the additional property that the measure of the whole space is equal to one. Secondly, we require so called random variables:

Definition 1.8 (Random variable)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a measurable space E with σ–algebra $\mathcal{E}$ of E. Then we call a function $X : \Omega \to E$ a random variable if

$$\forall B \in \mathcal{E} : \; X^{-1}(B) \in \mathcal{F}, \quad \text{where } X^{-1}(B) := \{\omega \in \Omega \mid X(\omega) \in B\}.$$

Hence, a random variable is a function, which allows us to use a more comfortable description of properties or measurements of a sample, i.e. if B is an interval $[a, b]$ or the property "lottery player", then we identify the corresponding event $X^{-1}(B)$ in the σ–algebra $\mathcal{F}$.

Now, we can introduce the expected value, sometimes also called mean, first moment or expectation:

Definition 1.9 (Expected value or mean)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a random variable X defined on that triple. Then, the expected value $\mathrm{E}(X)$ or mean of X is defined as the Lebesgue integral

$$\mathrm{E}(X) := \int_{\Omega} X \, d\mathrm{P} = \int_{\Omega} X(\omega) \, d\mathrm{P}(\omega) \tag{1.12}$$

whenever the integral exists.

Note that since the integral may not converge absolutely, not all random variables have a finite expected value, and for some it is not defined at all (e.g., Cauchy distribution).

In order to define the second important number, the covariance, we first introduce the notion of moments:

Definition 1.10 (Moment)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$, a natural number $n \in \mathbb{N}$ and a random variable X defined on that triple. Then, the n–th moment is given by

$$m_n := \mathrm{E}(X^n). \tag{1.13}$$

Hence, the mean is also the first moment. Regarding the covariance, we require the second moment to describe, how much two random variables in one probability space change together, i.e. what the nature of their connection and how strong this connection is:

Definition 1.11 (Covariance)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and two random variables X and Y defined on that triple. Then, the covariance $\mathrm{Cov}(X, Y)$ is defined as

$$\mathrm{Cov}(X, Y) := \mathrm{E}((X - \mathrm{E}(X))(Y - \mathrm{E}(Y))) \tag{1.14}$$

whenever the second moments of X and Y exist.
If $X = Y$, then covariance is called variance and we obtain $\mathrm{Cov}(X, X) = \sigma^2(X)$.

Higher moments describe the skewness and curtosis of the probability function P, which can be interpreted as a deviation measure from the normal distribution and a deviation measure from a symmetric distribution respectively.

The following notion of a so called probability density function uses the nice property of a random variable to be a transformation to an easily interpretable space. I.e., it describes the relative likelihood for this random variable to take on a given value (evaluated in the image space of the random variable):

Definition 1.12 (Probability density function)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a random variable $X : \Omega \to E$ defined on that triple, where the set E equipped with measure μ and $\mathcal{E}$ is a σ–algebra of E. Then, any measurable function $f : \mathcal{E} \to \mathbb{R}_0^+$, which satisfies

$$\Pr(X \in B) \left(= \int\limits_{X^{-1}(B)} d\mathrm{P} \right) = \int\limits_B f \, d\mu \tag{1.15}$$

for any measurable set $B \in \mathcal{E}$ is called a probability density function.

One of the most famous probability density functions induces the so called *Gaussian* random variables.

Definition 1.13 (Gaussian (or normal) distribution)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a random variable $X : \Omega \to E$ defined on that triple, where the set E equipped with measure μ and $\mathcal{E}$ is a σ–algebra of E. Suppose that the parameters $\mu, \sigma \in \mathbb{R}$ with $\sigma > 0$ define the density function

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{(x-\mu)}{2\sigma^2}} . \tag{1.16}$$

of the random variable X. Then X is called a Gaussian random variable, also written $X \in \mathcal{N}(\mu, \sigma^2)$, and f is called Gaussian distribution.

Last, we require some convergence concepts to formally describe what we observed in Figures 1.4 and 1.5. There are several several convergence concepts for different purposes: Some of these concepts are stronger, i.e. exhibit more requirements. The advantage of a strong concept is that, if a convergence can be shown for a method using the strong concept, then we also obtain convergence in the weak one. A schematic illustration of the convergence concepts we consider here is given in Figure 1.8, and their relation is shown in Figure 1.9.

Convergence in distribution is the most weak concept, but it is suffers from a major disadvantage: It is very hard, if not impossible, to show that the required conditions hold:

Definition 1.14 (Convergence in distribution)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$, a measurement vector $z \in \mathbb{R}^N$ and a sequence of random variables $X(N)$, $N \in \mathbb{N}$ and a random variable X, both defined on that triple. The respective probability distribution functions are denoted by f_N and f. Then, we call $X(N)$ **to converge to X in distribution** if

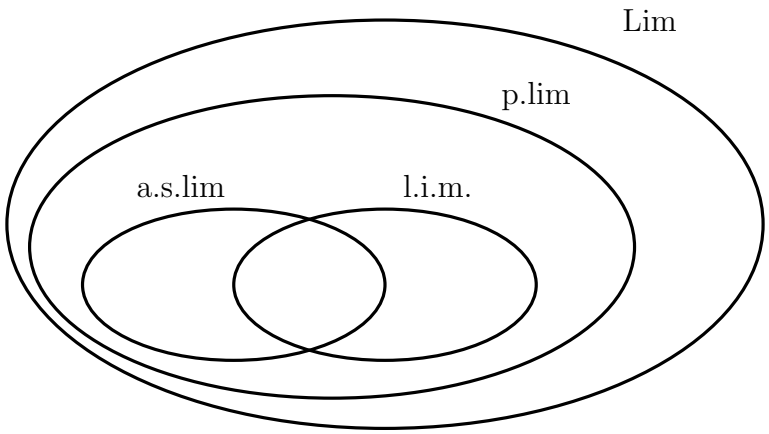


Figure 1.8: Schematic illustration of the convergence areas for stochastic limits.

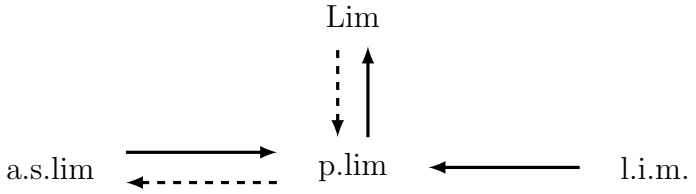


Figure 1.9: Inclusions between stochastic limits.

- $\lim\limits_{N\to\infty} f_N(z,(X(N))(\omega)) \to f\,(z,X(\omega))\ \forall\omega\in\Omega$ where f is continuous.

For short, we write $\operatorname{Lim}\limits_{N\to\infty} X(N) = X$.

Showing this property is particularly hard due to the non–uniqueness of the probability density function, cf. Definition 1.12. Hence, we would have to find a suitable probability density function across the sequence of random variables.

Incorporating the probability function, we obtain a more strict and more easily provable convergence criterion:

Definition 1.15 (Convergence in probability)
Consider a probability space $(\Omega,\mathcal{F},\mathrm{P})$ and a sequence of random variables $X(N)$, $N\in\mathbb{N}$ and a random variable X, both defined on that triple. Then, we call $X(N)$ **to converge to X in probability** if

- $\forall\varepsilon,\delta>0:\ \exists N_0\in\mathbb{N}:\ \mathrm{P}\left(|X(N)-X|\leq\varepsilon\right)>1-\delta\ \ \forall N>N_0$.

For short, we write $\operatorname{p.lim}\limits_{N\to\infty} X(N) = X$.

Using convergence in probability, we need to show existence of bounds N_0 for all pairs ε,δ. Although this is a tricky task, it may be solved using knowledge of the probability function P and of the random variables, which are also functions, cf. Definition 1.8. Hence, this may also be difficult.

Neglecting the probability function P, i.e. impose more restrictions, we can solely focus on the random variables:

Definition 1.16 (Convergence with probability 1)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a sequence of random variables $X(N)$, $N \in \mathbb{N}$ and a random variable X, both defined on that triple. Then, we call $X(N)$ **to converge to** X **with probability** 1 if

- $\lim\limits_{N\to\infty} (X(N))(\omega) = X(\omega)$ for almost all $\omega \in \Omega$.

For short, we write $\underset{N\to\infty}{\text{a.s.lim}}\, X(N) = X$ or $\mathrm{P}\left(\lim\limits_{N\to\infty} X(N) = X\right) = 1$.

For the convergence with probability 1 concept, we still need to check the criterion for almost all $\omega \in \Omega$, which can be done by exploiting properties like continuity etc. of the random variables. Hence, this concept is appropriate for our forthcoming analyses.

Another nice concept is based on distinct properties of the random variables, i.e. of its first and second moment:

Definition 1.17 (Mean square convergence)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a sequence of random variables $X(N)$, $N \in \mathbb{N}$ and a random variable X, both defined on that triple. Then, we call $X(N)$ **to converge to** X **in mean square** if

- $\mathrm{E}\left(|X|^2\right) < \infty$,
- $\mathrm{E}\left(|X(N)|^2\right) < \infty$ for all $N \in \mathbb{N}$, and
- $\lim\limits_{N\to\infty} \mathrm{E}\left(|X(N) - X|^2\right) = 0$.

For short, we write $\underset{N\to\infty}{\text{l.i.m.}}\, X(N) = X$.

Again, this a checkable concept, which we will consider within the identification process.

Within modeling, we will additionally require the concept of a *stochastic differential equation*.

Definition 1.18 (Stochastic differential equation)
Consider deterministic functions $a, b : \mathbb{R} \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$, a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a random variable $X : \mathbb{R} \times \Omega \to \mathbb{R}^{n_x}$ to be given. Then we call

$$\dot{x}(t) = a(t, x(t)) + b(t, x(t))X(t, \cdot) \tag{1.17}$$

a stochastic differential equation.

In contrast to ordinary differential equations defined in Definition 1.1, the introduction of the random variable X causes possibly multiple solutions to exists. Since the realization of the random variable $X(\cdot, \omega)$ is depending on chance, the solution is also depending on chance.

In turn, once the realization $\omega \in \Omega$ is fixed, (1.17) is an ordinary differential equation with a unique solution, i.e. for each realization which is also called a *path*, there exists one solution.

Here, we have a more close look at a specific path, the so called *Wiener process*.

Definition 1.19 (Wiener process)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ and a random variable $W : \mathbb{R} \times \Omega \to \mathbb{R}^{n_x}$ to be given. We call W a Wiener process if the following conditions are satisfied:

1. $W(t, \cdot)$ is a Gaussian random variable with $\mathrm{E}\left(W(t, \cdot)\right) = 0$ and $\sigma^2\left(W(t, \cdot)\right) = t$.
2. For $t_1 \geq t_0 \geq 0$ the *increments* $W(t_1, \cdot) - W(t_0, \cdot)$ are Gaussian random variables with $\mathrm{E}\left(W(t_1, \cdot) - W(t_0, \cdot)\right) = 0$ and $\sigma^2\left(W(t_1, \cdot) - W(t_0, \cdot)\right) = t_1 - t_0$.
3. For $t_3 \geq t_2 \geq t_1 \geq t_0 \geq 0$ the increments $W(t_3, \cdot) - W(t_2, \cdot)$ and $W(t_1, \cdot) - W(t_0, \cdot)$ are Gaussian random variables.

A path $W(t, \omega)$ of W is one of many possible arbitrary functions, which (in the whole) satisfy the conditions above. Indeed, one can show that these paths are almost surely continuous in t, i.e. the event $A = \{\omega \in \Omega \mid X(t, \omega) \text{ is continuous in } t\}$ exhibits probability $\Pr(A) = 1$, and almost surely nowhere differentiable.

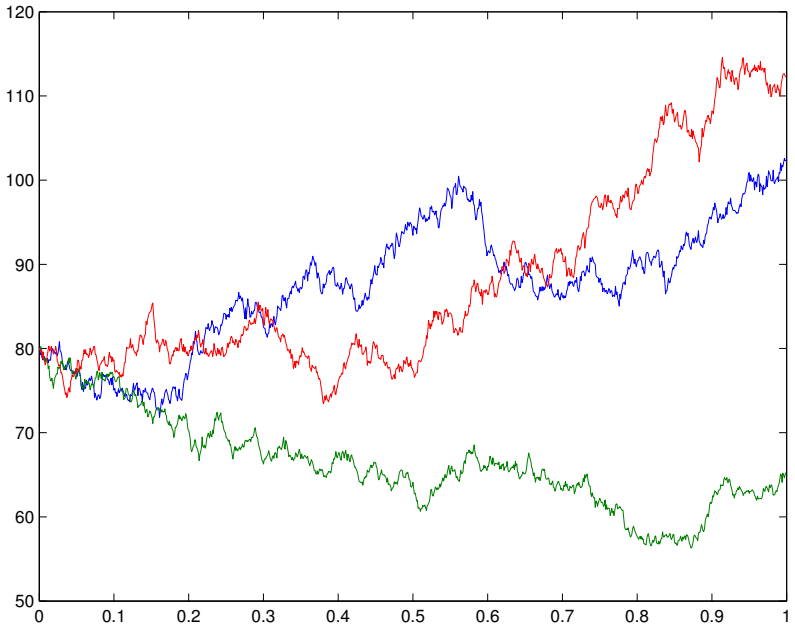


Figure 1.10: Different paths of a Wiener process

Condition 1 in Definition 1.19 states that the spreading of values of paths $W(t, \omega)$ grows larger if t grows larger. The mean, however, stays 0 at all times. Condition 2 reveals that a stochastic process $\tilde{W}(t) := W(t - t_0, \cdot) - W(t_0, \cdot)$ is a Wiener process, i.e. all tails of Wiener processes with translated initial condition 0 are Wiener processes. Last, condition 3 states that from the knowledge of the path for an interval $[t_0, t_1]$, no conclusions for future intervals $[t_2, t_3]$ can be drawn. Hence, a wiener process is memory free, and paths could at any time move upwards and downwards with exactly the same probability, no matter the past development.

Part I

Modeling

Chapter 2

Deterministic processes

Deterministic models contain a number of applications such as growth processes, biological reactions or spread of diseases. These models can also be used in different areas such as market forecast or product displacements. One of the classical systems modeling growth processes is also termed the logistic equation. Within this chapter, we will stick to the classical applications and analyze models with one or several species in more detail, with and without resource limitations.

2.1 Population dynamics for one species

The analysis of growth of one or more species within an ecosystem is called population dynamics. Within this section, we concentrate on the case of one species and analyze this problem in detail.

2.1.1 From Difference to Differential Equation

In general, one first needs to ask whether ordinary differential equations are the right instrument for modeling. Indeed, a differential equation by definition also "lives" on a continuous set. Population dynamics, however, are discrete in nature: The size of a population is usually measured by the number of individuals, which is an natural number. This problem is solved in almost all models by measuring the size of a population by its biomass x instead of the number of individuals. The biomass x is a non negative real number, and we can model its development over time by a differential equation.

The next problem is the right choice of a time axis. Biological measurements are never done continuously for $t \in [t_0, t_1]$, but at discrete instances in time $t_1 < t_2 < \ldots$. The increase or decrease of a population is henceforth given for these discrete time instances. Hence, a general discrete model of a population dynamics for the biomass x is given by

$$x(t_{k+1}) = x(t_k) + \Delta B(t_k) - \Delta D(t_k) + \Delta M(t_k) \tag{2.1}$$

where we use the denotation

$\Delta B(t_k)$	Number of births in the time interval $[t_k, t_{k+1}]$
$\Delta D(t_k)$	Number of deaths in the time interval $[t_k, t_{k+1}]$
$\Delta M(t_k)$	Number of migrations in the time interval $[t_k, t_{k+1}]$

Equations of type (2.1) are called difference equations, and such equation can be used (and are used) to analyze impacts of certain changes. Here, however, we will focus on differential equations and derive a respective model from the difference equation displayed above. The

reason for choosing differential equations is that many analysis tools are available for differential equations, which are either much more involved in the difference equation case, or even don't exist. Regarding modeling, both differential and difference equations can considered to be equal.

To obtain a differential equation from (2.1), we assume that all time instances t_k are equally distributed, i.e. $t_{k+1} - t_k =: \Delta t$ for all $k \in \mathbb{N}$. Hence, we obtain

$$\frac{x(t+\Delta t) - x(t)}{\Delta t} = \frac{\Delta B(t)}{\Delta t} - \frac{\Delta D(t)}{\Delta t} + \frac{\Delta M(t)}{\Delta t}.$$

Note that ΔB, ΔD and ΔM depend on Δt, even if this is not explicitly mentioned in our notation. Letting $\Delta t \to 0$, we obtain

$$\dot{x}(t) = b(t) - d(t) + m(t). \tag{2.2}$$

One could try to derive the functions b, d and m from ΔB, ΔD and ΔM via

$$b(t) = \lim_{\Delta t \to 0} \frac{\Delta B(t)}{\Delta t}, \quad d(t) = \lim_{\Delta t \to 0} \frac{\Delta D(t)}{\Delta t} \quad \text{and} \quad m(t) = \lim_{\Delta t \to 0} \frac{\Delta M(t)}{\Delta t}.$$

Proceeding this way would be a good idea if ΔB, ΔD and ΔM were known. Here, we do not follow this route but instead deduce b and d from model assumptions directly. We will not consider migration, and henceforth set $m \equiv 0$.

2.1.2 Simple growth model

The development of a model is typically done in two steps: First, structural assumptions on the right hand side of the differential equation are made. This means that we fix the vector field f to a special form, which follows from known laws or from heuristic considerations. Within this form, there are certain free parameters. In the second step, these parameters are identified to fit the model to reality. The identification step is done in the second part of this lecture. Here, we focus on the first step, the derivation of a model from structural assumptions.

The most simple model of a population dynamic for one species is given by the following assumptions:

1. The birth rate is linearly proportional to the current size of the population:

$$b(t) = \gamma x(t) \qquad \text{for some } \gamma \in \mathbb{R}$$

2. The death rate is linearly proportional to the current state of the population:

$$d(t) = \sigma x(t) \qquad \text{for some } \sigma \in \mathbb{R}$$

3. There is no migration:

$$m(t) \equiv 0$$

This leads to the differential equation

$$\dot{x}(t) = \lambda x(t) \tag{2.3}$$

where $\lambda = \gamma - \sigma$ represents the growth rate. One can easily see that solutions of (2.3) with initial condition $x(t_0) = x_0$ are given by

$$x(t; t_0 x_0) = x_0 \exp^{\lambda(t-t_0)} .$$

Note that $x(t)$ denotes the size of the population. Hence, we can only allow for $x(t) \geq 0$, and in particular $x_0 \geq 0$. Here and in the following, we use the abbreviation $\mathbb{R}^+ = \{x \in \mathbb{R} \mid x > 0\}$ and $\mathbb{R}_0^+ = \mathbb{R}^+ \cup \{0\}$.

Although this model is very simple, it still describes some growth phenomena pretty well. Figure 2.1 shows the size of the world population between 1950 and 2010 in billions, and a respective solution of (2.3). The values $x_0 = 2.5747$ and $\lambda = 0.0172$ were identified using given data using a linear Least Square Estimator, which we will discuss in detail in Chapter 6. The respective program is shown in Program A.3.

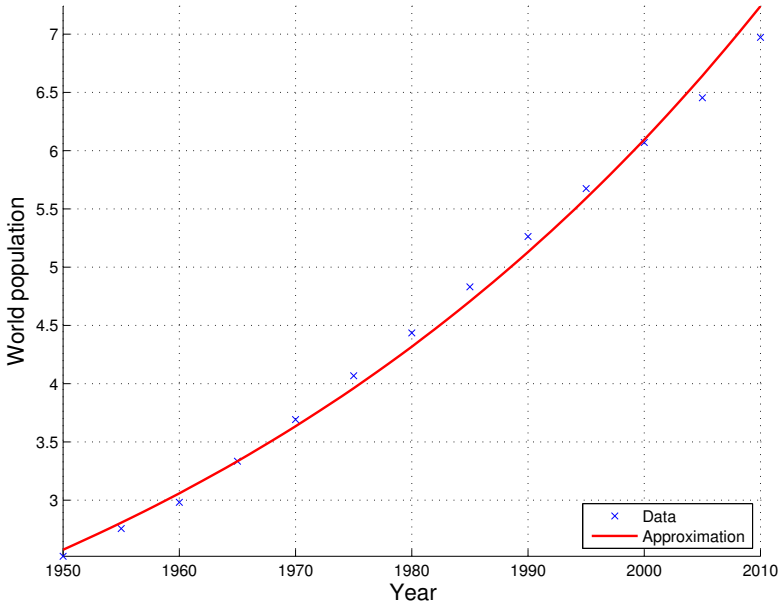


Figure 2.1: Growth of the world population and solution of (2.3) for identified parameters

The development of the world population growth is represented quite well. Other growth processes, however, are not described well by this model. One example for this is the development of the population in Europe, which has stalled throughout the last decades. The reason for this can be seen directly by the structure of the solution: From $\lambda > 0$ we have that $\exp^{\lambda t} \to \infty$ as $t \to \infty$. Hence, for $x_0 > 0$ the population grows exponentially over all bounds. The choice $\lambda < 0$, i.e. more deaths than births, cannot repair this problem. In this case we have $\exp^{\lambda t} \to 0$ as $t \to \infty$, which again does not reflect the current data correctly, cf. Figure 2.2.

2.1.3 Logistic growth model

To model such a slowed down growth, we have to extend equation (2.3) by a growth boundary, which we model by an upper bound $C > 0$ for the size of the population. C represents the capacity of an environment. This capacity is subject to the available resources such as food, water etc. To this end, we incorporate a factor $g(x)$ with the following properties:

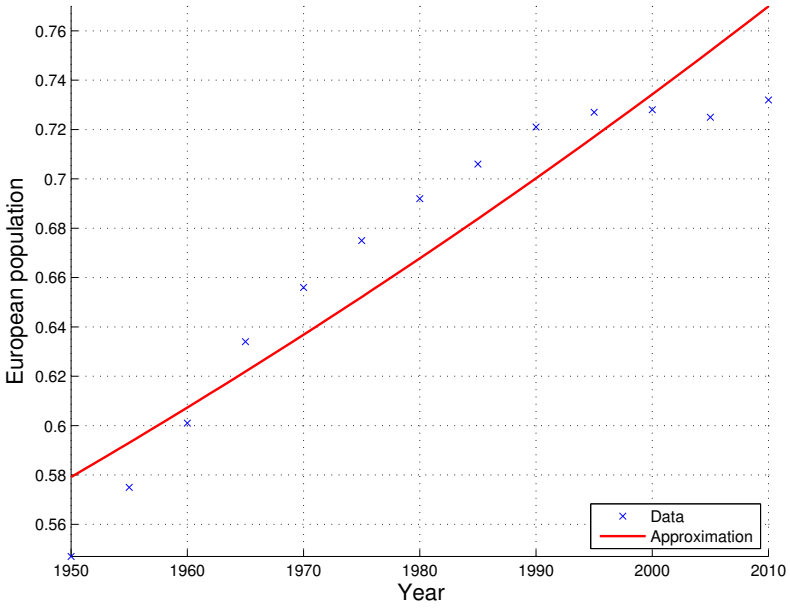


Figure 2.2: Growth of the European population and solution of (2.3) for identified parameters

1. If $x < C$, then we have $g(x) > 0$ reflect available space for growth.
2. If $x > C$, then we have $g(x) < 0$ reflecting negative growth.

The simplest function, which exhibits such a behavior, is the linear function $g(x) = C - x$. Applying this function, we obtain

$$\dot{x}(t) = \lambda\,(C - x(t))\,x(t), \tag{2.4}$$

which is also called the logistics equation. The expression $\lambda\,(C - x)$ is the now nonlinear growth rate. For this differential equation, the explicit solution is known and given by

$$x(t; t_0, x_0) = \frac{C}{1 + \left(\frac{C}{x_0} - 1\right) \exp^{-\lambda C(t - t_0)}}. \tag{2.5}$$

Now, one can analyze the behavior of the solution using this expression. Here, for an exercise, we want to pursue a different approach, and verify our results using the explicit solution. To this end, we first introduce some important terms for differential equations.

Definition 2.1 (Equilibrium)
A point $x^\star \in \mathbb{R}^{n_x}$ is called equilibrium (or fixed point) of a differential equation (1.1) if $x(t; t_0, x^\star) = x^\star$ for all $t, t_0 \in \mathbb{R}$.

One can easily see that a point $x^\star$ is an equilibrium if and only if $f(t, x^\star) = 0$ for all $t \in \mathbb{R}$. For our model (2.4) the zeros of $f(x^\star) = \lambda(C - x)x$ are given by $x^\star = 0$ and $x^+ = C$.

Equilibria are of particular interest due to their potential in analyzing the long term behavior of solutions. Regarding model (2.4) we can see that solutions $x(t; t_0, x_0)$ are growing strictly monotone between the two equilibria, i.e. $\dot{x}(t) > 0$ if $x(t) \in (0, C)$, and $\dot{x}(t) < 0$ if $x(t) > C$.

Since the solutions in positive time are bounded by the equilibrium solution $x(t) = x^+ = C$ and cannot intersect due to uniqueness, cf. Theorem 1.5, they are monotone and bounded, and therefore they converge. Using the following theorem, which is a special case of Barbalat's theorem, we can characterize possible equilibria.

Theorem 2.2 (Equilibrium)
Consider differential equation (1.1) *where f is autonomous. Moreover, the solution $x(t; t_0, x_0)$ converges to a point $x^\star \in \mathbb{R}^{n_x}$ for $t \to \infty$ or $t \to -\infty$. Then $x^\star$ is an equilibrium.*

Proof. Consider the case $t \to \infty$, the case $t \to -\infty$ follows analogously. Since the solution $x(t; t_0, x_0)$ converges to $x^\star$, we have that $f(x(t; t_0, x_0)) \to f(x^\star)$. Now suppose that for given $\varepsilon > 0$ the time $t^* > 0$ is chosen sufficiently large such that

$$\|x(t; t_0, x_0) - x^\star\| \leq \varepsilon \quad \text{and} \quad \|f(x(t; t_0, x_0)) - f(x^\star)\| \leq \varepsilon$$

holds for all $t > t^*$. Then we have that

$$\begin{aligned}\|x(t; t_0, x_0) - x(t^*; t_0, x_0)\| &= \left\| \int_{t^*}^{t} f(x(\tau; t_0, x_0)) d\tau \right\| \\ &\geq \left\| \int_{t^*}^{t} f(x^\star) d\tau \right\| - \left\| \int_{t^*}^{t} f(x(\tau; t_0, x_0)) - f(x^\star) d\tau \right\|\end{aligned}$$

holds for all $t > t^*$. Hence, we can conclude that

$$\begin{aligned}(t - t^*)\|f(x^\star)\| &= \left\| \int_{t^*}^{t} f(x^\star) d\tau \right\| \\ &\leq \|x(t; t_0, x_0) - x(t^*; t_0, x_0)\| + \left\| \int_{t^*}^{t} f(x(\tau; t_0, x_0)) - f(x^\star) d\tau \right\| \\ &\leq \|x(t; t_0, x_0) - x^\star\| + \|x^\star - x(t^*; t_0, x_0)\| + \int_{t^*}^{t} \|f(x(\tau; t_0, x_0)) - f(x^\star)\| \, d\tau \\ &\leq 2\varepsilon + (t - t^*)\varepsilon.\end{aligned}$$

Since the last inequality holds for all $t > t^*$, it also holds for $t = t^* + 1$ which gives us

$$\|f(x^\star)\| < 3\varepsilon.$$

Since $\varepsilon > 0$ was chosen arbitrarily, we can take the limit $\varepsilon \to 0$ and obtain $\|f(x^\star)\| = 0$, i.e. $f(x^\star) = 0$. Hence, $x^\star$ is an equilibrium of the differential equation (1.1). □

An important consequence of Theorem 2.2 is of particular importance for the analysis of differential equations: In the autonomous case equilibria represent all possible limits of solutions.

For our model (2.4) we can conclude via monotonicity that all solutions with $x(t_0) > 0$ converge to $x^+ = C$ for $t \to \infty$. In backwards time we can use an identical monotonicity argument to obtain that all solutions with $x(t_0) \in [0, C)$ converge to 0 for $t \to -\infty$. The solutions with $x(t_0) > C$, however, diverge to $x(t) \to \infty$ for $t \to -\infty$. The reason for that latter is that if the solution was converging, then by Theorem 2.2 another equilibrium $x^\star > C$ would have to exist, which is not the case for our model (2.4).

In the onedimensional case we can use monotonicity to discuss limits of solutions. For higher dimensions this doesn't work in general. Hence, we need other techniques. The basis is the following definition, which describes possible convergence situations for general differential equations in a neighborhood of an equilibrium.

Definition 2.3 (Exponential Stability)
Consider a differential equation (1.1).

1. An equilibrium $x^\star \in \mathbb{R}^{n_x}$ is called (locally) exponentially stable, if there exists a neighborhood $\mathcal{N}$ of $x^\star$ and parameters λ, $\theta > 0$ such that

$$\|x(t; t_0, x_0) - x^\star\| \leq \theta \exp^{-\lambda(t-t_0)} \|x_0 - x^\star\|$$

holds for all $x_0 \in \mathcal{N}$, $t_0 \in \mathbb{R}$ and all $t \geq t_0$.

2. An equilibrium $x^\star \in \mathbb{R}^{n_x}$ is called exponentially unstable, if parameter λ, $\theta > 0$ and a neighborhood $\mathcal{N}$ of $x^\star$ exist such that within each neighborhood $\mathcal{N}_0 \subset \mathcal{N}$ of $x^\star$ there exists a point $x_0 \in \mathcal{N}_0$ which satisfies

$$\|x(t; t_0, x_0) - x^\star\| \geq \theta \exp^{-\lambda(t-t_0)} \|x_0 - x^\star\|$$

for all $t \geq t_0$ for which $x(t; t_0, x_0) \in \mathcal{N}$ holds.

3. An equilibrium $x^\star \in \mathbb{R}^{n_x}$ is called exponentially antistable, if parameter λ, $\theta > 0$ and a neighborhood $\mathcal{N}$ of $x^\star$ exist such that for all $x_0 \in \mathcal{N}$ with $x_0 \neq x^\star$ and all $t_0 \in \mathbb{R}$ the inequality

$$\|x(t; t_0, x_0) - x^\star\| \geq \theta \exp^{-\lambda(t-t_0)} \|x_0 - x^\star\|$$

for all $t \geq t_0$ for which $x(t; t_0, x_0) \in \mathcal{N}$ holds.

Hence, for $t \to \infty$ and Case 1, all solutions from a neighborhood $\mathcal{N}$ of the equilibrium $x^\star$ converge to the equilibrium $x^\star$. In Case 3, all solutions move away from $x^\star$ for growing t, i.e. convergence is not possible. In Case 2 there exist solutions which start arbitrarily close to $x^\star$ but move away from it. However, there may exist initial values $x_0 \neq x^\star$, for which the solution $x(t; t_0, x_0)$ converges to $x^\star$.

Note that Cases 1–3 do not describe all possible scenarios. For example, a function $\beta(\|x_0 - x^\star\|, t)$ may exist, which converges to zero slower than $\theta \exp^{-\lambda(t-t_0)} \|x_0 - x^\star\|$ and that instead of Case 1 the inequality

$$\|x(t; t_0, x_0) - x^\star\| \leq \beta(\|x_0 - x^\star\|, t)$$

holds. The reason for choosing the definition of the (restricted case of) exponential estimates lies in the simplicity of checking these criteria — at least for the case of autonomous differential equations.

Theorem 2.4 (Exponential Stability)
Consider an equilibrium $x^\star \in \mathbb{R}^{n_x}$ of a differential equation (1.1) *with autonomous vector field $f : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$. Suppose f is continuously differentiable in a neighborhood of $x^\star$ and that $Df(x^\star) \in \mathbb{R}^{n_x \times n_x}$ represents the Jacobian of f at $x^\star$. Then the following holds:*

1. *The equilibrium $x^\star$ is (locally) exponentially stable if and only if the real parts of all Eigenvalues $\lambda_i \in \mathbb{C}$ of $Df(x^\star)$ are negative.*

2. *The equilibrium $x^\star$ is exponentially unstable if and only if there exists one Eigenvalue $\lambda_i \in \mathbb{C}$ of $Df(x^\star)$ with positive real part.*

3. *The equilibrium $x^\star$ is exponentially antistable if and only if the real part of all Eigenvalues $\lambda_i \in \mathbb{C}$ of $Df(x^\star)$ are positive.*

Proofs for these results can be found in the book [2]. The Jacobian $Df(x^\star)$ is also often called the *linearization* of (1.1) at $x^\star$.

Now, we want to apply and illustrate this result for our model (2.4) and test, whether it complies with the results from our monotonicity observations. As stated before, we have

$$f(x) = \lambda(C - x)x$$

and the equilibria are given by $x^\star = 0$ and $x^+ = C$. Since the differential equation is onedimensional, the Jacobian of f is real valued. Using the product rule we obtain

$$Df(x) = \lambda(C - x) - \lambda x \quad \Rightarrow \quad Df(x^\star) = \lambda C \text{ and } Df(x^+) = -\lambda C.$$

The Eigenvalues of these 1×1 matrices are given by their real values themselves, i.e we have $\lambda C > 0$ for $x^\star = 0$ and $-\lambda C < 0$ for $x^+ = C$. Hence, the equilibrium $x^\star = 0$ is exponentially antistable and the equilibrium $x^+ = C$ is exponentially stable. This perfectly fits our observations so far (as was to be expected). We can conclude that $x^+ = C$ is a possible limit value of the state $x(t)$ for $t \to \infty$, and $x^\star = 0$ is not such a limit value.

Once a locally exponentially stable equilibrium like $x^+ = C$ for our model (2.4), the next step in the analysis is to compute the set of initial values for which solutions $x(t)$ converge to this equilibrium $x^+ = C$. This is called the *Bassin of attraction*. In general, the bassin of attraction is a locally exponentially stable equilibrium $x^\star$ for an autonomous differential equation is given by

$$\mathcal{D}(x^\star) := \left\{ x_0 \in \mathbb{R}^{n_x} \mid \lim_{t \to \infty} x(t; x_0) = x^\star \right\}.$$

Moreover, since all solutions which move to this neighborhood converge to $x^\star$ according to (1.11) and vice versa all solutions, which converge to $x^\star$ must move to a neighborhood $\mathcal{N}$, we can conclude that for a neighborhood $\mathcal{N}$ from Definition 2.3 we have

$$\mathcal{D}(x^\star) = \{ x_0 \in \mathbb{R}^{n_x} \mid x(t; x_0) \in \mathcal{N} \text{ for some } t \geq 0 \}.$$

In $\mathbb{R}^{n_x}$ the computation of $\mathcal{D}$ is a complicated and often unsolvable task. In the onedimensional case this is much simpler since we can apply monotonicity arguments. Indeed, the bassins of (2.4) are almost completely described in the discussion after Theorem 2.4.

Here, we have seen that all solutions with $x(t_0) > 0$ converge to $x^+ = C$. Hence, we have $\mathcal{D}(x^+) \subset (x^\star, \infty) = (0, \infty)$. Since solutions with $x(t_0) \leq x^\star = 0$ will not converge to $x^+ = C$ as they would have to cross an equilibrium which they cannot leave anymore, we can conclude equality $\mathcal{D}(x^+) = (0, \infty)$.

To summarize the results for our model (2.4), we have the following:

1. There are two equilibria $x^\star = 0$ and $x^+ = C$. The equilibrium $x^+ = C$ is exponentially stable, the equilibrium $x^\star = 0$ is exponentially antistable.

2. Exactly those solutions with initial value $x_0 \in (0, \infty)$ converge to $x^+ = C$.

3. All solutions with initial value $x_0 \in [0, C)$ converge in backwards time to $x^\star = 0$, all solutions with initial value $x_0 > C$ diverge in backwards time to ∞.

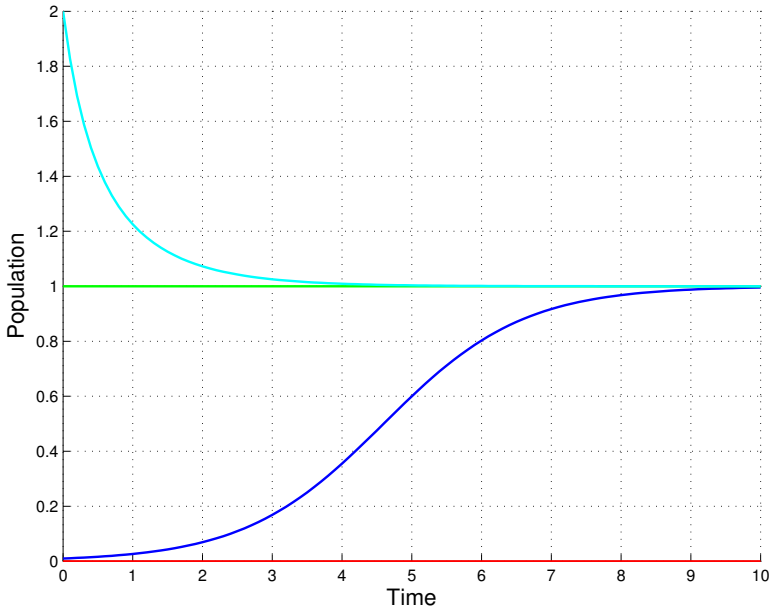


Figure 2.3: Solutions of the logistics equation (2.4)

Note that initial values $x_0 < 0$ make no sense for our model, which is why we don't consider them here.

In Figure 2.3 we display solutions using the explicit formula (2.5) with $C = \lambda = 1$ for initial values $x_0 \in \{0, 0.1, 1, 2\}$. The figure nicely illustrates our results nicely.

Note that also the logistic growth can be adapted to real data of the worldwide population.

Remark 2.5
The logistic growth (2.4) *is not the only model for bounded growth. To model cellular growth also the differential equation*

$$\dot{x}(t) = \lambda x(t) \ln\left(\frac{C}{x(t)}\right) \tag{2.6}$$

is used, the so called Gompertz-Growth which reflects clinical results nicely. For this equation, an explicit solution is unknown. With the methodology displayed above, one can show equivalent solutions properties as for the logistic growth (2.4).

To conclude this section, we observed that the model (2.4) is well suited to describe growth under ideal circumstances and that results for lab experiments can be reproduced nicely. In real applications there are a number of further influencing parameters, which are not included in (2.4):

- Environmental conditions are variable and not constant in general, i.e. influenced by summer and winter. In our model, these are all constant and may require time dependent or stochastic parameters, which we will discuss in the financial processes chapter.
- The spatial distribution of both the population and the resources is not modeled. This could be done using a partial differential equation, which allows a location dependent modeling of population.

- The birth and death rates are directly connected to the size of the population. Factors like age distribution are not considered. These could be included using delay differential equations.
- The impact of other species is not considered, which will be our central concern in the next section.

2.2 Population dynamics for several species

In this section we extend the model (2.3) to the case of several species. To this end, we first focus on the case with two species where the first one represents a food source (prey) for the second species (predator). The case of limited resources (2.4) can be treated similarly.

To extend our model (2.3) to two species, we denote the population of the first species, the prey, by x_1 and the second species, the predators, by x_2. For our model, we make the following assumptions:

1. The prey population x_1 evolves according to (2.3) with $\lambda = \gamma - \sigma$. Here, the birthrate γ is constant and the deathrate is given by $\sigma = \tilde{\sigma} + bx_2$. The deathrate consists of a constant term $\tilde{\sigma} \in (0, \gamma)$ representing the natural deaths, and a proportional term representing the death by predators bx_2. Hence, for $x_2 = 0$ the population x_1 grows exponentially. Here, we set $a = \gamma - \tilde{\sigma}$.
2. The predator population x_2 also evolves according to (2.3) with $\lambda = \gamma - \sigma$. Here, the deathrate σ is constant and the birthrate $\gamma = \tilde{\gamma} + dx_1$ consists of the natural birthrate $\tilde{\gamma} \in (0, \sigma)$ and a proportional term with cofactor $d > 0$. Hence, the birthrate is affin linearly depending on the number of preys x_1. For $x_1 = 0$ the predator population is dieing out since $\sigma > \tilde{\gamma}$. Here, we set $c = \sigma - \tilde{\gamma}$.

Combined, we obtain the two dimensional differential equation

$$\begin{aligned} \dot{x}_1(t) &= \quad ax_1(t) - bx_1(t)x_2(t) \\ \dot{x}_2(t) &= -cx_2(t) + dx_1(t)x_2(t) \end{aligned} \tag{2.7}$$

with parameters $a, b, c, d > 0$. This model is called the *Lotka–Volterra Model*.

For the analysis of (2.7), we first reduce the number of parameters. To this end, we apply the coordinate transformations $x_1 \to \frac{d}{c}x_1$ and $x_2 \to \frac{b}{a}x_2$. This gives us

$$\begin{aligned} \dot{x}_1(t) &= \quad ax_1(t)(1 - x_2(t)) \\ \dot{x}_2(t) &= -cx_2(t)(1 - x_1(t)) \end{aligned} \tag{2.8}$$

Note that the solutions of $\tilde{x}(t; t_0, x_0)$ of (2.7) and $x(t; t_0, x_0)$ of (2.8) are related via

$$x(t; t_0, x_0) = A\tilde{x}(t; t_0, x_0) \quad \text{and} \quad \tilde{x}(t; t_0, x_0) = Ax(t; t_0, x_0) \quad \text{with} \quad A = \begin{pmatrix} \frac{d}{c} & 0 \\ 0 & \frac{b}{a} \end{pmatrix}.$$

Hence, all solutions of (2.7) can be computed from (2.8) and vice versa. For this reason, the two differential equation systems are called equivalent.

For our analysis, we first compute the equilibria of (2.8), i.e. the zeros of the vector field

$$f(x) = \begin{pmatrix} ax_1(t)(1 - x_2(t)) \\ cx_2(t)(1 - x_1(t)) \end{pmatrix}.$$

One can easily see that the points

$$x^\star = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \text{and} \quad x^+ = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

are the only equilibria. To determine the stability property of these equilibria, we compute

$$Df(x^\star) = \begin{pmatrix} a(1-x_2^\star) & -ax_1^\star \\ cx_2^\star & -c(1-x_1^\star) \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & -c \end{pmatrix} \quad \text{and} \quad Df(x^+) = \begin{pmatrix} 0 & -a \\ c & 0 \end{pmatrix}.$$

The Eigenvalues of these matrices are given by a and $-c$ for $x^\star$ and $\pm\sqrt{-ca}$ for x^+. From Theorem 2.4 we can then conclude exponential unstability (not antistability) of $x^\star$. This can be interpreted as follows: For initial values $x_0 = (x_1, 0)^\top$, i.e. no predators, with $x_1 \neq 0$ the solution grows exponentially, i.e. it diverges from $x^\star = 0$. The set of all points which exponentially diverge is called *unstable manifold* $M_u(x^\star)$ of $x^\star$ — in our case this is the subspace $M_u(x^\star) = \{(1,0)^\top\}$. For initial values $x_0 = (0, x_2)$, i.e. no prey, with $x_2 \in \mathbb{R}$, the solutions exponentially converge to $x^\star = 0$. This is the so called *stable manifold* $M_s(x^\star) = \{(0,1)^\top\}$.

In case of x^+ we have that the real parts of the Eigenvalues are 0 due to $ca > 0$. Hence, none of the cases of Theorem 2.4 applies. Therefore, we can conclude that the solutions neither exponentially converge nor diverge. To get a feeling for what happens here, we consider a numerical solution of the system, which is shown in Figure 2.4 for $a = c = 1$.

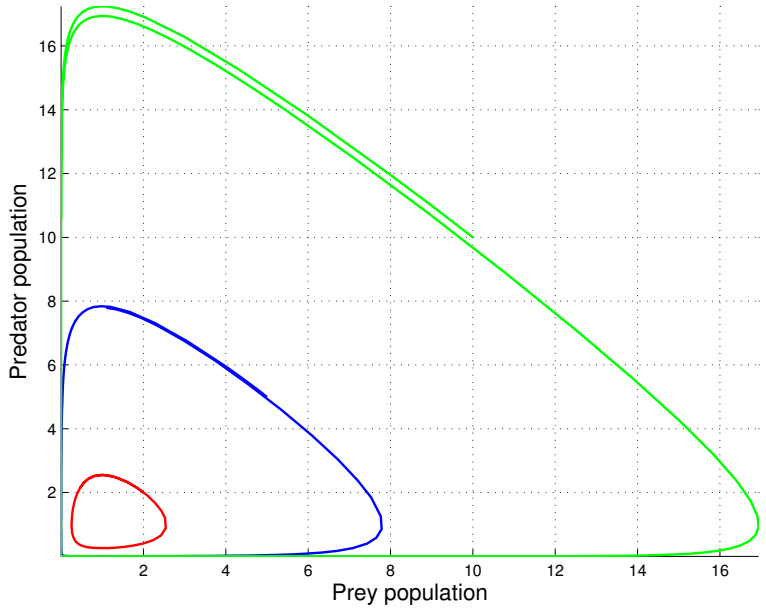


Figure 2.4: Solutions for the predator–prey model (2.8) with $a = c = 1$

From Figure 2.4 we can see why the equilibrium $x^+ = (1,1)^\top$ is neither exponentially stable nor unstable: All solutions, which do not lie on $M_s(x^\star)$ or $M_u(x^\star)$ are moving along periodic orbits around x^+. More formally, we can state the following:

Definition 2.6 (Periodicity)
A solution $x(t; t_0, x_0)$ is called periodic, if there exists a $T > 0$ such that

$$x(t; t_0, x_0) = x(t + T; t_0, x_0)$$

holds for all $t \in \mathbb{R}$. The time T is called the period of the solution.

Note that the solution of an autonomous differential equation is periodic if and only if there exist two time instances $t_1 < t_2 \in \mathbb{R}$ such that $x(t_1) = x(t_2) = x_P$. This follows directly from the identity $x(t) = x(t; t_1, x_P) = x(t; t_2, x_P)$, which gives us $x(t + t_2 - t_1) = x(t)$ for all $t \in \mathbb{R}$, i.e. periodicity for $T = t_2 - t_1$.

For our model (2.8) we want to show the numerical observation of periodicity also rigorously. To this end, we consider the quotient

$$\frac{\dot{x}_2(t)}{\dot{x}_1(t)} = \frac{-cx_2(t)(1 - x_1(t))}{ax_1(t)(1 - x_2(t))}.$$

From this equality it follows that

$$ax_1(t)\dot{x}_2(t) - ax_1(t)x_2(t)\dot{x}_2(t) = -cx_2(t)\dot{x}_1(t) + cx_2(t)x_1(t)\dot{x}_1(t)$$

and hence

$$c\dot{x}_1(t) - c\frac{\dot{x}_1(t)}{x_1(t)} + a\dot{x}_2(t) - a\frac{\dot{x}_2(t)}{x_2(t)} = 0.$$

Note that these equations only hold if all divisors are nonzero, i.e. only for solutions $x(t) \in \mathbb{R}^+ \times \mathbb{R}^+$ and which are no equilibria.

Integrating this equation from 0 to t reveals

$$cx_1(t) - c\ln(x_1(t)) + ax_2(t) - a\ln(x_2(t)) = k(x(0))$$

with $k(x(0)) = cx_1(0) - c\ln(x_1(0)) + ax_2(0) - a\ln(x_2(0))$. Now we define the function $V : \mathbb{R}^+ \times \mathbb{R}^+ \to \mathbb{R}^+$ via

$$V(x) = cx_1 - c\ln(x_1) + ax_2 - a\ln(x_2). \tag{2.9}$$

This function is constant along solutions, i.e. we have

$$V(x(t; t_0, x_0)) = V(x_0) \qquad \text{for all } t \geq t_0$$

and

$$\frac{d}{dt}V(x(t; t_0, x_0)) = 0.$$

The function V is called the *first integral* or *constant of motion* for our model (2.8). The solutions of (2.8) with initial value $x_0 \in \mathbb{R}^+ \times \mathbb{R}^+$ are moving along contour lines $V^{-1}(\ell) := \{x \in \mathbb{R}^+ \times \mathbb{R}^+ \mid V(x) = \ell\}$ of V. We say that a contour line $V^{-1}(\ell)$ is an *invariant set* with respect to (2.8). Note that V exhibits a global minimum at x^+ with $V(x^+) = c + a$.
To conclude periodicity, we divide the contour lines in four segments

$$\begin{aligned}
S_1 &= \{x \in V^{-1}(\ell) \mid x_1 \leq x_2 \leq 2 - x_1\} \\
S_2 &= \{x \in V^{-1}(\ell) \mid x_2 \leq x_1 \leq 2 - x_2\} \\
S_3 &= \{x \in V^{-1}(\ell) \mid x_1 \geq x_2 \geq 2 - x_1\} \\
S_4 &= \{x \in V^{-1}(\ell) \mid x_2 \geq x_1 \geq 2 - x_2\}
\end{aligned}$$

From the form of the contour lines it follows that there exists $\alpha > 0$ such that $|x_1 - 1| \geq \alpha$ for all $x \in S_1$ and $x \in S_3$, and $|x_2 - 1| \geq \alpha$ for all $x \in S_2$ and $x \in S_4$. Moreover, we have that there exists $\beta > 0$ such that $x_1 > \beta$ and $x_2 > \beta$ for all $x \in V^{-1}(\ell)$. Hence, from (2.8) we obtain

$$\begin{aligned}
\dot{x}_2(t) &< -c\beta\alpha, && \text{if } x(t) \in S_1 \\
\dot{x}_1(t) &> \quad a\beta\alpha, && \text{if } x(t) \in S_2 \\
\dot{x}_2(t) &> \quad c\beta\alpha, && \text{if } x(t) \in S_3 \\
\dot{x}_1(t) &< -a\beta\alpha, && \text{if } x(t) \in S_4.
\end{aligned}$$

In each sector one of the components is strictly monotone increasing or decreasing. Therefore, the solution must leave each sector in finite time in the sequence $S_1 \to S_2 \to S_3 \to S_4 \to S_1$. Therefore, the solutions are indeed periodic.

To interpret the solutions of a model, we need to consider them depending on the time component t. An exemplary solution is given in Figure 2.5. We can see that both populations

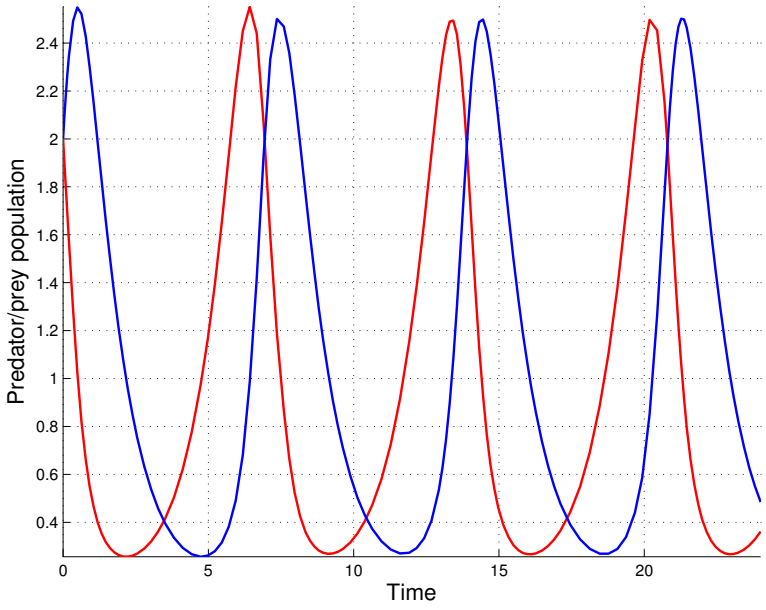


Figure 2.5: Time to state plot for the predator–prey model (2.8) with $a = c = 1$ and initial value $x_0 = (2, 2)^\top$

are oscillating periodically. If (like at the beginning) many predators and many preys are present, the number of the predators increases while the number of preys reduces up to a certain point until both populations are decaying. Once the number predators is sufficiently small, the number of preys is increasing again, and once the number of preys is large enough then also the number of predators starts to increase. Such a periodic behavior can also be spotted in reality.

2.2.1 Predator–prey model with limited resources

Since we generalized our model (2.3) to the two species case, the resulting model (2.7) possesses the unrealistic property that the prey population can grow unbounded if no predators are

present. Similar to the one species case, we switch to the more realistic model (2.4). For simplicity of notation in the two species case, we set $\mu = \lambda C$ and $e = \lambda$ and obtain

$$\dot{x}_1(t) = \mu x_1(t) - e x_1(t)^2. \tag{2.10}$$

Hence, we modify our model assumption 1 as follows:

1'. The prey population x_1 evolves according to (2.10) with $\mu = \gamma - \sigma$ and $e > 0$. Here, the birthrate γ and the bounding rate e are constant and the deathrate is given by $\sigma = \tilde{\sigma} + bx_2$. There only exist bounded resources for the prey and the deathrate consists of a constant term $\tilde{\sigma} \in (0, \gamma)$ representing the natural deaths, and a proportional term representing the death by predators bx_2. Hence, for $x_2 = 0$ the population x_1 approaches $C = a/e$ with $a = \gamma - \tilde{\sigma}$.

Hence, we obtain the equation system

$$\begin{aligned} \dot{x}_1(t) = &\quad a x_1(t) - b x_1(t) x_2(t) - e x_1(t)^2 \\ \dot{x}_2(t) = &-c x_2(t) + d x_1(t) x_2(t) \end{aligned} \tag{2.11}$$

with parameters a, b, c, d and $e > 0$. Similar to (2.7), we first reduce the number of parameters. To this end, we apply the coordinate transformations $x_1 \to \frac{d}{c} x_1$ and $x_2 \to \frac{bd}{da-ec} x_2$. This gives us

$$\begin{aligned} \dot{x}_1(t) = &\quad \alpha x_1(t)(1 - x_2(t)) + \beta x_1(t)(1 - x_1(t)) \\ \dot{x}_2(t) = &-c x_2(t)(1 - x_1(t)) \end{aligned} \tag{2.12}$$

with $\alpha = a - ec/d$ and $\beta = ec/d$. Here, we have to be careful that positive x_1 and x_2 are mapped on positive values. Since a, b, c, d and $e > 0$ this is the case if and only if $\frac{bd}{da-ec} > 0$, i.e. if $da > ec$.

For $da \leq ec$ one can show that the predator population is going to die out for $t \to \infty$. Here, we want to treat the more interesting case of two coexisting species. Henceforth $da > ec$, which is a necessary condition for the respective setting.

We now obtain the equilibria $x^\star = (0,0)^\top$, $x^{\star\star} = ((\alpha+\beta)/\beta, 0)^\top$ and $x^+ = (1,1)^\top$. Only x^+ is an element of $\mathbb{R}^+ \times \mathbb{R}^+$, for the other equilibria the populations of the predators or both species are zero.

The linearization of the dynamic reveals

$$Df(x) = \begin{pmatrix} \alpha(1 - x_2) + \beta(1 - 2x_1) & -\alpha x_1 \\ c x_2 & -c(1 - x_1) \end{pmatrix},$$

which gives us

$$Df(x^+) = \begin{pmatrix} -\beta & -\alpha \\ c & 0 \end{pmatrix}.$$

The Eigenvalues of this matrix are

$$\lambda_{1/2} = -\frac{\beta}{2} \pm \sqrt{\frac{\beta^2}{4} - ca}.$$

If the root is complex, then the real part $-\beta/2$ is negative. If the root is real valued, then $\lambda_{1/2}$ is real valued and we have

$$\lambda_{1/2} \leq -\frac{\beta}{2} + \sqrt{\frac{\beta^2}{4} - ca} < \frac{\beta}{2} + \sqrt{\frac{\beta^2}{4}} = 0,$$

i.e. in either case we obtain negative real parts. Therefore, x^+ is locally exponentially stable.

Now we know that there exists a neighborhood of x^+ such that all solutions within this neighborhood are converging to x^+. To compute this bassin $\mathcal{D}(x^+)$, we utilize the first integral given by

$$V(x) = cx_1 - c\ln(x_1) + ax_2 - a\ln(x_2). \tag{2.9}$$

In contrast to (2.7), this function is not constant along solutions of (2.12). Instead, for each solution $x(t)$ the following holds:

$$\begin{aligned}
\frac{d}{dt}V(x(t)) &= c\dot{x}_1(t) - c\frac{\dot{x}_1(t)}{x_1(t)} + \alpha\dot{x}_2(t) - \alpha\frac{\dot{x}_2(t)}{x_2(t)} \\
&= (c\alpha x_1(t)(1 - x_2(t) + c\beta x_1(t)(1 - x_1(t)))\left(1 - \frac{1}{x_1(t)}\right) \\
&\quad - \alpha c x_2(t)(1 - x_1(t))\left(1 - \frac{1}{x_2(t)}\right) \\
&= c\beta(1 - x_1(t))(x_1(t) - 1) \\
&= -c\beta(x_1(t) - 1)^2
\end{aligned}$$

Hence, the function $V(x(t))$ is monotone decreasing, for $x_1(t) \neq 1$ even strictly. Note that $V(x)$ exhibits a global minimum at $x = x^+$, and there exist no further local minima. In *Stability Theory*, such a function is called a *Lyapunov function*. Here, we face the special case of a semidefinite Lyapunov function, since its derivative is not strictly decreasing along a solution, but instead we only have ≤ 0.

Next we show that $x(t) \to x^+$ for $t \to \infty$. Since $V(x(t))$ is monotone decreasing and bounded from below, we know that $V(x(t))$ converges to a value V_∞. Similar to the proof of Theorem 2.2 we see that $\frac{d}{dt}V(x(t)) \to 0$ for $t \to \infty$. Hence, $x_1(t) \to 1$ for $t \to \infty$ must hold. The latter is only possible if $x_2(t) \to 1$. To see that, consider $|x_2(t) - 1| \geq \delta$. Hence, by (2.12) we obtain for $x_1(t)$ that $\dot{x}_1(t) > \varepsilon$ or $\dot{x}_1(t) < -\varepsilon$ holds for a neighborhood of 1. This contradicts convergence of $x_1(t) \to 1$. Hence, $x_2(t) \to 1$ for $t \to \infty$ and therefore also $x(t) \to x^+$ for $t \to \infty$. We can conclude that all solutions for the definition set of the Lyapunov function $V(x)$ converge to x^+, and that the bassin is given by $\mathcal{D}(x^+) = \mathbb{R}^+ \times \mathbb{R}^+$, which is also exemplary verified in Figure 2.6. The argumentation used here is also called *Lasalle's principle of invariance*.

In order to interpret the model, the time plot of a solution is useful, cf. Figure 2.7. The solution does show oscillations similar to Figure 2.5, but the solutions converge for growing t towards the equilibrium x^+. Such equilibria between two coexisting species are regularly observed in nature, and also the oscillations are know if the system is "pushed" out of its equilibrium.

2.2.2 Generalization to multiple species

We can generalize the model (2.11) to n different species x_1 to x_n. If we consider identical model assumptions for all species, i.e. dynamic (2.10), where the growth rate μ depends affine linearly on the other species, we obtain the model

$$\dot{x}_i(t) = k_i x_i(t) + b_i^{-1}\sum_{j=1}^{n} a_{ij}x_i(t)x_j(t), \qquad i = 1, \ldots, n \tag{2.13}$$

with $k_i \neq 0$, $a_{ii} \leq 0$ and $b_i > 0$. Via a_{ij} we define the matrix A. The coefficient a_{ii} corresponds to the factor e from (2.10) and models the limited resources. The values a_{ij} with $i \neq j$ represent

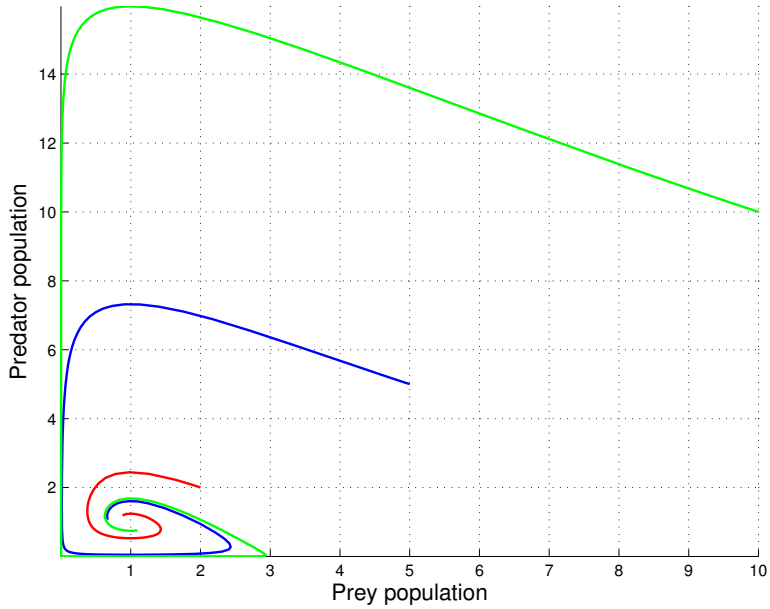


Figure 2.6: Solutions for the predator–prey model (2.12) with $a = c = 1$ and $\beta = 0.5$

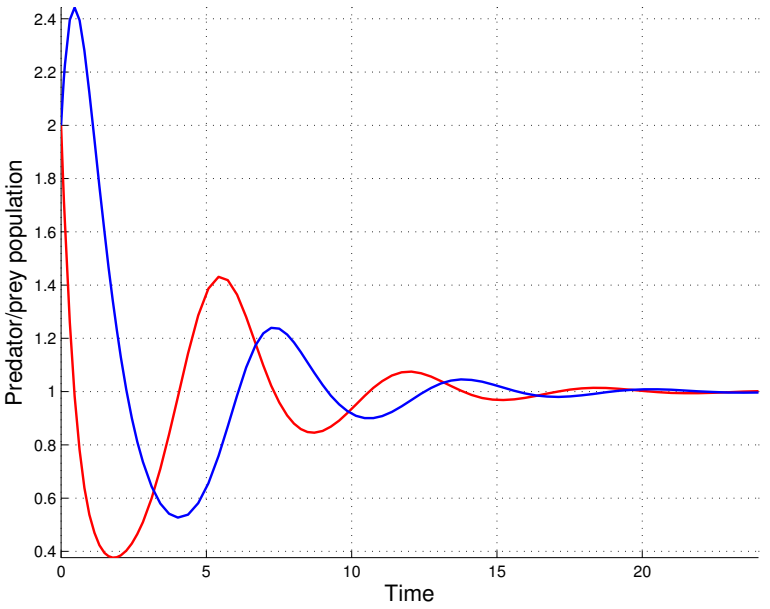


Figure 2.7: Time to state plot for the predator–prey model (2.12) with $a = c = 1$ and initial value $x_0 = (2, 2)^\top$

the interaction of species. For prey x_i and predator x_j we require $a_{ij} < 0$ and $a_{ji} > 0$. The strange notation b_i^{-1} is due to the original, a little bit different denotation of the model. Note that models (2.4) and (2.11) are special cases of this model.

The special case $a_{ii} = 0$ and $a_{ij} = -a_{ji}$ is called a *Volterra ecology*. In that case, the matrix $A = (a_{ij})$ is anti–symmetric, i.e. $x^\top Ax = 0$ for all $x \in \mathbb{R}^{n_x}$.

Similar to the previous section we are interested in equilibria x^+, for which all species coexist, i.e. $x_i^+ > 0$. Here, we obtain

$$k_i x_i^+ + b_i^{-1} \sum_{j=1}^{n} a_{ij} x_i^+ x_j^+ = 0 \qquad \Longrightarrow \qquad b_i k_i + \sum_{j=1}^{n} a_{ij} x_j^+ = 0.$$

Therefore, the equilibria are given as solutions of a linear equation system. If A is invertible, at most one such equilibrium exists. Note that in the latter case there exists exactly one solution to the linear equation system, but this solutions not necessarily satisfies $x_i^+ > 0$.

The construction of the first integral can be generalized for this model. Suppose an equilibrium x^+ with $x_i^+ > 0$, $i = 1, \ldots, n$ exists. Then the function

$$V(x) = \sum_{i=1}^{n} b(x_i - x_i^+ \ln(x_i)))$$

satisfies the equation

$$\frac{d}{dt} V(x(t)) = (x(t) - x^+)^\top A (x(t) - x^+).$$

If A is negative semidefinite, then the derivative is negative semidefinite and we can generalize the argumentation from the two species case to the n–dimensional model. For the Volterra ecology, A is anti–symmetric, which gives us $\frac{d}{dt} V(x(t)) = 0$. Again, we will obtain similar periodic phenomena.

Chapter 3

Mechanical processes

The mathematical foundations of modeling in classical mechanics were given by the works of Isaac Newton[1], Jean Baptiste le Rond d'Alembert[2], Joseph–Louis Lagrange[3] and William R. Hamilton[4]. Newton developed the elementary equations of motion (and by that the differential equation itself). Lagrange and Hamilton invented continuing methods for modeling and analysis, which we will discuss in Section 3.2.

3.1 Technical elements

Within this first section, we will introduce an approach which is know as *d'Alembert Principle*. It represents a modularization and combination of mechanical systems. Each of the modules (or elements) is described by a graphical symbol and a respective equation of motion, which not always corresponds to a differential equation. Within our models and formulas, we will use the denotation given in Figure 3.1.

Variable	Meaning	Unit	
m	Mass	kg	[kilogramm]
h	Height	m	[meter]
g	Gravitation	m/s^2	[meter per second square]
E	Energy	$kg\,m^2/s^2$	[Joule]

Table 3.1: Denomination for technical elements and models

One distinguishes between two different kinds of motion, which we will discuss in the following. The approach itself is constructive an — in principle — allows us to model arbitrarily complex mechanical system at very low mathematical costs. Yet, the approach is impracticable for complex system. To cope with this issue, we discuss mathematically more sophisticated methods later.

3.1.1 Translational models

Here, we consider elements of motion, which allow for a movement along a stright line, i.e. a one–dimensional movement. We will use the denomination displayed in Table 3.2.

[1] English mathematician and physicist, 1642 – 1727
[2] French mathematician and physicist, 1717 – 1783
[3] French mathematician, 1736 – 1813
[4] Irish mathematician, 1805 – 1865

Variable	Meaning	Unit	
y	Location, dilation	m	[meter]
v	Velocity	m/s	[meter per second]
a	Acceleration	m/s^2	[meter per second square]
F	Force	$N = kg\,m/s^2$	[Newton]

Table 3.2: Denomination for translational models

Mass element

A mass element consists of a (constant in time) mass m, a force F applied to this mass and the velocity v of the mass. The symbol for a mass element is depicted in Figure 3.1.

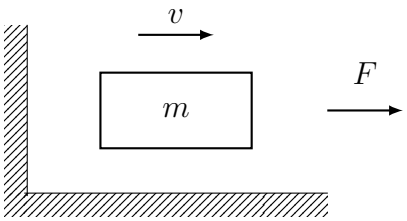


Figure 3.1: Symbol for a mass element

Utilizing Newton's second law, the differential equation for the mass element is given by

$$F(t) = ma(t) = m\dot{v}(t). \tag{3.1}$$

Note that the force F and the velocity v have to point into the same direction. Otherwise, we have to replace F by $-F$, which is a popular source for sign errors.

There are different sources of energy that are stored within a mass, the kinetic and potential energy.

- If a mass is in motion, then its kinetic energy is given by

$$E_k(t) = \frac{m}{2}v(t)^2.$$

- If a mass is caught in a gravity field, then its potential energy is given by

$$E_p(t) = mgh(t).$$

Spring element

The spring (or more generally the elasticity) element is a deformable object, for which the dilation y is a function of the applied force F. The symbol for a mass element is given in Figure 3.2.

For the ansatz of a linear model we use Hook's law to describe the spring element. Hence, we have

$$sy(t) = F(t) \tag{3.2}$$

where $y = y_2 - y_1$ is the dilation of the spring and $s > 0$ the spring constant. By convention, y_2 is the point of action in positive direction, and y_1 for negative direction.

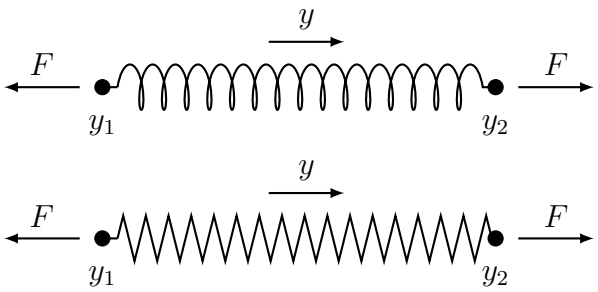


Figure 3.2: Symbols for a spring element

This model describes a real life spring sufficiently well for small dilations. For more realistic models, a nonlinear mapping between F and y is applied, which we will not cover here. Independent from the modeling of this mapping, pure spring elements are an idealization by themselves. In reality, there exists no spring without mass and damper. Note that for $y = 0$, the spring is in a position of rest, hence the dilation can be either positive or negative within this model.

Similar to mass elements, also spring elements can store potential energy. If equation (3.2) is supposed to hold, then this energy is given by

$$E_p(t) = \frac{s}{2} y(t)^2.$$

Damper element

A damper or damping element is a mechanical element, which cannot store energy, but instead converts the received energy into heat and releases the latter. This is referred to as a dissipator. The symbol for a damper element is given in Figure 3.3.

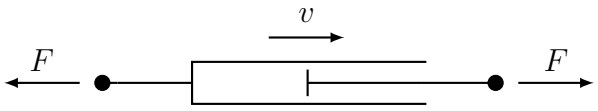


Figure 3.3: Symbol for a damper element

Again, we consider the linear model given by

$$F(t) = d\dot{v}(t), \tag{3.3}$$

where v is the relative velocity of the body (which corresponds to the piston in the cylinder), F the attacking force and $d > 0$ the damping constant. If a force F is applied, then the velocity dv will be reached. The relative velocity v is computed via $v = v_+ - v_-$, where v_+ denotes the velocity of the terminal point in positive direction, and v_- the velocity of the terminal point in negative direction.
This model is also called *viscosity model* or *viscous friction*. Other models are given by, e.g., *dry friction* or *drag/air resistance*. In the first case, the force F is increasing for slower velocities, in the latter the force quadratically depends on the velocity via $F = dv|v|$. Even more complex connections arise in the case of *stiction*, which cannot be modeled by a classical function, but required hysteresis models instead.

The absorbed energy of a damping element at time t is the product $F(t)v(t)$. Hence, in the time interval $[t_0, t_1]$, a damping element absorbs the energy given by the integral over the power, i.e.

$$E_a = \int_{t_0}^{t_1} F(t)v(t)dt.$$

3.1.2 Rotational models

So far, we consider elements of motion, which allow for a movement along a stright line. In the following, we will introduce three analog elements for rotations. We will use the denomination displayed in Table 3.3.

Variable	Meaning	Unit	
θ	Angle	rad	[radiant]
ω	Angular velocity	rad/s	[radiant per second]
α	Angular acceleration	rad/s^2	[radiant per second square]
τ	Torque	Nm	[Newton meter]
J	Moment of inertia	kgm^2	[kilogramm meter square]

Table 3.3: Denomination for rotational models

The torque describes the force, which is applied to a rotating body: Consider $F = (F_1, F_2, 0)$ to be a directed force and a body, which is rotating around the x_3 axis. The force is applied at the body at point $x = (x_1, x_2, 0)$ as illustrated in Figure 3.4. The vector x can be interpreted

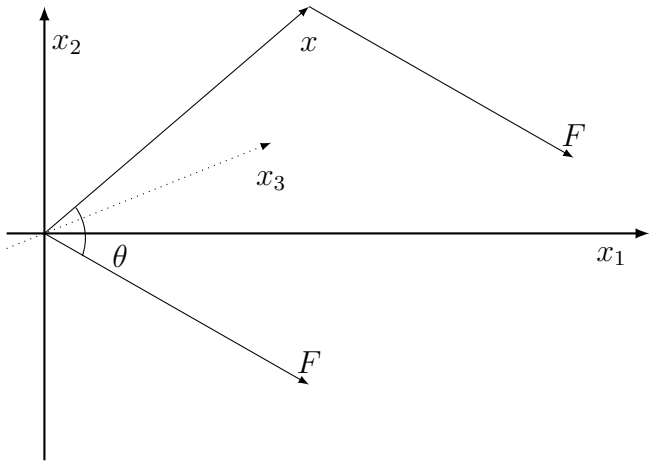


Figure 3.4: Schematic illustration of torque

as a leverage of the body. The resulting torque is given by

$$\tau = x_1F_2 - x_2F_1 = \|x\|\|F\|\sin(\theta), \tag{3.4}$$

where θ is the angle between x and F. Again, the sign is important. Positive direction must be chosen such that both expressions in (3.4) coincide.

Note that the force F is now a vector in a coordinate system. In contrast to the translational models, the information regarding direction is contained in F, hence we can compute contact forces without having to take care of directions.

Mass element

The mass element for rotations consists of a mass, which is rotating around an axis. The respective formula is given by

$$\tau(t) = J\alpha(t) = J\dot{\omega}(t) \tag{3.5}$$

where J represents the moment of inertia, which is given by the mass of the object and its distribution around the rotation axis. Figure 3.5 give the symbol for the rotational mass element.

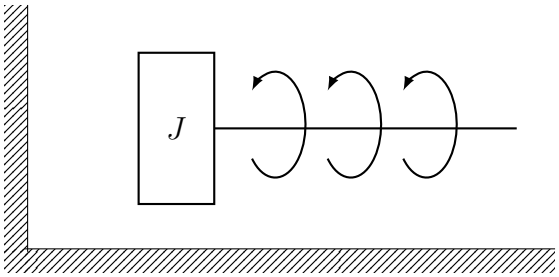


Figure 3.5: Schematic illustration of rotational mass element

For a rotating body $B \subset \mathbb{R}^3$ with mass m and density $\rho : B \to \mathbb{R}_0^+$ we have

$$J = \int_B r(x)^2 \rho(x)\, dx$$

where $r(x)$ is the distance of x to the rotation axis.

In special cases, a closed formula is known. A rotating point mass with mass m and distance r to the rotation axis possesses the moment of inertia

$$J = mr^2.$$

In general, the Parallel Axis Theorem (also know as Steiner's Theorem) holds:

Theorem 3.1 (Parallel Axis Theorem)
Consider a body $B \subset \mathbb{R}^3$ of mass m with density distribution $\rho : B \to \mathbb{R}_0^+$ and point of mass

$$\overline{x} = \frac{1}{m} \int_B x\rho(x)\, dx \in \mathbb{R}^3.$$

Let J be the moment of inertia of the body around an arbitrary axis, and J' be the moment of inertia of the body around a parallel axis containing the point of mass. Then the equality

$$J = J' + mR^2$$

holds where R denote the distance between the axis.

Spring/torsion element

The spring and the following damper element are completely analog to their translational counterparts. Similarly, we consider the linear models only. For the rotational spring element

the equation reads

$$s\theta(t) = \tau(t). \tag{3.6}$$

Figure 3.6 gives the respective symbol.

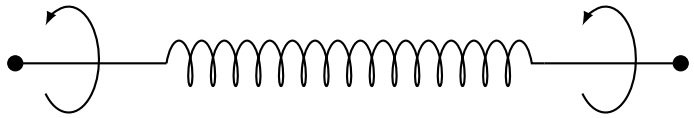


Figure 3.6: Symbols for a rotational spring element

Damper element

For the damper element, the following equation

$$d\alpha(t) = \tau(t) \tag{3.7}$$

holds and the symbol of the damper element is given in 3.7

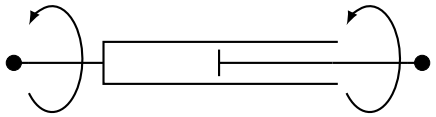


Figure 3.7: Symbol for a rotational damper element

3.1.3 Building complex models

In the previous sections, we discussed basic modules for mechanical systems. The ansatz to build more complex system is given by the following procedure:

1. Model the mechanical system using mass, spring and damping elements
2. Prepare the respective equations of motion
3. Formulate the connecting laws / contact forces

The basis for this ansatz is given by Newton's 3rd Law *actio = reactio*: In each mass, the sum of forces is zero. If additionally an external force is present, then the sum of internal forces is equal to the external force. Note that the direction of the force needs to be taken into account using a respective sign.

Here, we will exemplary discuss how this procedure works using a simple quarter car model.

Example 3.2 (Quarter Car Model)
For our model depicted in Figure 3.8, we make the following assumptions:

- *We consider vertical movements only.*
- *We model one wheel only.*

- *The chassis is modeled as mass m_1 at position y_1, the suspension is modeled using a spring and a damper element s_1, d_1.*
- *The wheel and the axis are modeled as mass m_2 at position y_2, the wheel is modeled using a spring and a damper element s_2, d_2.*
- *Road undulations are modeled via the road height function $u(t)$.*

From the equations of motion, we obtain the individual forces

$$m_i \dot{v}_i^m(t) = F_i^m(t), \quad d_i \dot{v}_i^d(t) = F_i^d(t), \quad s_i y_i^s(t) = F_i^s(t)$$

for $i = 1, 2$ where we used

$$\begin{aligned} v_1^m(t) &= \dot{y}_1(t), & \dot{v}_1^d(t) &= \dot{v}_1(t) - \dot{v}_2(t), & v_1^s(t) &= y_1(t) - y_2(t), \\ v_2^m(t) &= \dot{y}_2(t), & \dot{v}_2^d(t) &= \dot{v}_2(t) - \dot{u}(t), & v_2^s(t) &= y_2(t) - u(t). \end{aligned}$$

To combine the equations, we need to describe the forces in all masses. To this end, the direction of the forces has to be treated carefully. In m_1, the force F_1^m points into the upwards direction: Since m_1 is the upper end of the attached spring and damper, F_1^d and F_1^s also point upwards. Hence, in m_1 we obtain

$$F_1^m + F_1^d + F_1^s = 0.$$

In m_2, forces F_1^d, F_1^s point downwards, all other forces upwards and we obtain

$$F_2^m - F_1^d - F_1^s + F_2^d + F_2^s = 0.$$

Combined, we have

$$\begin{aligned} 0 &= F_1^m + F_1^d + F_1^s \\ &= m_1 \dot{v}_1^m(t) + d_1 \dot{v}_1^d(t) + s_1 y_1^s(t) \\ &= m_1 \ddot{y}_1(t) + d_1(\ddot{y}_1(t) - \ddot{y}_2(t)) + s_1(y_1(t) - y_2(t)) \end{aligned}$$

and

$$\begin{aligned} 0 &= F_2^m - F_1^d - F_1^s + F_2^d + F_2^s \\ &= m_2 \dot{v}_2^m(t) - d_1 \dot{v}_1^d(t) - s_1 y_1^s(t) + d_2 \dot{v}_2^d(t) + s_2 y_2^s(t) \\ &= m_2 \ddot{y}_2(t) - d_1(\dot{y}_1(t) - \dot{y}_2(t)) - s_1(y_1(t) - y_2(t)) + d_2(\dot{y}_2(t) - \dot{u}(t)) + s_2(y_2(t) - u(t)). \end{aligned}$$

These equations display two second order differential equations and can be reformulated as a system of four first order differential equations.

Example 3.3 (Pendulum)
In this example, we utilize rotational elements to generate a model of a pendulum. Here, we impose the following assumptions:

- *The pendulum is a point mass m which is mounted on a massless rod of length ℓ.*

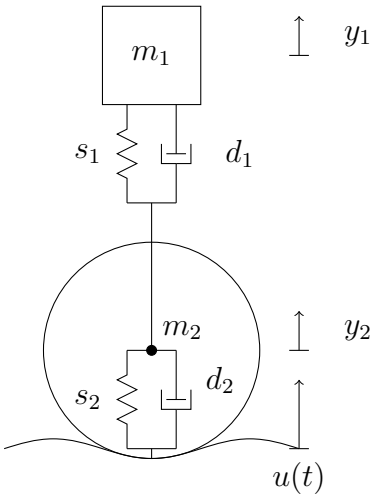


Figure 3.8: Schematic drawing of a quarter car test bench

- *There is no friction.*

A schematic sketch of the model is given in Figure 3.9. Let $x(t) = (x_1(t), x_2(t))^\top$ *be the endpoint of the pendulum. The rotation axis is located at* x_A*, and we set* $x_A = 0$*. As usual, the coordinates* x_1*,* x_2 *are increasing rightwards and upwards respectively. The point* $x(t)$ *can be calculated from the length* ℓ *and the angle* $\theta(t)$ *via*

$$x(t) = (\ell \sin(\theta(t)), -\ell \cos(\theta(t)))^\top .$$

Due to earth's gravitation, the force F *acting in* $x(t)$ *is given by* $F = (0, -mg)^\top$*. Utilizing* (3.4) *we obtain the torque*

$$\tau_F(t) = x_1(t) \cdot (-mg) + x_2(t) \cdot 0 = -mgx_1(t) = -mg\ell \sin(\theta(t)).$$

Moreover, for the mass element we obtain from equation (3.5)

$$\tau_J(t) = J\ddot{\theta}(t) = m\ell^2\ddot{\theta}(t).$$

Setting $\tau_F = \tau_J$*, we get*

$$m\ell^2\ddot{\theta}(t) = -mg\ell \sin(\theta(t)),$$

which gives a second order differential equation. Via $\omega(t) = \dot{\theta}(t)$*, we arrive at the system of first order differential equations*

$$\begin{aligned} \dot{\theta}(t) &= \omega(t) \\ \dot{\omega}(t) &= -\frac{g}{\ell} \sin(\theta(t)). \end{aligned}$$

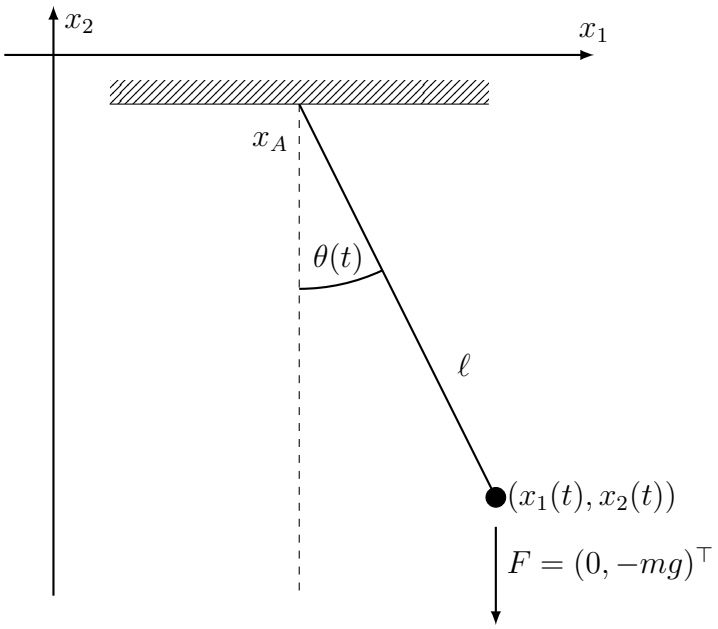


Figure 3.9: Schematic drawing of a pendulum

3.2 Lagrange–Equations

Within the last section we discussed a method to combine basic translational and rotational models with their forces. For large systems, this procedure is rather complex. The reason lies in the number of connection laws and contact forces for many points, each resulting in a single equation. This leads to large equation systems, which are difficult to solve.

An alternative is the so called *energy based* method using *Lagrange–Equations*.

The idea of the Lagrange–Equations utilizes the energy of a system. We restrict ourselves to the case of a system with n points of mass m_i at locations $r_i = (x_i, y_i, z_i)^\top$, $i = 1, \ldots, n$. The kinetic energy of this system is given by

$$E_k = \sum_{i=1}^{n} \frac{m_i}{2} \|v_i\|^2.$$

The mechanical structure with its J connections induces constraints of the form

$$C_n(r_1, \ldots, r_n, t) = 0 \qquad \forall n = 1, \ldots, J,$$

where $r_i = (x_i, y_i, z_i)^\top \in \mathbb{R}^3$ marks the positions of the points of mass.

Example 3.4
Consider a pendulum fixed at the origin with point mass m at point $r(t) = (x(t), y(t), z(t))^\top$ of length ℓ, which is swinging in the $x - y$ plain. All possible positions of $r(t)$ are the given by the equation

$$C_1 = \|r\|^2 - \ell^2 \quad \textit{and} \quad C_2(r) = z.$$

Let us now assume that we can parametrize the *manifold of compatible configurations* — which is implicitly given by the set

$$M = \left\{ (r_1, \ldots, r_n)^\top \mid C_j(r_1, \ldots, r_n, t) = 0 \; \forall j = 1, \ldots, J \right\}$$

— by coordinates $q(t) = (q_1(t), \dots, q(t)) \in Q$, where $Q \subset \mathbb{R}^{n_q}$. This means that there exists continuously differentiable functions $r_i(q,t)$ with

$$M = \left\{ (r_1(q(t),t), \dots, r_n(q(t),t))^\top \mid q(t) \in Q \right\}.$$

Additionally, we assume that the partial derivatives $\frac{\partial r}{\partial q_k}(q(t),t)$, $k = 1, \dots, n_q$ are linearly independent. The parameters $q_1, \dots, q_k$ are called *generalized coordinates*.

Example 3.5
For the pendulum we have

$$r(q(t)) = \begin{pmatrix} \ell \sin(q(t)) \\ -\ell \cos(q(t)) \\ 0 \end{pmatrix}$$

with $q(t) = q_1(t) \in Q = (-\varepsilon, 2\pi) \subset \mathbb{R}$ for arbitrary $\varepsilon > 0$. Note that q describes the angle of the pendulum, which is denoted by θ in the previous section.

Now we can describe our system using the generalized coordinates $q(t)$. Via the chain rule, we can also express the velocity in terms of $q(t)$. We obtain

$$v_i(t) = \frac{d}{dt} r_i(q(t),t) = \sum_{j=1}^{J} \frac{\partial r_i}{\partial q_j}(q(t),t)\dot{q}_j(t) + \frac{\partial r_i}{\partial t}(q(t),t), \quad i = 1, \dots, n.$$

Due to linear independence of the partial derivatives, this equation system can be solved for $\dot{q}(t)$. The variables $\dot{q}_1, \dots, \dot{q}_{n_q}$ are called *generalize velocities*.

Example 3.6
For the pendulum we have

$$v(t) = \begin{pmatrix} \ell \cos(q(t)) \\ \ell \sin(q(t)) \\ 0 \end{pmatrix} \dot{q}(t).$$

Now, we can write the kinetic energy using q and $\dot{q}$ via

$$E_k = \sum_{i=1}^{n} \frac{m_i}{2} \|v_i\|^2 = \sum_{i=1}^{n} \frac{m_i}{2} \left\| \sum_{j=1}^{J} \frac{\partial r_i}{\partial q_j}(q(t),t)\dot{q}_j(t) + \frac{\partial r_i}{\partial t}(q(t),t) \right\|^2 =: \mathcal{T}(q(t), \dot{q}(t), t)$$

which is also denoted by $\mathcal{T}(q(t), \dot{q}(t), t)$.

Example 3.7
For the pendulum we have

$$\mathcal{T}(q(t), \dot{q}(t), t) = \frac{m}{2} \ell^2 \dot{q}(t)^2.$$

For forces $F_i(t) \in \mathbb{R}^3$, $i = 1, \dots, n$, which are applied at the ith point of mass, we define the the so called *generalized forces* via

$$f_k(t) = \sum_{i=1}^{n} \left\langle F_i(t), \frac{\partial r_i}{\partial q_k}(q(t), t) \right\rangle, \quad k = 1, \dots, n_q.$$

Furthermore, we call a mechanical system *conservative*, if there exists a function $\mathcal{W}(r(q(t), t), t)$ such that

$$F_i(t) = -\frac{\partial \mathcal{W}}{\partial r_i}(r(q(t), t), t) =: -\nabla_i \mathcal{W}(r_1(q(t), t), \dots, r_n(q(t), t), t)$$

holds. For the generalized forces, we have to compute

$$f_k(t) = -\frac{\partial \mathcal{W}}{\partial q_k}(q(t), t)$$

with $\mathcal{W}(q(t), t) = \mathcal{W}(r(q(t), t), t)$, which gives us $f(t) = -\nabla_q \mathcal{W}(q(t), t)$. The function $\mathcal{W}$ is typically interpreted as potential energy of the system. This is why one typically adds a suitable constant to arrive at $\min_q \mathcal{W}(q(t), t) = 0$.

Example 3.8
Utilizing the pendulum example without friction, the force $F(t) = (0, -mg, 0)^\top$ applies to the pendulum, which can be written as $f(t) = -\nabla_q \mathcal{W}(q(t), t)$ with $\mathcal{W}(q(t), t) = mgy(t)$. Inserting $r(q(t)) = (\ell \sin(q(t)), -\ell \cos(q(t)), 0)^\top$, we have $\mathcal{W}(q(t), t) = -mg\ell \cos(q(t))$. To satisfy $\min_q \mathcal{W}(q(t), t) = 0$, we add $mg\ell$ to the expression and obtain

$$\mathcal{W}(q(t), t) = -mg\ell \cos(q(t)) + mg\ell.$$

Having defined the notation above, we are now ready to define the Lagrangian:

Definition 3.9 (Lagrangian)
Consider a conservative mechanical system. Then we call the function

$$\mathcal{L}(q(t), \dot{q}(t), t) = \mathcal{T}(q(t), \dot{q}(t), t) - \mathcal{W}(q(t), t) \tag{3.8}$$

the Lagrangian of the system.

Utilizing the Lagrangian, we can derive the equations of motion of the system: The so called *Lagrangian Equation*

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_k}(q(t), \dot{q}(t), t)\right) - \frac{\partial \mathcal{L}}{\partial q_k}(q(t), \dot{q}(t), t) = 0, \quad k = 1, \dots, n_q \tag{3.9}$$

holds. Note that this equation can be obtained from the physical condition that the functional

$$\mathcal{I}(q(t)) = \int_{t_0}^{t_1} \mathcal{L}(q(t), \dot{q}(t), t)\, dt$$

is minimal along solutions q. Setting $g(\alpha) = \mathcal{I}(q + \alpha z)$ for an arbitrary differentiable function z with $z(t_0) = z(t_1) = 0$, then we have $\dot{g}(0) = 0$. After some computations we obtain

$$\dot{g}(0) = \sum_{k=1}^{n_q} \int_{t_0}^{t_1} \left(\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_k}(q(t), \dot{q}(t), t) \right) - \frac{\partial \mathcal{L}}{\partial q_k}(q(t), \dot{q}(t), t) \right) z(t)\, dt$$

revealing (3.9).

Example 3.10
Considering the pendulum without friction, we obtain

$$\mathcal{L}(q(t), \dot{q}(t), t) = \frac{m}{2} \ell^2 \dot{q}(t)^2 + mg\ell \cos(q(t)) - mg\ell.$$

Hence, we have

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \dot{q}}(q(t), \dot{q}(t), t) &= m\ell^2 \dot{q}(t) \\ \frac{\partial \mathcal{L}}{\partial q}(q(t), \dot{q}(t), t) &= -mg\ell \sin(q(t)). \end{aligned}$$

Hence, we obtain the equations of motion via (3.9)

$$\begin{aligned} 0 &= \frac{d}{dt} \left(m\ell^2 \dot{q}(t) \right) + mg\ell \sin(q(t)) \\ &= m\ell^2 \ddot{q}(t) + mg\ell \sin(q(t)). \end{aligned}$$

Since $\ell > 0$ and $m > 0$, the latter simplifies to

$$0 = \ell \ddot{q}(t) + g \sin(q(t)),$$

which corresponds to our earlier results with $q = \theta$.

Remark 3.11
The Lagrangian approach we presented here is given for conservative systems, i.e. systems without loss of energy, e.g., via friction. To integrate such effects gives us a so called dissipative system. Within the modeling, a dissipation rate needs to be defined and translated into a generalized friction force. Then, we can add this force to the right hand side of the Lagrangian Equation (3.9) *and solve the latter.*

Chapter 4

Stochastic processes

Financial processes are a rather young field of research, which received quite some attention during the "New Economy". Its limitation may also have been a source for the dot-com bubble in the late 1990s. Since then, the field was less attractive, but since complex financial products will also be traded in the future, respective models and research will still have their place.

These complex products not only arise in the context of financial speculations, but also as safeguards for changes of currency exchange rates, i.e. daily business of international companies. Here, we will focus on the latter, and particularly discuss the simplest form known as the *European Option*. These derivatives depend on the underlying portfolio, that is a mixture of stock prices and currency exchange rates. Our aim is to compute the value of such a derivative at a given time instant. Since the value depends on the unknown future development of the stocks and rates, we require a respective model of these.

Since nobody can honestly claim to be able to predict the future development of stock prices and exchange rates, we will not use *deterministic* differential equations but *stochastic* ones instead, cf. Definition 1.18. For each initial condition, stochastic differential equations exhibit a number of possible solutions, which depend on chance. The idea of these stochastic differential equations is to approximate possible future developments such that known statistical values from past data (such as expected value or variance) are best modeled.

Within this chapter, we first provide the means of modeling and analyzing a model via the Ito stochastic differential equation and the Ito integral. Thereafter, we introduce one of the most simple task in finance, the assessment of options, and derive models for the stock development. Last, we show two practical methods which allow for computing prices of options.

4.1 Ito integral

In the introductory Chapter 1, we observed in Figure 1.10 that the paths of a Wiener process look similar to stock prices. Yet, even for a very simple modeling of stocks, the Wiener process is too simple. The reason for the latter is the absence of parameters, which can be fit to adapt the model to real data. The Wiener process lacks this structure.

However, the Wiener process is ideally suited to comprise as an ingredient in the definition of a stochastic differential equation in (1.17). Indeed, we will use the Wiener process to describe the derivative of the random variable X in (1.17). To discuss the resulting mathematical problems, we focus on the most simple stochastic differential equation first, and extend our findings to the more general case afterwards.

Since the Wiener process (cf. Definition 1.19) is a stochastic function, the solutions of a stochastic differential equation (1.17) based on a Wiener process are again stochastic functions.

Here, we continue to use the notation $t \in \mathbb{R}$ for time and $x(t) \in \mathbb{R}^{n_x}$ for the state of the stochastic process and $x_0 \in \mathbb{R}^{n_x}$ for the initial value. Each state x can be a vector, i.e. $x = (x_1, x_2, \ldots, x_N)^\top$ where each x_i is a real valued stochastic process. Each of these solutions x_i is connected to one Wiener process $W(t, \omega)$. The solution of the stochastic process for a given path $W(t, \omega)$ is then denoted by $x(t; t_0, x_0, \omega)$.

The main technical difficulty in formulating a stochastic differential equation arises already in a seemingly trivial task — the formulation of a problem where the Wiener process is the solution of that problem. The task is only seemingly trivial since we consider the Wiener process to be given and might be tempted to use

$$\dot{x}(t) = \dot{W}(t, \omega) \tag{4.1}$$

with initial condition $x_0 = W(0, \omega)$ at initial time $t_0 = 0$. Yet, what is $\dot{W}(t, \omega)$? One might think of it as a pathwise derivative, i.e. computing the derivative of each path $W(t, \omega)$ separately. As noted before, a typical path is nowhere differentiable.

To circumvent that problem, we can write (4.1) in form of an integral equation

$$x(t; t_0, x_0, \omega) = x_0 + \int_0^t \dot{W}(t, \omega) dt. \tag{4.2}$$

Now we can formally integrate, yet it doesn't answer the question "what is $\dot{W}(t, \omega)$". For the integral in (4.2) we will use the abbreviation $\int_0^t dW_t$. This denotation already shows the way, which we want to follow to solve our problem: Instead of analyzing the derivative $\dot{W}(t, \omega)$, we state a mathematical definition of the integral which satisfies the following properties:

- $\int_0^t dW_t$ is well defined.
- $\int_0^t dW_t$ provides the desired result $x(t; t_0, x_0, \omega) = W(t, \omega)$.
- A generalization to

$$I(F) := \int_0^t F(t) dW_t \tag{4.3}$$

  is possible, which allows for formulation of more complex stochastic differential equations. Note that F is again a stochastic process.

Here, we want to state such a concept for integrals of form (4.3). The idea of this concept is to approximate the integral for each pair of paths $F(t, \omega)$ and $W(t, \omega)$ by the limit of a suitable sum.

Definition 4.1 (Ito Integral)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$, a random variable $F : \mathbb{R} \times \Omega \to \mathbb{R}^{n_x}$, $N \in \mathbb{N}$ and a sequence of time instances $\tau_k^{(N)}$, $\tau_k = 0, 1, \ldots, N$ with

$$t_0 = \tau_0^{(N)} < \tau_1^{(N)} < \ldots < \tau_N^{(N)} = t_1$$

to be given. For each $\omega \in \Omega$ we define

$$I^{(N)}(F)(\omega) := \sum_{k=0}^{N-1} F(\tau_k^{(N)}, \omega) \cdot \left(W(\tau_{k+1}^{(N)}, \omega) - W(\tau_k^{(N)}, \omega) \right).$$

For a family of sequences $\left(\tau_k^{(N)}\right)_{N\in\mathbb{R}}$ with $\lim_{N\to\infty}\{\max_{k=1,\dots,N}\tau_k^{(N)}-\tau_{k-1}^{(N)}\}=0$ we call

$$I(F) := \operatorname*{l.i.m.}_{N\to\infty} I^{(N)}(F) \tag{4.4}$$

the Ito integral of the stochastic process F.

Here, the question arises whether or not the limit of the integral sequence exists. The trick of Ito is to *not* consider the limit to be understood for each path — i.e. anticipating $\lim_{N\to\infty} I^{(N)}(F)(\omega)$ for each sample $\omega\in\Omega$ — but instead to consider the values $I^{(N)}(F)$ and the integral $I^{(N)}$ as random variables $I^{(N)}(F):\Omega\to\mathbb{R}$ and $I^{(N)}:\Omega\to\mathbb{R}$. Utilizing the concept of Mean Square Convergence, cf. Definition 1.17, we can show that given respective assumptions on F the sequence $(I^{(N)}(F))_{N\in\mathbb{N}}$ converges and (4.4) is well defined.

Now, we can formalize the definition of a stochastic differential equation from Definition 1.18 in the sense of Ito:

Definition 4.2 (Ito Stochastic differential equation)
Consider deterministic functions $a,b:\mathbb{R}\times\mathbb{R}^{n_x}\to\mathbb{R}^{n_x}$, a probability space $(\Omega,\mathcal{F},\mathrm{P})$ and a random variable $W:\mathbb{R}\times\Omega\to\mathbb{R}^{n_x}$ to be given. Then we call

$$dx(t) = a(t,x(t))dt + b(t,x(t))dW_t \tag{4.5}$$

an Ito stochastic differential equation.

Note that this is only a symbolic notation. Equation (4.4) relates to the (longer) integral formulation

$$x(t) = x_0 + \int_{t_0}^{t} a(t,x(t))dt + \int_{t_0}^{t} b(t,x(t))dW_t$$

where the second integral is the Ito integral. The *deterministic* part $a(t,x(t))$ of the equation is called *drift*, and the *stochastic* part $b(t,x(t))$ is referred to as *diffusion*.

Remark 4.3
Equation (4.4) *can be extended in many ways, i.e. by inserting various independent Wiener processes* $W^1, W^2, \dots$. *Among other properties, we can show that*

$$\mathrm{E}\left(\int_{t_0}^{t_1} F(t)dW_t\right) = 0, \tag{4.6}$$

which follows directly from the independence of the random variables F *and* $W(s)-W(t)$ *for* $s>t>0$ *via*

$$\mathrm{E}\left(F(t)(W(s)-W(t))\right) = \mathrm{E}\left(F(t)\right)\underbrace{\mathrm{E}\left(W(s)-W(t)\right)}_{=0} = 0$$

and going to the limit $I(F)$.

To calculate with the Ito integral in general, we require respective rules. Ito's Lemma provides an extension of the chain rule for stochastic differential equations, and is also sometimes referred to as the stochastic chain rule.

Lemma 4.4 (Ito's Lemma)
Consider a function $g : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ which is twice continuously differentiable and suppose $x(t)$ to be the solution of a stochastic differential equation of form (4.5). Then $\tilde{x}(t) = g(t, x(t))$ satisfies

$$\begin{aligned} d\tilde{x}(t) = &\left(\frac{\partial g}{\partial t}(t, x(t)) + \frac{\partial g}{\partial x}(t, x(t))a(t, x(t)) + \frac{1}{2}\frac{\partial^2 g}{\partial x^2}(t, x(t))b(t, x(t))^2 \right) dt \\ &+ \frac{\partial g}{\partial x}(t, x(t))b(t, x(t))dW_t, \end{aligned} \tag{4.7}$$

where W is the Wiener process of the stochastic differential equation satisfied by $x(t)$. Formula (4.7) is also called Ito formula.

4.2 Options

The assessment of options is one of the simpler tasks in finance, yet it is far from trivial. Here, we will define what an option is. Then, we will shortly discuss an important formula from stochastic analysis, and develop models mimicking stock development.

An option is a contract, which provides the holder with the possibility (but not the obligation) to sell or buy a share at a future time instant for a fixed price. The price is referred to as *strike price*, the selling option is also called a *put* and the buying option is called a *call*.

Here, we consider the *European option*. The difference to other options is that the strike time is apriori fixed, and we denote it by T. The task now is the following:

What is the value of the option itself at time $t < T$?

This question arises if, e.g., a bank wants to emit such an option, or if a holder wants to sell it prior to the strike time. Focusing on the call, we denote the (known) base value of a share at a certain time $t \in [0, T]$ by S, and the value of the option (which we like to compute) by $V(t, S)$. Furthermore, let K the fixed strike price.

For $t = T$, we obtain that if $S > K$, then the value of the option equals the profit $V(T, S) = S - K > 0$. If $S \leq K$, then we would have to buy the share for a higher price than we would sell it for using the option. Hence, we don't use the option and get $V(T, S) = 0$. Combined, we obtain

$$V(T, S) = \max\{S - K, 0\} =: (S - K)^+.$$

The put case is inverted, i.e. we have

$$V(T, S) = \max\{K - S, 0\} =: (K - S)^+.$$

To compute the value of the option at any time $t < T$ we require

(1) a rule for computing $V(t, S)$ from $V(T, S(T))$ if $S(T)$ is known, and

(2) an estimate of the base value $S(T)$ at time T depending on the base value S at time t.

If (1) and (2) are available, then we can estimate $V(T, S(T))$ via (2) and apply the rule (1) to this estimate.

Regarding (1): We assume that $S(T)$ is known, and henceforth also $V(T, S(T))$ is known. Now we could simply set $V(t, S(t)) = V(T, S(T))$, which however doesn't fit the economic reality. Instead, we have to add a discount factor $\exp^{-r(T-t)}$, where $r > 0$ is the interest rate for a risk free fund. The discount is motivated by a general assumption in modeling of financial processes — no-arbitrage bounds. Arbitrage is the benefit from a risk free fund. The respective postulate assumes that if a product is traded at two markets at different prices, then the prices would converge immediately, rendering arbitrage to be impossible. Although this doesn't hold in practice, it is an accepted assumption. Considering the absence of arbitrage, the payoff of an option at time T is given by $B(T) = \exp^{r(T-t)} V(t, S(t))$. If we consider the value $V(T, S(T))$ to be known and if $V(t, S(t)) > \exp^{-r(T-t)} V(T, S(T))$, then we could sell the option immediately and invest the payoff risk free. Hence, we obtain

$$B(T) = \exp^{r(T-t)} V(t, S(t)) > V(T, S(T))$$

and our risk free profit is given by $B(T) - V(T, S(T)) > 0$. Vice versa, if $V(t, S(t)) < \exp^{-r(T-t)} V(T, S(T))$, then we could buy that option for $B(t) = V(t, S(t))$ and at strike time get the return

$$B(T) = V(T, S(T)) > \exp^{r(T-t)} V(t, S(t)).$$

Now the risk free profit is given by $B(T) - \exp^{r(T-t)} V(t, S(t)) > 0$. Since the postulate of no-arbitrage bounds excludes risk free profits, the following equality holds:

$$V(t, S(t)) = \exp^{-r(T-t)} V(T, S(T))$$

Regarding (2): We model the typical stock development using a stochastic differential equation of form (4.5) and set $S(T) = x(T; t, S(t))$. Note that $S(T)$ is not a fixed value but a random variable. The value of $V(T, S(T))$ can be estimated via the expected value $\mathrm{E}\left(V(T, x(T; t, S(t)))\right)$.

Combining (1) and (2), we obtain the equation

$$V(t, S(t)) = \exp^{-r(T-t)} \mathrm{E}\left(V(T, x(T; t, S(t)))\right), \tag{4.8}$$

which allows us to compute the value of the option at time t based on the value of $S(t)$.

The minimal requirements for modeling stock development are the parameters *trend* $\mu \in \mathbb{R}$ and the *spreading* $\sigma > 0$. The first parameter μ gives the general direction of the stock development, either up, down or leveling, while the second parameter σ corresponds to the variance/jitter of the stock development around the general direction.

The simplest stochastic differential equation model satisfying these requirements is given by

$$dx(t) = \mu x(t)dt + \sigma x(t)dW_t. \tag{4.9}$$

The solutions of this equation a called *geometric Brownian motion*. In finance, the parameters μ and σ are also termed *rate of return* and *volatility*.

Despite its simplicity, the model (4.9) is the basis of many applications regarding the modeling of stocks. The beauty of this simple equation is the fact that the solutions can be computed

analytically. Using Ito's Lemma 4.4 and the uniqueness of the solution of (4.9), we can show that

$$x(t;t_0,x_0) = x_0 \exp^{\left(\mu-\frac{1}{2}\sigma^2\right)t+\sigma W(t)} \tag{4.10}$$

is the solution of (4.9). For $\sigma = 0$ we reobtain the solution of the linear differential equation $\dot{x}(t) = \mu x(t)$ and its solution $x(t;t_0,x_0) = x_0 \exp \mu(t-t_0)$. By this equation, the expected value $\mathrm{E}\left(x(t;t_0,x_0)\right)$ is given. For the solution of (4.9) we have

$$\begin{aligned}\mathrm{E}\left(x(t;t_0,x_0)\right) &= \mathrm{E}\left(x_0\right) + \mathrm{E}\left(\int_{t_0}^{t} \mu x(\tau;t_0,x_0)dt\right) + \underbrace{\mathrm{E}\left(\int_{t_0}^{t} \sigma x(t;t_0,x_0)dW_t\right)}_{=0 \text{ due to (4.6)}} \\ &= \mathrm{E}\left(x_0\right) + \int_{t_0}^{t} \mathrm{E}\left(\mu x(\tau;t_0,x_0)\right)dt. \end{aligned} \tag{4.11}$$

Hence, the function $e(t) = \mathrm{E}\left(x(t;t_0,x_0)\right)$ satisfies $\dot{e}(t) = \mu e(t)$ with initial condition $e(0) = \mathrm{E}\left(x_0\right) = x_0$, which gives us $\mathrm{E}\left(x(t;t_0,x_0)\right) = x_0 \exp^{\mu(t-t_0)}$.

Similar to the expected value, we can also compute the variance of the solutions of (4.9), which is given by

$$\sigma^2\left(x(t;t_0,x_0)\right) = x_0^2 \exp^{2\mu(t-t_0)}\left(\exp^{\sigma^2(t-t_0)} - 1\right). \tag{4.12}$$

The parameters trend and spreading are typically estimated using past values. This shows, that this type of model is not entirely suited for generating prediction of stock developments. For risk neutral assessment, we set $\mu = r$.

4.3 Monte–Carlo method

As described before, the problem of assessing the value of an option can be played back to the computation of $V(t,S(t))$ via (4.8). Due to the stochastic processes involved in the problem, the main difficulty now is to compute the expected value $\mathrm{E}\left(V(T,x(T;t,S(t)))\right)$. Here, we discuss the Monte–Carlo Method to solve this problem.

The Monte–Carlo method is a direct and very versatile method to compute the expected value of complex expressions. Similar to the name giving casino in Monaco, the Monte–Carlo method utilizes a vast number of random experiments. Instead of hoping for a prize, we calculate an estimate of the expected value based on the results of the random experiments. The random experiments themselves are performed by computers according to the following algorithm, and the solution is therefore a numerical and not an analytic one.

Algorithm 4.5 (Monte–Carlo Method)
Given a stochastic differential equation (4.9), an initial time t, a strike time T, a risk free interest rate r and the function $V(T,S(T))$ from Section 4.2.

1. Use a random number generator to create (approximations of) paths $W(t,\omega_k)$, $k = 1,2,\ldots,N$ of a Wiener process.
2. Apply a numerical method to solve the stochastic differential equation to (approximatively) obtain $x(\tau;t,S(t,\omega_k))$. Set $\widetilde{S}_k(T) = x(\tau;t,S(t,\omega_k))$.

3. Compute the approximation of the expected value via

$$\widetilde{\mathrm{E}}(V(T,S(T))) = \frac{1}{N}\sum_{k=1}^{N} V(T,\widetilde{S}_k(T)).$$

4. Evaluate the estimate $\widetilde{V}(t,S(t)) = \widetilde{\mathrm{E}}(V(T,S(T)))\exp^{-r(T-t)}$.

Note that for the simple model (4.9), we can utilize the solution formula (4.10) instead of a numerical approximation. To this end, not the entire paths of the Wiener process need to be simulated, only the values $W(T,\omega_k)$ as $\mathcal{N}(0,T)$–distributed random variables.

The Monte–Carlo method is popular due to several reasons:

1. The method itself is intuitive and easy to understand.
2. It allows to consider complex stochastic processes and more general functions $V(T,S(T))$.
3. Additionally, the interest rate $r = r(t)$ may be time varying.

However, there are some drawbacks as well:

1. The solution produced by the method converges very slowly.
2. The method only computes the value of the option for a fixed base value $S(t)$ at a fixed time t. To evaluate $V(t,S(t))$ as a function of $S(t)$ and t, many runs of the method must be executed.

A different approach to assess options is given by methods based on partial differential equations, i.e. the Black–Scholes equation.

4.4 Numerical illustration

To illustrate the outcome the presented methods, we consider the following example:

Example 4.6
Consider model (4.9)

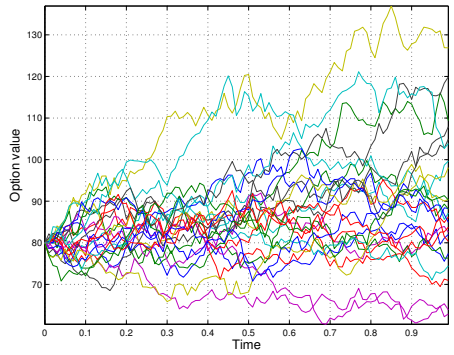
$$dx(t) = \mu x(t)dt + \sigma x(t)dW_t$$

with $\mu = 0.08$ *and* $\sigma = 0.2$*, initial time* $t = 0$ *and initial value* $S = 80$*, and strike price* $K = 100$ *at time* $T = 1$ *and suppose the risk free interest rate to be* $r = 0.08$*. The payoff is given by*

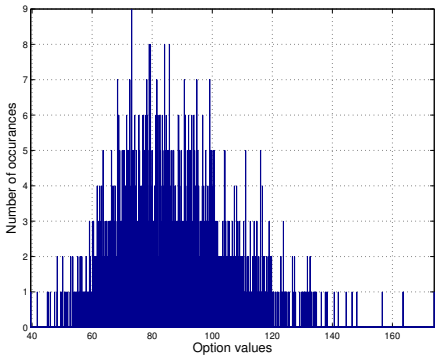
$$B = \max\{0, x(T;t,S) - K\}.$$

Applying Algorithm 4.5, we generate 2000 sample paths of the Wiener process in the first step.
In the second step, we compute the related 2000 solutions of (4.9), cf. Figure 4.1a for a few of these solutions. The large number of samples allows us to approximate the probability density

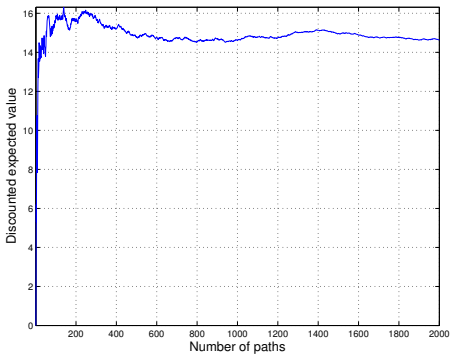
function via a histogram of solutions at strike time T, which is displayed in Figure 4.1b. Based on these solutions, we can compute the expected value of the underlying stock at strike time T in the third step. Applying (4.8), we obtain the discounted value of the option displayed in Figure 4.1c. The figure illustrates nicely that for large numbers of samples, the solution generated by the Monte–Carlo method converges. Yet, we also observe that quite a large number of samples is required to reduce the fluctuations in the discounted expected value.



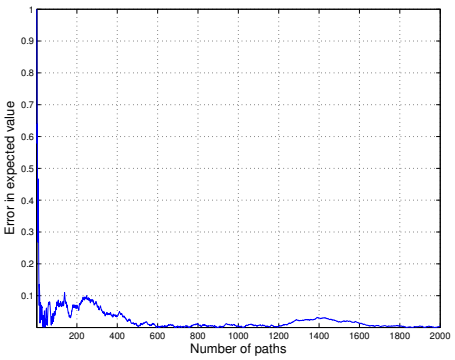
(a) Several solutions paths



(b) Histogram of values $x(T;t,S)$



(c) Discounted expected value of the option



(d) Error of the discounted expected value

Figure 4.1: Numerical results from Example 4.6

The last Figure 4.1d show the difference between the true solution, which we evaluated using the Black–Scholes equation, and the approximation computed by the Monte–Carlo method. Again, we observe convergence of the Monte–Carlo results, and that the approximation actually tends towards the correct value.

The Black–Scholes equation also allows us to display the continuum of solutions for different initial conditions $(t, S(t))$ for our Example 4.6. Figure 4.2 illustrates respective values showing the connection between the initial condition and the value of the option.

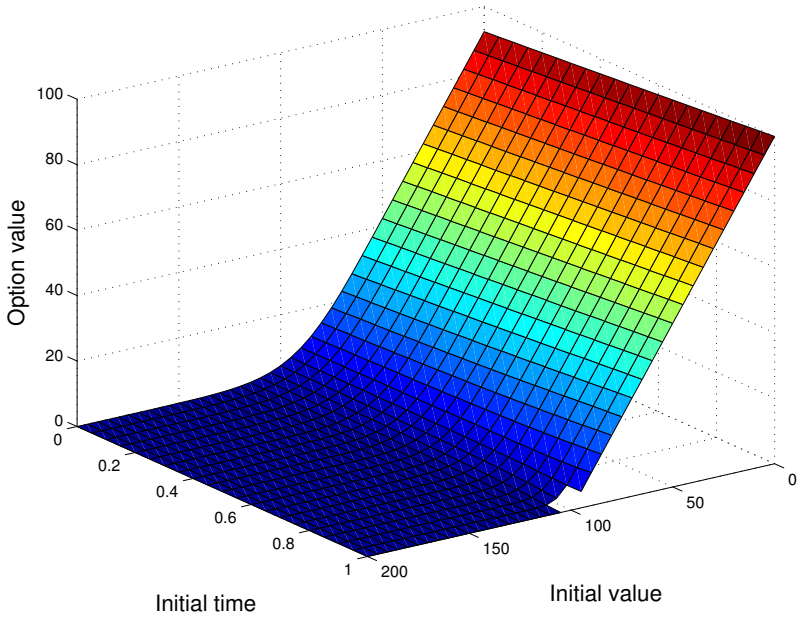


Figure 4.2: Option value for Example 4.6 for various initial conditions

Part II

Identification

Chapter 5

Structure of the identification process

Within the Chapter 1, we introduced the notions from stochastic analysis, which we require to study the modeling and identification process. Within the current chapter, we will first discuss the general design sequence of a system identification, which is also called an estimator. Thereafter, we focus on the properties which we are looking for in an estimator. Exemplarily, we will check these properties for one of the estimators given in the circuit example from Chapter 1. Assessing these estimators will show that there is a clear need for an in deep analysis of properties of estimators.

5.1 Basic design of estimators

Each identification process consists of a series of basic steps:

1. Collect information on the system
2. Select a model to represent the system
3. Choose an optimization criterion
4. Fit the model parameters to the measurements accordingly
5. Validate the computed model

Note that some of the steps may be hidden from the user or selected without being aware of a choice, which may result in suboptimal or even poor performance.

5.1.1 Step 1: Gathering information

In order to identify a process, we first need to build a model of that part of the system, which we are interested in. To this end, we need to gather information about the process. This step can be done either by observing natural fluctuations, but it is by far more efficient to set up dedicated experiments that actively excite the system via known inputs. While a good example of the first are default fluctuations in demand for a supply chain, the latter can be interpreted as a stress test of a supply chain by uncommon and/or extreme demands. Additionally, the controlled second approach allows for optimization of information gathering goals, such as minimum cost and time, measurement accuracy over a certain bandwidth or other possible aims. Note that the quality of the total identification process may heavily depend on these choices.

5.1.2 Step 2: Selecting the model structure

The model structure is the most variable part of the identification. It not only depends on the problem of identification itself, but may be subject to the further use of the model. For example, an approximation of the elasticity of a wheel via a PDE may give a good dynamical model. Yet, if the model is to be used in a feedback loop, the required computing time to evaluate the model is larger than the sampling time of the loop. Hence, a coarser (or worse) model is necessary for the subsequent task. Keeping this in mind, we distinguish the following:

Parametric vs. nonparametric models

In a parametric model, the system is described by a small number of characteristic quantities. These quantities are called parameters of the model. Regarding our simple electrical circuit example, the expected value is one parameter of the model, the variance the second one. An alternative example is given by the transfer function, e.g. of a filter, which is described by its poles and zeros.

A nonparametric model is given by measurements of a system function at a large number of points. Reviewing the transfer function example, a description via an impulse response at a large number of points is such a characterization.

Note that it is usually simpler to create a nonparametric model than a parametric one because the modeler needs less knowledge about the system itself in the first case. Yet, insight into the problem and concentration of information in a few characteristics is more substantial for parametric models and make the problem simulation faster.

White box vs. black box models

In a white box model, the internal functioning of the system is – at least to some degree – understood. In particular, skills of the experimenter as well as connections between components such as physical laws can be used, whose availability and applicability depend on such an insight. Here, a loudspeaker illustrates the need for extensive understanding of mechanical, electrical and acoustical phenomena in order to derive an appropriate model.

In contrast to the white box idea of using insight into the system, the black box approach uses a brute force modeling. To this end, a mathematical model is proposed, which allows the description of any observed input and output measurements, but may not even be connected to the real system. Regarding the loudspeaker, a high order transfer function may be used as such a model.

Again, the choice depends on the further aim. While the white box idea provides a better insight gain into the working principles of the system, the black box model may be sufficient for simulations/predictions. Note that it is typically a good idea to include as much knowledge as possible during modeling, yet that may not always be easy to accomplish. Analyzing a stable system for example, it is not simple to express this information if the polynomial coefficients are used as parameters of the model.

Linear vs. nonlinear models

In almost all cases, real life applications are nonlinear. Unfortunately, theory of nonlinear systems is quite involved and may be difficult to understand for a user unfamiliar with this theory. A nonlinear approach describes the system over its complete operating range and covers also rare and unusual phenomena.

Linear systems, on the other hand, are (almost) completely understood, nice to handle and can be evaluated quickly. Unfortunately, as stated above, real life is typically nonlinear. Therefore, linear systems commonly represent approximations of nonlinear systems within some region – assuming the region can be linearized. Within such a so called operating region, the linear part of the system can be regarded as dominant, i.e. the nonlinear part can be neglected without changing the behavior of the system.

Similar to the other choices, the scope of the problem is relevant to make an appropriate choice. For example, a nonlinear model is needed to describe the distortion of an amplifier, but a linear model is sufficient to represent its transfer characteristics if the operating range is small enough.

Linear-in-parameter vs. nonlinear-in-parameter models

The last choice has to be made between linear and nonlinear influence of parameters of the model. A model is called linear-in-parameter if there exists a linear relation between these parameters and the error that is minimized. Note that linear-in-parameters does not imply a linear model. For example, $e = y-(au^2+bu+c)$ is linear in a, b and c, but the model is nonlinear. Likewise, $e = y - (a + bj\omega)/(c + dj\omega)u$ is a linear model, but it is nonlinear-in-parameter in c and d.

The impact of this choice can be seen, e.g., for the least square estimator. If the model is linear-in-parameter, then the minimization problem of the least squares can be solved analytically, and does not require an iterative optimization method. Hence, the complexity of a linear-in-parameter model is much lower.

5.1.3 Step 3: Choose optimization criterion

After choosing a model, it must be matched to the available measurements of the process. To this end, one typically introduces a criterion, which measures the goodness of fit, i.e. the distance between the computed and the measured values. Note that the choice of this criterion is important regarding the outcome of the identification process as it determines the stochastic properties of the estimator. Regarding our simple resistor example, there are several choices which lead to estimators with different properties, cf. Section 5.3.

The cost criterion can be chosen arbitrarily. Yet, it typically resides on ad hoc intuitive insight. In the following Chapter 6, we provide a more systematic approach based on stochastic arguments to obtain such a criterion.

Remark 5.1
There exist tests on the cost criterion to check – even before deriving the estimator – if the resulting estimator can be consistent. These are necessary conditions, which are outside the scope of this lecture.

5.1.4 Step 4: Fitting model parameters

In the ensuing step of fitting the parameters, the design work is done and the computations start. Within this step, numerical or symbolic methods are applied to solve the minimization problem arising from the cost criterion in Step 3 subject to the model chosen in Step 2 with respect to the measurement derived in Step 1. Although this step seems to be the essential one, we can already see that the most of the work is the design. This is due to the fact that

nowadays, computing power is cheap and there exist a wide area of methods to solve certain problems. The actual art is to design the problem such that it is easily solvable but satisfies the constraints, which bound the model in its further use.

5.1.5 Step 5: Validating obtained model

In the final step, the validity of the obtained model shall be tested. Here, the following question are essential:

- Does the model describe the available data properly?
- Are there indications that some parts of the model are not well designed or flawed?

Note that, as mentioned before, the model with the smallest error is not always the preferred one in practice. Instead, a simpler model may be better suited if it describes the system within user-specified error bounds.

Within the validation process, errors should be separated into different classes such as un-modeled linear dynamics or nonlinearity distortions. Such information shall allow further improvements of the model if necessary. During the validation, the application should be kept in mind, i.e. conditions similar to reality are to be used. Note that extrapolation should be avoided as the errors of extrapolation increase drastically if many measurements are used, which is the typical case for estimator design.

Now that we have seen the general structure of an identification process, we are now interested in properties such an estimator shall offer.

5.2 Properties of estimators

Here, we start of with the claim that a good estimation of a system should exhibit the same characteristics, i.e. the same probability density function. Since the probability density function completely defines the properties of a system, such an estimation would do this as well. Unfortunately, as discussed in the context of Definition 1.14, without additional conditions it is very hard to show the respective convergence in distribution. But we also learned that certain properties of the expectation value are sufficient to guarantee mean square convergence, cf. Definition 1.17, which is in turn sufficient for convergence in distribution — the property we like to have.

Hence, our first demand for an estimator is that it reflects an identical expectation value.

Definition 5.2 (Unbiased estimator)
Suppose a probability space $(\Omega, \mathcal{F}, \mathrm{P})$, a measurable space E with σ–algebra $\mathcal{E}$ of E and an estimator (random variable) $\hat{\theta} : \Omega \to E$ for the parameter $\theta \in E$ to be given. If

$$\mathrm{E}\left(\hat{\theta}\right) = \theta \qquad \forall \theta \in E \tag{5.1}$$

holds true, then we call the estimator $\hat{\theta}$ unbiased. If

$$\lim_{N\to\infty} \mathrm{E}\left(\hat{\theta}(N)\right) = \theta \qquad \forall \theta \in E \tag{5.2}$$

holds, then we call the estimator $\hat{\theta}$ asymptotically unbiased. Otherwise, it is called biased.

Note that, if the estimator is unbiased, its mean converges towards the mean of the model or model parameters. Yet, since we design the model to represent only a certain part of reality, the model is typically not exact. Hence, the „ideal“ situation is not realistic and we have to think about generalizations. One possibility is to suppose that we evaluate the estimator in a noiseless situation to obtain an approximation. Then, these reference values are compared to results with noise. The final step is to eliminate the influence of the disturbance such that the estimator converges to its reference.

Unfortunately, it is very difficult if not impossible to find the expected value by analytical means. And for some probability density functions, the expected value does not exist. And last, we may face the problem that while the expected values are identical, i.e. the estimator is unbiased according to Definition 5.2, the probability density functions are very different and coincide only in the expected value. If such an estimator were used, the outcome of a system may be very different from the real one. To avoid such a problem, we introduce the concept of consistency:

Definition 5.3 (Weak and strong consistency)
Suppose an estimator $\hat{\theta}$ and parameters θ to be given. If $\hat{\theta}$ converges in probability to θ,

$$\operatorname*{p.lim}_{N\to\infty} \hat{\theta}(N) = \theta, \tag{5.3}$$

then the estimator $\hat{\theta}$ is called weakly consistent.
If $\hat{\theta}$ converges almost surely to θ,

$$\operatorname*{a.s.lim}_{N\to\infty} \hat{\theta}(N) = \theta, \tag{5.4}$$

then the estimator $\hat{\theta}$ is called strongly consistent.

The advantage of this concept is that we can prove consistency much easier than unbiasedness. Since the limit operator may be interchanged with a continuous function ($\operatorname{p.lim} f(x) = f(\operatorname{p.lim}(x))$) if both limits exist, the consistency idea also exhibits nice calculation properties.

Apart from unbiasedness and consistency, we are also interested in obtaining an estimator, which shows minimal errors only. In particular, we want to minimize the scatter range of the estimator around its limiting value. That gives us the concept of efficiency:

Definition 5.4 (Efficiency)
Suppose an unbiased estimator $\hat{\theta}$ of parameter θ to be given. If for any unbiased estimator $\hat{\theta}_1$ of parameter θ the inequality

$$\operatorname{Cov}\left(\hat{\theta}, \hat{\theta}\right) \leq \operatorname{Cov}\left(\hat{\theta}_1, \hat{\theta}_1\right) \tag{5.5}$$

holds, then the estimator $\hat{\theta}$ is called efficient.

Since we can rely on a finite number of noisy measurements only, it is clear that there are limits on the accuracy and precision that can be reached by the estimator. The connection between measurements and accuracy is given by the so called Cramer-Rao rule:

Theorem 5.5 (Cramer-Rao rule)
Consider a probability space $(\Omega, \mathcal{F}, \mathrm{P})$ *and a random variable* $X : \Omega \to E$ *defined on that triple, where the set* E *equipped with measure* μ *and* $\mathcal{E}$ *is a* σ*–algebra of* E*. Let* $f(z, \theta)$ *be the probability density function of the measurements* $z \in \mathbb{R}^N$*. Assume that* $f(z, \theta)$ *and its first and second derivatives w.r.t.* θ *exist for all* θ *and that the boundaries of the domain of* $f(z, \theta)$ *w.r.t.* z *are independent of* θ*. Then, the Cramer–Rao lower bound on the mean square error of any estimator* $\hat{G}(\hat{\theta}(z))$ *of the function* $G(\theta) \in \mathbb{C}^r$ *is*

$$\mathrm{MSE}\left(\hat{G}(\hat{\theta}(z))\right) \geq \left(\frac{\partial G(\theta)}{\partial \theta} + \frac{\partial b_G}{\partial \theta}\right) \mathrm{Fi}(\theta)^{+} \left(\frac{\partial G(\theta)}{\partial \theta} + \frac{\partial b_G}{\partial \theta}\right)^{H} + b_g b_g^H \tag{5.6}$$

where b_G *denotes the expected value bias given by*

$$b_G := \mathrm{E}\left(\hat{G}(\hat{\theta}(z))\right) - G(\theta)$$

and $\mathrm{Fi}(\theta)$ *represents the Fisher information matrix of the parameters* θ

$$\mathrm{Fi}(\theta) := \mathrm{E}\left(\left(\frac{\partial \ln f(z, \theta)}{\partial \theta}\right)^{\top} \left(\frac{\partial \ln f(z, \theta)}{\partial \theta}\right)\right) = -\mathrm{E}\left(\frac{\partial^2 \ln f(z, \theta)}{\partial \theta^2}\right).$$

We like to stress that the Cramer–Rao rule requires knowledge of the true parameter θ, which may not be at hand. An approximation can still be calculated by replacing θ in (5.6) by its estimated value $\hat{\theta}$. Similarly, the probability density function $f(z, \theta)$ can be approximated using available measurements z only.

The Cramer–Rao rule gives us a very simple way to check efficiency:

Corollary 5.6 (Efficiency)
If a given estimator $\hat{\theta}$ *reaches the Cramer-Rao bound* (5.6)*, then it is efficient.*

Remark 5.7 (Special cases)
There are a few special cases we like to point out:

1. *Inequality* (5.6) *becomes an equality if and only if there exists a matrix* Γ *such that*

$$\hat{G}(\hat{\theta}(z)) - \mathrm{E}\left(\hat{G}(\hat{\theta}(z))\right) = \Gamma \left(\frac{\partial \ln f(z, \theta)}{\partial \theta}\right)^{\top}.$$

2. *If* $G(\theta) = \theta$*,* $b_G = 0$ *and* $\mathrm{Fi}(\theta)$ *is regular, then we obtain the Cramer–Rao lower bound for unbiased estimators*

$$\mathrm{Cov}\left(\hat{\theta}(z), \hat{\theta}(z)\right) \geq \mathrm{Fi}(\theta)^{-1}.$$

3. *If* $G(\theta) = \theta$*,* $b_G \neq 0$ *and* $\mathrm{Fi}(\theta)$ *is regular, then we find the Cramer–Rao lower bound on the mean square error of biased estimators*

$$\mathrm{MSE}\left(\hat{\theta}(z)\right) \geq \left(Id + \frac{\partial b_G}{\partial \theta}\right) \mathrm{Fi}(\theta)^{-1} \left(Id + \frac{\partial b_G}{\partial \theta}\right)^{\top} + b_G b_G^{\top}.$$

5.3 Exemplary analysis

Recall the simple resistor example from Section 1.3. There, we have already seen that the simple approach estimator R_{SA} does not reveal good results, cf. Figures 1.4 and 1.6. To keep the analysis simple, we consider the following assumption:

Assumption 5.8 (Noise)
The measurements are disturbed by additive random variables, i.e.

$$i(k) = i_0 + X_i(k) \qquad \text{and} \qquad u(k) = u_0 + X_u(k) \tag{5.7}$$

with the properties that

- each random variable has zero mean and variance σ_u^2, σ_i^2,
- each random variable is independently and identically distributed (iid),
- each random variable exhibits a symmetric distribution, and
- the random variables are mutually independent.

5.3.1 Unbiasedness

Analysis of $\hat{\theta}_{\mathbf{EV}} = R_{\mathbf{EV}}$:

Using the model (5.7) within formula (1.4) we directly see

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{EV}}\right) = \lim_{N\to\infty} \hat{\theta}_{\mathrm{EV}}(N) = \lim_{N\to\infty} \frac{\frac{1}{N}\sum_{k=1}^{N} u(k)}{\frac{1}{N}\sum_{k=1}^{N} i(k)} = \lim_{N\to\infty} \frac{\frac{1}{N}\sum_{k=1}^{N} u_0 + X_u(k)}{\frac{1}{N}\sum_{k=1}^{N} i_0 + X_i(k)}$$

$$= \lim_{N\to\infty} \frac{u_0 + \frac{1}{N}\sum_{k=1}^{N} X_u(k)}{i_0 + \frac{1}{N}\sum_{k=1}^{N} X_i(k)} = \frac{u_0 + \lim_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N} X_u(k)}{i_0 + \lim_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N} X_i(k)}.$$

Now, we can apply the zero mean and iid property of X_u and X_i, that is

$$\mathrm{E}\left(X_u\right) = \lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} X_u(k) = 0, \qquad \mathrm{E}\left(X_i\right) = \lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} X_i(k) = 0. \tag{5.8}$$

Hence, we obtain

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{EV}}\right) = \frac{u_0 + \overbrace{\lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} X_u(k)}^{=0}}{i_0 + \underbrace{\lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} X_i(k)}_{=0}} = \frac{u_0}{i_0} = R_0 \tag{5.9}$$

which shows that the error-in-variable estimator is unbiased.

Analysis of $\hat{\theta}_{\mathbf{LS}} = R_{\mathbf{LS}}$:

Again, we first compute the expectation value of the estimator. Here, we obtain

$$\begin{aligned}
\mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right) &= \lim_{N\to\infty} \hat{\theta}_{\mathrm{LS}}(N) = \lim_{N\to\infty} \frac{\sum_{k=1}^{N} u(k)\cdot i(k)}{\sum_{k=1}^{N} i(k)^2} \\
&= \frac{\lim_{N\to\infty} \sum_{k=1}^{N} (u_0 + X_u(k))\cdot(i_0 + X_i(k))}{\lim_{N\to\infty} \sum_{k=1}^{N} (i_0 + X_i(k))^2} \cdot \frac{\frac{1}{N}}{\frac{1}{N}} \\
&= \frac{\lim_{N\to\infty} u_0 i_0 + \frac{u_0}{N}\sum_{k=1}^{N} X_i(k) + \frac{i_0}{N}\sum_{k=1}^{N} X_u(k) + \frac{1}{N}\sum_{k=1}^{N} X_u(k)X_i(k)}{\lim_{N\to\infty} i_0^2 + \frac{2i_0}{N}\sum_{k=1}^{N} X_i(k) + \frac{1}{N}\sum_{k=1}^{N} X_i(k)^2}
\end{aligned}$$

Now, from Assumption 5.8 we use the zero mean and iid property properties (5.8) of the random variables X_u and X_i as well as their mutually independency and the variance assumption, i.e.

$$\forall k = 1,\ldots,N:\ X_u(k)X_i(k) = 0, \qquad \sigma^2(X_i) = \lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} X_i(k)^2 = \sigma_i^2 \tag{5.10}$$

to obtain

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right) = \frac{u_0 i_0}{i_0^2 + \sigma_i^2} = \frac{R_0}{1 + \sigma_i^2/i_0^2}.$$

Hence, the least square estimator will always underestimate the magnitude of the value it is supposed to approximate. Note that the noise is removed from the nominator, but is always present in the denominator. Utilizing Definition 5.2, the least square estimator is biased. The bias depends on the *signal-to-noise ration (SNR)* of the measurements i_0/σ_i.

Analysis of $\hat{\theta}_{\mathbf{SA}} = R_{\mathbf{SA}}$:

If we take a closer look at the simple approach estimator $\hat{\theta}_{\mathrm{SA}}$ and incorporate the structural assumptions (5.7), we obtain

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) = \lim_{N\to\infty} \hat{\theta}_{\mathrm{SA}}(N) = \lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} \frac{u(k)}{i(k)} = \lim_{N\to\infty} \frac{1}{N}\sum_{k=1}^{N} \frac{u_0 + X_u(k)}{i_0 + X_i(k)}$$

Rewriting this equation, we have

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) = \lim_{N\to\infty} \frac{1}{N}\frac{u_0}{i_0}\sum_{k=1}^{N} \frac{1 + X_u(k)/u_0}{1 + X_i(k)/i_0} = \lim_{N\to\infty} R_0 \frac{1}{N}\left(\sum_{k=1}^{N} \frac{1}{1 + X_i(k)/i_0} + \sum_{k=1}^{N} \frac{X_u(k)/u_0}{1 + X_i(k)}\right)$$

Applying the Taylor series

$$\frac{1}{1 + X_i(k)/i_0} = \sum_{j=0}^{\infty} (-1)^j \left(\frac{X_i(k)}{i_0}\right)^j$$

we obtain

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) = \lim_{N\to\infty} R_0 \frac{1}{N}\left(\sum_{k=1}^{N}\sum_{j=0}^{\infty}(-1)^j\left(\frac{X_i(k)}{i_0}\right)^j + \sum_{k=1}^{N}\sum_{j=0}^{\infty}(-1)^j\left(\frac{X_i(k)}{i_0}\right)^j \frac{X_u(k)}{u_0}\right)$$

Applying the mutually independent property (5.8), we can shorten this expression to

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) = \lim_{N\to\infty} R_0 \frac{1}{N}\left(\sum_{k=1}^{N}\sum_{j=0}^{\infty}(-1)^j\left(\frac{X_i(k)}{i_0}\right)^j + \sum_{k=1}^{N} 1\cdot\frac{X_u(k)}{u_0}\right)$$

which can again be shortened using the zero mean property of X_u displayed in (5.8) to

$$\begin{aligned}\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) &= \lim_{N\to\infty} R_0 \frac{1}{N}\sum_{k=1}^{N}\sum_{j=0}^{\infty}(-1)^j\left(\frac{X_i(k)}{i_0}\right)^j = \lim_{N\to\infty} R_0 \frac{1}{N}\sum_{k=1}^{N}\left(1+\sum_{j=1}^{\infty}(-1)^j\left(\frac{X_i(k)}{i_0}\right)^j\right)\\ &= \lim_{N\to\infty} R_0\left(1+\frac{1}{N}\sum_{j=1}^{\infty}\sum_{k=1}^{N}(-1)^j\left(\frac{X_i(k)}{i_0}\right)^j\right).\end{aligned}$$

Since X_i is symmetric due to Assumption 5.8, we have $\sum_{k=1}^{N}(-1)^j\left(X_i(k)/i_0\right)^j = 0$ for odd numbers j. Therefore, the limiting value of the expectation value is

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) = R_0\left(1+\lim_{N\to\infty}\left(\frac{1}{N}\sum_{j=1}^{\infty}\sum_{k=1}^{N}\left(\frac{X_i(k)}{i_0}\right)^{2j}\right)\right) = R_0\left(1+\sum_{j=1}^{\infty}\frac{m_j^{2j}(X_i(k))}{i_0^{2j}}\right)$$

where we used the moments of $X_i(k)$ according to Definition 1.10. For small disturbances $|X_i(k)/i_0| < 1$, we can neglect the moments of order 4 and higher and finally obtain

$$\mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right) = R_0\left(1+\frac{\sigma^2\left(X_i\right)}{i_0^2}\right) = R_0\left(1+\frac{\sigma_i^2}{i_0^2}\right).$$

Similar to the least square estimator $\hat{\theta}_{\mathrm{LS}} = R_{\mathrm{LS}}$, the analysis shows that the estimator converges to a value larger than the desired one. Note that the problem for group A here is that the Taylor series expansion cannot be performed.

5.3.2 Consistency

Note that we have already done these computations during the computation of the expected values since we have been using the concept of convergence with probability 1, which is a stronger concept than convergence in probability. In particular, for the error–in–variables approach we have

$$\operatorname*{p.lim}_{N\to\infty}\hat{\theta}_{\mathrm{EV}} = \operatorname*{p.lim}_{N\to\infty}\frac{\frac{1}{N}\sum_{k=1}^{N}u(k)}{\frac{1}{N}\sum_{k=1}^{N}i(k)} = \frac{\operatorname*{p.lim}_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N}u_0+X_u(k)}{\operatorname*{p.lim}_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N}i_0+X_i(k)} = \frac{u_0}{i_0} = R_0.$$

Hence, $\hat{\theta}_{\mathrm{EV}}$ is a weakly consistent estimator. As we can see, it is much easier to check consistency than (asymptotic) unbiasedness.

Additionally, the concept is typically used based on a cost function interpretation of an estimator. Hence, it reveals an insight into the anticipated errors.

Interpretation of $\hat{\theta}_{\mathbf{LS}} = R_{\mathbf{LS}}$:

Consider, for example, the least square estimator $\hat{\theta}_{\mathrm{LS}}$ from (1.5). The idea is to **minimize the equation errors** $e(k) := u(k) - Ri(k)$ in the model equation in a least square sence. This gives us the cost function

$$J_{\mathrm{LS}}(N) := \sum_{k=1}^{N} e^2(k, R)$$

and allows us to restate the estimation problem via

$$\hat{\theta}_{\mathrm{LS}}(N) = \operatorname*{argmin}_{R \in \mathbb{R}} J_{\mathrm{LS}}(N, R).$$

Interpretation of $\hat{\theta}_{\mathbf{SA}} = R_{\mathbf{SA}}$:

Similarly, the simple approach estimator can be rewritten using the resistor estimates $R(k) := u(k)/i(k)$ defining the cost function $J_{\mathrm{SA}}(N, R) := \sum_{k=1}^{N} \left(R(k) - R\right)^2$ and the estimation problem

$$\hat{\theta}_{\mathrm{SA}}(N) = \operatorname*{argmin}_{R \in \mathbb{R}} J_{\mathrm{SA}}(N, R).$$

The difference compared to the least square estimator $\hat{\theta}_{\mathrm{LS}}$ is that here, we do not consider a model of the problem, which is subject to disturbances. Instead, the **measurement quotient** $R(k)$ is **considered as disturbed**.

Interpretation of $\hat{\theta}_{\mathbf{EV}} = R_{\mathbf{EV}}$:

Last, the error–in–variable approach **assumes that each variable by itself is disturbed**, not the quotient as for the simple approach. Here, the cost function is defined via

$$J_{\mathrm{EV}}(N, R, i_p, u_p) := \sum_{k=1}^{N} \left(u(k) - u_p\right)^2 + \sum_{k=1}^{N} \left(i(k) - i_p\right)^2,$$

which gives us the estimation problem

$$\begin{aligned} \hat{\theta}_{\mathrm{EV}}(N) = & \operatorname*{argmin}_{R, i_p, u_p, N} J_{\mathrm{EV}}(N, R, i_p, u_p) \\ & \text{subject to } u_p = Ri_p. \end{aligned}$$

Note that one typically considers a quadratic cost criterion, but basically one is free to choose any other costs as well. The advantage of the quadratic cost is the simplicity in minimization. Moreover, one can show that normally distributed disturbing noise leads to a quadratic criterion.

5.3.3 Efficiency

In order to analyze efficiency of an estimator, we need to calculate the second moment of it. Alternatively, as we have seen in Corollary 5.6, the Cramer–Rao rule (5.6) can be evaluated. For the present example, the probability density functions are not known exactly and may only be approximated using respective measurements. For this reason, we focus on a manual calculation of the variance of the estimators.

Analysis of $\hat{\theta}_{\mathbf{EV}} = R_{\mathbf{EV}}$:

Regarding the variance, we apply Definition 1.11, that is

$$\sigma^2\left(\hat{\theta}_{\mathrm{EV}}\right) = \mathrm{E}\left(\left(\hat{\theta}_{\mathrm{EV}} - \mathrm{E}\left(\hat{\theta}_{\mathrm{EV}}\right)\right)^2\right).$$

To compute this value, we reconsider $\hat{\theta}_{\mathrm{EV}}$ and — since we are interested in the second moment only — neglect all second order contributions within such as X_i^2 or $X_u X_i$ in this term, i.e.

$$\begin{aligned}
\hat{\theta}_{\mathrm{EV}} &= \frac{u_0 + \frac{1}{N}\sum_{k=1}^{N} X_u(k)}{i_0 + \frac{1}{N}\sum_{k=1}^{N} X_i(k)} = \frac{u_0 + \frac{1}{N}\sum_{k=1}^{N} X_u(k)}{i_0 + \frac{1}{N}\sum_{k=1}^{N} X_i(k)} \cdot \frac{i_0 - \frac{1}{N}\sum_{k=1}^{N} X_i(k)}{i_0 - \frac{1}{N}\sum_{k=1}^{N} X_i(k)} \\
&\overset{\text{neglect 2nd order}}{\approx} \frac{u_0 i_0 + \frac{i_0}{N}\sum_{k=1}^{N} X_u(k) - \frac{u_0}{N}\sum_{k=1}^{N} X_i(k)}{i_0^2} \\
&= \frac{u_0}{i_0} + \frac{1}{i_0 N}\sum_{k=1}^{N} X_u(k) - \frac{u_0}{i_0^2 N}\sum_{k=1}^{N} X_i(k) \\
&= R_0\left(1 + \frac{1}{N}\sum_{k=1}^{N}\frac{X_u(k)}{u_0} - \frac{1}{N}\sum_{k=1}^{N}\frac{X_i(k)}{i_0}\right).
\end{aligned}$$

Hence, we obtain

$$\begin{aligned}
\sigma^2\left(\hat{\theta}_{\mathrm{EV}}\right) = \mathrm{E}\left(\left(\hat{\theta}_{\mathrm{EV}} - R_0\right)^2\right) &= \mathrm{E}\left(\left(R_0\left(\frac{1}{N}\sum_{k=1}^{N}\frac{X_u(k)}{u_0} - \frac{1}{N}\sum_{k=1}^{N}\frac{X_i(k)}{i_0}\right)\right)^2\right) \\
&\overset{\text{mutually ind.}}{=} \mathrm{E}\left(\frac{R_0^2}{N^2}\sum_{k=1}^{N}\frac{X_u(k)^2}{u_0^2} + \frac{R_0^2}{N^2}\sum_{k=1}^{N}\frac{X_i(k)^2}{i_0^2}\right) \\
&\overset{\text{linearity}}{=} \frac{R_0^2}{N^2}\left(\mathrm{E}\left(\sum_{k=1}^{N}\frac{X_u(k)^2}{u_0^2}\right) + \mathrm{E}\left(\sum_{k=1}^{N}\frac{X_i(k)^2}{i_0^2}\right)\right) \\
&= \frac{R_0^2}{N^2}\left(\frac{\sigma^2(X_u)}{u_0^2} + \frac{\sigma^2(X_i)}{i_0^2}\right) = \frac{R_0^2}{N^2}\left(\frac{\sigma_u^2}{u_0^2} + \frac{\sigma_i^2}{i_0^2}\right)
\end{aligned}$$

Analysis of $\hat{\theta}_{\mathbf{LS}} = R_{\mathbf{LS}}$:

Considering the variance, Definition 1.11 reveals

$$\sigma^2\left(\hat{\theta}_{\mathrm{LS}}\right) = \mathrm{E}\left(\left(\hat{\theta}_{\mathrm{LS}} - \mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right)\right)^2\right).$$

Similar to the estimator $\hat{\theta}_{\mathrm{EV}}$, we first reconsider $\hat{\theta}_{\mathrm{LS}}$ and approximate it using only zero and first order terms:

$$\hat{\theta}_{\mathrm{LS}} \approx \frac{u_0 i_0 + \frac{u_0}{N}\sum_{k=1}^{N} X_i(k) + \frac{i_0}{N}\sum_{k=1}^{N} X_u(k)}{i_0^2 + \frac{2i_0}{N}\sum_{k=1}^{N} X_i(k)} \cdot \frac{i_0^2 - \frac{2i_0}{N}\sum_{k=1}^{N} X_i(k)}{i_0^2 - \frac{2i_0}{N}\sum_{k=1}^{N} X_i(k)}$$

$$\overset{\text{neglect 2nd order}}{=} \frac{u_0 i_0^3 + \frac{u_0 i_0^2}{N}\sum_{k=1}^{N} X_i(k) + \frac{i_0^3}{N}\sum_{k=1}^{N} X_u(k) - \frac{2u_0 i_0^2}{N}\sum_{k=1}^{N} X_i(k)}{i_0^4}$$

$$= \frac{u_0}{i_0} + \frac{1}{N i_0}\sum_{k=1}^{N} X_u(k) - \frac{u_0}{N i_0^2}\sum_{k=1}^{N} X_i(k)$$

$$= R_0\left(1 + \frac{1}{N}\sum_{k=1}^{N}\frac{X_u(k)}{u_0} - \frac{1}{N}\sum_{k=1}^{N}\frac{X_i(k)}{i_0}\right)$$

This expression is identical to $\hat{\theta}_{\mathrm{EV}}$, and we therefore obtain

$$\sigma^2\left(\hat{\theta}_{\mathrm{LS}}\right) = \sigma^2\left(\hat{\theta}_{\mathrm{EV}}\right) = \frac{R_0^2}{N^2}\left(\frac{\sigma_u^2}{u_0^2} + \frac{\sigma_i^2}{i_0^2}\right).$$

Analysis of $\hat{\theta}_{\mathbf{SA}} = R_{\mathbf{SA}}$:

Computing the variance via Definition 1.11, that is

$$\sigma^2\left(\hat{\theta}_{\mathrm{SA}}\right) = \mathrm{E}\left(\left(\hat{\theta}_{\mathrm{SA}} - \mathrm{E}\left(\hat{\theta}_{\mathrm{SA}}\right)\right)^2\right).$$

Similar to the estimator $\hat{\theta}_{\mathrm{EV}}$, we first reconsider $\hat{\theta}_{\mathrm{SA}}$ and approximate it using only zero and first order terms:

$$\hat{\theta}_{\mathrm{SA}} = \frac{1}{N}\sum_{k=1}^{N}\frac{u_0 + X_u(k)}{i_0 + X_i(k)} \cdot \frac{i_0 - X_i(k)}{i_0 - X_i(k)}$$

$$\overset{\text{neglect 2nd order}}{=} \frac{1}{N}\sum_{k=1}^{N}\frac{u_0 i_0 + i_0 X_u(k) - u_0 X_i(k)}{i_0^2}$$

$$= R_0\left(1 + \frac{1}{N}\sum_{k=1}^{N}\frac{X_u(k)}{u_0} - \frac{1}{N}\sum_{k=1}^{N}\frac{X_i(k)}{i_0}\right)$$

This expression is identical to $\hat{\theta}_{\mathrm{EV}}$ and $\sigma^2\left(\hat{\theta}_{\mathrm{EV}}\right)$, and we therefore obtain

$$\sigma^2\left(\hat{\theta}_{\mathrm{SA}}\right) = \sigma^2\left(\hat{\theta}_{\mathrm{EV}}\right) = \sigma^2\left(\hat{\theta}_{\mathrm{LS}}\right) = \frac{R_0^2}{N^2}\left(\frac{\sigma_u^2}{u_0^2} + \frac{\sigma_i^2}{i_0^2}\right).$$

5.3.4 Assessment

Unbiasedness

The simple approach estimator $\hat{\theta}_{\mathrm{SA}}$ continuously overestimates the true value of the parameter, hence it is biased. The error–in–variable estimator $\hat{\theta}_{\mathrm{EV}}$ approximate the true value of the parameter and is therefore unbiased. Last, the least square estimator $\hat{\theta}_{\mathrm{LS}}$ underestimates the true value of the parameter. Like $\hat{\theta}_{\mathrm{SA}}$, it is biased.
Hence, from a bias point of view, the error–in–variable estimator is preferable.

Consistency

Within our analysis, we found that all three estimators are converging in distribution. Therefore, they also converge in probability and are consistent.

Efficiency

Last, we have seen, all three estimators show the same second moment. Hence, none of them is more efficient.

Concluding, the error–in–variable estimator is the choice at hand given the presented alternatives, since it performs as well as the other estimators in all three categories and gives better results in terms of bias.

Chapter 6

Least square estimation

Within this chapter, we pursue a systematic approach to the parameter estimation problem. In particular, we ask what criterion should be used to match the model to the data. To answer this question, we use a statistical approach to select a criterion to measure the quality of the resulting fit. After defining the problem at hand, we discuss two estimators here, the least square and the weighted least square estimator. Note that it is also possible to use other estimator types such as the least absolute values.

6.1 Problem definition

Let a input-output model be given by

$$y_0(k, \theta) = g\left(u_0(k), \theta\right) \tag{6.1}$$

where $k \in \mathbb{N}_0$ represents the measurement index, $y_0(k) \in \mathbb{Y} = \mathbb{R}^{n_y}$ the output, $u(k) \in \mathbb{U} = \mathbb{R}^{n_u}$ the input and $\theta \in \Theta = \mathbb{R}^{n_\theta}$ the true parameter vector.

The aim is to estimate the parameters from noisy observations at the output of the systems. To this end, we assume that the output is separated into a deterministic and a probabilistic part $y_0(\cdot)$ and $X_y(\cdot)$:

Assumption 6.1
Given an input output model, noise disturbances only occur within the output observations

$$y(k, X_y) = y_0(k) + X_y(k) \tag{6.2}$$

where $y(k, X_y)$ and $y_0(k)$ represent the modeled and nominal output and $X_y(k)$ denotes the random output variable.

To achieve the described goal, we minimize the errors

$$e(k, \theta) = z_k - y_0(k, \theta) \tag{6.3}$$

between the measured and the estimated/modeled values z_k and $y_0(k, \theta)$ respectively.
The quality of a fit can then be expressed via a cost criterion. One such criterion is given by the so called nonlinear least squares (NLS), which is derived from the minimization of the sum of squared values:

Definition 6.2 (Least Square estimator)
The least square estimator $\hat{\theta}_{\mathrm{NLS}}(N)$ is given by

$$\hat{\theta}_{\mathrm{NLS}}(N) = \underset{\theta}{\operatorname{argmin}}\, J_{\mathrm{NLS}}(N,\theta), \quad \text{with } J_{\mathrm{NLS}}(N,\theta) := \frac{1}{2}\sum_{k=1}^{N} e^2(k,\theta). \tag{6.4}$$

Alternatively, one may also use the sum of absolute values

$$\hat{\theta}_{\mathrm{NLA}}(N) = \underset{\theta}{\operatorname{argmin}}\, J_{\mathrm{NLA}}(N,\theta), \quad \text{with } J_{\mathrm{NLA}}(N,\theta) := \frac{1}{2}\sum_{k=1}^{N} |e(k,\theta)| \tag{6.5}$$

The least square estimator (6.4) is the most popular one. Yet, by choosing the cost function arbitrarily as we have done it here, it is not at all clear that the result is not necessarily optimal. Least squares, however, are strongly motivated by numerical aspects. This is due to the fact that minimizing a least squares cost function is usually less involved than alternative cost functions. Here, the quadratic nature can be exploited which reveal that the necessary first order conditions for an optimal are also sufficient. Still, we like to mention that the nonlinear least absolute values (6.5) are less sensitive to outliers in the data and may for this reason be interesting in certain applications as well.

As we have seen in Section 5.3, even within the class of least squares different estimators can be designed which lead to results with different properties. In context of an optimal outcome with respect to the properties presented in Chapter 5, it is important to see where the noise enters into the raw data. Thereafter, a cost function should be selected that explicitly accounts for these errors.

6.2 Linear least square

If the model is chosen to be linear-in-parameter θ, equations (6.1) and (6.3) simplify to

$$y_0(\theta) = K(u_0)\,\theta \tag{6.6}$$

with input/output matrix $K(u) \in \mathbb{R}^{N\times n_\theta}$, input vector $u_0 \in \mathbb{R}^N$ and output vector $y_0 \in \mathbb{R}^N$. Hence, the error can be rewritten as

$$e(\theta) = z - K(u_0)\,\theta \tag{6.7}$$

where $z \in \mathbb{R}^N$ represents the vector of measurements. The quality criterion $J_{\mathrm{NLS}}(N,\theta)$ reduces to a linear one given by

$$\begin{aligned} J_{\mathrm{LS}}(N,\theta) &:= \frac{1}{2}e(\theta)^\top e(\theta) = \frac{1}{2}\left(z - K(u_0)\,\theta\right)^\top \left(z - K(u_0)\,\theta\right) \\ &= \frac{1}{2}\sum_{k=1}^{N}\left(z_k - K(u_0(k))\,\theta\right)^2. \end{aligned} \tag{6.8}$$

Hence, we obtain the following:

Definition 6.3 (Linear least square estimation problem)
The linear least square estimate $\hat{\theta}_{\mathrm{LS}}(N)$ is given by

$$\hat{\theta}_{\mathrm{LS}}(N) = \underset{\theta}{\operatorname{argmin}}\, J_{\mathrm{LS}}(N,\theta) \tag{6.9}$$

with $J_{\mathrm{LS}}(N,\theta)$ according to (6.8).

Since J_{LS} is quadratic, we can compute the minimizer of this loss function explicitly via

$$\frac{\partial J_{\mathrm{LS}}(N,\theta)}{\partial \theta} = 0.$$

This gives us

$$0 = \frac{\partial J_{\mathrm{LS}}(N,\theta)}{\partial \theta} = e(\theta)^{\top}\frac{\partial e(\theta)}{\theta} = e(\theta)^{\top}(-K\left(u_0\right)) = -K\left(u_0\right)^{\top} e(\theta).$$

Hence, we have to solve the equation

$$-K\left(u_0\right)^{\top}\left(z - K\left(u_0\right)\theta\right) = 0$$

for θ which reveals the solution

$$\hat{\theta}_{\mathrm{LS}}(N) = \theta = \left(K\left(u_0\right)^{\top} K\left(u_0\right)\right)^{-1} K\left(u_0\right)^{\top} z.$$

Concluding, we have shown the following:

Theorem 6.4 (Solution of $\hat{\theta}_{\mathrm{LS}}$)
The solution to the linear least square estimation problem (6.9), (6.8)

$$\hat{\theta}_{LS}(N) = \underset{\theta}{\operatorname{argmin}}\, J_{LS}(N,\theta) \quad \textit{with} \quad J_{LS}(N,\theta) = \frac{1}{2}\left(z - K\left(u_0\right)\theta\right)^{\top}\left(z - K\left(u_0\right)\theta\right)$$

is given by

$$\hat{\theta}_{LS}(N) = \left(K\left(u_0\right)^{\top} K\left(u_0\right)\right)^{-1} K\left(u_0\right)^{\top} z. \tag{6.10}$$

Here, we like to note that one typically does not compute the least square estimator via formula (6.10), but instead solves the linear equation

$$\left(K\left(u_0\right)^{\top} K\left(u_0\right)\right)\hat{\theta}_{\mathrm{LS}}(N) = K\left(u_0\right)^{\top} z$$

and avoids inverting the matrix $K\left(u_0\right)^{\top} K\left(u_0\right)$. Unfortunately, the matrix $K\left(u_0\right)^{\top} K\left(u_0\right)$ causes numerical instability since eigenvalues are raised by the power of two. There are, however, ways to compute the solution of the linear least square estimation problem (6.9), (6.8) by other, more stable algorithms such as the QR decomposition.

In order to generate the matrix K, one has to reformulate the model of the problem (6.6) combined for the available inputs and outputs $u(k)$ and $y(k)$, $k = 1, \ldots, N$. Let us consider two examples:

Example 6.5
The simplest model is given by

$$y_0 = \theta,$$

which is independent from the input. Combining all available outputs $y(k)$, $k = 1, \ldots, N$ *this reads*

$$\begin{aligned} y_0(1) &= \theta \\ &\vdots \\ y_0(N) &= \theta \end{aligned}$$

and reveals the matrix

$$K = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

Using formula (6.10) *we obtain the estimator*

$$\begin{aligned} \hat{\theta}_{LS}(N) &= \left(K^\top K\right)^{-1} K^\top z \\ &= \left((1, \ldots, 1) \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \right)^{-1} (1, \ldots, 1)\, z \\ &= (N)^{-1} (1, \ldots, 1)\, z = \frac{1}{N} \sum_{k=1}^{N} z(k). \end{aligned}$$

To illustrate the result, we chose measurements of the form

$$z = \theta \quad \textit{with} \quad \theta = 1 + 0.2 X_y,$$

where X_y *is normally independently distributed with mean* 0 *and standard deviation* 1, *i.e.* $X_y \in \mathcal{N}(0, 1)$ *and* $\theta \in \mathcal{N}(1, 0.2)$. *Considering* 100 *such measurements, we obtain the result display in Figure 6.1. The respective program is displayed in Program A.14.*

Example 6.6
Given the model

$$y = u_1 \theta_1 + u_2^2 \theta_2$$

we can combine inputs and outputs to obtain

$$\begin{aligned} y(1) &= u_1(1)\theta_1 + u_2^2(1)\theta_2 \\ &\vdots \\ y(N) &= u_1(N)\theta_1 + u_2^2(N)\theta_2. \end{aligned}$$

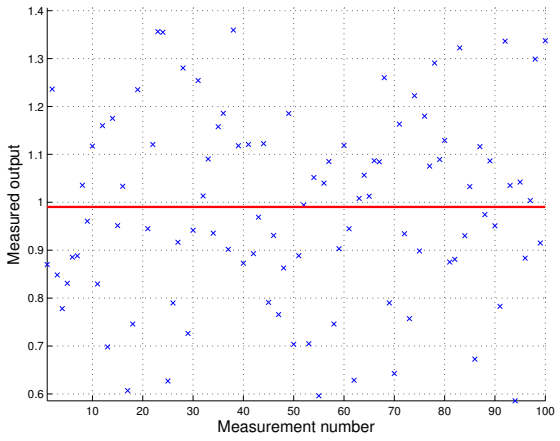


Figure 6.1: Sample measurements and estimation for Example 6.5

Hence, we have (6.6)*, i.e.*

$$y_0 = K(u_0)\,\theta$$

with

$$y_0 = \begin{pmatrix} y(1) \\ \vdots \\ y(N) \end{pmatrix}, \quad u_0 = \begin{pmatrix} u_1(1) \\ u_2(1) \\ \vdots \\ u_1(N) \\ u_2(N) \end{pmatrix}, \quad \textit{and} \quad K(u_0) = \begin{pmatrix} u_1(1) & u_2^2(1) \\ \vdots & \vdots \\ u_1(N) & u_2^2(N) \end{pmatrix}.$$

Now, we can apply formula (6.10) *to obtain the estimator*

$$\begin{aligned}
\hat{\theta}_{LS}(N) &= \left(K(u_0)^\top K(u_0)\right)^{-1} K(u_0)^\top z \\
&= \left(\begin{pmatrix} u_1(1) & u_2^2(1) \\ \vdots & \vdots \\ u_1(N) & u_2^2(N) \end{pmatrix}^\top \begin{pmatrix} u_1(1) & u_2^2(1) \\ \vdots & \vdots \\ u_1(N) & u_2^2(N) \end{pmatrix}\right)^{-1} \begin{pmatrix} u_1(1) & u_2^2(1) \\ \vdots & \vdots \\ u_1(N) & u_2^2(N) \end{pmatrix}^\top z \\
&= \left(\begin{pmatrix} u_1(1) & \dots & u_1(N) \\ u_2^2(1) & \dots & u_2^2(N) \end{pmatrix} \begin{pmatrix} u_1(1) & u_2^2(1) \\ \vdots & \vdots \\ u_1(N) & u_2^2(N) \end{pmatrix}\right)^{-1} \begin{pmatrix} u_1(1) & \dots & u_1(N) \\ u_2^2(1) & \dots & u_2^2(N) \end{pmatrix} z \\
&= \begin{pmatrix} \sum_{k=1}^{N} u_1^2(k) & \sum_{k=1}^{N} u_1(k)u_2^2(k) \\ \sum_{k=1}^{N} u_1(k)u_2^2(k) & \sum_{k=1}^{N} u_2^4(k) \end{pmatrix}^{-1} \begin{pmatrix} \sum_{k=1}^{N} u_1(k)z_k \\ \sum_{k=1}^{N} u_2^2(k)z_k \end{pmatrix}.
\end{aligned}$$

The estimator can be computed by solving the two–dimensional linear equation $A\hat{\theta}_{LS}(N) = b$

with

$$A = \begin{pmatrix} \sum_{k=1}^{N} u_1^2(k) & \sum_{k=1}^{N} u_1(k)u_2^2(k) \\ \sum_{k=1}^{N} u_1(k)u_2^2(k) & \sum_{k=1}^{N} u_2^4(k) \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} \sum_{k=1}^{N} u_1(k)z_k \\ \sum_{k=1}^{N} u_2^2(k)z_k \end{pmatrix}.$$

To illustrate the result, we chose N inputs

$$u_1(k) = 1 + \frac{k}{N-1},$$
$$u_2(k) = 2 + \frac{10k}{N-1},$$

which gives us u_0 and $K(u_0)$. Then, we generated measurements of the form

$$z = K(u_0)\,\theta$$

with

$$\theta_1 = 1 + 2X_{y,1}$$
$$\theta_2 = 2 + 1X_{y,2}$$

where $X_{y,1}, X_{y,2}$ are normally independently distributed with mean 0 and standard deviation 1, i.e. $\theta_1 \in \mathcal{N}(1,2)$ and $\theta_2 \in \mathcal{N}(2,1)$. Considering 100 such measurements, we obtain the result display in Figure 6.2. The respective program is displayed in Program A.15.

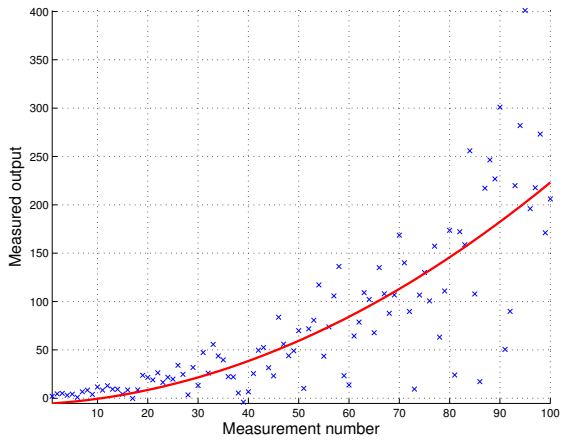


Figure 6.2: Sample measurements and estimation for Example 6.6

6.2.1 Properties of the linear least square estimator

Note that we did not formulate any assumptions on the behavior of the noise X_y to compute formula (6.10), but instead calculated it directly from the measurements and the model without

bothering about the noise behavior. However, in order to make statements about the properties of the estimator, it is necessary to give some specifications on the noise behavior.

The expected value of the estimator $\hat{\theta}_{\mathrm{LS}}$ regarding model outputs, i.e. by considering $z = y(X_y)$, can be computed via

$$\begin{aligned}
\mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right) &\overset{(6.10)}{=} \mathrm{E}\left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top y(X_y)\right) \\
&\overset{(6.2)}{=} \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top \mathrm{E}\left(y_0 + X_y\right) \\
&= \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top y_0 + \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top \mathrm{E}\left(X_y\right) \\
&\overset{(6.6)}{=} \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top K\left(u_0\right)\theta + \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top \mathrm{E}\left(X_y\right) \\
&= \theta + \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top \mathrm{E}\left(X_y\right).
\end{aligned}$$

Now, in order for $\hat{\theta}_{\mathrm{LS}}$ to be unbiased, we require $\mathrm{E}\left(X_y\right) = 0$.

Corollary 6.7 (Unbiasedness of $\hat{\theta}_{\mathrm{LS}}$)
If the model is linear-in-parameter and the probabilistic part of the output satisfies $\mathrm{E}\left(X_y\right) = 0$*, then the least square estimator* $\hat{\theta}_{LS}$ *is unbiased.*

The second interesting characteristic is the covariance matrix of the estimator $\hat{\theta}_{\mathrm{LS}}$. Here, we see the following:

$$\begin{aligned}
\mathrm{Cov}\left(\hat{\theta}_{\mathrm{LS}}, \hat{\theta}_{\mathrm{LS}}\right) &= \mathrm{E}\left(\left(\hat{\theta}_{\mathrm{LS}} - \mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right)\right)\left(\hat{\theta}_{\mathrm{LS}} - \mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right)\right)^\top\right) \\
&= \mathrm{E}\left(\left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top \mathrm{E}\left(X_y\right)\right)\left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top \mathrm{E}\left(X_y\right)\right)^\top\right) \\
&= \left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top\right) \mathrm{E}\left(X_y X_y^\top\right) \left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top\right)^\top \\
&= \left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top\right) \mathrm{Cov}\left(X_y, X_y\right) \left(\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top\right)^\top
\end{aligned}$$

Similar to Corollary 6.7, we can make the following conclusion regarding the covariance matrix of the estimator $\hat{\theta}_{\mathrm{LS}}$.

Corollary 6.8 (Covariance of $\hat{\theta}_{\mathrm{LS}}$)
Consider a linear-in-parameter model. If the disturbing noise X_y *is white and uncorrelated, i.e.* $\mathrm{Cov}\left(X_y, X_y\right) = \sigma^2\left(X_y\right) Id_{n_\theta}$*, then the covariance matrix of the estimator* $\hat{\theta}_{LS}$ *is given by*

$$\mathrm{Cov}\left(\hat{\theta}_{LS}, \hat{\theta}_{LS}\right) = \sigma^2\left(X_y\right)\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} \tag{6.11}$$

Else, the covariance matrix can be computed via

$$\mathrm{Cov}\left(\hat{\theta}_{LS}, \hat{\theta}_{LS}\right) = L\,\mathrm{Cov}\left(X_y, X_y\right) L^\top. \tag{6.12}$$

where $L := \left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} K\left(u_0\right)^\top$.

To illustrate this, let us reconsider Examples 6.5 and 6.6.

Example 6.9
Given Assumption 6.1, consider the model

$$y_0 = \theta$$

and suppose the noise X_y to be white and uncorrelated. Using $K = (1, \dots, 1)^\top$, we can evaluate (6.11) *and obtain*

$$\mathrm{Cov}\left(\hat{\theta}_{LS}, \hat{\theta}_{LS}\right) = \frac{1}{N}\sigma^2\left(X_y\right).$$

Example 6.10
Consider the model

$$y = u_1\theta_1 + u_2^2\theta_2$$

and again assume Assumption 6.1 to hold and the noise X_y to be white and uncorrelated, i.e. $\mathrm{Cov}\left(X_y, X_y\right) = \sigma^2\left(X_y\right) Id_{n_\theta}$. *Then, we obtain*

$$\begin{aligned}\mathrm{Cov}\left(\hat{\theta}_{LS}, \hat{\theta}_{LS}\right) &= \sigma^2\left(X_y\right)\left(K\left(u_0\right)^\top K\left(u_0\right)\right)^{-1} \\ &\overset{\text{Example 6.6}}{=} \sigma^2\left(X_y\right)\begin{pmatrix} \sum_{k=1}^{N} u_1^2(k) & \sum_{k=1}^{N} u_1(k)u_2^2(k) \\ \sum_{k=1}^{N} u_1(k)u_2^2(k) & \sum_{k=1}^{N} u_2^4(k) \end{pmatrix}^{-1}.\end{aligned}$$

Here, we like to note that within the least square estimator (6.10)

$$\left(K\left(u_0\right)^\top K\left(u_0\right)\right)\hat{\theta}_{\mathrm{LS}}(N) = K\left(u_0\right)^\top z$$

the multiplication $K\left(u_0\right)^\top z$ includes an $N \times n_\theta$ and a $n_\theta \times 1$ matrix. To this sum, we can apply the central limit theorem we gives us that the estimator $\hat{\theta}_{\mathrm{LS}}$ asymptotically converges to a Gaussian distribution **even** if X_y is not Gaussian distributed, that is

$$\lim_{N\to\infty} \hat{\theta}_y = \mathcal{N}\left(\mathrm{E}\left(\hat{\theta}_{\mathrm{LS}}\right), \mathrm{Cov}\left(\hat{\theta}_{\mathrm{LS}}, \hat{\theta}_{\mathrm{LS}}\right)\right).$$

6.3 Weighted least square

So far, we have only been looking at equally weighted measurements in (6.4) (and (6.5)). However, it may be desirable to change this property, e.g. to suppress measurements with high uncertainty and to emphasize those with low uncertainty. To design such a weighting, the covariance matrix can be used.

In practice, it is not always clear which weighting should be used. Yet certain indicators can be used to improve the estimator. For example, if it is known that the model exhibits errors, then utilizing the covariance matrix may not be a good idea. Instead, the user may prefer to

put a dedicated wighting in order to keep the model errors small in some specific operation regions.

Definition 6.11 (Weighted Least Square estimator)
The weighted least square estimator $\hat{\theta}_{\text{WLS}}(N)$ is given by

$$\hat{\theta}_{\text{WLS}}(N) = \underset{\theta}{\operatorname{argmin}}\, J_{\text{WLS}}(N,\theta), \quad \text{with } J_{\text{WLS}}(N,\theta) := \frac{1}{2} e(\theta)^\top W e(\theta) \tag{6.13}$$

where $W \in \mathbb{R}^{N\times N}$ is symmetric and positive definite.

Again we can utilize the quadratic nature of J_{WLS} to compute the minimizer of this loss function explicitely via

$$\frac{\partial J_{\text{WLS}}(N,\theta)}{\partial \theta} = 0.$$

This gives us

$$0 = \frac{\partial J_{\text{WLS}}(N,\theta)}{\partial \theta} = e(\theta)^\top W^\top \frac{\partial e(\theta)}{\theta} = e(\theta)^\top W^\top (-K(u_0)) = -K(u_0)^\top W e(\theta).$$

Solve the equation

$$-K(u_0)^\top W (z - K(u_0)\,\theta) = 0$$

for θ reveals

$$\hat{\theta}_{\text{WLS}}(N) = \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W z.$$

Hence, we have shown the following:

Theorem 6.12 (Solution of $\hat{\theta}_{\text{WLS}}$)
The solution to the weighted linear least square estimation problem (6.13) *is given by*

$$\hat{\theta}_{WLS}(N) = \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W z. \tag{6.14}$$

Let us reconsider the more elaborate Example 6.6 for the weighted linear least square estimator:

Example 6.13
Recall the model

$$y = u_1\theta_1 + u_2^2\theta_2.$$

Combining inputs and outputs

$$\begin{aligned} y(1) &= u_1(1)\theta_1 + u_2^2(1)\theta_2 \\ &\vdots \\ y(N) &= u_1(N)\theta_1 + u_2^2(N)\theta_2 \end{aligned}$$

we have (6.6)

$$y_0 = K(u_0)\,\theta$$

with

$$y_0 = \begin{pmatrix} y(1) \\ \vdots \\ y(N) \end{pmatrix}, \quad u_0 = \begin{pmatrix} u_1(1) \\ u_2(1) \\ \vdots \\ u_1(N) \\ u_2(N) \end{pmatrix}, \quad \textit{and} \quad K(u_0) = \begin{pmatrix} u_1(1) & u_2^2(1) \\ \vdots & \vdots \\ u_1(N) & u_2^2(N) \end{pmatrix}.$$

Now, we choose

$$W = diag(0, \frac{1}{N-1}, \frac{2}{N-1}, \ldots, 1) \in \mathbb{R}^{N \times N},$$

i.e., measurements with larger index k are associated with higher weights.
To illustrate the difference between the $\hat{\theta}_{LS}$ and $\hat{\theta}_{WLS}$, we again choose N inputs

$$u_1(k) = 1 + \frac{k}{N-1},$$
$$u_2(k) = 2 + \frac{10k}{N-1},$$

which gives us u_0 and $K(u_0)$. Then, we generated measurements of the form

$$z = K(u_0)\,\theta$$

with

$$\theta_1 = 1 + 2X_{y,1}$$
$$\theta_2 = 2 + 1X_{y,2}$$

where $X_{y,1}, X_{y,2}$ are normally independently distributed with mean 0 and standard deviation 1, i.e. $\theta_1 \in \mathcal{N}(1,2)$ and $\theta_2 \in \mathcal{N}(2,1)$. Considering 100 such measurements, we obtain the result display in Figure 6.3. The respective program is displayed in Program A.16. Here, we see that the estimated curve deviates for measurements with small index k. This is to be expected since the respective weights are very small.

6.3.1 Properties of the weighted linear least square estimator

Turn toward the bias of the weighted linear least square estimator, we can utilize $z = y(X_y)$ to compute

$$\begin{aligned} \mathrm{E}\left(\hat{\theta}_{\mathrm{WLS}}\right) &\overset{(6.14)}{=} \mathrm{E}\left(\left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W y(X_y)\right) \\ &\overset{(6.2)}{=} \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W \mathrm{E}(y_0 + X_y) \\ &= \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W y_0 + \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W \mathrm{E}(X_y) \end{aligned}$$

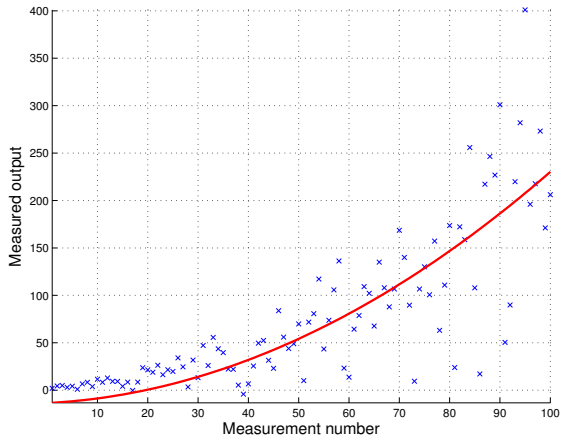


Figure 6.3: Sample measurements and estimation for Example 6.13

$$\overset{(6.6)}{=} \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W K(u_0)\,\theta + \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W \mathrm{E}(X_y)$$
$$= \theta + \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W \mathrm{E}(X_y).$$

Now, in order for $\hat{\theta}_{\mathrm{WLS}}$ to be unbiased, we require $\mathrm{E}(X_y) = 0$.

Corollary 6.14 (Unbiasedness of $\hat{\theta}_{\mathrm{WLS}}$)
If the model is linear-in-parameter and the probabilistic part of the output satisfies $\mathrm{E}(X_y) = 0$, *then the least square estimator* $\hat{\theta}_{WLS}$ *is unbiased.*

Similarly, we can compute the covariance matrix of the estimator $\hat{\theta}_{\mathrm{WLS}}$ using the arguments from the unweighted case. Here, we use the abbreviation $K := K(u_0)$.

$$\mathrm{Cov}\left(\hat{\theta}_{\mathrm{WLS}}, \hat{\theta}_{\mathrm{WLS}}\right) = \mathrm{E}\left(\left(\hat{\theta}_{\mathrm{WLS}} - \mathrm{E}(\theta)\right)\left(\hat{\theta}_{\mathrm{WLS}} - \mathrm{E}(\theta)\right)^\top\right)$$
$$\overset{(6.14)}{=} \mathrm{E}\left(\left(\left(K^\top W K\right)^{-1} K^\top W X_y\right)\left(\left(K^\top W K\right)^{-1} K^\top W X_y\right)^\top\right)$$
$$= \left(\left(K^\top W K\right)^{-1} K^\top W\right) \mathrm{E}\left(X_y X_y^\top\right) \left(\left(K^\top W K\right)^{-1} K^\top W\right)^\top$$
$$= \left(\left(K^\top W K\right)^{-1} K^\top W\right) \mathrm{Cov}(X_y, X_y) \left(\left(K^\top W K\right)^{-1} K^\top W\right)^\top$$

Hence, we can conclude the following about the covariance of $\hat{\theta}_{\mathrm{WLS}}$:

Corollary 6.15 (Covariance of $\hat{\theta}_{\mathrm{WLS}}$)
Consider a linear-in-parameter model. Then the covariance matrix of the estimator $\hat{\theta}_{WLS}$ *is given by*

$$\mathrm{Cov}\left(\hat{\theta}_{WLS}, \hat{\theta}_{WLS}\right) = L\,\mathrm{Cov}(X_y, X_y)\,L^\top \tag{6.15}$$

where $L := \left(K(u_0)^\top W K(u_0)\right)^{-1} K(u_0)^\top W$.

This result allows for a very interesting conclusion shown in [1], namely that the covariance matrix can be minimized if the weight is chosen as the inverse of the covariance matrix of the random variable X_y, that is $W = \mathrm{Cov}\left(X_y, X_y\right)^{-1}$.

Corollary 6.16 (Minimal covariance of $\hat{\theta}_{\mathrm{WLS}}$)
Consider a linear-in-parameter model. If the weighting matrix of the weighted linear least square estimator $\hat{\theta}_{WLS}$ is chosen as $W = \mathrm{Cov}\left(X_y, X_y\right)^{-1}$, then the covariance matrix of $\hat{\theta}_{WLS}$ is minimal and given by

$$\mathrm{Cov}\left(\hat{\theta}_{WLS}, \hat{\theta}_{WLS}\right) = \left(K\left(u_0\right)^{\top} W K\left(u_0\right)\right)^{-1}. \tag{6.16}$$

Chapter 7

Maximum likelihood and Bayes estimator

As we have seen in the previous Section 6.3, the covariance matrix of the noise may be used as weighting matrix to incorporate prior knowledge about the noise of the measurements. Yet, a full stochastic characterization requires the probability density function of the noise distortions. Given such a knowledge, it may be possible to obtain results, which are even better than those of the weighted linear least square estimator. The maximum likelihood estimator offers a theoretical framework to incorporate the knowledge about the distribution of the noise distortions in the estimator.

The Bayes estimator extends the maximum likelihood estimator by incorporating knowledge on the probability distribution function of the parameter itself. Hence, if such information is at hand, the Bayes estimator supersedes the maximum likelihood estimator. Unfortunately, in practical applications the probability density function of the parameter is hardly ever known, which renders the estimator to be impractical.

7.1 Maximum likelihood estimator

The probability density function f_{X_y} of the noise determines the conditional probability density function $f(y_0 \mid \theta)$ of the model of the measurements stated in the previous Chapter 6

$$y_0(k) = g\left(u_0(k), \theta\right) \tag{6.1}$$

describing the system and the inputs that excite the system. Similarly, Assumption 6.1 shall hold, i.e. the noise enters the model additively

$$y(k, X_y) = y_0(k) + X_y(k) \tag{6.2}$$

where $y(k, X_y)$ and $y_0(k)$ represent the modeled and nominal output and $X_y(k)$ denotes the random output variable. Then, the likelihood function becomes

$$f(y(k, X_y) \mid u_0, \theta) := f_{X_y}\left(y(k, X_y) - g\left(u_0(k), \theta\right)\right). \tag{7.1}$$

The maximum likelihood procedure now consists of two steps:

Algorithm 7.1 (Maximum likelihood procedure)
Input: Probability density function f_{X_y} and measurements z

- Plug actual measurements z into (7.1) for variable $y(k, X_y)$.
- Consider θ as the free variable and maximize the conditional probability density function

Output: Maximum likelihood estimate

$$\hat{\theta}_{\mathrm{ML}}(N) = \underset{\theta}{\operatorname{argmax}} f(z \mid u_0, \theta) \tag{7.2}$$

At first sight, this algorithm may appear simple, and indeed its applications is easy:

Example 7.2
Consider again the simplest model displayed in Example 6.5

$$y_0 = g(u_0, \theta) = \theta,$$

which is independent of the input. Assume that f_{X_y} is normal with zero mean and variance $\sigma^2(X_y)$, i.e.

$$f_{X_y}(x) = \frac{1}{\sqrt{2\pi\sigma^2(X_y)}} e^{-\frac{x^2}{2\sigma^2(X_y)}}.$$

Then, for each measurement z we obtain the conditional probability density function

$$\begin{aligned} f(z \mid u_0, \theta) &= f_{X_y}(z - g(u_0(k), \theta)) = f_{X_y}(z - \theta) \\ &= \frac{1}{\sqrt{2\pi\sigma^2(X_y)}} e^{-\frac{(z-\theta)^2}{2\sigma^2(X_y)}}. \end{aligned}$$

To maximize this expression, we have to minimize the exponent $\frac{(z-\theta)^2}{2\sigma^2(X_y)}$. Since $\sigma^2(X_y)$ is constant, this results in $\hat{\theta}_{ML} = z$.

Only one measurement is a very small batch, but we can see that the maximum likelihood estimator correctly identifies the measurement value. Next, we incorporate a whole series of measurements:

Example 7.3
Again we consider the model

$$y_0 = g(u_0, \theta) = \theta,$$

which is independent of the input and assume that f_{X_y} is normal with zero mean and variance $\sigma^2(X_y)$. Incorporating multiple independent measurements $z_1, \ldots, z_N$, the likelihood function is

$$\begin{aligned} f(z \mid u_0, \theta) &= f(z_1 \mid u_0, \theta) \cdot \ldots \cdot f(z_N \mid u_0, \theta) \\ &= f_{X_y}(z_1 - g(u_0(1), \theta)) \cdot \ldots \cdot f_{X_y}(z_N - g(u_0(N), \theta)) \\ &= f_{X_y}(z_1 - \theta) \cdot \ldots \cdot f_{X_y}(z_N - \theta). \end{aligned}$$

Hence, we obtain

$$f(z \mid u_0, \theta) = \left(\frac{1}{\sqrt{2\pi\sigma^2 (X_y)}} \right)^N e^{-\frac{1}{2\sigma^2(X_y)} \sum_{k=1}^{N} (z_k - \theta)^2}$$

and the minimizer becomes

$$\hat{\theta}_{ML}(N) = \frac{1}{N} \sum_{k=1}^{N} z_k$$

Note that in the previous example, the maximum likelihood estimator and the (weighted) least square estimator coincide. This is only the case for normally distributed errors.

Corollary 7.4
If $f_{X_y} \in \mathcal{N}(0, \sigma^2(X_y))$, then the $\hat{\theta}_{ML}(N) = \hat{\theta}_{LS}(N)$.

In general, we have the following:

Corollary 7.5
If f_{X_y} is normal and measurements z_k are iid, then the Maximum likelihood estimator problem is a linear least square problem.

7.1.1 Properties for normally independent distributions

Now that we have seen how the maximum likelihood estimator can be computed, let us consider some of its properties such as expected values of the estimated mean and the estimated variance. Here, we will focus on normally independent distributions. The computations, however, can also be performed for other distributions. In the upcoming Section 7.1.2, we provide some general statements on the maximum likelihood estimator.

Consider N samples z_k, $k = 1, \ldots, N$, which are normally independently distributed with mean μ and standard deviation σ. Then the likelihood function becomes

$$f(z \mid u_0, \theta) = f(z_1 \mid u_0, \theta) \cdot \ldots \cdot f(z_N \mid u_0, \theta) = \frac{1}{\sqrt{2\pi\sigma_y^2}} e^{-\frac{1}{2\sigma_y^2} \sum_{k=1}^{N} (z_k - \mu)^2}$$

and the loglikelihood function is

$$\ln f(z \mid u_0, \theta) = -\frac{N}{2} \ln\left(2\pi\sigma_y^2\right) - \frac{1}{2\sigma_y^2} \sum_{k=1}^{N} (z_k - \mu)^2 .$$

Then, we can compute the derivatives with respect to μ and σ^2, which gives us the necessary first order optimality conditions

$$\frac{\partial}{\partial \mu} \ln f(z \mid u_0, \theta) = \frac{1}{\sigma_y^2} \sum_{k=1}^{N} (z_k - \mu) \stackrel{!}{=} 0$$

$$\frac{\partial}{\partial \sigma_y^2} \ln f(z \mid u_0, \theta) = -\frac{N}{2\sigma_y^2} + \frac{1}{2\sigma_y^4} \sum_{k=1}^{N} (z_k - \mu)^2 \stackrel{!}{=} 0.$$

Solving these equations reveals

$$\mu\left(\hat{\theta}_{\mathrm{ML}}\right) = \frac{1}{N} \sum_{k=1}^{N} z_k \tag{7.3}$$

$$\sigma_y^2\left(\hat{\theta}_{\mathrm{ML}}\right) = \frac{1}{N} \sum_{k=1}^{N} \left(z_k - \mu\left(\hat{\theta}_{\mathrm{ML}}\right)\right)^2. \tag{7.4}$$

From (7.3), we directly obtain

$$\mathrm{E}\left(\mu\left(\hat{\theta}_{\mathrm{ML}}\right)\right) = \frac{1}{N} \sum_{k=1}^{N} \mathrm{E}\left(z_k\right) = \mu,$$

which shows that the mean of the maximum likelihood estimator is unbiased.

Corollary 7.6 (Unbiasedness of $\hat{\theta}_{\mathrm{ML}}$)
If the output of a system is normally independently distributed, then the maximum likelihood estimator $\hat{\theta}_{ML}$ given by

$$\mathrm{E}\left(\hat{\theta}_{ML}\right) = \mu\left(\hat{\theta}_{ML}\right) = \frac{1}{N} \sum_{k=1}^{N} z_k \tag{7.3}$$

is unbiased.

Moreover, we can utilize (7.4) to see

$$\begin{aligned}
\mathrm{E}\left(\sigma_y^2\left(\hat{\theta}_{\mathrm{ML}}\right)\right) &= \frac{1}{N} \sum_{k=1}^{N} \mathrm{E}\left(\left(z_k - \mu\left(\hat{\theta}_{\mathrm{ML}}\right)\right)^2\right) \\
&= \frac{1}{N} \sum_{k=1}^{N} \mathrm{E}\left(\left((z_k - \mu) - \left(\mu\left(\hat{\theta}_{\mathrm{ML}}\right) - \mu\right)\right)^2\right) \\
&= \frac{1}{N} \sum_{k=1}^{N} \mathrm{E}\left((z_k - \mu)^2 - 2\left(z_k - \mu\right)\left(\mu\left(\hat{\theta}_{\mathrm{ML}}\right) - \mu\right) + \left(\mu\left(\hat{\theta}_{\mathrm{ML}}\right) - \mu\right)^2\right) \\
&= \frac{1}{N} \sum_{k=1}^{N} \Big(\mathrm{E}\left((z_k - \mu)^2\right) - 2\mathrm{E}\left(\left(z_k - \mu\right)\left(\mu\left(\hat{\theta}_{\mathrm{ML}}\right) - \mu\right)\right) \\
&\qquad + \mathrm{E}\left(\left(\mu\left(\hat{\theta}_{\mathrm{ML}}\right) - \mu\right)^2\right)\Big) \\
&= \frac{1}{N} \sum_{k=1}^{N} \left(\sigma_y^2 - 2\mathrm{E}\left(\left(z_k - \mu\right)\left(\frac{1}{N}\sum_{k=1}^{N} z_k - \mu\right)\right) + \mathrm{E}\left(\left(\frac{1}{N}\sum_{k=1}^{N} z_k - \mu\right)^2\right)\right) \\
&= \frac{1}{N} \sum_{k=1}^{N} \left(\sigma_y^2 - 2\frac{1}{N}\mathrm{E}\left(\left(z_k - \mu\right)^2\right) + \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} \mathrm{E}\left(\left(z_i - \mu\right)\left(z_j - \mu\right)\right)\right)
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{N}\sum_{k=1}^{N}\left(\sigma_y^2 - 2\frac{1}{N}\sigma_y^2 + \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{E}\left((z_j-\mu)^2\right)\right)\\
&= \frac{1}{N}\sum_{k=1}^{N}\left(\sigma_y^2 - 2\frac{1}{N}\sigma_y^2 + \frac{1}{N}\sigma_y^2\right)\\
&= \frac{\sigma_y^2}{N}\sum_{k=1}^{N}\left(1-\frac{2}{N}+\frac{1}{N}\right) = \sigma_y^2\left(1-\frac{1}{N}\right).
\end{aligned}
$$

Hence, the variance of the maximum likelihood estimator is biased.

Corollary 7.7 (Covariance of $\hat{\theta}_{\mathrm{ML}}$)
If the output of a system is normally independently distributed, then the covariance matrix of the estimator $\hat{\theta}_{ML}$ given by

$$\mathrm{Cov}\left(\hat{\theta}_{ML},\hat{\theta}_{ML}\right) = \frac{1}{N}\sum_{k=1}^{N}\left(z_k - \mu\left(\hat{\theta}_{ML}\right)\right)^2 \tag{7.4}$$

is biased by a factor $1-1/k$.

As we can directly see, $\lim_{k\to\infty} 1-1/k = 1$ holds which gives us the following efficiency results:

Corollary 7.8 (Efficiency)
If the output of a system is normally independently distributed, then the covariance matrix of the estimator $\hat{\theta}_{ML}$ reaches the Cramer-Rao lower bound for $k\to\infty$.

From this analysis of the restricted normally independently distributed case, we now show some more general results.

7.1.2 General properties of the maximum likelihood estimator

In the literature, a series of important properties is tabled assuming well-defined experimental conditions. If these conditions are met, then the user knows in advance what properties the estimator will have without going through the complete development process. The results here are only stated. Full proofs can be found in, e.g., [3].

The following invariance principle is a very powerful tool. In particular, this principle allows us to condense the measurements, i.e. to lower the dimension of the measurement vector, without compromising the maximum likelihood property of the estimator. Additionally, transformations of estimators given by g can be analyzed easily.

Theorem 7.9 (Principle of invariance)
If $\hat{\theta}_{ML}$ is a maximum likelihood estimator of θ and $g:\mathbb{R}_\theta^n\to\mathbb{R}^{n_g}$ is a function with $n_g\le n_\theta<\infty$, then $\hat{\theta}_g = g(\hat{\theta}_{ML})$ is a maximum likelihood estimator of $g(\theta)$.

Regarding the properties we discussed in Chapter 5, one can show the consistency and efficiency of the maximum likelihood estimator.

Theorem 7.10 (Consistency)
If $\hat{\theta}_{ML}$ is a maximum likelihood estimator based on N iid random variables with n_θ independent of N, then $\hat{\theta}_{ML}(N)$ converges to y_0 almost surely, i.e.

$$\underset{N\to\infty}{\text{a.s.lim}}\, \hat{\theta}_{ML}(N) = \theta.$$

Theorem 7.11 (Asymptotic efficiency)
If $\hat{\theta}_{ML}(N)$ is a maximum likelihood estimator based on N iid random variables with n_θ independent of N, then $\hat{\theta}_{ML}(N)$ is asymptotically efficient, i.e. $\operatorname{Cov}\left(\hat{\theta}_{ML}(N), \hat{\theta}_{ML}(N)\right)$ *asymptotically reaches the Cramer–Rao lower bound.*

A last property, which we see here, is the so called asymptotic normality. The importance of this property is not only that it allows one to calculate uncertainty bounds on the estimates, but that it also guarantees that most of the probability mass gets more and more unimodally concentrated around its limiting value.

Theorem 7.12 (Asymptotic normality)
If $\hat{\theta}_{ML}(N)$ is a maximum likelihood estimator based on N iid random variables with n_θ independent of N, then $\hat{\theta}_{ML}(N)$ converges in law to a normal random variable.

7.2 Bayes estimator

In contrast to the maximum likelihood estimator, the Bayes estimator requires knowledge on the probability density function of both the noise on the measurements and the unknown parameters. The kernel of the Bayes estimator is the conditional probability density function of the unknown parameters θ with respect to the measurements z denoted by $f_\theta(\theta \mid u_0, z)$. This probability density function contains complete information about the parameters θ, given a set of inputs u_0 and respective measurements z. This makes it possible for the experimenter to determine the best estimate of θ for the given situation. To select this best value, it is necessary to lay down an objective criterion, i.e. the minimization of a risk function $C(\theta \mid \theta_0)$. The risk function then describes the cost of selecting the parameter θ if θ_0 is the true but unknown parameter. The estimated parameter $\hat{\theta}$ is found as the minimizer of the risk function weighted with the probability density function $f_\theta(\theta \mid u, z)$ over the range $\mathbb{D}$ of the parameter θ, i.e.

$$\hat{\theta}(N) = \underset{\theta_0}{\operatorname{argmin}} \int_{\theta\in\mathbb{D}} C(\theta \mid \theta_0)\, f_\theta(\theta \mid u, z)\, d\theta. \tag{7.5}$$

If the cost criterion is chosen in the form

- $C(\theta \mid \theta_0) = |\theta - \theta_0|^2$ (which leads to the mean value) or
- $C(\theta \mid \theta_0) = |\theta - \theta_0|$ (results in the median, which is less sensitive to outliers since these contribute less to the second criterion compared to the first one),

then a closed solution of (7.5) is known. In contrast to this „minimum risk“ estimators, one may also choose the criterion

$$\hat{\theta}_{\mathrm{BA}}(N) = \underset{\theta}{\operatorname{argmax}}\, f_\theta(\theta \mid u, z), \tag{7.6}$$

which reveals the Bayes estimator. In practice, it is very difficult to select the best out of these variants.

Here, we study the Bayes estimator in more detail. To search for the maximizer of (7.6), the Bayes rule

$$f_\theta(\theta \mid u, z) = \frac{f(z \mid \theta, u) f_\theta(\theta)}{f(z)}$$

is applied. In order to maximize the right hand side of this equation, it is sufficient to maximize its numerator as the denominator is independent of θ. Hence, the solution is given by looking for the maximum of

$$f(z \mid \theta, u) f_\theta(\theta).$$

As we can already see, a lot of a priori information is required to use the Bayes estimator, i.e. $f(z \mid \theta, u)$, which is also used in the maximum likelihood estimator in (7.1), and $f_\theta(\theta)$. Note that in many problems the probability density function $f_\theta(\theta)$ is unavailable, which renders the Bayes estimator to be barely used in practice.

Example 7.13
Let us reconsider Example 6.5 with modifications from Example 7.2, i.e. the model is given by

$$y_0 = g(u_0, \theta) = \theta,$$

which is independent of the input and f_{X_y} is normal with zero mean and variance $\sigma^2(X_y)$, i.e.

$$f_{X_y}(x) = \frac{1}{\sqrt{2\pi\sigma^2(X_y)}} e^{-\frac{x^2}{2\sigma^2(X_y)}}.$$

Additionally, the probability density function of θ is given by its mean w and standard deviation σ_w, that is

$$f_\theta(x) = \frac{1}{\sqrt{2\pi\sigma_w^2}} e^{-\frac{(x-w)^2}{2\sigma_w^2}}.$$

Since we have

$$f(z \mid \theta, u) = f(z \mid \theta) = f_{X_y}(X_y) = f_{X_y}(z - \theta),$$

the Bayes estimator is found by maximizing the expression

$$f(z \mid \theta, u) f_\theta(\theta) = \frac{1}{\sqrt{2\pi\sigma^2(X_y)}} e^{-\frac{(z-\theta)^2}{2\sigma^2(X_y)}} \frac{1}{\sqrt{2\pi\sigma_w^2}} e^{-\frac{(\theta-w)^2}{2\sigma_w^2}}$$

with respect to θ and the estimate becomes

$$\hat{\theta}_{BA} = \frac{z/\sigma^2(X_y) + w/\sigma_w^2}{1/\sigma^2(X_y) + 1/\sigma_w^2}.$$

The example shows two things: For one, if the quality of the a priori information w is high compared with the quality of the measurements, that is $\sigma_w^2 \ll \sigma^2(X_y)$, then the estimate is determined mainly by the a priori information. Conversely, if $\sigma_w^2 \ll \sigma^2(X_y)$, then the estimate is dominated by the information gained by measurements.

Example 7.14
Example 7.13 can be extended in a manner similar to Example 7.3 by considering N independent measurements $z_1, \ldots, z_N$. Then, the Bayes estimator becomes

$$\hat{\theta}_{BA}(N) = \frac{\sum_{k=1}^{N} z_N/\sigma^2(X_y) + w/\sigma_w^2}{N/\sigma^2(X_y) + 1/\sigma_w^2}.$$

From this last example, we can make the following conclusion:

Corollary 7.15
If f_{X_y} and f_θ are normal and measurements z_k are iid, then the Bayes estimator problem is a linear least square problem.

Chapter 8

Kalman filtering

The famous Kalman filter belongs to the class of so called recursive identification methods. The idea of such methods is to iteratively update the estimate utilizing new measurements at hand. Following this approach, an online processing of the results is possible. Additionally, one could generalize the approach by introducing a „forgetting factor“ to the cost function, which renders the method to be adaptive by design.

In contrast to our previous analysis of input–output models, the Kalman filter is designed for state space systems of the form

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k), \end{aligned} \tag{8.1}$$

which are subject to equation and measurement noise. Hence, we have a systematic distinction between the internal state x and the externally viewable measurements y. Here, we discuss basic properties of the Kalman filter and construct a respective algorithm. Since the Kalman filter idea is quite involved, we start of by a simple introduction into recursive identification based on the mean value calculation.

8.1 Recursive identification

There exist two systematically different ways to compute an estimator: In the first case, the optimization is postponed till all measurements are available. The second case, on the other hand, evaluates the estimate each time a new sample is available. So far, we have seen the postponement approach, but now we focus on the second recursive case.

A straightforward solution to generate such a procedure is to redo all the calculations after each sample. Such an approach is numerically robust and requires no further insight, yet it may be computationally expensive depending on the number of samples and the complexity of the computation process. For example, it is simple to recompute the mean value, but it is a complex task to solve a nonlinear optimization problem for a dynamical model. Hence, reformulating the problem such that only the newly required calculations are made, recuperating all the previous results, may allow us to generate a more efficient solution method.

Before coming to a more elaborate variant of this approach, we consider the simple example of the mean value computation

$$\hat{\theta}(N) = \frac{1}{N} \sum_{k=1}^{N} z_k.$$

Using this formula, we can recompute the mean value once a new measurement is available via

$$\hat{\theta}(N+1) = \frac{1}{N+1}\sum_{k=1}^{N+1} z_k.$$

To recuperate the previous sum, we can equivalently evaluate

$$\begin{aligned}\hat{\theta}(N+1) &= \frac{1}{N+1}\sum_{k=1}^{N} z_k + \frac{1}{N+1} z_{N+1} \\ &= \frac{N}{N+1}\hat{\theta}(N) + \frac{1}{N+1} z_{N+1}.\end{aligned}$$

Although this form already meets our requirements of reusing previous computations, it is possible to rearrange it to a more suitable expression:

$$\hat{\theta}(N+1) = \hat{\theta}(N) + \frac{1}{N+1}\Big(z_{N+1} - \hat{\theta}(N)\Big)$$

Although this expression is very simple, it is very informative because almost every recursive algorithm can be reduced to a similar form. The following observations can be made:

- The new estimate $\hat{\theta}(N+1)$ equals the old estimate $\hat{\theta}(N)$ plus a correction term, that is $\frac{1}{N+1}\Big(z_{N+1} - \hat{\theta}(N)\Big)$.
- The correction term consists of two terms by itself: a gain factor $\frac{1}{N+1}$ and an error term.
- The gain factor decreases towards zero as more measurements are already accumulated in the previous estimate. This means that in the beginning of the experiment, less importance is given to the old estimate $\hat{\theta}(N)$, and more attention is paid to the new incoming measurements. When N starts to grow, the error term becomes small compared to the old estimate. The algorithm relies more and more on the accumulated information in the old estimate $\hat{\theta}(N)$ and it does not vary it that much for accidental variations of the new measurements. The additional bit of information in the new measurement becomes small compared with the information that is accumulated in the old estimate.
- The second term $z_{N+1} - \hat{\theta}(N)$ is an error term. It incorporates the difference between the predicted value of the next measurement on the basis of the model and the actual measurement z_{k+1}.
- When properly initiated, i.e. $\hat{\theta}(1) = z_1$, this recursive result is exactly equal to the non recursive implementation. However, from a numerical point of view, it is a very robust procedure as calculation errors etc. are compensated in each step.

8.2 Construction of the Kalman filter

As stated earlier, the Kalman filter we discuss here deals with state space models of the form (8.1), which are excited by the known input signal u and disturbed by the equation noise source X_x. Additionally, the output quantities y are disturbed by a measurement noise source X_y. The aim of the Kalman filter is to estimate the state x of the system from the measurements z. As we will see in this chapter, the Kalman filter operates by propagating the mean and covariance of the state through time. Our approach to deriving the Kalman filter will involve the following steps:

1. First, we discuss a mathematical description of the model dynamics whose states we want to estimate. Here, we focus on LTI state space models of the form (8.1).
2. Next, we implement equations that describe the propagation of the mean and the covariance of the state with time respectively. These equations form a dynamic system by themselves.
3. These equations are used to implement a recursive algorithm and form the basis for the derivation of the Kalman filter because:
   (a) The mean of the state is the Kalman filter estimate of the state.
   (b) The covariance of the state is the covariance of the Kalman filter state estimate.
4. Every time we receive a new measurement, we update the mean and covariance of the state similar to the simple example displayed in Section 8.1.

In order to classify the Kalman filter problem, we first require a formal distinction of problems regarding information and time dependency:

- $x(k-k)$: an interpolation problem,
- $x(k)$: a filtering problem,
- $x(k+k)$: an prediction (or extrapolation) problem.

The Kalman filter is not only a classical filtering problem, i.e. the data is not computed based on current information. Instead, an internal dynamic for the mean value is constructed and propagated, such that new information can be integrated recursively. To this end, we have to distinguish between an a priori and an a posteriori estimate of the expected value.

8.2.1 Model dynamics and assumptions

The Kalman filter system can be stated in both continuous and discrete time. Here, we focus on the discrete time version given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + X_x(k) \\ y(k) &= Cx(k) + X_y(k), \end{aligned} \tag{8.2}$$

where x, u, X_x, y and X_y are vectors and A, B and C are matrices, see also Figure 8.1 for a corresponding block diagram. Here, we suppose the following to hold:

Assumption 8.1
Regarding system (8.2) we have that

- the matrices A, B and C are known,
- the matrix B satisfies $B = 0$,
- the random variables X_x and X_y are independent variables,
- the probability density functions f_{X_x} and f_{X_y} are normal distributions,
- the expected values satisfy $\mathrm{E}\left(X_x(k)\right) = 0$ and $\mathrm{E}\left(X_y(k)\right) = 0$ and
- the covariance matrices are given by

$$\mathrm{Cov}\left(X_x(k), X_x(j)\right) = R_x\delta_{kj} \quad \text{and} \quad \mathrm{Cov}\left(X_y(k), X_y(j)\right) = R_y\delta_{kj}.$$

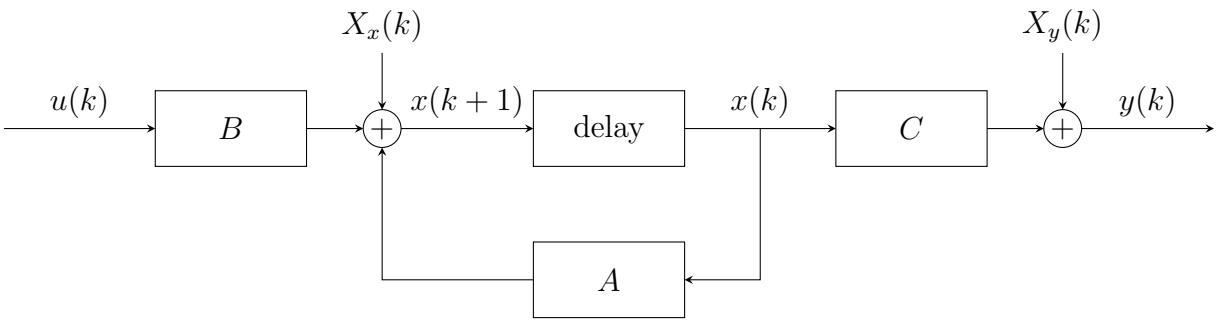


Figure 8.1: Block diagram of the state space system (8.2)

To shorten the notation, we introduce the vector

$$Y(k) := \{y(1), \dots, y(k)\}$$

and denote

$$P(k) := \mathrm{Cov}(x(k) \mid Y(k)) = \mathrm{E}\left([x(k) - \mathrm{E}(x(k) \mid Y(k))] [x(k) - \mathrm{E}(x(k) \mid Y(k))]^\top \right)$$
$$Q(k) := AP(k)A^\top + R_x(k).$$

Given this problem setting, we can now start to derive internal dynamic of the Kalman filter, i.e. the dynamic of the mean value and the covariance.

8.2.2 Propagation of mean and covariance

To write down the Kalman filter dynamics, we first need to construct the propagation of the mean value and the covariance regarding past information. Casually speaking, we need to know how these properties evolve regarding past information without new measurements.

Lemma 8.2
Given a system (8.2) *such that Assumption 8.1 holds. Then we have*

$$\mathrm{E}(x(k+1) \mid Y(k)) = A\mathrm{E}(x(k) \mid Y(k)). \tag{8.3}$$

Proof. Since we have

$$\begin{aligned} \mathrm{E}(x(k+1) \mid Y(k)) &= \mathrm{E}(Ax(k) + X_x(k) \mid Y(k)) \\ &= A\mathrm{E}(x(k) \mid Y(k)) + \mathrm{E}(X_x(k) \mid Y(k)) \\ &= A\mathrm{E}(x(k) \mid Y(k)). \end{aligned}$$

the assertion follows directly. □

Lemma 8.3
Given a system (8.2) *such that Assumption 8.1 holds. Then we have*

$$\mathrm{Cov}(x(k+1) \mid Y(k)) = AP(k)A^\top + R_x = Q(k). \tag{8.4}$$

Proof. Incorporating the definition of the notation, we see

$$\begin{aligned}
&\mathrm{Cov}(x(k+1) \mid Y(k)) = \mathrm{Cov}(Ax(k) + X_x(k) \mid Y(k)) \\
&= \mathrm{E}\left(\left(Ax(k) + X_x(k) - \mathrm{E}\left(Ax(k) + X_x(k) \mid Y(k)\right)\right)\right. \\
&\qquad \left.\left(Ax(k) + X_x(k) - \mathrm{E}\left(Ax(k) + X_x(k) \mid Y(k)\right)\right)^\top\right) \\
&= \mathrm{E}\left(\left(Ax(k) + X_x(k) - A\mathrm{E}\left(x(k) \mid Y(k)\right)\right)\left(Ax(k) + X_x(k) - A\mathrm{E}\left(x(k) \mid Y(k)\right)\right)^\top\right) \\
&= \mathrm{E}\left(A\left(x(k) - \mathrm{E}\left(x(k) \mid Y(k)\right)\right)\left(x(k) - \mathrm{E}\left(x(k) \mid Y(k)\right)\right)^\top A^\top\right) \\
&\quad + \mathrm{E}\left(X_x(k)x(k)^\top \mid Y(k)\right)A^\top + A\mathrm{E}\left(x(k)X_x(k)^\top \mid Y(k)\right) \\
&\quad - \mathrm{E}\left(X_x(k)\mathrm{E}\left(x(k) \mid Y(k)\right)^\top \mid Y(k)\right)A^\top + A\mathrm{E}\left(\mathrm{E}\left(x(k) \mid Y(k)\right)X_x(k)^\top \mid Y(k)\right) \\
&\quad + \mathrm{E}\left(\mathrm{E}\left(X_x(k)\right)\mathrm{E}\left(X_x(k)\right)^\top \mid Y(k)\right) \\
&= A\mathrm{E}\left(\left(x(k) - \mathrm{E}\left(x(k) \mid Y(k)\right)\right)\left(x(k) - \mathrm{E}\left(x(k) \mid Y(k)\right)\right)^\top\right)A^\top + R_x \\
&= AP(k)A^\top + R_x = Q(k).
\end{aligned}$$

which concludes the proof. □

Now that we know the estimate of the mean value and the covariance under the system dynamics, we can move forward to integrate a new measurement.

8.2.3 Derivation of the Kalman dynamics

To derive an update formula of the estimate of the mean value and the covariance computed in the previous section, we need to construct the probability density function of $x(k+1)$. The idea here is to compute an estimate of $x(k+1)$ such that the probability of a respective realization after the measurement of $y(k+1)$ is maximal. This probability density function, in turn, requires an extension of Bayes' rule, which can be derived from the conditional probability density functions

$$\begin{aligned}
f(a,b,c) &= f(a \mid b,c)f(b,c) = f(a \mid b,c)f(b \mid c)f(c) \\
f(a,b,c) &= f(a,b \mid c)f(c).
\end{aligned}$$

Combining these two equations, we obtain

$$f(a \mid b,c) = \frac{f(a,b,c)}{f(b \mid c)f(c)} = \frac{f(a,b \mid c)f(c)}{f(b \mid c)f(c)} = \frac{f(a,b \mid c)}{f(b \mid c)}.$$

Substituting $a = x(k+1)$, $b = y(k+1)$ and $c = Y(k)$ reveals

$$\begin{aligned}
f(x(k+1) \mid y(k+1), Y(k)) &= \frac{f(x(k+1), y(k+1) \mid Y(k))}{f(y(k+1) \mid Y(k))} \\
&= \frac{f(y(k+1) \mid x(k+1), Y(k))f(x(k+1) \mid Y(k))}{f(y(k+1) \mid Y(k))} \\
&= \frac{f_{X_y}(y(k+1) - CAx(k))f(x(k+1) \mid Y(k))}{f(y(k+1) \mid Y(k))}
\end{aligned} \tag{8.5}$$

where we have used that given $Y(k)$, we obtain $x(k+1) = Ax(k)$. Expression (8.5) is very informative. On the left hand side, we find the socalled „a posteriori“ probability density

function of $x(k+1)$, which includes the knowledge obtained from the measurement $y(k+1)$. The a posteriori pdf is calculated form the „a priori“ pdf by taking the latest measurement $y(k+1)$ into account.

In the following part, we are going to determine $x(k+1)$ such that the probability of realizing $x(k+1)$ after the measurement $y(k+1)$ is maximal. Note that we imposed the limitation that the probability density function of the noise X_x and X_y are normal distributions, cf. Assumption 8.1. Since the covariance matrix $\mathrm{Cov}(x(k+1) \mid Y(k))$ is given by Lemma 8.3 and R_x, R_y are given by Assumption 8.1, the probability density functions f_{X_x} and f_{X_y} are determined completely. The denominator of (8.5) is independent of $x(k+1)$ and can therefore be considered as constant when finding the maximum. Hence, we have

$$\begin{aligned}
&\max_{x(k+1)} f(x(k+1) \mid y(k+1), Y(k)) = \\
&= \max_{x(k+1)} e^{-\frac{1}{2}(y(k+1)-CA\mathrm{E}(x(k)|Y(k)))^\top R_y^{-1}(y(k+1)-CA\mathrm{E}(x(k)|Y(k)))} \\
&\qquad \cdot e^{-\frac{1}{2}(x(k+1)-A\mathrm{E}(x(k)|Y(k)))^\top Q^{-1}(k+1)(x(k+1)-A\mathrm{E}(x(k)|Y(k)))} \\
&= \max_{x(k+1)} e^{-\frac{1}{2}(x(k+1)-A\mathrm{E}(x(k)|Y(k)))^\top \left(Q^{-1}(k+1)+C^\top R_y^{-1}C\right)(x(k+1)-A\mathrm{E}(x(k)|Y(k)))}
\end{aligned}$$

From this equation, we directly obtain

$$\mathrm{Cov}\left(x(k+1) \mid Y(k+1)\right) = P(k+1) = Q(k+1)^{-1} + C^\top R_y^{-1} C. \tag{8.6}$$

In order to compute the maximizer of $f(x(k+1) \mid y(k+1), Y(k))$, it is sufficient to minimize the exponent of the above expression. Considering the necessary first order condition, we obtain

$$\left(Q^{-1}(k+1) + C^\top R_y^{-1} C\right)\left(x(k+1) - A\mathrm{E}\left(x(k) \mid Y(k)\right)\right) = 0$$

In order to obtain stationarity of the evolution, we require $x(k+1) = \mathrm{E}\left(x(k+1) \mid Y(k+1)\right)$. Inserting this into the necessary condition reveals

$$\begin{aligned}
&\left(Q^{-1}(k+1) + C^\top R_y^{-1} C\right) \mathrm{E}\left(x(k+1) \mid Y(k+1)\right) \\
&\quad = Q^{-1}(k+1) A \mathrm{E}\left(x(k) \mid Y(k)\right) + C^\top R_y^{-1} C A \mathrm{E}\left(x(k) \mid Y(k)\right)
\end{aligned}$$

Now, we can use the matrix inverse lemma

$$P = \left(Q^{-1} + C^\top R_y C\right)^{-1} = Q - QC^\top \left(CQC^\top + R_y\right)^{-1} CQ$$

and the relation

$$\left(Q + C^\top R_y^{-1} C\right)^{-1} C^\top R_y^{-1} = QC^\top \left(CQC^\top + R_y\right)^{-1}$$

to obtain

$$\begin{aligned}
&\mathrm{E}\left(x(k+1) \mid Y(k+1)\right) = \\
&\quad = A\mathrm{E}\left(x(k) \mid Y(k)\right) + Q(k+1)C^\top \left(CQ(k+1)C^\top + R_y\right)^{-1} \left(y(k+1) - CA\mathrm{E}\left(x(k) \mid Y(k)\right)\right)
\end{aligned} \tag{8.7}$$

8.2.4 Integration of mean and covariance into a recursive algorithm

To shorten notation, we introduce the abbreviation

$$X(k) := \mathrm{E}\left(x(k) \mid Y(k)\right),$$

we can formulate the following recursive algorithm:

Algorithm 8.4 (Kalman filter for LTI systems without external input)
Given a given LTI model with initial conditions R_x, R_y and $X(1)$, set $P(1) = R_x$.
For $k = 1, \dots$ do

$$Q(k+1) = AP(k)A^\top + R_x \tag{8.8}$$
$$K(k+1) = Q(k+1)C^\top \left(CQ(k+1)C^\top + R_y\right)^{-1} \tag{8.9}$$
$$P(k+1) = (\mathrm{Id} - K(k+1)C)\, Q(k+1) \tag{8.10}$$
$$X(k+1) = AX(k) + K(k+1)\left(y(k+1) - CAX(k)\right) \tag{8.11}$$

The algorithm contains several factors, which exhibit a good interpretation regarding the computations made earlier in this chapter. Here, the time component plays an important role.

- The matrix $Q(k+1) = P(k+1 \mid k)$ represents the a priori covariance matrix of $X(k+1) = \mathrm{E}\left(x(k+1) \mid Y(k+1)\right)$ using k measurements only.
- Similarly, the matrix $P(k+1)$ corresponds to the a posteriori covariance matrix of $X(k+1) = \mathrm{E}\left(x(k+1) \mid Y(k+1)\right)$ using $k+1$ measurements.
- Considering the dynamic of the system, the vector $AX(k)$ reveals the extrapolated state variable based on the model dynamics A and k measurements.
- Projecting on the output, the vector $CAXk$ represents the expected output given the extrapolated state of the system.

We like to note that, within the algorithm, the matrices Q, P and K are independent of the measurements. For this reason, they can be computed beforehand which lowers the computational complexity of the filter. Additionally, the method remains usable when the noise is not normally distributed. In that case, however, the solution found by the filter is no longer an optimal one.

Similar to the case defined by Assumption 8.1, we can consider the more general LTI case with external inputs, i.e. $B \neq 0$. Recall, that the remaining assumptions are still in place, that is

Assumption 8.5
Regarding system (8.2) we have that

- the matrices A, B and C are known,
- the random variables X_x and X_y are independent variables,
- the probability density functions f_{X_x} and f_{X_y} are normal distributions,
- the expected values satisfy $\mathrm{E}\left(X_x(k)\right) = 0$ and $\mathrm{E}\left(X_y(k)\right) = 0$ and
- the covariance matrices are given by

$$\mathrm{Cov}\left(X_x(k), X_x(j)\right) = R_x\delta_{kj} \quad \text{and} \quad \mathrm{Cov}\left(X_y(k), X_y(j)\right) = R_y\delta_{kj}.$$

Given these assumptions, the computations displayed before in this chapter can be modified and the following algorithm can be derived:

Algorithm 8.6 (Kalman filter for LTI systems with external input)
Given a given LTI model with initial conditions R_x, R_y and $X(1)$, set $P(1) = R_x$.
For $k = 1, \ldots$ do

$$Q(k+1) = AP(k)A^\top + R_x \tag{8.12}$$
$$K(k+1) = Q(k+1)C^\top \left(CQ(k+1)C^\top + R_y\right)^{-1} \tag{8.13}$$
$$P(k+1) = (\mathrm{Id} - K(k+1)C)\, Q(k+1) \tag{8.14}$$
$$X(k+1) = AX(k) + Bu(k) + K(k+1)\left(y(k+1) - CAX(k) - CBu(k)\right) \tag{8.15}$$

8.3 Example

Consider an inertial measurement unit (IMU) to be given, which is capable of measuring all three angular velocities around the body fixed coordinate (BFC) axis of the unit as well as the three acceleration forces in the directions of the BFC axis. The measurements are obtained from gyros and accelerometers respectively. IMUs are typically used to maneuver aircraft, including unmanned aerial vehicles (UAVs), among many others, and spacecraft, including satellites and landers. This data can then be used by a computer to continually calculate the vehicle's current position. One way to do this is to integrate over time the sensed acceleration, together with an estimate of gravity, to calculate the current velocity for each of the six degrees of freedom. In a second step, one can integrate the velocity to calculate the current position, which leads to a typical double integrator system.

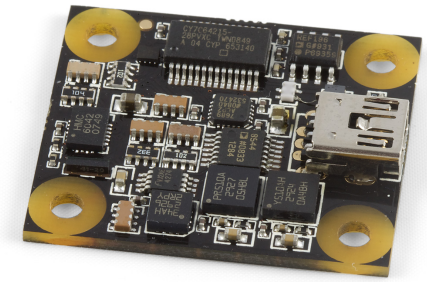
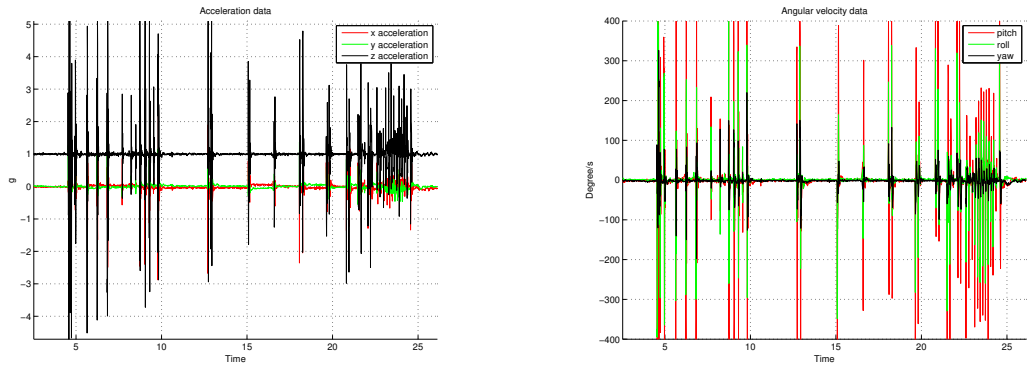


Figure 8.2: Inertial measurement unit (IMU)

Unfortunately, such a method suffers from accumulated error. The reason for this error lies in the construction of the method: The sensors detect accelerations and velocity only once within a sampling interval. Hence, these states may change within the interval and the method cannot recognize that change. Instead, the integration accumulates the error, which may grow exponentially over time. Note that reducing the length of the sampling intervals will not solve the problem as the error is systemic.

A sample of such measurements are displayed in Figures 8.3 and 8.4. Figure 8.3 shows details for common results in car experiments for bumpy roads, where we can see high vertical accelerations. The data set was recorded for a 1:8 model car running an off road track. From the data, we can see that the vertical forces are extremely large, up to $5g$ in upwards and $-4g$ downwards. The angular velocity rates, on the other hand, are rather small. Note that the high values for pitch and roll are due to a singularity in the sensors, which provide data from

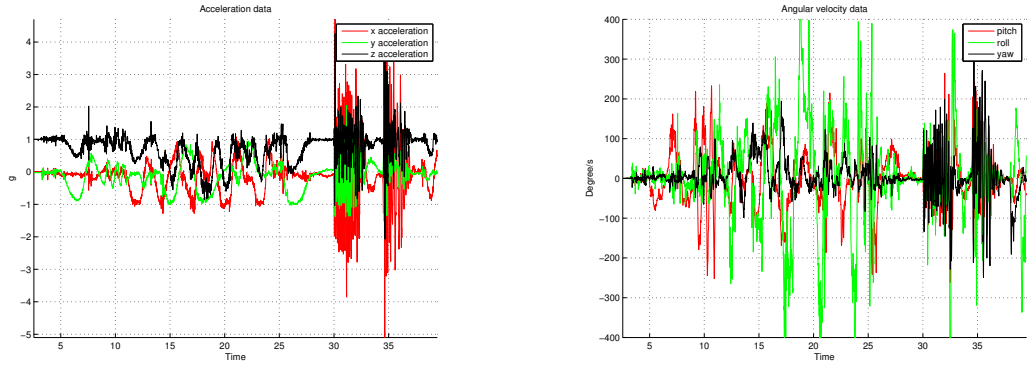
a $[0, 360]$ degree interval.



(a) IMU acceleration data in BFC (b) IMU velocity data in BFC

Figure 8.3: IMU measurement data from gyros and accelerometers for sudden strikes

The second measurement data was obtained during quadcopter flights, where curvy maneuvers in all three axis occurred. From the data, one can see that the maneuvers were rather extreme and the copter crashed twice at the end. Here, the data changes are much smaller compared to the first case, at least until the crash occurred.



(a) IMU acceleration data in BFC (b) IMU velocity data in BFC

Figure 8.4: IMU measurement data from gyros and accelerometers for continuous changes

In both cases, we were interested in the current state of the system. For the sake of simplicity, we focus on the pitch angle. Here, we like to note that the accelerometer data typically jitters and is not that accurate. The gyros on the other hand give us quite good data, but as we have discussed before, a simple integration may result in exponential errors illustrated in Figure 8.5.

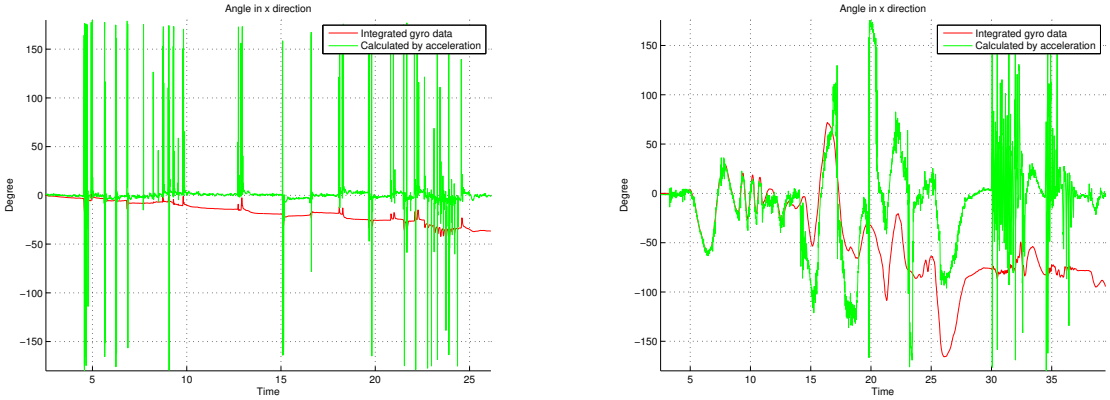
Within Figure 8.5, we displayed the results of two different computations: For one, we used the accelerometer data to evaluate

$$\hat{\theta}_1 = \frac{180^\circ}{\pi} \cdot \arctan 2(\ddot{x}_{3,\mathrm{BFC}}, \ddot{x}_{2,\mathrm{BFC}}). \tag{8.16}$$

Since this estimate is based on accelerometer data only, there is no drift in the result. Secondly, we used a simple integration of the timestamped angular velocity data

$$\hat{\theta}_1(k+1) = \hat{\theta}_{\mathrm{pitch}}(k) + (t_{k+1} - t_k)\, \dot{x}_{1,\mathrm{BFC}}. \tag{8.17}$$

We observe that in both cases the angle computed by (8.17) diverges from the result of (8.16).



(a) Results for angles using one sensor family only

(b) Results for angles using one sensor family only

Figure 8.5: IMU angular results for using sensor families separately

Now, we apply the Kalman filter to this problem to fuse the advantages of the gyro (no jitter) and the accelerometer (no drift). To this end, we define the model dynamics (8.1) by

$$x(k) = \begin{pmatrix} x_{1,\mathrm{BFC}}(k) \\ \dot{x}_{1,\mathrm{BFC}}(k) \end{pmatrix}, \qquad u(k) = \frac{\pi}{180^\circ} \cdot \dot{x}_{1,\mathrm{BFC}}(k)$$
$$A(k) = \begin{pmatrix} 1 & -(t_{k+1} - t_k) \\ 0 & 1 \end{pmatrix}, \qquad B(k) = \begin{pmatrix} (t_{k+1} - t_k) \\ 0 \end{pmatrix}, \qquad C = \begin{pmatrix} 1 & 0 \end{pmatrix}$$

which gives us the system

$$x(k+1) = \begin{pmatrix} 1 & -(t_{k+1} - t_k) \\ 0 & 1 \end{pmatrix} x(k) + \begin{pmatrix} (t_{k+1} - t_k) \\ 0 \end{pmatrix} u(k) \tag{8.18}$$
$$y(k) = \begin{pmatrix} 1 & 0 \end{pmatrix} x(k) \tag{8.19}$$

The Kalman filter is initialized using the accelerometer data to generate an initial value of the estimator

$$x(0) = \begin{pmatrix} \frac{180^\circ}{\pi} \cdot \arctan 2(\ddot{x}_{3,\mathrm{BFC}}, \ddot{x}_{2,\mathrm{BFC}}) \\ 0 \end{pmatrix},$$

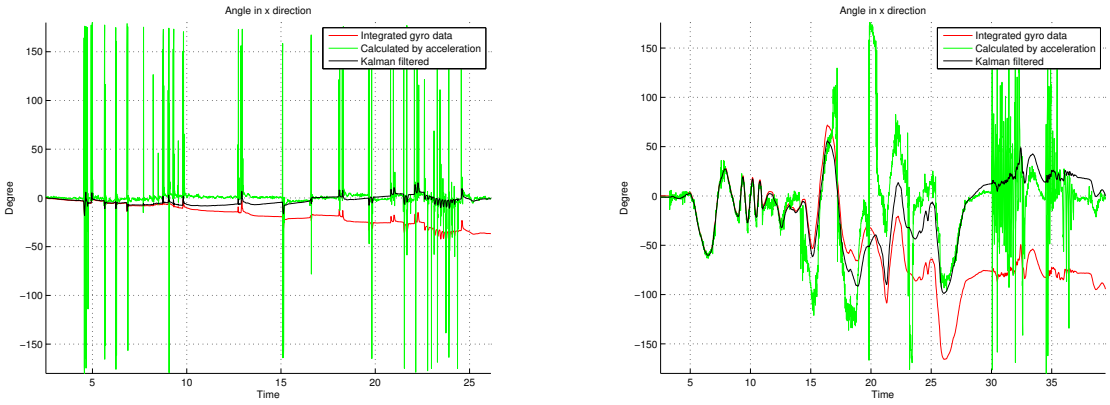
and the approximated covariance matrices of the disturbances

$$R_x = \begin{pmatrix} \mathrm{E}\left(\frac{\pi}{180^\circ} \cdot 0.0257 \cdot (t_{k+1} - t_k)^2\right) & 0 \\ 0 & 10^{-8} \end{pmatrix}, \qquad R_y = \frac{\pi}{180^\circ} \cdot 15,$$

which are based on physical properties of the sensors and a freely chosen bias correction value for $R_{x2,2}$.

The resulting estimates of the Kalman filter based on the above are shown in Figure 8.6.

Within this figure, we can clearly see the improvement of the Kalman filter sensor fusion. Note only is the jitter of the accelerometer reduced drastically, but also the drift of the gyros is eliminated after a certain startup phase. Within the startup, the Kalman filter estimate resembles the gyro results. Then, the filter reaches a point where it recognizes the growing error and compensates for it. Within Figure 8.6, we can identify this point by the starting divergence of the filter result from the integration one. From that point forward, the Kalman filter estimate converges towards the accelerometer result, yet without it strong pulses.



(a) Results for Kalman filter using data from Figure 8.3

(b) Results for Kalman filter using data from Figure 8.4

Figure 8.6: IMU Kalman filter fusion results in comparison to single sensor family results

Appendices

Appendix A

Programs

Here, we display the programs used throughout the script. The programs may be used for a better personal understanding of the estimation process and the involved notions.

A.1 From Chapter 1: Motivating example of the electric circuit

```
clear all;
close all;

N = 100;

% Generating the measurements
% Group A
rng(1);
voltage = 1+0.2*randn(1,N);
rng(1);
current = 1+0.3*randn(1,N);

% % Group B
% rng(1);
% voltage = 1+0.2*randn(1,N);
% rng(1);
% current = 1+0.3*(rand(1,N)-0.5);

% Plotting measured voltages
figure(1);
hold on;
plot(1:N,voltage);
axis on;
axis ([1 N 0 2]);
xlabel('Measurement number', 'Fontsize', 14);
ylabel('Measured voltage (V)', 'Fontsize', 14);
grid on;
hold off;

% Plotting measured currents
figure(2);
hold on;
plot(1:N,current);
axis on;
```

```
35 axis ([1 N 0 2]);
36 xlabel('Measurement␣number', 'Fontsize', 14);
37 ylabel('Measured␣current␣(I)', 'Fontsize', 14);
38 grid on;
39 hold off;
40
41 % Computing and plotting the resistance
42 for i=1:size(voltage,2)
43     resistance(i) = voltage(1,i)/current(1,i);
44 end
45 figure(3);
46 hold on;
47 plot(1:N, resistance);
48 axis on;
49 axis ([1 100 0 5]);
50 xlabel('Measurement␣number', 'Fontsize', 14);
51 ylabel('Measured␣value␣(R)', 'Fontsize', 14);
52 grid on;
53 hold off;
```

Program A.1: Generating measurements for the electric circuit problem

```
1  clear all;
2  close all;
3
4  Nmax = 10000;
5
6  % Group A
7  % rng(1);
8  % voltage = 1+0.2*randn(1,Nmax);
9  % rng(1);
10 % current = 1+0.3*randn(1,Nmax);
11
12 % Group B
13 rng(1);
14 voltage = 1+0.2*randn(1,Nmax);
15 rng(1);
16 current = 1+0.3*(rand(1,Nmax)-0.5);
17
18 % Compute different estimators
19 for i=1:Nmax
20     R_SA(i) = 1/i * sum(voltage(1,1:i)./current(1,1:i));
21     R_EV(i) = sum(voltage(1,1:i))/sum(current(1,1:i));
22     R_LS(i) = (1/i * sum(voltage(1,1:i).*current(1,1:i)))/(1/i * sum(current
           (1,1:i).*current(1,1:i)));
23 end
24
25 % Plot estimated resistances
26 figure(1);
27 semilogx(1:Nmax,R_SA,'-b', ...
28       1:Nmax,R_EV,'-r', ...
29       1:Nmax,R_LS,'-g');
30 xlabel('Number␣of␣measurements', 'Fontsize', 14);
31 ylabel('Estimated␣resistance', 'Fontsize', 14);
32 grid on;
33 legend('R_{SA}','R_{EV}','R_{LS}');
34
35 % Compute frequencies of estimation results
```

```
R_SA_min = min(R_SA);
R_SA_max = max(R_SA);
R_SA_pdf_x = linspace(R_SA_min,R_SA_max,100);
%histc(R_SA,R_SA_pdf_x);
R_SA_pdf_y=histc(R_SA,R_SA_pdf_x);

R_EV_min = min(R_EV);
R_EV_max = max(R_EV);
R_EV_pdf_x = linspace(R_EV_min,R_EV_max,100);
%hist(R_EV,R_EV_pdf_x);
R_EV_pdf_y=histc(R_EV,R_EV_pdf_x);

R_LS_min = min(R_LS);
R_LS_max = max(R_LS);
R_LS_pdf_x = linspace(R_LS_min,R_LS_max,100);
%hist(R_LS,R_LS_pdf_x);
R_LS_pdf_y=histc(R_LS,R_LS_pdf_x);

% Plot frequencies of estimation results
figure(2);
plot(R_SA_pdf_x,R_SA_pdf_y,'-b', ...
     R_EV_pdf_x,R_EV_pdf_y,'-r',...
     R_LS_pdf_x,R_LS_pdf_y,'-g');
xlabel('Estimated resistance', 'Fontsize', 14);
ylabel('frequency', 'Fontsize', 14);
grid on;
axis on;
axis ([0.9 1.1 0 90000]);
legend('R_{SA}','R_{EV}','R_{LS}');

% Comparing frequency development of estimation results
for i=3:4
    R_SA_min = min(R_SA(1:10^i));
    R_SA_max = max(R_SA(1:10^i));
    R_SA_pdf_x = linspace(R_SA_min,R_SA_max,100);
    R_SA_pdf_y = histc(R_SA(1:10^i),R_SA_pdf_x);

    R_EV_min = min(R_EV(1:10^i));
    R_EV_max = max(R_EV(1:10^i));
    R_EV_pdf_x = linspace(R_EV_min,R_EV_max,100);
    R_EV_pdf_y = histc(R_EV(1:10^i),R_EV_pdf_x);

    R_LS_min = min(R_LS(1:10^i));
    R_LS_max = max(R_LS(1:10^i));
    R_LS_pdf_x = linspace(R_LS_min,R_LS_max,100);
    R_LS_pdf_y = histc(R_LS(1:10^i),R_LS_pdf_x);

    figure(i);
    plot(R_SA_pdf_x,R_SA_pdf_y,'-b', ...
         R_EV_pdf_x,R_EV_pdf_y,'-r',...
         R_LS_pdf_x,R_LS_pdf_y,'-g');
    xlabel('Estimated resistance', 'Fontsize', 14);
    ylabel('frequency', 'Fontsize', 14);
    grid on;
    axis on;
    axis ([0.9 1.1 0 9000]);
    legend('R_{SA}','R_{EV}','R_{LS}');
end

```

```
%%%%%%%%%%%

% Computing standard deviation
for i=1:Nmax
    R_SA_var(i)=var(R_SA(1:i));
    R_EV_var(i)=var(R_EV(1:i));
    R_LS_var(i)=var(R_LS(1:i));
end

% Plotting standard deviation
figure(5);
loglog(1:Nmax,R_SA_var,'-b', ...
     1:Nmax,R_EV_var,'-r', ...
     1:Nmax,R_LS_var,'-g');
xlabel('Number of measurements', 'Fontsize', 14);
ylabel('Standard deviation', 'Fontsize', 14);
grid on;
legend('R_{SA}','R_{EV}','R_{LS}');

% Plotting analysis of realizations
figure(6);
current_min = min(current);
current_max = max(current);
current_x = linspace(current_min,current_max,100);
hist(current,current_x);
xlabel('Current', 'Fontsize', 14);
ylabel('# Realization', 'Fontsize', 14);
grid on;
```

Program A.2: Analyzing the outcome of the electric circuit estimation problem

A.2 From Chapter 2: Growth of the world population

```
clear all;
close all;

% Set data
population_world(1950) = 2.519;
population_world(1955) = 2.756;
population_world(1960) = 2.982;
population_world(1965) = 3.335;
population_world(1970) = 3.692;
population_world(1975) = 4.068;
population_world(1980) = 4.435;
population_world(1985) = 4.831;
population_world(1990) = 5.263;
population_world(1995) = 5.674;
population_world(2000) = 6.070;
population_world(2005) = 6.454;
population_world(2010) = 6.972;
population_europe(1950) = 0.547;
population_europe(1955) = 0.575;
population_europe(1960) = 0.601;
population_europe(1965) = 0.634;
population_europe(1970) = 0.656;
```

```
population_europe(1975) = 0.675;
population_europe(1980) = 0.692;
population_europe(1985) = 0.706;
population_europe(1990) = 0.721;
population_europe(1995) = 0.727;
population_europe(2000) = 0.728;
population_europe(2005) = 0.725;
population_europe(2010) = 0.732;
times = 1950:5:2010;

% Set sample size
N = 13;

% Generate inputs in logarithmic form for linearity
% log(x(t)) = log(x0) + lambda(t-t0)
rng(1);
input(:,1) = ones(1,N);
input(:,2) = times-1950;

% Define model
K = [input(:,1), input(:,2)];
% Evaluate linear least square formula
estimate = inv(transpose(K)*K)*transpose(K)*log(population_world(times))';

% Retrieving parameters from logarthmic form
x0 = exp(estimate(1));
lambda = estimate(2);
times_plotting = 1950:1:2010;

% Plot results
figure(1);
hold on;
plot(times,population_world(times(:)),'xb');
plot(times_plotting,x0*exp(lambda*(times_plotting-1950)),'-r','LineWidth',1.5);
axis on;
axis tight;
xlabel('Year', 'Fontsize', 14);
ylabel('World population', 'Fontsize', 14);
legend('Data','Approximation','Location','SouthEast');
grid on;
hold off;

% Evaluate linear least square formula
estimate = inv(transpose(K)*K)*transpose(K)*log(population_europe(times))';

% Retrieving parameters from logarthmic form
x0 = exp(estimate(1));
lambda = estimate(2);
times_plotting = 1950:1:2010;

% Plot results
figure(2);
hold on;
plot(times,population_europe(times(:)),'xb');
plot(times_plotting,x0*exp(lambda*(times_plotting-1950)),'-r','LineWidth',1.5);
axis on;
axis tight;
xlabel('Year', 'Fontsize', 14);
ylabel('European population', 'Fontsize', 14);
```

```
legend('Data','Approximation','Location','SouthEast');
grid on;
hold off;
```

Program A.3: Identification and evaluation of the worldwide population growth

```
clear all;
close all;

x0 = [0.01, 1, 2];
lambda = 1;
C = 1;
times_plotting = 0:0.1:10;

% Plot results
figure(1);
hold on;
plot(times_plotting,zeros(size(times_plotting)),'-r','LineWidth',1.5);
plot(times_plotting,C./(1 + (C/x0(1) - 1) * exp(-lambda*C*times_plotting)),'-b',
    'LineWidth',1.5);
plot(times_plotting,C./(1 + (C/x0(2) - 1) * exp(-lambda*C*times_plotting)),'-g',
    'LineWidth',1.5);
plot(times_plotting,C./(1 + (C/x0(3) - 1) * exp(-lambda*C*times_plotting)),'-c',
    'LineWidth',1.5);
axis on;
axis tight;
xlabel('Time', 'Fontsize', 14);
ylabel('Population', 'Fontsize', 14);
grid on;
hold off;
```

Program A.4: Solution of the logistics equation (2.4)

```
function biology_twospeciesunconstrained
clear all;
close all;

T=8;
x01 = [2, 2];
[t1,y1]=ode45(@f,[0,T],x01,'');
T=12;
x02 = [5, 5];
[t2,y2]=ode45(@f,[0,T],x02,'');
T=22;
x03 = [10, 10];
[t3,y3]=ode45(@f,[0,T],x03,'');

% Plot results
figure(1);
hold on;
plot(y1(:,1),y1(:,2),'-r','LineWidth',1.5);
plot(y2(:,1),y2(:,2),'-b','LineWidth',1.5);
plot(y3(:,1),y3(:,2),'-g','LineWidth',1.5);
axis on;
axis tight;
xlabel('Prey␣population', 'Fontsize', 14);
```

```
ylabel('Predator population', 'Fontsize', 14);
grid on;
hold off;

T=24;
x01 = [2, 2];
[t1,y1]=ode45(@f,[0,T],x01,'');

figure(2);
hold on;
plot(t1,y1(:,1),'-r','LineWidth',1.5);
plot(t1,y1(:,2),'-b','LineWidth',1.5);
axis on;
axis tight;
xlabel('Time', 'Fontsize', 14);
ylabel('Predator/prey population', 'Fontsize', 14);
grid on;
hold off;

end


% Dynamics
function y=f(t,x)
a = 1;
c = 1;

y(1,1) = a * x(1) * ( 1 - x(2) );
y(2,1) = - c * x(2) * ( 1 - x(1) );
end
```

Program A.5: Solution of the two species problem with unlimited resources (2.8)

```
function biology_twospeciesunconstrained
clear all;
close all;

T=8;
x01 = [2, 2];
[t1,y1]=ode45(@f,[0,T],x01,'');
T=12;
x02 = [5, 5];
[t2,y2]=ode45(@f,[0,T],x02,'');
T=22;
x03 = [10, 10];
[t3,y3]=ode45(@f,[0,T],x03,'');

% Plot results
figure(1);
hold on;
plot(y1(:,1),y1(:,2),'-r','LineWidth',1.5);
plot(y2(:,1),y2(:,2),'-b','LineWidth',1.5);
plot(y3(:,1),y3(:,2),'-g','LineWidth',1.5);
axis on;
axis tight;
xlabel('Prey population', 'Fontsize', 14);
ylabel('Predator population', 'Fontsize', 14);
grid on;
```

```
hold off;

T=24;
x01 = [2, 2];
[t1,y1]=ode45(@f,[0,T],x01,'');

figure(2);
hold on;
plot(t1,y1(:,1),'-r','LineWidth',1.5);
plot(t1,y1(:,2),'-b','LineWidth',1.5);
axis on;
axis tight;
xlabel('Time', 'Fontsize', 14);
ylabel('Predator/prey population', 'Fontsize', 14);
grid on;
hold off;

end


% Dynamics
function y=f(t,x)
a = 1;
c = 1;
beta = 0.5;

y(1,1) = a * x(1) * ( 1 - x(2) ) + beta * x(1) * ( 1 - x(1) );
y(2,1) = - c * x(2) * ( 1 - x(1) );
end
```

Program A.6: Solution of the two species problem with limited resources (2.12)

A.3 From Chapter 4: Financial Processes

```
% gaussvar(mu,sigma)
%
% Gives back a N(mu,sigma^2)-distributed random number
%
% Input :  mu    (Mean)
%          sigma (Standard deviation)
%
% Output: N(mu,sigma^2)-distributed random number

function result = finance_gaussvar(mu,sigma);

u1 = rand(1);
u2 = rand(1);

result = sqrt(-2*log(u1))*cos(2*pi*u2)*sigma+mu;
```

Program A.7: Generating a $\mathcal{N}(\mu(,)\sigma^2)$ distributed random variable

```
function W=finance_wienerprozess(T,n,numberofsteps);

```

```
h=T/n;
mu=0;
sigma=sqrt(h);

for i=1:numberofsteps
    W(1,i)=0;
    for j=2:n
        dW = finance_gaussvar(mu,sigma);
        W(j,i) = W(j-1,i) + dW;
    end
end
```

Program A.8: Generating a Wiener process

```
function W=finance_wienerprozess(T,n,numberofsteps);

h=T/n;
mu=0;
sigma=sqrt(h);

for i=1:numberofsteps
    W(1,i)=0;
    for j=2:n
        dW = finance_gaussvar(mu,sigma);
        W(j,i) = W(j-1,i) + dW;
    end
end
```

Program A.9: Generating a path of a Wiener process

```
function X = finance_eulermaruyama(T,n,N,X0,a,b)

% Set stochastic variable of Wiener process
h = T / n;
X = zeros(n,N);

% Set initial value
X(1,:) = X0;

% Evaluate Wiener process
W = finance_wienerprocess(T,n,N);

% Evaluate path
for i = 1:n-1
    X(i+1,:) = X(i,:) .* (1 + a*h + b.*(W(i+1,:)-W(i,:)));
end
```

Program A.10: Solving a stochastic differential equation

```
function result = finance_put(S,t,K,r,sigma,T)

d1 = (log(S/K) + (r+0.5*sigma^2)*(T-t))/(sigma*sqrt(T-t));
d2 = d1 - sigma*sqrt(T-t);
n1 = 0.5*(1 + erf(-d1/sqrt(2)));
n2 = 0.5*(1 + erf(-d2/sqrt(2)));
```

```
result = K*exp(-r*(T-t))*n2 - S*n1;
```

Program A.11: Compute value of a European put via Black–Scholes solution

```
function finance_montecarlo

terminalTime = 1;
numberOfIntervals = 100;
numberOfIterations = 5000;
initialValue = 80;
terminalValue = 100;
interestRate = 0.08;
risk = 0.2;

Y = finance_eulermaruyama(terminalTime,numberOfIntervals,numberOfIterations,
    initialValue,interestRate,risk);

% ------------------------------------------------------------------
% Plot of Euler-Maruyama solutions of all paths
for i = 1:numberOfIntervals
    T(i) = (i - 1) * terminalTime / numberOfIntervals;
end
figure(1)
subplot(2,2,1);
plot(T,Y)
axis tight;
grid on;
xlabel('Time', 'Fontsize', 14);
ylabel('Option value', 'Fontsize', 14);
histogram = sort(Y(numberOfIntervals,:));
subplot(2,2,2);
hist(histogram,numberOfIterations)
axis tight;
xlabel('Option value', 'Fontsize', 14);
ylabel('Number of occurences', 'Fontsize', 14);
grid on;

% ------------------------------------------------------------------
% Compute payoff

payoff = max(0, terminalValue - Y(numberOfIntervals,:));

% ------------------------------------------------------------------
% Compute and plot discounted expected value
V0 = exp(-interestRate*terminalTime) * (cumsum(payoff)./(1:numberOfIterations));
subplot(2,2,3);
plot(V0)
xlabel('Number of paths', 'Fontsize', 14);
ylabel('Discounted expected value', 'Fontsize', 14);
axis tight;
grid on;

% ------------------------------------------------------------------
% Error if expected value
Vexact = finance_put(initialValue,0,terminalValue,interestRate,risk,terminalTime
    );
subplot(2,2,4);
plot(abs(V0 - Vexact*ones(size(V0)))/Vexact)
```

```
53 xlabel('Number␣of␣paths', 'Fontsize', 14);
54 ylabel('Error␣in␣expected␣value', 'Fontsize', 14);
55 axis tight;
56 grid on
57
58 figure(2);
59 plot(T,Y(:,:))
60 xlabel('Time', 'Fontsize', 14);
61 ylabel('Option␣value', 'Fontsize', 14);
62 axis tight;
63 grid on;
64
65
66 figure(3);
67 hist(histogram,numberOfIterations)
68 xlabel('Option␣values', 'Fontsize', 14);
69 ylabel('Number␣of␣occurances', 'Fontsize', 14);
70 axis tight;
71 grid on;
72
73 figure(4);
74 plot(V0)
75 xlabel('Number␣of␣paths', 'Fontsize', 14);
76 ylabel('Discounted␣expected␣value', 'Fontsize', 14);
77 axis tight;
78 grid on;
79
80 figure(5);
81 plot(abs(V0 - Vexact*ones(size(V0)))/Vexact)
82 xlabel('Number␣of␣paths', 'Fontsize', 14);
83 ylabel('Error␣in␣expected␣value', 'Fontsize', 14);
84 axis tight;
85 grid on;
```

Program A.12: Evaluate Monte–Carlo method

```
1  function finance_blackscholes
2
3  terminalTime = 1;
4  terminalValue = 100;
5  interestRate = 0.08;
6  risk = 0.2;
7
8  initialTime=[0:0.05:1]';
9  initialValue=[0:5:200]';
10
11 % ------------------------------------------------------------
12 % Error if expected value
13 for t=1:size(initialTime,1)
14     for S=1:size(initialValue,1)
15         Vexact(t,S) = finance_put(initialValue(S),initialTime(t),terminalValue,
               interestRate,risk,terminalTime);
16     end
17 end
18 figure(1);
19 surf(initialValue,initialTime,Vexact);
20 xlabel('Initial␣value', 'Fontsize', 14);
21 ylabel('Initial␣time', 'Fontsize', 14);
```

```
zlabel('Option value', 'Fontsize', 14);
axis tight;
grid on;
```

Program A.13: Evaluating Black–Scholes equation for various initial conditions

A.4 From Chapter 6: Least square estimation

```
clear all;
close all;

% Set sample size
N = 100;

% Generate inputs (here: no inputs)

% Get measurements (in case of real measurements delete random parameters
% and replace measurement data)
% Generate random parameters
rng(1);
parameter(:,1) = 1 + 0.2 * randn(N,1);
% Generate measurements
measurement = parameter(:,1);

% Define model
K = ones(N,1);
% Evaluate linear least square formula
estimate = inv(transpose(K)*K)*transpose(K)*measurement;

% Plot results
figure(1);
hold on;
plot(1:N,measurement,'xb');
plot(1:N,K*estimate,'-r','LineWidth',2);
axis on;
axis tight;
xlabel('Measurement number', 'Fontsize', 14);
ylabel('Measured output', 'Fontsize', 14);
grid on;
hold off;
```

Program A.14: Computing the linear least square estimator for Example 6.5

```
clear all;
close all;

% Set sample size
N = 100;

% Generate inputs
rng(1);
input(:,1) = linspace(1,2,N)';
input(:,2) = linspace(1,10,N)';

```

```
% Get measurements (in case of real measurements delete random parameters
% and replace measurement data)
% Generate random parameters
rng(1);
parameter(:,1) = 1 + 2.0 * randn(N,1);
parameter(:,2) = 2 + 1.0 * randn(N,1);
% Generate measurements
rng(1);
measurement = input(:,1) .* parameter(:,1) + input(:,2).^2 .* parameter(:,2);

% Define model
K = [input(:,1), input(:,2).^2];
% Evaluate linear least square formula
estimate = inv(transpose(K)*K)*transpose(K)*measurement;

% Plot results
figure(1);
hold on;
plot(1:N,measurement,'xb');
plot(1:N,K * estimate,'-r','LineWidth',2);
axis on;
axis tight;
xlabel('Measurement number', 'Fontsize', 14);
ylabel('Measured output', 'Fontsize', 14);
grid on;
hold off;
```

Program A.15: Computing the linear least square estimator for Example 6.6

```
clear all;
close all;

% Set sample size
N = 100;

% Generate inputs
input(:,1) = linspace(1,2,N)';
input(:,2) = linspace(1,10,N)';

% Get measurements (in case of real measurements delete random parameters
% and replace measurement data)
% Generate random parameters
rng(1);
parameter(:,1) = 1 + 2.0 * randn(N,1);
parameter(:,2) = 2 + 1.0 * randn(N,1);
% Generate measurements
measurement = input(:,1) .* parameter(:,1) + input(:,2).^2 .* parameter(:,2);

% Define model
K = [input(:,1), input(:,2).^2];
% Define weighting matrix
W = diag(linspace(0,1,N));
% Evaluate linear least square formula
estimate = inv(transpose(K)*W*K)*transpose(K)*W*measurement;

% Plot results
figure(1);
hold on;
```

```
plot(1:N,measurement,'xb');
plot(1:N,K * estimate,'-r','LineWidth',2);
axis on;
axis tight;
xlabel('Measurement␣number', 'Fontsize', 14);
ylabel('Measured␣output', 'Fontsize', 14);
grid on;
hold off;
```

Program A.16: Computing the linear least square estimator for Example 6.13

A.5 From Chapter 8: Kalman filtering

```
clear all;
close all;

% Read data
%data = load('kalman_data/data-zero');
%data = load('kalman_data/data-360x');
%data = load('kalman_data/data-x');
%data = load('kalman_data/data-x2');
data = load('kalman_data/data-smallhits');

% Reassign data to arrays
discreteTime = data(:,1); % in s
acceleration_x = data(:,2); % in G = 9.81 m/s^2
acceleration_y = data(:,3); % in G = 9.81 m/s^2
acceleration_z = data(:,4); % in G = 9.81 m/s^2
angular_velocity_x = data(:,5); % in degree/s
angular_velocity_y = data(:,6); % in degree/s
angular_velocity_z = data(:,7); % in degree/s
numberOfSamples = length(discreteTime);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute angle by integration of velocity data from gyros
angle_x_version1 = zeros(numberOfSamples,1);
for n=1:numberOfSamples-1
    timestep = discreteTime(n+1) - discreteTime(n);
    angle_x_version1(n+1) = angle_x_version1(n) + timestep * angular_velocity_x(
       n);
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute angle via accelerometer data
angle_x_version2 = 180/pi * atan2( acceleration_y, acceleration_z);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Kalman Filter

% Set measurement noise of output
Ry_z = 15*pi/180; % standard deviation of measurement noise


```

```
% Initial state vector ( phi_x, bias_x )
x = [ atan2( acceleration_y(1), acceleration_z(1) ); 0 ];
% Set filter output for initial state
angle_x_version3 = zeros(numberOfSamples,1);
angle_x_version3(1) = 180/pi * x(1);

% Set equation error (process noise)
hApprox = 0.008;
varW1 = hApprox^2 * 0.0257*pi/180;
varW2 = 1e-8; % critical parameter for bias correction
Rx_z = [ varW1 0; 0 varW2];

% Initial covariance
P = Rx_z;

for n=1:numberOfSamples-1
    % System matrices
    timestep = discreteTime(n+1) - discreteTime(n);
    A = [ 1 -timestep; 0 1 ];
    B = [ timestep; 0];
    C = [ 1   0];

    % measurement in rad
    u = angular_velocity_x(n) / 180 * pi;
    y = atan2( acceleration_y(n+1), acceleration_z(n+1) );

    Q = A * P * A' + Rx_z;
    K = Q * C' * inv( C * Q * C' + Ry_z);
    P = (eye(2) - K * C) * Q;
    x = A * x + B * u + K * (y - C * A * x - C * B * u);

    angle_x_version3(n+1) = 180/pi * x(1);
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%5
% Plots
% Data plot
figure(1);
hold on;
plot(discreteTime,acceleration_x,'r-');
plot(discreteTime,acceleration_y,'g-');
plot(discreteTime,acceleration_z,'k-');
title('Acceleration data');
xlabel('Time');
ylabel('g');
legend('x acceleration', 'y acceleration', 'z acceleration');
axis tight;
grid on;

figure(2);
hold on;
plot(discreteTime,angular_velocity_x,'r-');
plot(discreteTime,angular_velocity_y,'g-');
plot(discreteTime,angular_velocity_z,'k-');
title('Angular velocity data');
xlabel('Time');
ylabel('Degree/s');
legend('pitch ', 'roll', 'yaw');
```

```
101 axis tight;
102 grid on;
103
104 % Results plot
105 figure(3);
106 hold on;
107 plot(discreteTime,angle_x_version1,'r-');
108 plot(discreteTime,angle_x_version2,'g-');
109 plot(discreteTime,angle_x_version3,'k-');
110 title('Angle in x direction');
111 xlabel('Time');
112 ylabel('Degree');
113 legend('Integrated gyro data', 'Calculated by acceleration', 'Kalman filtered');
114 axis tight;
115 grid on;
```

Program A.17: Evaluating the Kalman filter in Section 8.3

List of Tables

List of Figures

List of Programs

Bibliography

[1] A. Aitken. On Least Squares and Linear Combinations of Observations. *Proceedings of the Royal Society of Edinburgh*, 55:42–48, 1935.

[2] B. Aulbach. *Gewöhnliche Differenzialgleichungen*. Spektrum Akademischer Verlag, 2010.

[3] G. Goodwin and R. Payne. *Dynamic System Identification: Experiment Design and Data Analysis*. Academic Press Inc., 1977.

[4] L. Grüne. Modellierung mit Differentialgleichungen. Technical report, Universität Bayreuth, Bayreuth, 2008.

[5] H. Khalil. *Nonlinear Systems*. Prentice Hall PTR, 2002.

[6] L. Ljung. *System Identification: Theory for the User*. Pearson Education, 1998.

[7] J. Nocedal and S. Wright. *Numerical optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[8] J. Schoukens. System Identification. Technical report, Vrije Universiteit Brussel, 2013.

[9] J. Schoukens, R. Pintelon, and Y. Rolain. *Mastering System Identification in 100 Exercises*. John Wiley & Sons, 2012.

[10] E. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*. Springer, 1998.