INAUGURAL-DISSERTATION

zur
Erlangung der Doktorwürde
der
Naturwissenschaftlich-Mathematischen Gesamtfakultät
der
Ruprecht-Karls-Universität Heidelberg

vorgelegt von
Diplom-Mathematiker Christoph Zimmer
aus Kassel

Tag der mündlichen Prüfung:

Parameter estimation for stochastic models of biochemical reactions

Advisors:
Prof. Dr. Dr. h. c. Hans Georg Bock
Prof. Dr. Ursula Kummer

Abstract

Parameter estimation is central for the analysis of models in Systems Biology. Stochastic models are of increasing importance. However, parameter estimation for stochastic models is still in the early phase of development and there is need for efficient methods to estimate model parameters from time course data which is intrinsically stochastic, only partially observed and has measurement noise.

The thesis investigates methods for parameter estimation for stochastic models presenting one efficient method based on integration of ordinary differential equations (ODE) which allows parameter estimation even for models which have qualitatively different behavior in stochastic modeling compared to modeling with ODEs. Further methods proposed in the thesis are based on stochastic simulations. One of the methods uses the stochastic simulations for an estimation of the transition probabilities in the likelihood function. This method is suggested as an addition to the ODE-based method and should be used in systems with few reactions and small state spaces. The resulting stochastic optimization problem can be solved with a Particle Swarm algorithm. To this goal a stopping criterion suited to the stochasticity is proposed. Another approach is a transformation to a deterministic optimization problem. Therefore the polynomial chaos expansion is extended to stochastic functions in this thesis and then used for the transformation.

The ODE-based method is motivated from a fast and efficient method for parameter estimation for systems of ODEs. A multiple shooting procedure is used in which the continuity constraints are omitted to allow for stochasticity. Unobserved states are treated by enlarging the optimization vector or using resulting values from the forward integration. To see how well the method covers the stochastic dynamics some test functions will be suggested. It is demonstrated that the method works well even in systems which have qualitatively different behavior in stochastic modeling than in modeling with ODEs. From a computational point of view, this method allows to tackle systems as large as those tackled in deterministic modeling.

Zusammenfassung

Parameterschätzung ist sehr wichtig für die Analyse von Modellen in der Systembiologie. Stochastische Modelle sind von wachsender Bedeutung. Allerdings ist Parameterschätzung für stochastische Modelle noch im Anfangssstadium der Entwicklung und es besteht Bedarf an effizienten Methoden, Modellparameter zu schätzen auf Basis von Zeitreihendaten mit intrinsischer Stochastizität, die nicht vollständig beobachtbar sind und mit Messfehlern aufgezeichnet werden.

Die Arbeit untersucht Methoden für die Parameterschätzung und zeigt eine Methode auf, die auf Integration von gewöhnlichen Differentialgleichungen basiert, mit der man Parameter selbst von Modellen schätzen kann, deren Systemdynamik sich im stochastischen Modell qualitativ vom Differentialgleichungsmodell unterscheidet. Weitere Methoden basieren auf stochastischen Simulationen. Eine Methode nutzt die stochastischen Simulationen, die Übergangswahrscheinlichkeiten in der Likelihood Funktion zu schätzen. Diese Methode sei als Ergänzung zur auf Differentialgleichungen basierten Methode empfohlen für Systeme mit wenigen Reaktionen und kleinen Zustandsräumen. Das resultierende Optimierungsproblem ist stochastisch und kann mit Hilfe eines Particle Swarm Algorithmus gelöst werden. Dafür wird ein an die Stochastizität angepasstes Abbruchkriterium eingeführt. Ein weiterer Ansatz ist eine Transformation in ein deterministisches Optimierungsproblem. Dafür erweitert diese Arbeit die sogenannte polynomielle Chaos-Entwicklung auf stochastische Funktionen und nutzt sie für die Transformation.

Die auf Differentialgleichungen basierende Methode ist von einer schnellen und effizienten Methode für Parameterschätzung bei Systeme von Differentialgleichungen motiviert. Ein Mehrzielverfahren wird verwendet, in dem die Stetigkeitsbedingungen weggelassen werden, um die Stochastizät zu berücksichtigen. Falls es Systemzustände gibt, die nicht beobachtetet werden, wird der Optimierungsvektor vergrößert oder die Ergebnisse der Vorwärtsintegration verwendet. Um zu prüfen, wie gut die Methode die Stochastizität widerspiegelt, werden einige Kennzahlen vorgeschlagen. Es wird demonstriert, dass die Methode auch bei Systemen erfolgreich ist, deren dynamisches Verhalten bei der stochastischen Modellierung qualitativ verschieden ist vom Verhalten bei Modellierung mit Differentialgleichungen. Was die Rechenzeit betrifft, ermöglicht die Methode eine Behandlung von Systemen gleicher Größe, wie es bei der Parameterschätzung für Modelle von Differentialgleichungen geschieht.

Contents

Introduction

Computational modeling is a central approach in Systems Biology for studying increasingly complex biochemical systems. Parameter estimation is very important for the analysis of models in Systems Biology. Progress in experimental techniques, e.g. the possibility of measuring small numbers of molecules in single cells [1], highlights the need for stochastic modeling approaches to analyze this data. Simulation methods have been developed for decades since [2] and nowadays exist with a lot of variants [3]. Parameter estimation methods for stochastic models however are still in the early phase of development.

Approaches exist for time series data using stochastic simulations. Due to the Markov property of the time series the likelihood function factorizes into the product of transition probabilities. These transition probabilities are generally unknown in stochastic modeling. They can be estimated using stochastic simulations. This can be done with density estimation methods [4,5]. Another approach is the use of a reversible jump algorithm [6]. The parameter estimation is then performed with Bayesian methods. An alternative to that is the use of a stochastic gradient descent [7] and the use of a reversible jump Markov chain Monte Carlo method for the estimation of the transition probabilities. Using a surrogate probabilistic model as an approximation is faster from a computational point of view [8]. Another approximation is suggested in form of an approximate maximum likelihood method [9], where also a singular value decomposition likelihood method is described.
A second class of methods focuses on a numerical solution of the Chemical Master equation (CME), which describes the probability for each state in dependence of the time. These systems are generally high dimensional. To address this problem a state space truncation can be used [10] or moment-closure methods, which are an approximation focusing on a finite number of moments of the probability distribution [11,12]. [13,14] use an adaptive Galerkin method for the solution of the CME. If distribution information is available from measurement a finite state projection [15] can be used to solve the CME without simulations. The common challenge is the fact that the solution of the CME as well as simulation-based methods become very time-consuming as the number of states in the state space becomes larger.

This thesis will investigate methods for parameter estimation for stochastic models and test them on example systems from Systems Biology suggesting to use a method based on multiple shooting for the integration of the ordinary differential equations (ODE). This method is able to infer parameters even from partially observed data with measurement noise. It will be named multiple shooting for stochastic systems (MSS) in the following. Test functions are suggested to see how well the method covers the stochastic dynamics. This work has been submitted to PloS ONE and the description in this thesis is based on this article [16]. Methods based on stochastic simulations are computationally much more cost intensive. Nevertheless one of them is suggested as addition for systems with few reactions and small number of states in the state space.

The simulation-based objective functions are an addition in cases where the test functions for the MSS method advise against its use especially if only few reactions occur. As this is only the case for one of the considered designs in the Immigration-Death model they will be tested there.
In order to compare results with an exact estimation and to investigate the average behavior of the simulation-based methods the concept of the probability generating function is used to derive an exact solution for the CME, which is only possible in few simple systems. The solution is derived by building a power series with the CME, which satisfies a partial differential equation. For this a solution can be found, of which derivatives are calculated, which then can be shown to be a solution of the CME.
Then it is presented that a direct use of the least squares functional with stochastic simulation leads to a bias in the objective function landscape. The reason is that the least squares functional is optimal under the assumption of normally distributed measurement error, which is not satisfied by the intrinsic stochasticity. The effect of the violation is strong so the functional should not be used.
Therefore the use of the likelihood function is more appropriate. The transition probabilities in the likelihood function are estimated using a relative frequency of simulations. This objective function will be named likelihood simulation (LS) function. The functional is unbiased with respect to stochasticity in the data and simulations, which is tested with an exact solution for the transition probabilities gained with the probability generating function. Other methods for estimating the transition probabilities can be found in [4, 6]. However all approaches become computationally cost intensive with increasing state space. To account for larger state spaces a modification of the objective function is suggested. It will be named modified likelihood simulation (MLS) function.
For the estimation of the parameter vector a stochastic optimization problem has to be solved, which can be done using a black box optimizer such as Particle Swarm. To account for the stochasticity in the objective function a stopping criterion suited to the stochasticity is suggested. This stopping criterion should be also relevant for other black box optimization algorithms in context of stochastic functions. Another approach is a transformation to a deterministic problem using the polynomial chaos expansion. The polynomial chaos expansion is already successfully used for deterministic functions [17] and it is shown here to work for stochastic functions as well.

The MSS procedure is motivated by a method proposed by Bock [18] for the parameter estimation in systems of ODEs and further developed by [19] and already successfully applied to deterministic systems with chaotic behavior by [20, 21]. The MSS method can tackle models with fully observed and partially observed data sets. The fact that it works without stochastic simulations and without solving a high dimensional CME means that it is possible to tackle systems of a size as large as realistic models being tackled with ODEs. The method is based on short time ODE integration and performs successfully even on models which behave qualitatively different when modeled stochastically (see [22] for an example of such a model). The advantage of the MSS method is very high speed since neither solving a high dimensional CME system nor lots of stochastic simulations are required.
As the objective function of the MSS method is completely deterministic it is possible to apply derivative-based methods as well as methods without derivatives for the optimization. However, the focus of this thesis is the formulation of the optimization problem with a suitable objective function. The choice of the numerical optimization method and especially the question of local minima will not be the focus of this thesis. Concerning this question section 3.4 will refer to the literature.
The MSS method is described with equidistant time points of measurements here for notational simplicity. It is possible to apply it without changes to non-equidistant time points of measurements, which is very important for the applicability of optimum experimental design. Due to its

structure the method is easily able to handle measurement noise although it will of course reduce the accuracy of the estimation.
As the MSS method uses ODE integration for stochastic data it loses its theoretical maximum likelihood property. Nevertheless it is possible to check how much the theoretical assumptions are violated and the thesis will present that this is not problematic even in models with irregular stochastic oscillations. Furthermore the MSS method is successfully applied to models which are structurally not identifiable using "traditional" single shooting ODE methods (TSS method).

The Lotka-Volterra model describes the dynamical development of a predator and prey population. It is a proof of concept example for the MSS method as it demonstrates to be able to cope with partially observed stochastic time course data and measurement noise. The next model is a Calcium oscillation model [22]. Calcium signaling is important for cell development, fertilization and death [23]. The model furthermore serves as an example of a qualitatively different behavior in stochastic modeling than in ODE modeling. The only scenario in which there are test functions that advise against the use of the MSS method is a certain design in the Immigration-Death model, for which the MLS objective function is applied.

The thesis is structured as follows:

The first chapter shortly introduces modeling of biochemical reactions (section 1.1) and then describes why stochastic modeling is relevant and how systems can be modeled stochastically including a description of the Gillespie algorithm, which is a very well known method to simulate systems stochastically (section 1.2). The first chapter concludes with a short motivation why new methods for parameter estimation should be developed (section 1.3).

The second chapter proposes methods for parameter estimation based on stochastic simulations. The very first approach is a naive use of the least squares functional with stochastic simulations (section 2.1). To see the average behavior the concept of the probability generating function is used (section 2.3.1), which leads to the conclusion that the first functional is biased (section 2.3.2). Second is a likelihood function-based approach estimating the transition probabilities with simulations (LS function, section 2.2) also suggesting a modification (MLS function) for large state spaces.
The last section of this chapter is devoted to the solution of the stochastic optimization problem. One option is the use of a black box optimization algorithm, namely Particle Swarm (section 2.4.2). Another option is the transformation to a deterministic optimization problem using the polynomial chaos expansion (section 2.4.3), which is extended to stochastic functions (section 2.4.3).

The third chapter suggest the very fast MSS method based on multiple shooting for the ODE integration using residuals instead of the transition probabilities. This method is able to handle partially observed models (section 3.2) as well as measurement noise. As the theoretical maximum likelihood properties are lost due to the use of the residuals test functions are suggested (section 3.3) to see how strong the maximum likelihood assumptions are violated.

The application chapter applies the suggested methods to examples from systems biology. The first model is an instructive example: an Immigration-Death model (section 4.1). As in this model the exact solution can be calculated using the probability generating function it allows a comparison of the methods: the simulation-based method using the MLS function with two different Particle Swarm algorithms (section 4.1.1), the polynomial chaos expansion (section 4.1.2) and the MSS method (4.1.3). Concluding this section some remarks are made concerning the performance of

the estimation with increasing number of observations or molecules in the steady state.
The MSS method is then applied to a Lotka-Volterra model (section 4.2), where it performs successfully on partially observed data with measurement noise. The last model is a Calcium oscillation model (section 4.3) with qualitatively different behavior in deterministic modeling than in stochastic modeling.

Chapter 1

Stochastic Modeling

1.1 Modeling of Biochemical Reactions

Computational modeling is a key technique for the analysis of complex systems in the sciences. It allows creating testable hypotheses, which is of high importance for the verification or falsification of an assumed systems behavior. Furthermore computational modeling allows a prediction of the future systems behavior. According to the subject of research the right level of detail has to be chosen, which means that the model will be a simplification of the real world covering the essential details [24].
Systems of biochemical reactions are often modeled as systems of ODEs describing the time dependent development of the concentrations of the species involved. $[X] = ([X_1], \ldots, [X_D])$ denotes the D reactants and $v_1, \ldots, v_r$ the kinetics of the r reactions. The so called stoichiometric matrix

$$S = \begin{pmatrix} s_{11} & s_{12} & \ldots & s_{1r} \\ s_{21} & s_{22} & \ldots & s_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ s_{D1} & s_{D2} & \ldots & s_{Dr} \end{pmatrix} \tag{1.1}$$

is a $D \times r$ dimensional matrix where s_{ij} describes the gain or loss for substance X_i due to the j-th reaction. Therefore the system of reactions can be written as

$$s_{1j}^- X_1 + s_{2j}^- X_2 + \ldots + s_{Dj}^- X_D \longrightarrow s_{1j}^+ X_1 + s_{2j}^+ X_2 + \ldots + s_{Dj}^+ X_D, \quad \text{for } j = 1, \ldots, r$$

where

$$s_{ij}^- = \begin{cases} -s_{ij}, & s_{ij} < 0 \\ 0, & else \end{cases} \quad \text{and} \quad s_{ij}^+ = \begin{cases} s_{ij}, & s_{ij} > 0 \\ 0, & else. \end{cases} \tag{1.2}$$

In terms of ODEs this systems is written as

$$\frac{d[X](t)}{dt} = S\, v([X], t). \tag{1.3}$$

For more details on modeling biochemical reactions see amongst many others [24] or [25].

1.2 Stochastic models

1.2.1 Background

This section will give a short introduction into the idea of stochastic modeling. For an overview see [3] or for more details [2].
There are two possible sources of noise or stochasticity in modeling. The first is extrinsic stochasticity, for example the very well known measurement error or noise. Apart from this measurement noise systems of ODE evolve deterministically and will therefore be called deterministic models in this text. A second possible source of stochasticity in stochastic systems is systems intrinsic, or also named inherent. That means that a system evolves stochastically. Think of an radioactive decay with Poisson distributed decays. Other sources of intrinsic stochasticity can be the orientation or speed of the molecules.
Whilst the deterministic modeling describes the development of concentrations of substances, stochastic modeling focuses on single molecules. Therefore deterministic modeling is appropriate for systems with "high" concentrations of the reacting species and stochastic modeling for "low" concentrations. This can be stated even stronger as it is possible to show that for increasing number of molecules stochastic systems can be approximated with a so called τ-leap method, which then can be approximated by stochastic differential equations and finally by ODEs [3, 26].
First it will be presented how it is possible to simulate a system stochastically and then some remarks will be given.

Probability for collision

Consider a bi-molecular reaction

$$X_1 \ + \ X_2 \ \rightarrow ?$$

with a right hand side that is neglected right now. This equation describes a reaction in which a X_1-molecule collides with an X_2-molecule and reacts. What is the probability for such a reaction? Under the assumption of a well stirred reactor and randomly diffusing particles the probability for a collision of a single pair X_1, X_2 is simply the probability that these two molecules have a distance less the sum of their reacting distances δ:

$$P(|Pos_{X_1} - Pos_{X_2}| < \delta) \tag{1.4}$$

where Pos_{X_1} denotes the position of the X_1 particle. As the probability for any Pos_{X_2} position will be the same, the probability (1.4) is simply the probability that X_1 is in a sphere (around X_2) with radius r:

$$P(|Pos_{X_1} - Pos_{X_2}| < r) \ = \ \frac{4 \ \pi \ \delta^3}{3 \ V} \tag{1.5}$$

with a volume V. As the reactor is well stirred this probability is independent of time. Other influences are independent of V, for example the probability for a reaction, which depends on the energy of the molecules.

Mass action kinetics

Consider a system of chemical reactions as in equation (1.1) and assume that each of the r reactions R_j, $j = 1, \ldots, r$, has a stochastic rate constant c_j which combines the probability for a collision as in equation 1.5 and the probability for a reaction. The associated rate law or hazard function $h_j(x, c_j)$ describes the probability for a reaction R_i of any of the participating species in the next small time step $t, t + dt$ and is for

- zeroth-order mass action:

$$R_j : \varnothing \xrightarrow{c_j} X_i, \qquad \text{hazard function: } h_j(x, c_j) = c_j,$$

- first-order mass action:

$$R_j : X_i \xrightarrow{c_j} ?, \qquad \text{hazard function: } h_j(x, c_j) = c_j\ x_i,$$

- second-order mass action:

$$R_j : X_i + X_k \xrightarrow{c_j} ?, \qquad \text{hazard function: } h_j(x, c_j) = c_i\ x_i\ x_k,$$

  or

$$R_j : 2X_i \xrightarrow{c_j} ?, \qquad \text{hazard function: } h_j(x, c_j) = c_j\ \tfrac{x_i\ (x_i-1)}{2}.$$

For comments and modeling of higher order reactions or other kinetics see [24, 27].

Rate law conversion

Now the next question is how to compute the stochastic rate constants c_j from the deterministic rate constants. Denote with $[X]$ the concentration of the species X and with x the number of molecules, thus $x = [X]n_A V$ with the Avogadro constant n_A and a volume V.
For deterministic systems the rate laws are usually given with the measurement unit moles per liter and second or a multiple of it because the concentrations are measured in moles per liter.

- Zeroth-order mass action:
  A deterministic rate constant θ given in moles per liter and second corresponds to an influx of $n_A V\theta$ molecules per second, consequently the conversion is as follows

$$R_j : \varnothing \xrightarrow{c_j} X_i \quad c_j = n_A V\theta.$$

- First-order mass action:
  The deterministic decrease of $\theta[X_i]$ means a decrease of $n_A\theta[X_i]V = \theta x_i$ molecules per second and therefore

$$R_j : X_i \xrightarrow{c_j} ? \quad c_j = \theta.$$

- Second-order mass action :
  The deterministic rate constant means a decrease of $\theta[X_i][X_k]$ moles per second, which is $\theta n_A[X_i][X_k]V = \frac{\theta x_i x_k}{n_A V}$ molecules per second, which leads to

$$R_j : X_i + X_k \xrightarrow{c_j} ? \quad c_j = \frac{\theta}{n_A V}.$$

Remarks on the use of higher order reactions can be found in [28].

1.2.2 Gillespie's algorithm

Stochastic modeling of systems of biochemical reactions algorithmically means the following: The time τ until the next reaction event R_j and the choice which reaction is fired next are random events. Let c_j be a reaction constant for reaction R_j, $j = 1, \ldots, r$ and h_j the hazard function for reaction R_j such that the probability for a R_j reaction in a small time interval dt is $h_j dt$. Then the Gillespie algorithm allows to simulate a stochastic time course of the biochemical system: As stated above the time τ until the next reaction and which reaction is fired are random events. The basic idea of Gillespie's algorithm is to draw two random numbers: one determines the time until the next reaction and the other one chooses the reaction R_j to be fired.

(1) First, calculate the sum over all hazard functions h_j:

$$h_0 = \sum_{j=1}^{r} h_j(x, c_j).$$

(2) The stochastic time step is

$$\tau = -\frac{1}{h_0} \log\ u_1,$$

with a uniformly distributed random number $u_1 \sim U((0, 1])$.

(3) For the determination of the reaction, choose $\mu \in \mathbb{N}$ such that

$$\sum_{j=1}^{\mu-1} \frac{h_j}{h_0}\ < u_2\ \leq\ \sum_{j=1}^{\mu} \frac{h_j}{h_0}$$

is fulfilled with another uniformly distributed random number $u_2 \sim U((0, 1])$.

(4) Realize the reaction R_μ by updating the substrate numbers involved in the reaction and increase the time by τ. Go to (1) and repeat until the desired end time.

There are some ways of implementing the Gillespie's algorithm as the Direct Method described above, the First Reaction Method or the Next Reaction Method, which reduces the computational complexity making use of the data structure, see [3] for a review. For the following it is important to know that it is possible to simulate a stochastic time course given some initial conditions and reaction constants. Reaction constants and correspondingly the rate laws for the ODE representation will be named parameters. To conclude this paragraph some effects of stochastic modeling are demonstrated.

1.2.3 A first example and some effects

To illustrate the importance and the effects of such intrinsic stochasticity the example of an Immigration-Death model is used, which can, for example, model a substance diffusing in a cell:

$$\emptyset \xrightarrow{\theta_1} X$$
$$X \xrightarrow{\theta_2} \emptyset.$$

There are two reactions: R_1 is the immigration of a species and R_2 is the death reaction. Hence if θ_1 and θ_2 are given in moles per liter $c_1 = n_A V \theta_1$ and $c_2 = \theta_2$. Reaction R_1 means the immigration of a species to the system, which should happen independent of the systems state with a constant rate θ_1. Reaction R_2 means the death reaction depending on the systems state. The hazard functions are $h_1(x, c_1) = c_1$ and $h_2(x, c_2) = c_2 x$. Figure 1.1 shows some realizations of this process calculated using the Gillespie algorithm (section 1.2.2) with COPASI [28]. To illustrate

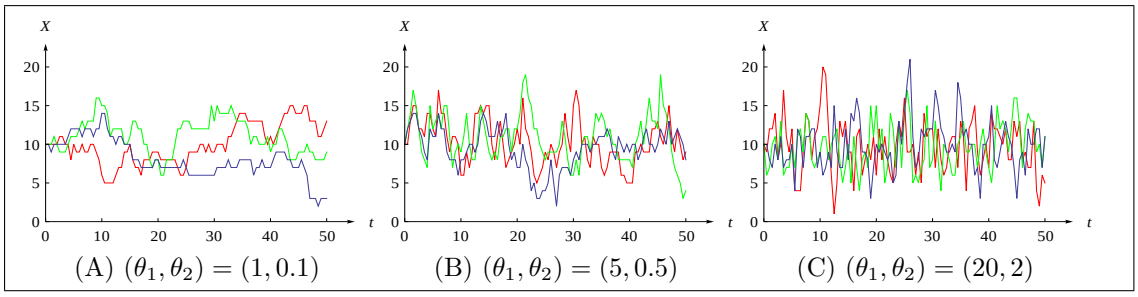


Figure 1.1: **Stochastic realization of the Immigration-Death model.** Three different realizations with 100 observations with $T = 50$ and $x_0 = 10$ for three different parameters.

the importance of stochastic modeling it is now compared to the modeling with ODEs. In terms of differential equations this systems would read as

$$\frac{d[X]}{dt} = \theta_1 - \theta_2[X], \quad [X](0) = [X]_0$$

with a solution

$$[X]_{(\theta_1,\theta_2)}(t) = \frac{\theta_1}{\theta_2} \text{ for all t}$$

for $[X]_0 = \frac{\theta_1}{\theta_2}$. The steady state is chosen as initial value mainly for illustration purpose. In addition for many systems transient dynamics is not readily available and therefore steady state data is typical of these systems. The systems behavior of the Immigration-Death model is then constant over time in ODE modeling and only depends on the quotient $\frac{\theta_1}{\theta_2}$ and not on the absolute values of θ_1 and θ_2. From point of stochastic modeling this means that either no reaction takes place or two reactions 1 and 2 always at the same time so canceling out the effects. This is in contradiction to the intuition from modeling on a reaction by reaction basis. Furthermore one would expect that the absolute value of θ_1 and θ_2 also has an effect on the systems behavior: this effect is not represented in ODE modeling but in stochastic modeling, see figure 1.1.

1.3 Why a different method for parameter estimation?

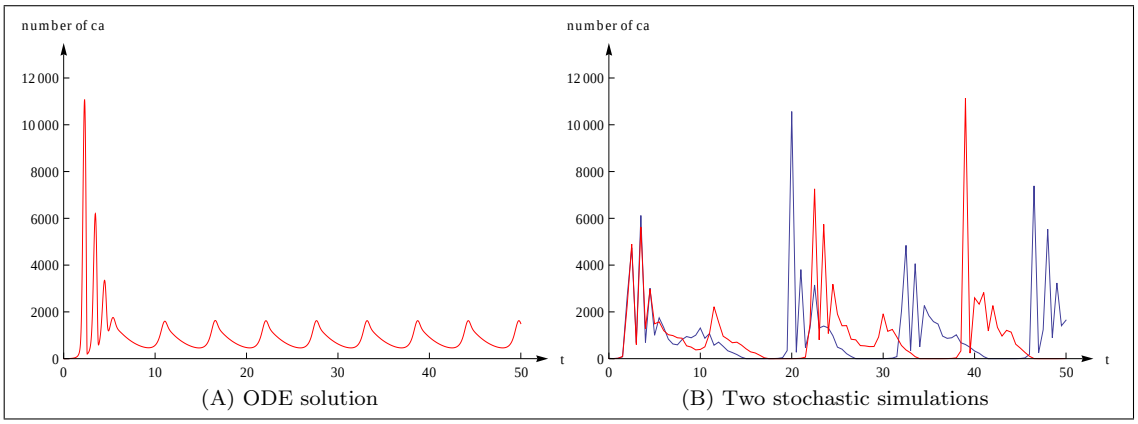
After the introduction to stochastic modeling in the previous section, this section is devoted to the introduction to parameter estimation questions. For modeling with ODEs and normally distributed measurement noise the least squares functional is the appropriate choice. The question is now how to perform a parameter estimation for data which is modeled stochastically.

Immigration-Death model

The Immigration-Death model with initial conditions in the steady state is not identifiable using traditional ODE techniques for parameter estimation as the solution of the ODE does in the steady state only depend on the quotient of the two parameters. Nevertheless if the Immigration-Death model is modeled stochastically it shows a different behavior for parameters with the same quotient but different absolute value, namely the reactivity of the systems changes, figure 1.1. Consequently both parameters should be identifiable, which indicates a need for different methods.

Calcium oscillation model

This can be seen even more distinctly in a Calcium oscillation model [22]. In this model stochastic modeling leads to systems behavior which is very different from the behavior of the ODE modeled system. Very different means that it is qualitatively different and can not be modeled as the ODE solution plus normally distributed measurement noise, see figure 1.2. This example suggests that the methods for parameter estimation in ODE models should not be used for stochastic models.



(A) ODE solution (B) Two stochastic simulations

Figure 1.2: **Qualitatively different behavior between ODE modeling and stochastic modeling.** (A) shows the ODE solution and (B) two stochastic realizations with 100 observations and $T = 50$. Simulation with the Gillespie algorithm (section 1.2.2) with the software COPASI [28]. The model equations and parameters can be found in subsection 4.3.1.

Chapter 2

Evaluation of the objective function based on stochastic simulations

2.1 Single Shooting with stochastic trajectory

Measurements $\nu = (\nu_0, \ldots, \nu_n)$ are taken at time points $t_0, \ldots, t_n$. The deterministic optimization problem with the well known least squares functional is

$$\hat{\theta} \;=\; \underset{\theta}{\operatorname{argmin}} \; F_d(\nu, \theta)$$

for normally distributed measurement error with variances σ_i^2:

$$F_d(\nu, \theta) = \sum_{i=1}^{n} \frac{\|\nu_i - h(t_i, \theta, \nu_0, t_0)\|_2^2}{\sigma_i^2} \tag{2.1}$$

with a model response function h describing the model response in dependence on the parameter θ, time t and initial values ν_0 at time t_0. The model response h can be, for example, the solution of a system of ODEs.
To underline its difference to other ODE-based approaches described in chapter 3 this objective function will be named "traditional single shooting" (TSS) function and correspondingly together with an optimization it will be named TSS method. Single shooting means that each trajectory is started in an initial value and then integrated (shot) until the last measurement.
To focus on the intrinsic stochasticity assume from now on that measurements are exact. As the underlying model is a stochastic model focusing on species' counts this means $\nu_i \in \mathbb{N}_0^D$ for $i = 1, \ldots, n$. The main objective of this section is to investigate the behavior of the least squares functional with stochastic single shooting in stochastic modeling.
The following points are different in stochastic modeling:

- What is the variance σ_i^2? Although there is no measurement noise the observation ν is stochastic and therefore has a variance. But as the distribution is in most cases not normal and even unknown this variance is also unknown.
- In contrast to the model response function $h(t_i, \theta, \nu_0, t_0)$ from (2.1), which is deterministic, the model response in stochastic modeling is a stochastic process. To underline this the

capital letter H will be used instead of the small letter h. Hence the value of the objective function F is now a random variable.

- The state space of the model response h is $\mathbb{R}_+^D$ where $\mathbb{R}_+$ denotes the positive real numbers and zero. This is different for H, which is the result of the Gillespie algorithm, which simulates on a single molecule basis. Therefore the state space is $\mathbb{N}_0^D$.
- Note that the assumptions for a maximum likelihood estimator are not fulfilled as the distribution is not Gaussian and consequently the theoretical properties are not guaranteed. How much this influences the estimation will be investigated in this section.

In statistics the law of large numbers states that the average of m independent identically distributed random variables converges with increasing m to the expectation if the expectation is finite. One possible approach is to make use of this property to reduce the stochastic fluctuation in H. Furthermore using more than one simulation of H would allow to estimate a variance from the simulations. How far this makes sense will be discussed later.
First there are two possibilities of taking the average: averaging the least squares differences calculated for each simulation,

$$F_{s,q}^{(m)}(\nu,\theta) \;=\; \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{n}\left\|\nu_i - H^{(j)}(t_i,\theta,\nu_0,t_0)\right\|_q^q, \tag{2.2}$$

or averaging the trajectories with a resulting objective functional,

$$F_{tr,q}^{(m)}(\nu,\theta) \;=\; \sum_{i=1}^{n}\left\|\nu_i \;-\; \frac{1}{m}\sum_{j=1}^{m}H^{(j)}(t_i,\theta,\nu_0,t_0)\right\|_q^q. \tag{2.3}$$

Remark 2.1 (Using averages of trajectories). One option would be to calculate a mean of the m observations and then to consider its squared difference to the measurement data as objective function value as done in $F_{tr,q}^{(m)}$, (2.3). This generally seems not to be a good choice because of oscillatory systems where the stochasticity dislocates the period. Then averaging could simply yield a constant time series, which is of no sense for the parameter estimation. How big this effect really is and if there are remedies will not be discussed here but is an open point.

The other option is taking the average of the least squares difference values for each simulation. This leads to a value close to the expectation of the least squares differences. Next point are the variances σ_i. One approach is to assume $\sigma_i{=}\sigma$, which makes the optimization independent of the variance, see equation (2.2). A drawback of this approach is that it ignores different sizes of the variances, which might occur for different time points of measurement. This can lead to inaccuracies in the estimation. Results will be shown later in section 2.3, figure 2.1. Another approach is to use an estimated variance of the simulations, hence $\widehat{\sigma_i}^2 = Var\left(\{H^{(j)}(t_i,\theta,t_0,\nu_0), j=1,\ldots,m\}\right)$, resulting in

$$F_{s,q}^{(m)}(\nu,\theta) \;=\; \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{n}\left\|\frac{\nu_i - H^{(j)}(t_i,\theta,\nu_0,t_0)}{\widehat{\sigma_i}^2}\right\|_q^q. \tag{2.4}$$

Then the variance will be estimated correctly if $\theta{=}\theta^{(0)}$ and otherwise not, which questions the accuracy of this approach. Results will also be shown later in section 2.3, see figure 2.2.
The resulting optimization problem is

$$\hat{\theta}_k \;=\; \underset{\theta}{\operatorname{argmax}}\; F_{k,q}^{(m)}(\nu,\theta) \tag{2.5}$$

with $k \in \{s, \tilde{s}\}$ which means $\sigma_i = \sigma$ or $\sigma_i = \widehat{\sigma}_i$.
To investigate the properties of that estimator the concept of the probability generating function will be used in section 2.3 to calculate the expectation of the objective function landscape. This calculation suggests that the estimator is biased. Therefore it will not be considered in the application chapter.

2.2 Likelihood Function with transition probabilities

2.2.1 Estimating the transition probabilities

In deterministic modeling with normally distributed measurement errors the least squares functional leads to a maximum likelihood estimator. In stochastic modeling the distribution of the random variables - namely the stochastic process of the reacting substances - is not known and not necessarily Gaussian. Therefore the likelihood function is used for the optimization

$$L(\nu, \theta) \;=\; \prod_{i=1}^{n} p_\theta(\nu_i, t_i | \nu_{i-1}, t_{i-1}) \tag{2.6}$$

under consideration that the data ν is a Markov process with the transition probability $p_\theta(\nu_i, t_i|\nu_{i-1}, t_{i-1})$ for a transition from state ν_{i-1} at time t_{i-1} to a state ν_i at time t_i. Hence the log-likelihood function is

$$l(\nu, \theta) \;=\; \sum_{i=1}^{n} \log\left(p_\theta(\nu_i, t_i|\nu_{i-1}, t_{i-1})\right). \tag{2.7}$$

This function will be maximized over θ, which means a search for the parameter θ that gives the highest probability for the realization ν. But as the probability $p_\theta(\nu_i, t_i|\nu_{i-1}, t_{i-1})$ is generally not known it has to be estimated. As the state space is discrete this can be done with, say m, simulations:

$$\widehat{p_\theta}^{(m)}(\nu_i, t_i|\nu_{i-1}, t_{i-1}) \;=\; \frac{\sum_{j=1}^{m} 1_{\{H^{(j)}(t_i, \theta, \nu_{i-1}, t_{i-1}) \,=\, \nu_i\}}}{m}, \tag{2.8}$$

where

$$1_{\{x \,=\, y\}} \;=\; \begin{cases} 0, & x \neq y \\ 1, & x = y. \end{cases}$$

Equation (2.8) uses the relative frequency of the m simulations starting at time t_{i-1} in ν_{i-1} and those resulting in state ν_i at time t_i as estimate. It is assumed that m is chosen large enough so that $\widehat{p_\theta}^{(m)} > 0$. To relax this assumption the next subsection suggests a modification. The optimization functional is then

$$F_L^{(m)}(\nu, \theta) \;=\; \sum_{i=1}^{n} \log\; \widehat{p}_\theta^{(m)}(\nu_i, t_i|\nu_{i-1}, t_{i-1}). \tag{2.9}$$

This method will be named likelihood simulation (LS) method.

2.2.2 Modification of transition probabilities

Example 2.1 (Example for the state space after short time). The objective function and therefore also the optimization will fail if for two time points t_{i-1}, t_i: $\widehat{p_\theta}^{(m)}(\nu_i, t_i|\nu_{i-1}, t_{i-1}) = 0$ holds, which means that the number of simulated trajectories starting in ν_{i-1} and ending in ν_i is zero. This might happen if the parameter is very far from the true parameter or the number of simulations m is not high enough or the space of possible values which a simulation can reach after $t_i - t_{i-1}$ is too large.
10 000 simulations are run for $t_1 - t_0 = 0.5$ using a Calcium oscillation model [22] with a parameter setting as in section 4.3. For each species the 10%-quantiles and 90%-quantiles of the resulting trajectories give a total three dimensional space of a range of 10^8. Trying to estimate $\hat{p}$ with equation (2.8) means trying to find a single point in a $3d$ range of 10^8 points. To address this point a weighted stochastic simulation algorithm [29] might be used. Nevertheless for a good estimate very many simulations are needed (see [30] for the original stochastic simulation algorithm).

A realization is only counted if it ends up at the right value. If not no information is gained for the objective function. But intuitively speaking one would guess that there is at least some information in it. If the parameter is completely wrong all simulations might end up far away from the data points. If the parameter is somehow close to the true value the simulations might end up somewhere around the data point. So the distance should give some information.

$$g_\theta^{(m)}(\nu_i, t_i|\nu_{i-1}, t_{i-1}) = \begin{cases} \widehat{p_\theta}^{(m)}(\nu_i, t_i|\nu_{i-1}, t_{i-1}), & \widehat{p_\theta}^{(m)}(\cdot|\cdot) > 0 \\ \left(m \sum_{j=1}^m \left\| H^{(j)}(t_i, \theta, \nu_{i-1}, t_{i-1}) - \nu_i \right\|_2^2\right)^{-1}, & else. \end{cases} \tag{2.10}$$

Note that also other norms would be possible as well as other switching criteria. Which is best is up to further research. Note that approximate Bayesian techniques [31] are similar to this with the difference that they "count" a simulation if their distance or a summary statistics is below a threshold.
Properties of the modification which will be named modified likelihood simulation (MLS) function:

- If $\widehat{p_\theta}^{(m)} > 0$ the function g is the same as the "old" function $p_\theta^{(m)}$ in (2.8).
- Any $\widehat{p_\theta}^{(m)} = 0$ will result in a lower value of the modification g than $\widehat{p_\theta}^{(m)} > 0$.
- g decreases with increasing distances of the simulations H to ν_i.
- Using a distance measure assumes a distribution of the random variables, but here only for the case $\widehat{p_\theta}^{(m)} = 0$.

Note that for this modification the number of simulations has to be constant within an optimization procedure. Otherwise the second point of the properties list will not hold any longer. As in section 2.1 it would also be possible to use other norms.
The resulting objective function is:

$$F_g^{(m)}(\nu, \theta) = \sum_{i=1}^n \log\ g_\theta^{(m)}(\nu_i, t_i|\nu_{i-1}, t_{i-1}) \tag{2.11}$$

and the optimization problem

$$\hat{\theta}_k = \underset{\theta}{\operatorname{argmax}}\ F_k^{(m)}(\nu, \theta). \tag{2.12}$$

It has to be kept in mind that $F_k^{(m)}(\nu, \theta),\ k = \{L, g\}$ for a fixed data set ν is a random variable in the simulations $\widehat{p_\theta}^{(m)}$. How to tackle this kind of stochastic optimization problem will be discussed in section 2.4.

2.3 The objective function landscapes

In this section the expectation value for the objective function $F_{s,q}^{(m)}$ from equation (2.2) and $F_L^{(m)}$ from equation (2.9) will be calculated for a simple test case in which it is possible to derive an exact solution for the transition probability, namely an Immigration-Death model.

2.3.1 The probability-generating function

To derive the transition probability the concept of the probability generating function is necessary. This subsection will motivate and derive the probability generating function and give some of their properties. The reaction system of the Immigration-Death model is

$$\begin{aligned} \emptyset &\xrightarrow{\theta_1} X \\ X &\xrightarrow{\theta_2 x} \emptyset, \end{aligned} \tag{2.13}$$

with the Chemical Master Equation system

$$\frac{\partial}{\partial t}p(j,t) \;=\; -(\theta_1 + j\theta_2)p(j,t) + (j+1)\theta_2 p(j+1,t) + \theta_1 p(j-1,t), \quad (j = 0, 1, \ldots) \tag{2.14}$$

with an initial distribution $p(j, 0)$ and t denoting time and j the number of species. The following derivation of the probability generating function for the Immigration-Death model is taken from [32], Example 4.5. Multiply equation (2.14) by z^j and sum up, which leads to the probability generating function

$$G(z,t) \;=\; \sum_{j=0}^{\infty} p(j,t) z^j \tag{2.15}$$

with $z \in \mathbb{R}$ and $|z| < 1$.

Remark 2.2 (Convergence of the series). The series $G(z, t)$ is convergent for all $|z| < 1$: As the p_j are probabilities it holds $|p(j, \cdot)z^j| \leq |p(j, \cdot)|\ |z^j| \leq \rho^j$ for a $\rho < 1$ and consequently it is majorised by the geometric series. As the p_j are from a probability density this is not the maximal radius of convergence.

If the series is convergent and the solution can be given in a close form than it is easily possible to calculate the probability $p(k, t)$ for k at time t by taking the $k - th$ derivative for $z = 0$:

$$\frac{1}{k!}\ \frac{\partial^k G(z,t)}{\partial z^k} \;=\; p(k,t).$$

Hence the limit of the power series has to be calculated: Therefore a partial differential equation is derived, for which a solution can be found, which is identical with the limit of equation (2.15).

Note first that the derivatives of $G(z,t)$ with respect to t and z are:

$$\begin{aligned}
\frac{\partial G(z,t)}{\partial t} &= \sum_{j=0}^{\infty} \frac{\partial p(j,t)}{\partial t} z^j \\
&= \sum_{j=0}^{\infty} \left(-(\theta_1 + j\theta_2)p(j,t) + (j+1)\theta_2 p(j+1,t) + \theta_1 p(j-1,t)\right) z^j \\
&= \underbrace{\sum_{j=0}^{\infty} -\theta_1 p(j,t) z^j}_{-\theta_1 G(z,t)} - \sum_{j=0}^{\infty} j\ \theta_2\ p(j,t) z^j + \sum_{j=0}^{\infty} (j+1)\ \theta_2 p(j+1,t) z^j + \\
&\quad \underbrace{\sum_{j=1}^{\infty} \theta_1 p(j-1,t) z^j}_{\theta_1 z G(t,z)}
\end{aligned}$$

(it is possible to change limit and differentiation inside the convergence radius) and

$$\frac{\partial G(z,t)}{\partial z} = \sum_{j=0}^{\infty} j p(j,t) z^{j-1}$$

and with that it follows

$$\begin{aligned}
&\frac{\partial G(z,t)}{\partial t} + \theta_2(z-1)\frac{\partial G(z,t)}{\partial z} \\
&= \theta_1(z-1)G(z,t) - \sum_{j=0}^{\infty} \theta_2\ j\ p(j,t) z^j + \sum_{j=0}^{\infty} \theta_2 (j+1) p(j+1,t) z^j + \theta_2(z-1)\sum_{j=0}^{\infty} j p(j,t) z^{j-1} \\
&= \theta_1(z-1)G(z,t) + \theta_2 \frac{1}{z}\sum_{j=0}^{\infty} (j+1) p(j+1,t) z^{j+1} - \theta_2 \sum_{j=0}^{\infty} j p(j,t) z^j + \theta_2(z-1)\frac{1}{z}\sum_{j=0}^{\infty} j p(j,t) z^j \\
&= \theta_1(z-1)G(z,t) + \underbrace{\theta_2 \frac{1}{z}\sum_{j=0}^{\infty} (j+1) p(j+1,t) z^{j+1} + \theta_2\left((z-1)\frac{1}{z} - 1\right)\sum_{j=0}^{\infty} j p(j,t) z^j}_{=0} \\
&= \theta_1(z-1)G(z,t),
\end{aligned}$$

which is a partial differential equation,

$$\frac{\partial G(z,t)}{\partial t} + \theta_2(z-1)\frac{\partial G(z,t)}{\partial z} = \theta_1(z-1)G(t,z). \tag{2.16}$$

This is a Lagrange's linear equation, for which an exact solution can be derived. The following heuristic ansatz is motivated by geometric arguments, see chapter 11 and 12 of [33]. The result will later be verified. For fixed t and z one gets the following auxiliary equations

$$\frac{dt}{1} = \frac{dz}{\theta_2(z-1)} = \frac{G(z,t)}{\theta_1(z-1)G(z,t)}.$$

From the first equality, $\frac{dt}{dz} = \frac{1}{\theta_2(z-1)}$, it follows by integrating over z and taking the exponential value $a_1 = \exp(-\theta_2 t)(z-1)$ with a constant $a_1 \in \mathbb{R}$ and from the second equality $\frac{dG}{dz} = \frac{\theta_1}{\theta_2} G$ by

solving this ODE $a_2 = G - \exp\left(\frac{\theta_1}{\theta_2} z\right)$ with a constant $a_2 \in \mathbb{R}$. This would also follow with a geometric argumentation, which would also motivate the following ansatz: An arbitrary function ϕ is a solution of the partial differential equation (2.16) if for a_1 and a_2 holds $\phi(a_1, a_2) = 0$. For the derivation of this argument see again [33]; here a verification that the result satisfies the partial differential equation shall suffice. Choose $\phi(a_1, a_2) = a_2 - \varphi(a_1)$ with a function φ to be determined by the initial condition and one gets the general solution

$$G(z,t) \;=\; \exp\left(\frac{\theta_1}{\theta_2} z\right)\; \varphi\left(\exp(-\theta_2 t)(z-1)\right). \tag{2.17}$$

Determine the function φ by the initial condition. Assume that at time $t = 0$ there are ν_0 species present. Then it holds with equation (2.17) and the fact that $p(j, 0) = 0$ for $j \neq \nu_0$ and $p(\nu_0, 0) = 1$:

$$\exp\left(\frac{\theta_1}{\theta_2} z\right) \varphi(z-1) \;=\; G(z,0) \;=\; z^{\nu_0}.$$

Define an auxiliary variable $\rho = z - 1$, thus $z = \rho + 1$, which leads to the form

$$f(\rho) = \frac{(\rho+1)^{\nu_0}}{\exp\left(\frac{\theta_1}{\theta_2}(z+1)\right)}.$$

Now choose $\rho = \exp(-\theta_2 t)(z-1)$ as argument for f, which leads to

$$\begin{aligned} G(z,t) \;&=\; \exp\left(\frac{\theta_1}{\theta_2}\left(z - \exp(-\theta_2 t)(z-1) - 1\right)\right) \quad \left(\exp(-\theta_2 t)(z-1) + 1\right)^{\nu_0} \\ &=\; \exp\left(\frac{\theta_1}{\theta_2}(z-1)(1 - e^{-\theta_2 t})\right) \left(1 + (z-1)e^{-\theta_2 t}\right)^{\nu_0}. \end{aligned} \tag{2.18}$$

Calculating the derivatives verifies that $G(t, z)$ from equation (2.18) is indeed a solution of the partial differential equation (2.16).
Now the probabilities can be derived. The probability for $\nu_t = k$ can be calculated by taking the k-th derivative with respect to z at $z = 0$ using the Leibniz-rule ([34]):

$$\begin{aligned} p_{(\theta_1,\theta_2)}(k, t|\nu_0, 0) \;&=\; \frac{1}{k!} \frac{\partial^k G(z,t)}{\partial^k z}\Big|_{z=0} \\ &= \frac{1}{k!} \sum_{j=0}^{k} \binom{k}{j} \left(\left(\frac{\theta_1}{\theta_2}\right)^j \left(1 - e^{-\theta_2 t}\right)^j \; e^{\frac{\theta_1}{\theta_2}(z-1)(1-e^{-\theta_2 t})} \right. \\ &\qquad \left. \left(1 + (z-1)e^{-\theta_2 t}\right)^{\nu_0-(k-j)} \left(e^{-\theta_2 t}\right)^{k-j} \prod_{i=0}^{k-j-1} (\nu_0 - i)\right)\Bigg|_{z=0} \\ &= \frac{1}{k!} \sum_{j=0}^{k} \binom{k}{j} \left(\left(\frac{\theta_1}{\theta_2}\right)^j \left(1 - e^{-\theta_2 t}\right)^j \; e^{-\frac{\theta_1}{\theta_2}(1-e^{-\theta_2 t})} \right. \\ &\qquad \left. \left(1 - e^{-\theta_2 t}\right)^{\nu_0-(k-j)} \left(e^{-\theta_2 t}\right)^{k-j} \prod_{i=0}^{k-j-1} (\nu_0 - i)\right). \end{aligned}$$

Inserting this in equation (2.14) verifies it as a solution. The solution is unique, see [35] for the uniqueness of CME solutions. This derivation of an exact solution is only possible in simple cases. A further example is an enzymatic reaction treated in [36].

Remark 2.3 (Properties for large and small t). Of high interest is the behavior for large and small t:

- **The steady state for $t \to \infty$:**
  It holds

$$p_{(\theta_1,\theta_2)}\left(\nu_t = k, t|\nu_0, 0\right) \xrightarrow{t\to\infty} \frac{\left(\frac{\theta_1}{\theta_2}\right)^k e^{-\frac{\theta_1}{\theta_2}}}{k!},$$

  which means $\nu_\infty \sim Pois\left(\frac{\theta_1}{\theta_2}\right)$ with a Poisson distribution $Pois$, as one would expect for the steady state, see [37].

- **Probability is concentrated in one point for $t = 0$:**
  For $t \to 0$ it holds:

$$p_{(\theta_1,\theta_2)}\left(\nu_t = k, t|\nu_0, 0\right) \xrightarrow{t\to 0} \begin{cases} 0 & k < \nu_0 \\ \frac{\prod_{l=0}^{k-1}(\nu_0 - l)}{k!} = 1 & k = \nu_0 \\ 0 & k > \nu_0. \end{cases}$$

  The second property means that for infinitesimal small t the probability for no reaction tends to one.

2.3.2 Expectation of the stochastic single shooting function

In section 2.3.1 a formula for the probability $p_\theta(\nu_1, t_1|\nu_0, t_0)$ to jump from state ν_0 at time t_0 to state ν_1 at time t_1 given a parameter $\theta = (\theta_1, \theta_2)$ was derived. Assume now ν_0 and t_0 fixed. ν_t is then a random variable with first moment as follows:

$$E_\theta[\nu_t|\nu_0] = \sum_{k=0}^{\infty} p_\theta(k, t|\nu_0, t_0)\ k \tag{2.19}$$

and similar second moment:

$$E_\theta[\nu_t^2|\nu_0] = \sum_{k=0}^{\infty} p_\theta(k, t|\nu_0, t_0)\ k^2. \tag{2.20}$$

Now, what we are interested in is the squared difference between the measurement data and the simulated data.
Choice $\sigma_i = \sigma$ (which means that the optimization is independent of σ):
Remember the functional $F_{s,2}^{(m)}$ from equation (2.2). There are two different sources of random effects: stochasticity inherent to the data and stochasticity in the simulation. The expectation value is taken over the set of possible measurement data with a true parameter $\theta^{(0)}$, $E_d[\cdot]$ and (!) the set of possible simulations with the corresponding parameter θ, $E_s[\cdot]$:

$$\begin{aligned} F_{s,2}^{E}(\theta) &= E_d\left[E_s\left[F_{s,2}^{(m)}(\nu, \theta)\right]\right] \\ &= \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{n} E_d\left[E_s\left[\nu_i^2 + H^{(j)}(t_i, \theta, t_0, \nu_0)^2 - 2\nu_i H^{(j)}(t_i, \theta, t_0, \nu_0)\right]\right] \\ &= \sum_{i=1}^{n} E_d[\nu_i^2] + E_s[H^{(1)}(t_i, \theta, t_0, \nu_0)^2] - 2E_d[\nu_i]E_s[H^{(1)}(t_i, \theta, t_0, \nu_0)]. \end{aligned} \tag{2.21}$$

In the first step the linearity of the expectation value is used. In the second the fact that the measurement data ν_i is constant with respect to the expectation E_s and that $H(t_i, \theta, t_0, \nu_0)$ is constant with respect to the expectation E_d. The term $\frac{1}{m}\sum_{j=1}^{m}$ cancels out as the expectation over the $H^{(j)}$ is independent of j. Now the expectation values E_s and E_d only differ in the parameter θ in the probability distribution, namely the expectation value E_d is with respect to the true value $\theta^{(0)}$ while E_s is with respect to the parameter θ. Hence

$$F^E_{s,2}(\theta^{(0)}, \theta) \;=\; \sum_{i=1}^{n} \left(E_{\theta^{(0)}}[\nu_i^2|\nu_0] + E_\theta[H(t_i, \theta, t_0, \nu_0)^2|\nu_0] - 2E_{\theta^{(0)}}[\nu_i|\nu_0]E_\theta[H(t_i, \theta, t_0, \nu_0)|\nu_0] \right).$$

The value $F^E_{s,2}(\theta)$ is the expectation of the fitness function $F^{(m)}_{s,2}$ for a parameter θ when the true parameter is $\theta^{(0)}$ and the expectation value is calculated over the set of possible data and the set of possible simulations.
For plotting the fitness landscapes for numerical reasons the following approximations for equation (2.19) and equation (2.20) are used:

$$E_\theta[X_t|\nu_0] \;\approx\; \sum_{k=0}^{\tilde{N}} p_\theta(k, t|\nu_0, t_0)k \text{ and} \tag{2.22}$$

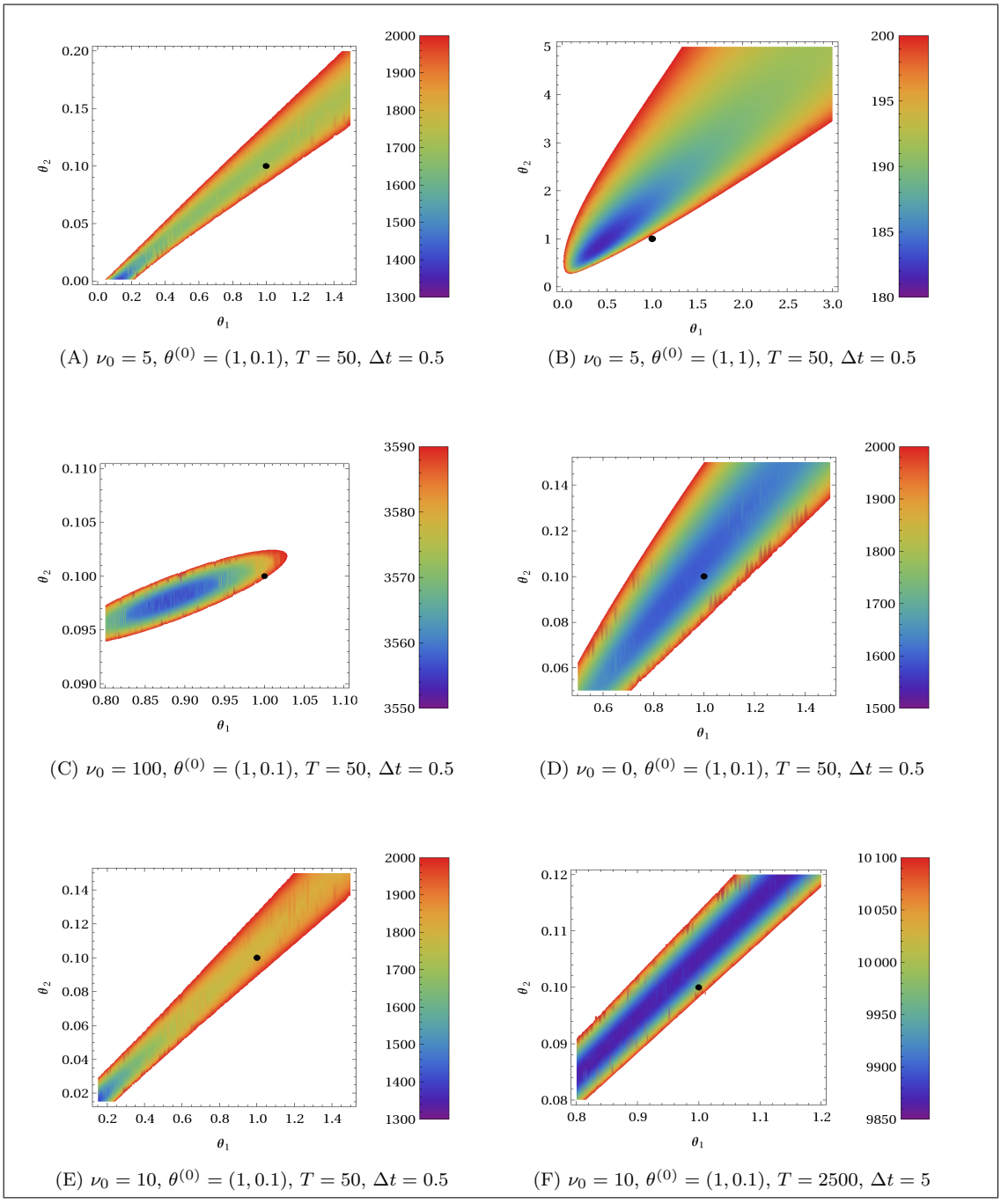
$$E_\theta[X_t^2|\nu_0] \;\approx\; \sum_{k=0}^{\tilde{N}} p_\theta(k, t|\nu_0, t_0)k^2 \tag{2.23}$$

with $\tilde{N}$ such that $\sum_{k=0}^{\tilde{N}} p_\theta(k, t|\nu_0, t_0) \geq 1 - 10^{-6}$. The reason is that the probability for jumps to very high k is very small. See figure 2.1 for the graphics. The graphics in figure 2.1-2.4, and 2.7 are plotted with the *ColorbarPlot* package, which is available in the Wolfram library archive of Mathematica [38]. The graphics show the landscape of the function encoded in a colored contour plot. The true parameter is marked with a black dot. The imprecise transitions in color in figures 2.1 (D), (F) or 2.2 (A) are graphical inaccuracies by the plot routine.
In none of the graphics of figure 2.1 the true parameter is in the area of the lowest function values. Therefore the functional $F^E_{s,2}$ is not unbiased. For further remarks see section 2.3.3.

Choice $\sigma_i = Var(H(\theta, t_i))$:

To calculate the expectation of the functional (2.4) keep in mind that only $E_d[\cdot]$ depends on ν:

$$\begin{aligned}
F^E_{\tilde{s},2}(\theta) \;&=\; E_s\left[E_d\left[\sum_{i=1}^{n} \frac{\frac{1}{m}\sum_{j=1}^{m}\left(\nu_i - H^{(j)}\left(t_i, \theta, \nu_{i-1}, t_{i-1}\right)\right)^2}{\widehat{\sigma_i}} \right] \right] \\
&=\; \sum_{i=1}^{n} \frac{1}{m} \sum_{j=1}^{m} E_s\left[\frac{E_d\left[\left(\nu_i - H^{(j)}\left(t_i, \theta, \nu_{i-1}, t_{i-1}\right)\right)^2\right]}{\widehat{\sigma_i}} \right] \\
&\approx\; \sum_{i=1}^{n} \frac{E_s\left[E_d\left[\left(\nu_i - H^{(j)}(t_i, \theta, t_{i-1}, \nu_{i-1})\right)^2\right]\right]}{E_s\left[H^{(1)}(t_i, \theta, t_{i-1}, \nu_{i-1})^2\right] \;-\; E_s\left[H^{(1)}(t_i, \theta, t_{i-1}, \nu_{i-1})\right]^2} \\
&=\; \sum_{i=1}^{n} \frac{E_{\theta^{(0)}}[\nu_i^2|\nu_0] + E_\theta[H(t_i, \theta, t_0, \nu_0)^2|\nu_0] - 2E_{\theta^{(0)}}[\nu_i|\nu_0]E_\theta[H(t_i, \theta, t_0, \nu_0)|\nu_0]}{E_\theta[H(t_i, \theta, t_0, \nu_0)^2|\nu_0] - E_\theta[H(t_i, \theta, t_0, \nu_0)|\nu_0]^2}.
\end{aligned} \tag{2.24}$$



(A) $\nu_0 = 5$, $\theta^{(0)} = (1, 0.1)$, $T = 50$, $\Delta t = 0.5$

(B) $\nu_0 = 5$, $\theta^{(0)} = (1, 1)$, $T = 50$, $\Delta t = 0.5$

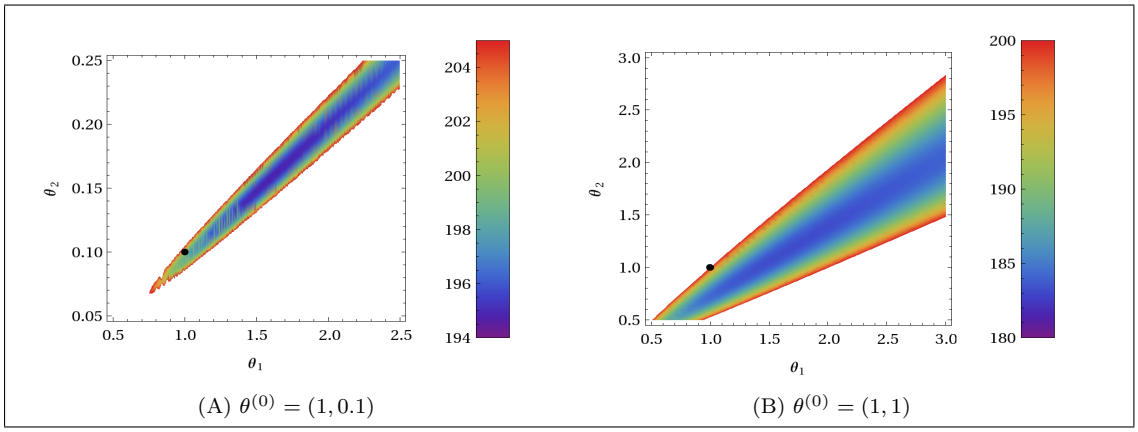
(C) $\nu_0 = 100$, $\theta^{(0)} = (1, 0.1)$, $T = 50$, $\Delta t = 0.5$

(D) $\nu_0 = 0$, $\theta^{(0)} = (1, 0.1)$, $T = 50$, $\Delta t = 0.5$

(E) $\nu_0 = 10$, $\theta^{(0)} = (1, 0.1)$, $T = 50$, $\Delta t = 0.5$

(F) $\nu_0 = 10$, $\theta^{(0)} = (1, 0.1)$, $T = 2500$, $\Delta t = 5$

Figure 2.1: **Expectation of the stochastic single shooting function.** $F^E_{s,2}$ from equation (2.21) of the stochastic single shooting function $F^{(m)}_{s,2}$, equation (2.2), with different initial values, parameters and designs. White stands for higher function values than assigned in the color bar.



(A) $\theta^{(0)} = (1, 0.1)$ (B) $\theta^{(0)} = (1, 1)$

Figure 2.2: **Expectation of the stochastic single shooting functional with estimated variances.** Expectation $F^E_{\tilde{s},2}$, (2.24) of $F^{(m)}_{\tilde{s},2}$, (2.4) with $\Delta t = 0.5$, $T = 50$, $\nu_0 = 5$. White stands for higher function values than assigned in the color bar.

For the approximation one uses the fact that the correlation between $(\nu_i - H)^2$ and $\widehat{\sigma_i}$ is small as only the i^{th} component is the dependent part. Furthermore it holds approximately $E_s[\widehat{\sigma_i^2}] \approx E_\theta\left[\left(H^{(1)}\right)^2\right] - E_\theta\left[H^{(1)}\right]^2$ as the expectation of the variance converges with $m \to \infty$ to the variance. Last the relations of equation (2.19) and equation (2.20) are inserted. For the results see figure 2.2. The landscape of the function $F^E_{\tilde{s},2}$ is encoded in a colored contour plot. The true parameter marked with a black dot is in none of the graphics in the area of the lowest function values, which means that the functional is not unbiased, see also section 2.3.3.

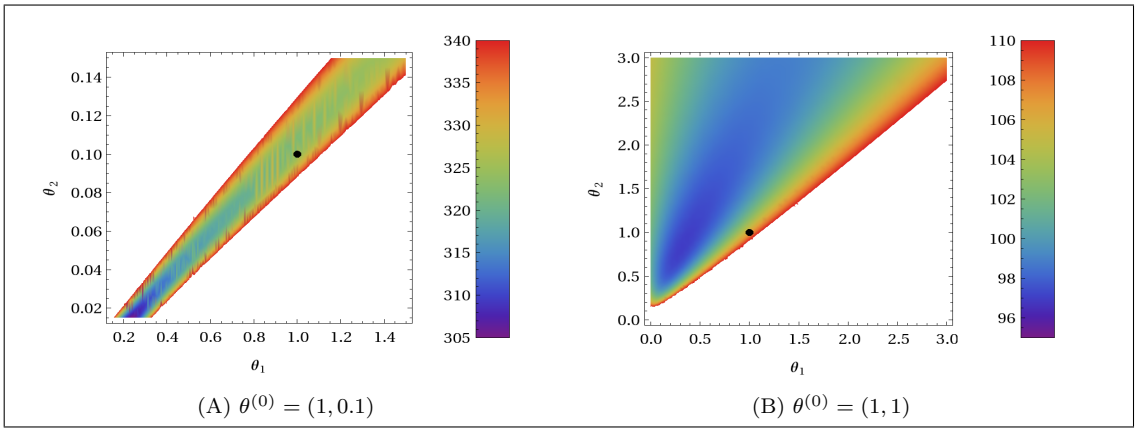
Choice L^1-norm:
Again as for the other functionals the expectation values of the fitness can be calculated and plotted using the relation:

$$F^E_{s,1}(\theta) \;=\; \sum_{i=1}^{m}\sum_{j=0}^{\infty}\sum_{k=0}^{\infty}\left(|k-j|p(k,t_i|\nu_0,t_0)p(j,t_i|\nu_0,t_0)\right).$$

Similar approximations as in equation (2.22) are used yielding the figure 2.3, which shows the landscape of $F^E_{\tilde{s},2}$ is encoded in a colored contour plot. As the true parameter marked with a black dot is not in the area of the lowest function values the functional is not unbiased.

2.3.3 Discussion of the performance of the stochastic single shooting functionals

The previous subsection was investigating the performance of the stochastic single shooting functional. To address this goal the expectation of the objective function value landscape was calculated. It should be noted that the formal statement of unbiasedness is $E[\text{argmin}_\theta\ \hat{\theta}] = \theta^{(0)}$ (see equation (2.5) for the estimator), which is slightly different to what was tested in the previous subsection. Nevertheless it can be stated that the landscapes of the objective function have a structural bias as this was the case for all tested parameters, initial values and designs. Consequently the stochastic single shooting functional should not be used for parameter estimation in stochastic models.



(A) $\theta^{(0)} = (1, 0.1)$ (B) $\theta^{(0)} = (1, 1)$

Figure 2.3: **Expectation of the L^1-based function.** Expectation $F^E_{s,1}$ of the objective function $F^{(m)}_{s,1}$, (2.2), with $\nu_0 = 5$, $\Delta t = 0.5$ and $T = 50$. White stands for higher function values than assigned in the color bar.

This leads to the question: Why does the least squares functional appear so poor in stochastic modeling while it is so successfully applied in deterministic modeling?
The least squares functional is constructed in deterministic modeling in order to get a maximum likelihood estimator under the assumption of a Gaussian error distribution. This assumption is crucial. The least squares functional is optimal if the underlying distribution is Gaussian. Otherwise the maximum likelihood property cannot be stated. In stochastic models for biological systems the distributions at a time point t is unknown due to the nonlinearity of the model and not necessarily Gaussian. This explains the poor behavior of the canonical least squares approach. There is a second explanation. Remember the decomposition of the variance of a random variable Y in statistics:

$$Var(Y) \;=\; E[Y^2] - E[Y]^2.$$

Hence $E[Y^2] \;=\; Var(Y) + E[Y]^2$. Now let Y be: $Y = \sum_{i=1}^{n} \frac{(\nu_i - h(t_i,\theta,\nu_{i-1},t_{i-1}))^2}{\widehat{\sigma_i}^2}$. In deterministic modeling $Var(Y)$ does not depend on the parameter value θ, consequently it holds: $\min_\theta \; E[Y^2] = min_\theta \; E[Y]^2$, which means that a trajectory is fitted in order to obtain minimal squared differences to the measurements. Now in stochastic modeling with $Y = \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{(\nu_i - H^{(j)}(t_i,\theta,\nu_{i-1},t_{i-1}))^2}{\widehat{\sigma_i}^2}$ the variance $Var(Y)$ depends on θ. Moving in the parameter space Θ there are directions in which the descent of $Var(Y)$ is greater than the ascent of $E[Y]^2$ resulting in a lower value of $E[Y^2]$, which causes the bias:

$\underline{\sigma_i^2 \text{ constant}}$:

- This is for example the case if both parameters become smaller (see figure 2.1(A)), which means that the system is less reactive (see for an illustration figure 1.1). Therefore the variance in the stochastic least squares term is smaller. As this effect is stronger than the increase of the $E[Y]^2$ term the function value of $F^E_{s,2}$ is smaller in this region than at the position of the true parameter.

- For higher θ_2 a similar effect can be seen. As the system then tends to lower particle numbers (the deterministic steady state is smaller) the variance is as well smaller, which again leads

to smaller function values (see figure 2.1(B)).

- Figures 2.1(E,F) with starting value $\nu_0 = 10$ in the deterministic steady state also show a bias, which reduces with increasing $\Delta t = t_i - t_{i-1}$ as the distribution then is approximately Poisson (see remark 2.3) and the least squares function is therefore adequate.

L^1-norm and $\hat{\sigma}_i$

- $\hat{\sigma}_i$ is estimated correctly for the true parameter. Higher parameter values mean higher reactivity in the system resulting in higher estimated variances ${\hat{\sigma}_i}^2$. Dividing by this too high ${\hat{\sigma}_i}^2$ compared to the true parameter leads to a low objective function value for these parameters. This causes the bias (see figure 2.2).

- Considering the L^1-norm the bias is still present although the L^1-norm is in general more robust with respect to outliers.

Whether it is possible to correct the bias is up to future research. Due to the results seen here even in this simple example model the estimator in this form will not be considered in the application chapter.

2.3.4 Expectation of the LS function

Remarks on the maximum likelihood properties of the estimator:

First of all equation (2.12) is an estimator which maximizes a likelihood function. Maximum likelihood estimators are often used because of their asymptotic properties.

- But as $\nu_i|\nu_{i-1}$ and $\nu_j|\nu_{j-1}$ are not necessarily identically distributed for $i \neq j$ one of the requirements for the asymptotic properties of a maximum likelihood estimator does not hold.

- Nevertheless for a "long" trajectory the state $\nu = x$ for a fixed x will "often" be reached. Thus the estimation can be done with the distribution information of the transitions from that single state x. The estimator has the asymptotic properties of a maximum likelihood estimator.

It will be necessary to test the estimator for a finite number of observations. One option would be to do simulation runs with an optimization algorithm for simulated data with a true parameter which is known. This will be done in section 4.1.1. This includes influence from three sources of stochasticity: optimization algorithm Particle swarm, evaluation of the objective function with the use of simulations and the stochastic data. To analyze, which effects come from the properties of the method and which are just stochastic influences, the expectation of the LS function landscapes is plotted using exact transition probabilities for the Immigration-Death model, equation (2.13). For the results see figure 2.4. Then the LS function landscapes are plotted using simulations to evaluate the transition probabilities, which allows for observing the influence of this source of stochasticity, see figure 2.5 and 2.6.
It should be noted that the formal statement of unbiasedness, $E[\hat{\theta}] = \theta^{(0)}$, is slightly different to comparing the minimal point of the expectation of the LS function landscape as the expectation value and argmin are not interchangeable with equality. Nevertheless the expectation of the LS function landscape gives a good evaluation of the performance of a functional.
Set $F_L^E(\theta) = E_d E_s[F_L^{(m)}(\nu, \theta)]$ as the expectation over data $E_d[\cdot]$ and simulations $E_s[\cdot]$ of the LS

function $F_L^{(m)}$:

$$\begin{aligned} E_s[F_L^{(m)}(\nu,\theta)] &= \sum_{i=1}^{n} E_s\left[\log\left(\widehat{p_\theta}^{(m)}(\nu_i,t_i|\nu_{i-1},t_{i-1})\right)\right] \\ &\approx \sum_{i=1}^{n} \log\left(p_\theta(\nu_i,t_i|\nu_{i-1},t_{i-1})\right). \end{aligned}$$

Note that the expectation and the logarithm are not interchangeable with equality. Nevertheless for large m the estimation $\hat{p}$ will be close to p, which allows the approximation. Now the expectation value over the data will be calculated. Let $p^{\nu}_{\theta^{(0)}}(\nu_1,\ldots,\nu_n)$ be the probability density function which gives the probability to get a realization $(\nu_1,\ldots,\nu_n)$. Set for abbreviation: $p_\theta(\nu_i,t_i|\nu_{i-1},t_{i-1}) = p_\theta(i|i-1)$:

$$\begin{aligned} F_L^E(\theta) &= E_d\left[\sum_{i=1}^{n}\log\left(p_\theta(i|i-1)\right)\right] \\ &= \sum_{\nu_1,\ldots,\nu_n}\left(\sum_{i=1}^{n}\log\left(p_\theta(i|i-1)\right)\right)p^{\nu}_{\theta^{(0)}}(\nu_1,\ldots,\nu_n) \\ &= \sum_{\nu_1}\cdots\sum_{\nu_n}\left(\sum_{i=1}^{n}\log\left(p_\theta(i|i-1)\right)\right)p_{\theta^{(0)}}(n|n-1)\ldots p_{\theta^{(0)}}(1|0) \\ &= \sum_{i=1}^{n}\left(\sum_{\nu_1}\cdots\sum_{\nu_n}\log\left(p_\theta(i|i-1)\right)p_{\theta^{(0)}}(n|n-1)\ldots p_{\theta^{(0)}}(1|0)\right) \\ &= \sum_{i=1}^{n}\sum_{\nu_{i-1}}\sum_{\nu_i}\log\left(p_\theta(i|i-1)\right)p_{\theta^{(0)}}(i|i-1) \\ &\quad \sum_{\nu_1}\cdots\sum_{\nu_{i-2}}\sum_{\nu_{i+1}}\cdots\sum_{\nu_n} p_{\theta^{(0)}}(n|n-1)\ldots p_{\theta^{(0)}}(i+1|i)p_{\theta^{(0)}}(i-1|i-2)\ldots p_{\theta^{(0)}}(1|0) \\ &= \sum_{i=1}^{n}\sum_{\nu_{i-1}}\sum_{\nu_i}\log\left(p_\theta(i|i-1)\right)p_{\theta^{(0)}}(i-1|i)p_{\theta^{(0)}}(i-1|0). \end{aligned}$$

In the second equality the Markov property is used, the third and fourth change the summation in a suitable order and the fifth uses the following relations:

$$\begin{aligned} \sum_{\nu_j} p_{\theta^{(0)}}(j|j-1) &= 1 \text{ for } j = i+1,\ldots,n, \\ \sum_{\nu_l} p_{\theta^{(0)}}(j+1|j)\; p_{\theta^{(0)}}(j|j-1) &= p_{\theta^{(0)}}(j+1|j-1) \text{ for } j = 1,\ldots,i-2. \end{aligned}$$

For computational reasons the lower bound for the summation over ν_i is zero and the upper is chosen such that jumps from ν_{i-1} to ν_i are taken into account if they have a probability larger than 10^{-6}. The landscapes of the LS function values $F_{E,L}(\theta)$ are plotted in figure 2.4. These graphics show the value of $F_{E,L}(\theta)$ encoded in a colored contour plot. The black dot marks the true parameter.
The optimum is at the position of the true parameter in all settings, which leads to the conclusion that the functional F_L is unbiased. For examples using simulations to evaluate the transition probabilities see the following subsection. For a simulation study of the estimator estimating the parameters of 25 different time series of the Immigration-Death model, see section 4.1.

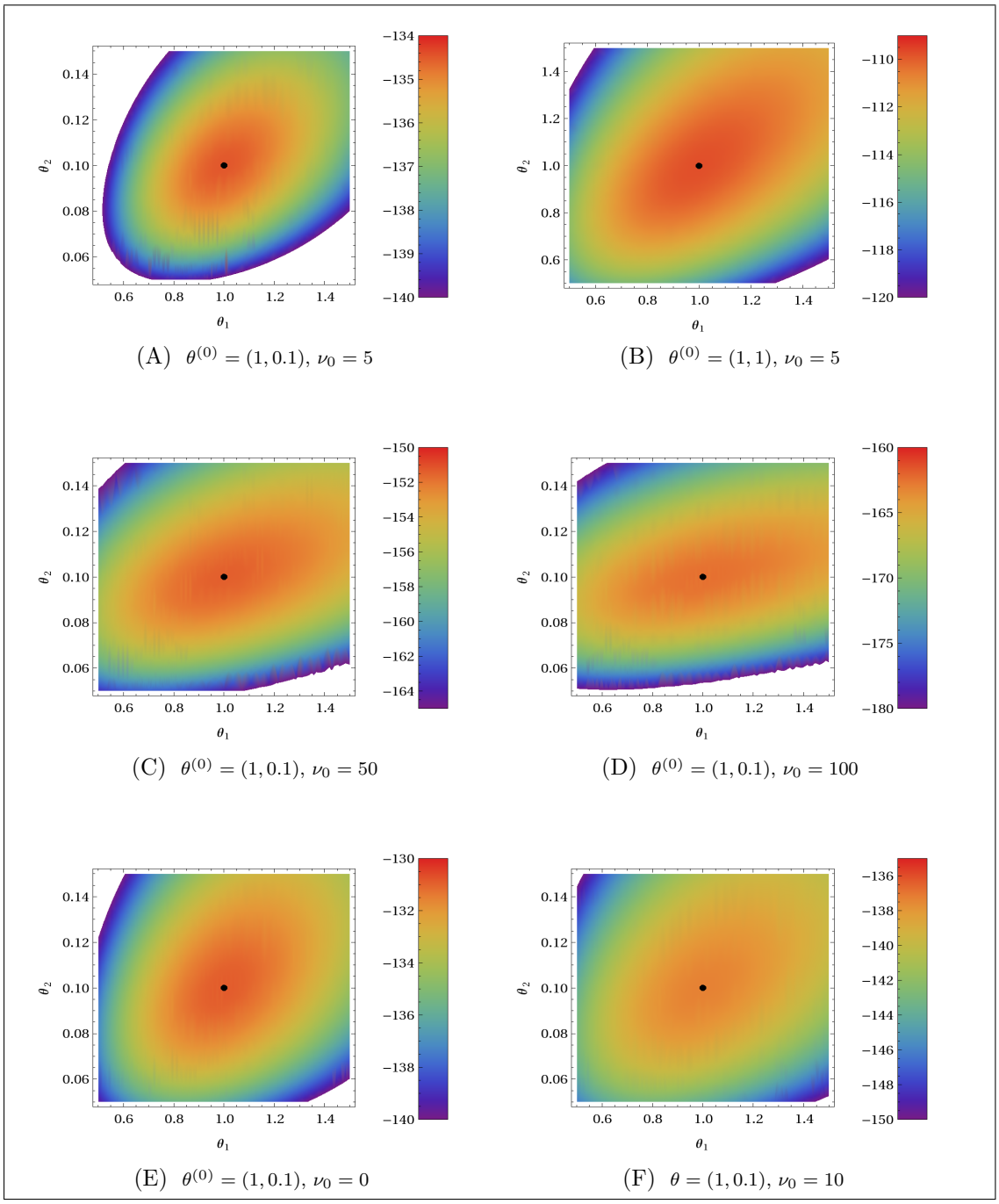


Figure 2.4: **Expectation of LS function.** Expectation value F_L^E for the LS function $F_L^{(m)}$ from equation (2.9). All with $\Delta t = t_i - t_{i-1} = 0.5$ and $T = 50$. True parameters are marked with points. White stands for higher function values than assigned in the color bar. As the likelihood function is negative the optimization problem is here a maximization problem.

2.3.5 The MLS function using simulations

The analytical investigation is only possible in very simple example systems like the Immigration-Death system used above. In more complex systems parameter estimation can only be done with simulations. Therefore simulation methods will be tested in the setting of the simple example system and compared to the theoretical results to see how well the approximation with simulations works. First for a given measurement $\nu_1, \ldots, \nu_m$ the value of the MLS function for a parameter θ is calculated using equation (2.11). In figures 2.5 and 2.6 evaluations of the MLS function are represented as colored dots for different parameter settings and different stochastic data sets.

Observations:

- The first observation is that the landscape is not smooth as in the plots with the expectation value but it has discontinuities. This is due to the stochasticity in the simulations. Even with $m = 1000$ simulations this leads to a remarkable variability in the objective function values even for close parameter values, see, for example, the green dots in the yellow (2.5 (A,E)) and even in the purple area, figure 2.5 (D).
- The coarse structure of the landscape is recognizable: there is a region of small function values in the proximity of the true parameter. The landscape can be considered as a noisy version of the landscapes in figure 2.4.
- Due to the stochasticity in the evaluation of the transition probabilities the meaning of minimum is not clearly defined as another realization of the objective function landscape can result in a different minimal point, see also remarks in section 2.4.1 and the evaluation of a stopping criterion suited to the stochasticity in section 2.4.2.
- The ground of the valley is sometimes at lower parameter values than the true parameter as in figure 2.5(A) or at higher parameter values as in figure 2.5(C), which is due to the stochasticity in the data. This effect can also be seen in deterministic modeling, where the effect comes from the measurement noise. Remember that an estimator is a random variable itself.
- This leads to the question of the distribution of the estimator with respect to the stochasticity in the data and the simulations. This is an open point suggested for further investigation.
- Figure 2.7 shows how strong the effect of the approximation in $F_g^{(m)}$ is. Around the true parameter one can observe that the approximation is identical to the exact value $F_L^{(m)}$. The approximation is mostly used "far away" from the true parameter.

For a conclusion it can be stated that the simulation results go along with the theoretical analysis stating that the MLS objective function is unbiased. This means that its expectation value over data and simulations has its minimum at the true parameter. This is an important property for the applicability of the MLS function.

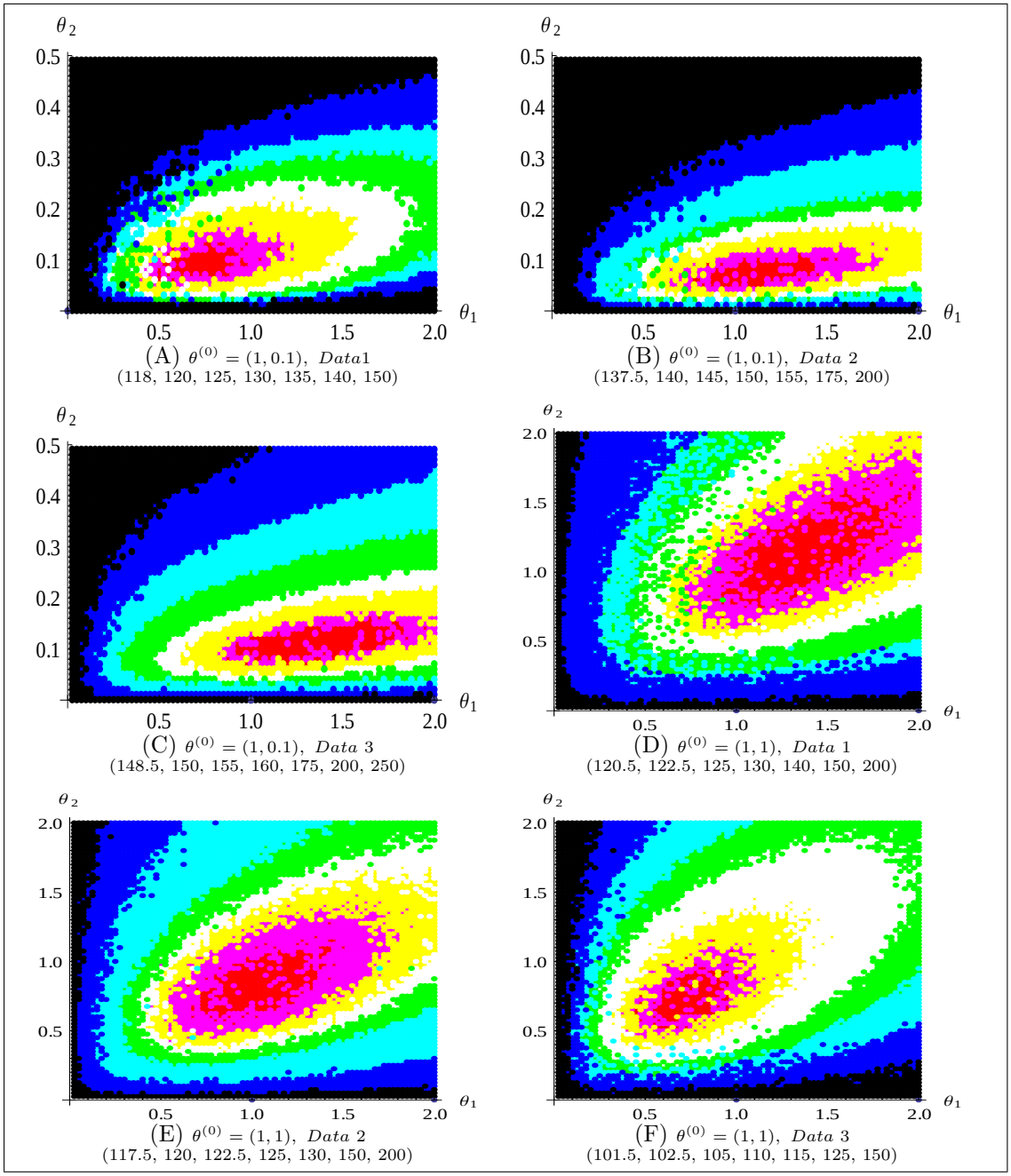


Figure 2.5: **Landscape of the MLS function with simulations I.** Absolute value of $F_g^{(1000)}$ from equation (2.11) for two different parameters and for each parameter three different data sets simulated with the Gillespie algorithm implemented in COPASI [28] with $\nu_0 = 5$, $\Delta t = 0.5$ and $n = 100$. Numbers in brackets are for color assignment: smaller values than first number colored in red, greater or equal than first and smaller than second colored in purple, ... and values greater than last number colored in black. Due to the absolute value the problem is again a minimization problem.

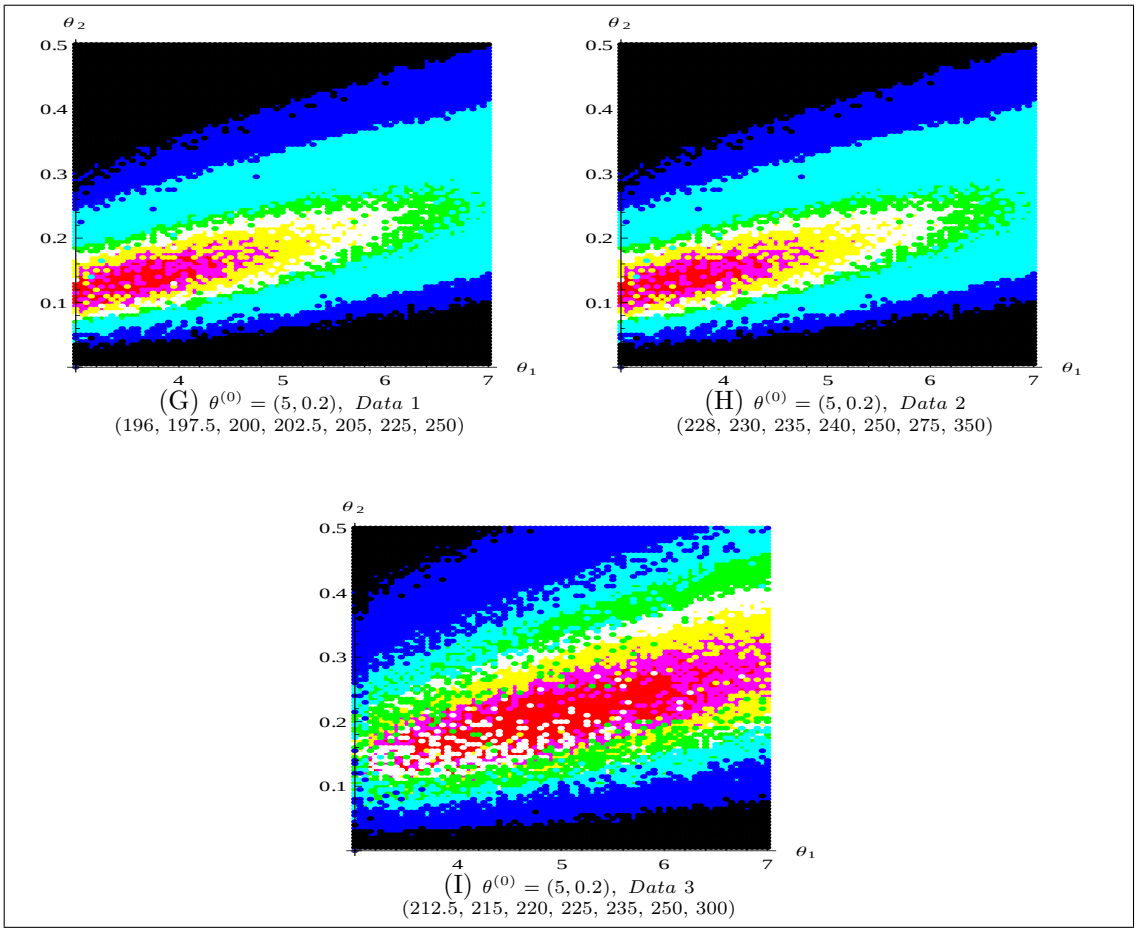


Figure 2.6: **Landscape of the MLS function with simulations II.** Absolute value of $F_g^{(1000)}$ from equation (2.11) for a third parameter with three different data sets simulated with the Gillespie algorithm implemented in COPASI [28] with $\nu_0 = 5$, $\Delta t = 0.5$ and $n = 100$. Coloring as in figure 2.5.

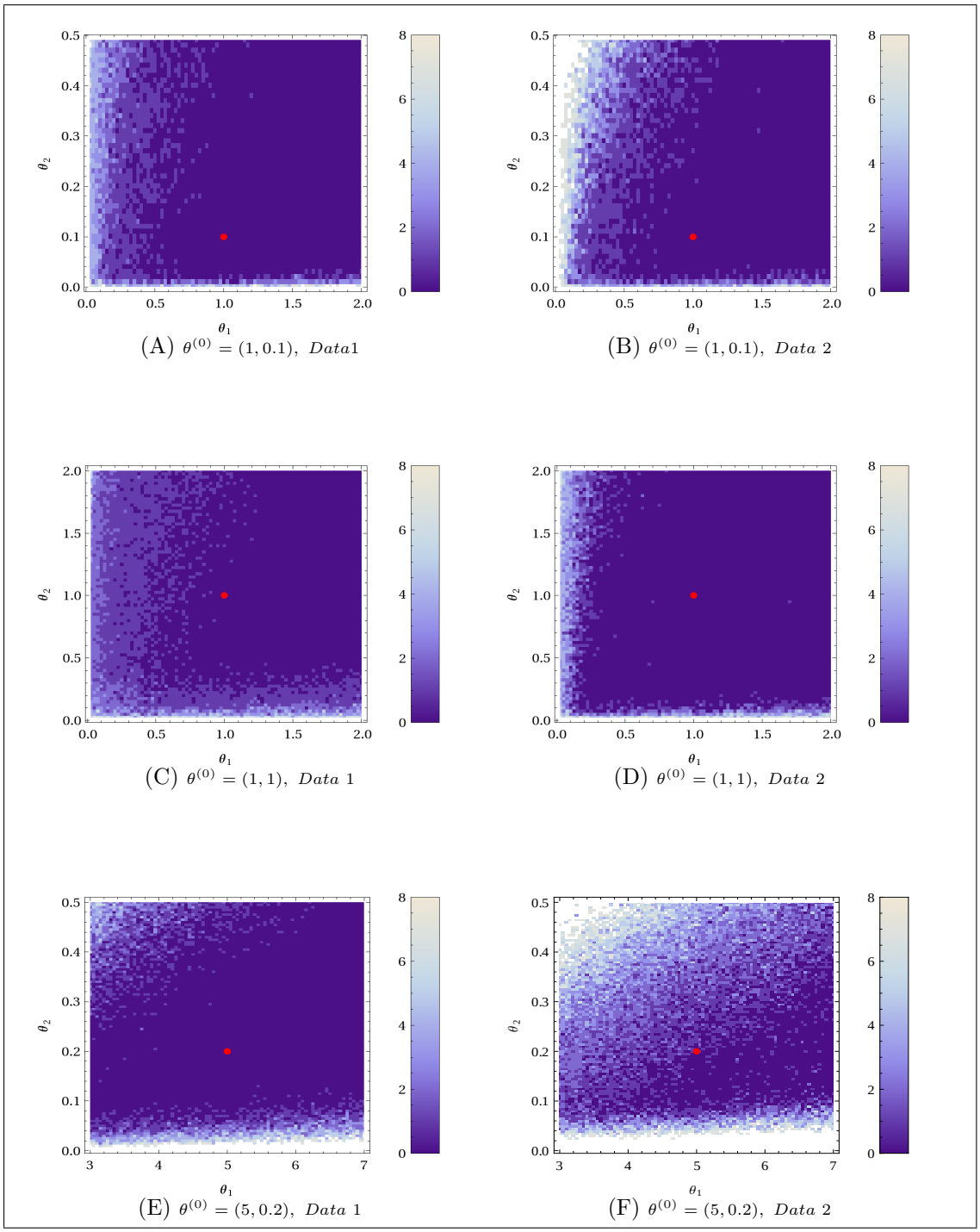


Figure 2.7: **Impact of the approximation MLS.** Number of transitions from $i-1$ to i where the approximation g, equation (2.10), plays a role in $F_g^{(1000)}$: $\nu_0 = 5$, $\Delta t = 0.5$, $n = 100$, $m = 1000$. Dark blue: zero, brighter colors: higher numbers.

2.4 Optimization of the stochastic objective function

2.4.1 Statement of the stochastic problem

To estimate the parameter the LS function $F_L^{(m)}$ in equation (2.9) or the MLS function $F_g^{(m)}$ in equation (2.11) has to be optimized:

$$\hat{\theta} \;=\; \underset{\theta}{\operatorname{argmin}}\; F_k^{(m)}(\nu,\theta), \quad k \in \{g, L\}. \tag{2.25}$$

Before considering algorithms to find a solution for this problem some remarks on its properties are given:

- To evaluate $F_k^{(m)}$, $k \in \{L, g\}$, Gillespie simulations are used:
  - The objective function is a random variable.
  - As the objective function is a random variable it is not possible to calculate derivatives and for the optimization it can be considered as a "black box". Black box means that it calculates a function value for a given input but the details of the calculation are not considered for the optimization.
- For increasing m the random variable $F_L^{(m)}$ will converge to a limit F_L.
- What does optimum mean for finite m?

From definition a minimum x_{min} of a function f is a point for which holds $f(x) \geq f(x_{min})$ for all x in a neighborhood $B_\delta(x_{min})$ of x_{min} with $B_\delta(x_{min}) = \{x|\ \|x - x_{min}\|_2 < \delta\}$ and $\delta > 0$. Directly used for stochastic problems this formulation does not make sense as for two random variables X, Y as the meaning of $X > Y$ is not well-defined and consequently an optimum is not well-defined. Hence one should not expect an exact single point as the result of the estimation problem for finite m but rather a range of points with low function values.
There is a huge amount of literature on optimization of stochastic quantities, for example [39] and also on optimization algorithms which are able to handle the so-called black box property, for example [40, 41]. Here a Particle Swarm optimization will be tested (section 2.4.2). Another approach is a transformation to a deterministic optimization problem (section 2.4.3). For this work every evaluation of F_k, $k \in \{L, g\}$, is computed with a new set of random numbers to reduce the influence of sets of random numbers with low probabilities. It is also possible to fix the set of random numbers and do several estimations for different so-called random seeds [4].

2.4.2 Particle Swarm

The likelihood function is calculated with simulations so the objective function can be considered as a black box.
One approach for this kind of functions is to use non-gradient-based optimization procedures. The algorithm Particle Swarm for the results shown later is from [42] and the COPASI implementation: At first a set of points - which will be named particles - will be randomly distributed on the range considered for optimization. For ranges containing one order of magnitude or less the particles will be uniformly distributed on the range. For ranges containing several orders of magnitude the probability for small values will be increased as described in figure 2.8.

- **Input:** number of iterations n_{it}, number of particles n_{pa}, number of simulations m, data ν, range of optimization: $\left[p_1^{lo}, p_1^{up}\right] \times \ldots \times \left[p_{dim}^{lo}, p_{dim}^{up}\right]$ with $p_{ij} \geq 0$ for $i = 1, \ldots, dim$ and $j \in \{lo, up\}$, objective function $F(\cdot)$.
- **Initialization**
  - Distribute initial swarm: Set $p_i^{lo} = \text{If}\left(p_i^{lo} > 0 \ , \ p_i^{lo} \ , \ 10^{-10} \right)$
    % or any small value greater zero
    $particles = \text{Table}[$
    $$\text{If}\left(\log_{10} p_i^{up} - \log_{10} p_i^{lo} < 1.8, \quad U([p_i^{lo}, p_i^{up}]), \quad p_i^{lo} \left(\frac{p_i^{up}}{p_i^{lo}}\right)^{U([0,1])} \right), \quad (i, 1, n_{pa})]$$
    $speed = Table\,[0, (i, 1, n_{pa})]$
  - function value: $fwert = \text{Table}\,[F(particle(i)),\ (i, 1, n_{pa})]$
  - best value: $gbestvalue = \min\,(fwert)$
  - best position: $gbest = (p_i | i = \text{argmin}_i(fwert(i)))$
  - best personal value: $pbestvalue = fwert$
  - best personal position: $pbest = particles$
  - constants: $w = \frac{1}{2 \log_{10} 2}, \ \ c = \frac{1}{2} + \log_{10} 2$
- **Iteration:** For $(i, 1, n_{it})$ (
  - $speed = \text{Table}[$
    $w\ speed(i)$
    $+\ c\ U\,([0,1])\ \ (pbest(i) - particle(i))$
    $+\ c\ U\,([0,1])\ \ (gbest - particle(i)),$
    $(i, 1, n_{pa})\ ]$
  - $particle = \text{Table}[$
    $\text{If}($
    $particle(i) + speed(i) \in [p_i^{lo}, p_i^{up}],$
    $particle(i) + speed(i),$
    $particle(i)$
    $),(i, 1, n_{pa})]$
  - $fwert = \text{Table}[\ F(particle(i)),\ (i, 1, n_{pa})]$
  - $gbestvalue = \min(fwert)$
  - $gbest = (p_i | i = \text{argmin}_i(fwert(i)))$
  - $pbestvalue = \text{Table}[\text{If}(pbestvalue(i) > fwert(i), fwert(i), pbestvalue(i)),\ (i, 1, n_{pa})]$
  - $pbest = \text{Table}[\ \text{If}(pbestvalue(i) > fwert(i), particle(i), pbest(i)),\ (i, 1, n_{pa})\ ]$
    $)$
- **Output:** $(gbest, gbestvalue)$

Figure 2.8: **Pseudo code for Particle Swarm.** $U([a, b])$ stands for a random variable uniformly distributed on the interval $[a, b]$.

The function will be evaluated for each of these particles. The best particle with its function value will be stored in *gbest* and *gbestvalue*. In *pbest* the best "personal" position a particle has reached during the iteration procedure will be stored with its function value. In each iteration the particle will move towards the *gbest* and *pbest* positions with a certain speed. The speed is updated as follows: The difference between *pbest* and the particles position is calculated and multiplied by a random number distributed uniformly on $[0, 1]$. The difference between *gbest* and the particles position is multiplied by another random number also distributed uniformly on $[0, 1]$. The sum of both values is then added to the old speed value. This update is then used for the update of the particles position. It is simply added to the old particles position and accepted if this value is inside the range. If not the old position is kept. Then the function will be evaluated with the new particle positions and *gbest* and *pbest* will be updated. After the last iteration the position *gbest* is considered as optimum.
This is a heuristic algorithm which performs successfully in many situations in practice but it has to be stated that there is no guarantee of reaching the global or even a local optimum.

- **Input:** number of iterations n_{it}, number of particles n_{pa}, number of simulations m, data ν, range of optimization: $[p_1^{lo}, p_1^{up}] \times \ldots \times [p_{dim}^{lo}, p_{dim}^{up}]$ with $p_{ij} \leq 0$ for $i = 1, \ldots, dim$ and $j \in \{lo, up\}$, objective function $F(\cdot)$, iteration number for termination criterion check n_{tc}.
- **Initialization**
- **Iteration:** For $(i, 1, n_{it})$ (
  - *speed*
  - *particle*
  - $fwert$
  - *gbestvalue*
  - *gbest*
  - *pbestvalue*
  - *pbest*
  - If(
    Divisible (i, n_{tc}),
    (
    tc = StdDev (*gbestvalue*, Table(F(*gbest*), $(i, 1, n_{pa})$)),
    If($tc > 0.5$ StdDev ($fwert$) , Break)
)
)
)
- **Output:** $(gbest, gbestvalue)$

Figure 2.9: **Pseudo code for modified Particle Swarm.** The Pseudo code is identical to the original Particle Swarm in figure 2.8 except the addition of the termination criterion conditions and the termination criterion. StdDev stands for Standard Deviation.

The algorithm is presented here without a stopping criterion although there is one in the COPASI implementation terminating the algorithm when the variance of the function values of the swarm as well as the variance of the swarm positions becomes small. For stochastic modeling it is difficult to use this criterion directly as it is not clear what can be considered to be small in comparison to the stochasticity in the landscape. In the following a modification will be described to improve this situation.
In stochastic problems it might happen that the algorithm quickly finds a range of low function values where the noise or stochasticity is much larger than the differences of the underlying landscape, see figure 2.5 and 2.6. The algorithm will continue to pick new points and to calculate more function evaluations. This might be very time-consuming, especially if a function evaluation is computationally intensive. Thus a stopping criterion should be introduced. Some stopping criteria are introduced in [43, 44]. Here a different stopping criterion is suggested: it lets the algorithm stop as soon as the stochasticity becomes larger than the differences in the underlying landscape. Evaluate every n_{tc}-th iteration the function in *gbest* more than once. Then calculate a variance of these evaluations. Compare this variance to the variance in $fwert$. If the last one is larger, this suggests that there is still a chance to reduce the objective function value. If both are about the same size, it suggests that the fluctuations can be explained by the stochasticity and hence the algorithm should terminate. A comparison of the different criteria is out of the scope of this thesis.

The modification will be tested in section 4.1.1 for an Immigration-Death model. This stopping criterion should be also useful for other global optimization algorithms but a deeper investigation of this question is suggested for further research.
Another option would be the use of a fixed random seed for optimization and a repetition of the optimization procedure.

2.4.3 Transformation to a deterministic landscape: polynomial chaos expansion

The aim of using the polynomial chaos expansion is a transformation of the stochastic landscape into a deterministic landscape. To this aim the stochastic function is evaluated at several points. These realizations of the stochastic function are then used to construct a deterministic approximating polynomial, which then can be easily optimized.
The term polynomial chaos was used already very early by Wiener [45]. The concept of the polynomial chaos expansion was applied to optimization in the context of parameter uncertainties, for example by [17]. Here it will be used in a slightly different context. But before explaining these differences in detail, the basic idea is explained:
Let P_i be a basis of orthogonal polynomials on $L^2(\Lambda)$ with respect to a weighting function ψ on an at first one-dimensional interval Λ. Then a function F on $L^2(\Lambda)$ can be expanded in a series of orthogonal polynomials P_i:

$$F(\theta) = \sum_{j=0}^{\infty} a_j P_j(\theta)$$

with

$$a_j = \int_{\Lambda} F(\theta)\ P_j(\theta)\ \psi(\theta)\ d\theta\ \approx\ \sum_{\gamma\in\Gamma} F(\gamma)\ P_j(\gamma)\ \psi(\gamma)$$

with a suitable integration grid Γ. In practice it is often enough to consider $F(\theta) = \sum_{j=0}^{N_{PC}} a_j P_j(\theta)$ with small $N_{PC} < 10$. It is important to note that for each a_j, $j = 1, \ldots, N_{PC}$, the same grid Γ can be used. Hence the number of function evaluations does not directly depend on N_{PC}.

Example 2.2 (Orthogonal polynomials). The Legendre polynomials on $L^2([-1,1])$ with the weighting function $\psi(\theta) = \frac{1}{2}1_{([-1,1])}(\theta)$ are an example of a complete orthogonal system in one dimension [46]. They satisfy the following Legendre ODE

$$(1-\theta^2)F''(\theta) - 2\theta F'(\theta) + j(j+1)F(\theta) = 0,\ j \in \mathbb{N}_0.$$

Other examples are the Hermite polynomials with the weighting function $\psi(\theta) = \exp^{-\frac{\theta^2}{2}}$ on $\mathbb{R}$ or the Laguerre polynomials on $[0,\infty)$ with a weighting function $\psi(\theta) = x^\alpha \exp^{-\theta}$ with an $\alpha > -1$, see also [47].

If the range of interest for a parameter is not identical to the interval of the space on which the polynomials form an orthogonal system, a suitable transformation has to be used.
The multidimensional case is an extension: For a multi index $\xi = (\xi_1, \ldots, \xi_d)$ define (see [17] for notation)

$$P_\xi(\theta) \ = \ \prod_{k=1}^{d} P_{\xi_k}(\theta_k)$$

with a weighting function

$$\psi(\theta) \ = \ \prod_{k=1}^{d} \psi(\theta_k)$$

and therefore the approximation

$$F(\theta) = \sum_{|\xi|=0}^{\infty} a_\xi P_\xi(\theta).$$

Remark 2.4 (Multidimensional orthogonal system). The system P_ξ, $|\xi| = 0, 1, \ldots$ is still orthogonal because for $\xi \neq \eta$ it holds:

$$\langle P_\xi, P_\eta \rangle \ = \ \left\langle \prod_{k=1}^{d} P_{\xi_k}, \prod_{k=1}^{d} P_{\eta_k} \right\rangle \ = \ \prod_{k=1}^{d} \int_{\Lambda_k} P_{\xi_k}(\theta_k)\ P_{\eta_k}(\theta_k)\ d\theta_k \ = \ 0$$

and the system is still a basis due to the construction as the product of basis elements [47].

Example 2.3 (Multidimensional polynomials). For example for a two-dimensional polynomial chaos expansion with degree $N_{PC} = 3$, the set $|\xi| = 2$ is $\{(2,0),(1,1),(0,2)\}$ and $P_{\{2,1\}}(\theta_1,\theta_2) = P_2(\theta_1)P_1(\theta_2)$.

Extension to stochastic functions

In the polynomial chaos expansion the function F is considered to be deterministic. The context is slightly different in stochastic optimization problems with the structure of equation (2.25): The function L is deterministic as in the setting above. But the approximation with the LS function $F_L^{(m)}$ is stochastic. Nevertheless a convergence property can be stated if "many" simulations m are used for the estimation of the transition probabilities in LS function $F_L^{(m)}$. Let Ω be the space on which the random variable $F_L^{(m)}$ is defined and μ the probability measure on Ω corresponding to the Gillespie algorithm. Now approximate the function $F_L^{(m)}(\theta,\omega)$ for every fixed $\omega \in \Omega$ with a series of orthogonal polynomials $G_k^m(\theta,\omega)$. The question now is if it holds $G_k^m(\theta,\omega) \ \to \ l(\theta)$ for $k \to \infty$ and $m \to \infty$.
The notation "(ω)" in the following will indicate that the function is a random variable.

Lemma 2.1. *For the approximation with the LS function $F_L^{(m)}(\theta,\omega)$, equation (2.9), of the likelihood function $l(\theta)$, equation (2.7), it holds if $p_\theta(\nu_i,t_i|\nu_{i-1},t_{i-1}) > 0$ for every $i=1,\ldots,n$:*

$$F_L^{(m)}(\theta,\omega) \rightarrow L(\theta) \text{ a. s. on } \Omega.$$

Proof. Use $p^{(m)}(i,\theta,\omega)$ as a short notation for the random variable $p_\theta(\nu_i,t_i|\nu_{i-1},t_{i-1})$. For $i=1,\ldots,n$ it holds: $\hat{p}^{(m)}(i,\theta,\omega) \rightarrow p(i,\theta)$ a. s. on Ω (see also [48], chapter 12.1). Thus for $p_i > 0$ also

$$\begin{aligned}\lim_{m\to\infty} F_L^{(m)}(\theta,\omega) &= \lim_{m\to\infty} \sum_{i=1}^{n} \log\ p^{(m)}(i,\theta,\omega) \\ &= \sum_{i=1}^{n} \log \lim_{m\to\infty} p^{(m)}(i,\theta,\omega) \\ &\overset{a.s.}{=} \sum_{i=1}^{n} \log\ p(i,\theta) \\ &= l(\theta)\end{aligned}$$

where a.s. means almost sure convergence. This leads to the desired convergence. □

Theorem 2.1 (Convergence of polynomial chaos expansion for stochastic functions)**.** *Define for each m a series of orthogonal polynomials $G_k^m(\theta,\omega)$ which converges to $F_L^{(m)}(\theta,\omega)$ for $k\to\infty$ in the L^2 sense. Then it holds $G_k^m(\theta,\omega) \rightarrow l(\theta)$ in L^2 as $m,k\to\infty$ if $p_i(\theta) > 0$ for all $i=1,\ldots,n$.*

Proof. As $F_L^{(m)}(\theta,\omega) \leq 1$ for all ω it is in $L^2(\Omega)$ for each θ. Next the theorem of dominated convergence allows to conclude the L^2 convergence of $F_L^{(m)}(\theta,\omega) \rightarrow l(\theta)$ from lemma 2.1. Then:

$$\begin{aligned}&\int_{\Lambda\times\Omega} |G_k^m(\theta,\omega) - l(\theta)|^2\ d(\mu\times\psi) \\ &\quad= \int_{\Lambda\times\Omega} \left|G_k^m(\theta,\omega) - F_L^{(m)}(\theta,\omega) + F_L^{(m)}(\theta,\omega) - l(\theta)\right|^2\ d(\mu\times\psi) \\ &\quad\leq \int_\Omega \underbrace{\int_\Lambda \left|G_k^m(\theta,\omega) - F_L^{(m)}(\theta,\omega)\right|^2 d\ \psi}_{\xrightarrow{k\to\infty} 0 \text{ in } L^2}\ d\ \mu + \int_\Lambda \underbrace{\int_\Omega \left|F_L^{(m)}(\theta,\omega) - l(\theta)\right|^2 d\ \mu}_{\xrightarrow{m\to\infty} 0 \text{ in } L^2} d\ \psi \\ &\quad\xrightarrow{k,m\to\infty} 0.\end{aligned}$$

The order of the integration may be exchanged due to the theorem of Fubini. □

The last two results basically say that using enough simulations and an order of approximation that is high enough the result is close to the underlying likelihood function. For results see the application chapter, in which the polynomials are normalized to orthonormal polynomials.

Sparse Grids

To make the polynomial chaos method as efficient as possible it is necessary to use a suitable integration grid Γ. It is generally not known what kind of Γ would be optimal for stochastic

optimization problems in systems biology. Hence a multidimensional grid is derived which is optimal with respect to the polynomial degree. Starting with the one-dimensional case

$$\int_\Lambda F(\theta)d\theta \;\approx\; \sum_{i=1}^{n_l} w_{li} F(\gamma_{li})$$

where $l \in \mathbb{N}$ is the level of the quadrature formula and w_{li} are weights: $\Gamma_l^1 = (\gamma_{l1}^1, \ldots, \gamma_{ln_l}^1)$. In the one-dimensional case examples for quadrature formulas are the one-dimensional Gauss quadrature formulas such as Gauss Legendre or Gauss Hermite quadrature rules [34]. One important subclass of quadrature formulas are nested quadrature formulas: $\Gamma_{l-1}^1 \subset \Gamma_l^1$. One option for deriving multidimensional sparse grids is presented by [49], which will be briefly reviewed here with a slightly different notation:
For a one-dimensional quadrature formula Γ_l^1 consider the difference grids:

$$\Xi_l^1 \;=\; \Gamma_l^1 \backslash \Gamma_{l-1}^1$$

with $\Gamma_0^1 = \{\}$. A sparse grid for the $\mathcal{D}$-dimensional case is built with:

$$\Gamma_l^{\mathcal{D}} \;=\; \biguplus_{|\eta| \le l+\mathcal{D}-1} \Xi_{\eta_1}^1 \;\times\; \ldots \;\times\; \Xi_{\eta_{\mathcal{D}}}^1$$

and $\eta \in \mathbb{N}^{\mathcal{D}}$. Note that the calligraphy letter $\mathcal{D}$ is used for the dimension of θ to avoid confusion with the dimension of the data, for which both d and D will be used later.
For the nested case - on which the focus will be in this text - the weight for a point $\gamma_{\eta\kappa}^{\mathcal{D}} = (\gamma_{\eta_1\kappa_1}^{\mathcal{D}}, \ldots, \gamma_{\eta_{\mathcal{D}}\kappa_{\mathcal{D}}}^{\mathcal{D}})$ can be calculated as follows:

$$w_{\eta\kappa} \;=\; \sum_{|\eta+\mathbf{q}| \le l+2\mathcal{D}-1} v_{(\eta_1+\mathbf{q}_1)\kappa_1} \cdots\; v_{(\eta_{\mathcal{D}}+\mathbf{q}_{\mathcal{D}})\kappa_{\mathcal{D}}}$$

with $\mathbf{q} \in \mathbb{N}^{\mathcal{D}}$ and

$$v_{(k+q)j} \;:=\; \begin{cases} w_{kj}, & q = 1 \\ w_{(k+q-1)r} - w_{(k+q-2)s} & q > 1,\; \gamma_{kj} = \gamma_{(k+q-1)r} \;=\; \gamma_{(k+q-2)s} \end{cases}$$

with r and s determined by the case by case analysis condition. Smolyak's algorithm ([49] and references therein) can now be written as

$$\int_\Lambda F(\theta)d\theta \;\approx\; \sum_{|\eta| \le l+\mathcal{D}-1} \sum_{\kappa_1=1}^{n_{\eta_1}} \cdots \sum_{\kappa_{\mathcal{D}}=1}^{n_{\eta_{\mathcal{D}}}} w_{\eta\kappa}\; F(\gamma_{\eta\kappa})$$

where n_l denotes the number of elements in Ξ_l^1.

Remark 2.5 (Number of points for a sparse grid). The number of points can be determined by

$$n_l^{\mathcal{D}} \;=\; \sum_{|\eta| \le l+\mathcal{D}-1} n_{\eta_1} \ldots n_{\eta_{\mathcal{D}}}.$$

For example for $\mathcal{D} = 2$ and $l = 5$, the sum is calculated over all $|\eta| \le 6$, which yields $n_5^2 = 33$.

For the following a delayed Kronrod-Patterson rule is used as one-dimensional grid. It is suggested for numerical computations in [50] (page 742) and is derived from a Kronrod-Patterson rule [50] (page 731). Points and weights for this rule are given in [51](chapter 3.2 and table M10 and M11). Further information on points for sparse grids can be found on the website belonging to [52].

Table 2.1: **Nodes for sparse grids.** Nodes of the one-dimensional delayed Kronrod-Patterson rule Γ_l^1

$l = 1$	$\Gamma_l^1 = \{\ 0\ \}$ $w_l = \{\ 2\ \}$
$l = 2, 3$	$\Gamma_l^1 = \{\ 0,\ \pm 0.7746\ \}$ $w_l = \{\ 0.8889,\ 0.5556\ \}$
$l = 4, 5, 6$	$\Gamma_l^1 = \{0,\ \pm 0.4342\ \pm\ 0.7746,\ \pm 0.9605\ \}$ $w_l = \{\ 0.4509,\ 0.4014,\ 0.2685,\ 0.1047\ \}$
$l = 7$ *to* 12	$\Gamma_l^1 = \{0, \pm 0.2234, \pm 0.4342, \pm 0.6211, \pm 0.7746, \pm 0.8885, \pm 0.9605, \pm 0.9938\}$ $w_l = \{\ 0.2255,\ 0.2192,\ 0.2006,\ 0.1751,\ 0.1344,\ 0.0929,\ 0.0516,\ 0.0170\ \}$

Chapter 3

Objective function based on short time ODE integration

This chapter will describe a method for parameter estimation based on short time ODE integration. The method will be named multiple shooting for stochastic systems (MSS).

3.1 Fully observed case

Denote the measurements at time points $t_0, \ldots, t_n$ with $\nu = (\nu_0, \ldots, \nu_n)$. Assume for this subsection that all states can be measured. As the time course in stochastic modeling is a continuous time Markov jump process the likelihood function can be factorized into the product of transition probabilities:

$$L(\nu, \theta) \;=\; \prod_{i=1}^{n} p_\theta(\nu_i, t_i | \nu_{i-1}, t_{i-1}).$$

The transition probability p_θ is generally not known and might either be estimated by means of simulations or calculated by using the solution of a high dimensional CME system, both of which is very time-consuming. The approach approximates the system on the short time interval $[t_{i-1}, t_i]$ with an ODE model. The fact that this is done only on a very short time interval is crucial. An ODE approximation is stronger than a linear noise approximation but as this approximation only has to hold on a short time interval it is much less restrictive than the usual linear noise approximation [53]. The proposed approach is motivated by the methods for parameter estimation in ODE systems introduced by Bock [18,54] and uses short time ODE integration: starting at time point t_{i-1} at state ν_{i-1} the initial value problem of the systems of ODEs is solved with initial value ν_{i-1} at time t_{i-1} until time t_i. The result, $h(t_i, \theta, \nu_{i-1}, t_{i-1})$, is compared to the data point ν_i and the residual is defined as

$$\epsilon_i \;=\; \nu_i - h(t_i, \theta, \nu_{i-1}, t_{i-1}), \tag{3.1}$$

which for the model leads to the description

$$\begin{pmatrix} \nu_1 \\ \nu_2 \\ \vdots \\ \nu_n \end{pmatrix} = \begin{pmatrix} h(t_1, \theta, \nu_0, t_0) \\ h(t_2, \theta, \nu_1, t_1) \\ \vdots \\ h(t_n, \theta, \nu_{n-1}, t_{n-1}) \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

If the residuals ϵ are independent, normally distributed random variables with mean zero and constant variance, the least squares estimator

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \; F_h(\nu, \theta)$$

with

$$F_h(\nu, \theta) \;=\; \sum_{i=1}^{n} \|h(t_i, \theta, \nu_{i-1}, t_{i-1}) - \nu_i\|_2^2 \tag{3.2}$$

is a maximum likelihood estimator. In general the distribution for a time point in a stochastic model is not known and not necessarily Gaussian. Hence theoretically the properties of a maximum likelihood estimator can not be guaranteed. In practice it is possible to test if the ϵ_i perform approximately like independent, normally distributed random variables with mean zero. If this is the case the estimator might still be quite powerful. Later in this chapter test functions will be suggested which describe the properties of the residuals. The structure of equation (3.2) allows handling normally distributed measurement noise as well without losing the desired properties.
Instead of using the solution of the initial value problem of the ODE system in equation (3.2) one can also use the mean of the stochastic simulations – the first moment – which can be calculated with a moment-closure [11] with only the first moment. Although calculating the moment-closure is fast compared to other calculations this would be slightly slower than the suggested MSS method. If it is more stable, especially in cases of partially observed models, has to be investigated.
Equation (3.2) corresponds to the multi-experiment setting described in [19]. Each time interval is regarded as one experiment and the parameter vector θ are the global variables common to all experiments. There are no local variables which are specific for only some of the experiments. For the optimization it is consequently possible to apply the efficient methods suggested in [19].

3.2 Partially observed models

Assume that only the first d components of the D-dimensional vector of species ν can be observed. At time t_0 the unobserved states $\nu_0^{(d+1)}, \ldots, \nu_0^{(D)}$ are also used as optimization variables. In fact the unobserved states are discrete numbers but for the optimization purpose this condition is relaxed and they are optimized as real numbers. For the solution of the initial value problem on $[t_{i-1}, t_i]$ instead of the full measurement ν the observed states $\nu_{i-1}^{(1)}, \ldots, \nu_{i-1}^{(d)}$ are used from the measurement and for the unobserved states the result of the initial value problem on the previous time interval is used, thus $h^{(d+1,\ldots,D)}(t_{i-1}, \theta, \cdot, t_{i-2})$. Furthermore it is possible and in some situations even necessary to enlarge the optimization vector even more and to include further unobserved states. Define an index set K with $\{t_0\} \subset K$ which contains all time points at which the unobserved states will be included in the optimization variable. Denote the unobserved states in the optimization vector at time t_j with $\nu_K^{(j)} = \left(\nu_j^{(d+1)}, \ldots, \nu_j^{(D)}\right)$ and the union of unobserved states at different time points with $\nu_K = \left(\nu_K^{(j)}\right)_{j \in K}$. Now define the completion of the observed measurement as $\tilde{\nu}$ with

$$\tilde{\nu}_j \;=\; \begin{cases} \nu_j^{(1)}, \ldots, \nu_j^{(d)}, \nu_K^{(j)} & : \quad t_j \in K \\ \nu_j^{(1)}, \ldots, \nu_j^{(d)}, h^{(d+1,\ldots,D)}(t_j, \theta, \tilde{\nu}_{j-1}, t_{j-1}) & : \quad t_j \notin K. \end{cases}$$

Then again as in the fully observed case a distance measure is used to compare the result of the integration with the data point:

$$F_K(\nu, \theta, \nu_K) \;=\; \sum_{i=1}^{n} \left\| h^{(1,\ldots,d)}(t_i, \theta, \tilde{\nu}_{i-1}, t_{i-1}) - \nu_i^{(1,\ldots,d)} \right\|_2^2. \tag{3.3}$$

In the multi-experiment setting referred to in the previous section $\nu_K^{(j)}$ can now be considered as local variables specific to the experiments connected to $[t_j, t_{j+1}]$.
The case $K = \{t_0\}$ means that just the first unobserved state is included in the optimization and the case $K = \{t_0, t_1, \ldots, t_n\}$ means that all unobserved states are included in the optimization.
Remarks on the numerical optimization of the function in equation (3.3) can be found in section 3.4.

3.3 Test functions for the validity of the approximation

As the method uses an approximation it is very important to investigate if a model satisfies the approximation. Therefore this section suggests test functions to see how well the approximation works. Although defined for arbitrary parameters the test functions are evaluated with the optimal $\hat{\theta}$ resulting from an optimization of the suggested functionals in order to see if the stochastic data can be represented with the ODE model together with the optimal parameter. To check the assumption on the mean of the residuals their average, $\bar{\epsilon} = \frac{1}{n}\sum_{i=1}^{n} \epsilon_i$, can be calculated. If the model is well approximated this should be small. Note that the assumption requires more, namely that the expectation of ϵ for every time point is zero, which can not be tested with a single trajectory of measurements.
To see if the residuals are approximately normally distributed calculate the Kullback-Leibler divergence of a density estimate from the ϵ_i and a centered normal distribution of variance σ_{KL}^2 restricted to the support of the density estimate. The Kullback-Leibler divergence is then minimized over σ_{KL}^2. If the Kullback-Leibler divergence for the optimal σ_{KL}^2 is close to zero this means that the ϵ_i can be approximated well by a normal distribution with constant variance. The point that the variance is constant is important as it is not possible to estimate the variance from only one observation per time point.
For more theoretical remarks on the relation of the Kullback-Leibler divergence to the goodness of an approximation by a normal distribution see [55,56]. If the system contains more than one species the dimension of ϵ is greater than one. In this case the analysis is done componentwise because the vector is approximately normally distributed if all components are approximately normally distributed. It would be interesting for future research to what extend accounting for possible correlation plays a role: for now it is enough to see that the ϵ could be a realization from a vector with independent normal distributions, see also the Cramér-Wold theorem [57].
Calculating the correlation of the residuals between every pair of time points is not possible if only one time course of measurements is recorded. Instead it will be calculated how long it takes until the residuals are uncorrelated. Estimate the autocorrelation of the residuals:

$$\hat{R}(k, \epsilon) \;=\; \frac{1}{(n-k)\hat{\sigma}_\epsilon^2} \sum_{i=1}^{n-k} (\epsilon_{t_i} - \bar{\epsilon})(\epsilon_{t_{i+k}} - \bar{\epsilon})$$

where k represents the time step and $\hat{\sigma}_\epsilon^2$ an estimate for the variance within the residuals ϵ. If ϵ is such that $\tilde{k} := \min(k | \hat{R}(k, \epsilon) < 0) \;>\; 0$ exists define the autocorrelation time as $act(\epsilon) = t_{\tilde{k}}$. If the total horizon of measurements $t_n - t_0$ is only of the same size as the autocorrelation time of the residuals $act(\epsilon)$ this indicates a strong violation of the assumption of uncorrelated residuals. It would also be possible to calculate a continuous autocorrelation time using an interpolation between the last positive and the first negative value but this does not give more information than the suggested procedure.
Furthermore it is an important question for the quality of the estimation how much information can be found with the MSS method in the intrinsic noisy system. Define a signal to noise ratio for

each of the observed components,

$$SNR(\nu,\theta) \;=\; \Big(SNR(\nu^{(1)},\theta),\ldots,SNR(\nu^{(d)},\theta)\Big) \text{ with} \tag{3.4}$$

$$SNR(\nu^{(l)},\theta) \;=\; \frac{\sum_{i=1}^{n}\; h^{(l)}(t_i,\theta,\nu_{i-1},t_{i-1})-\nu_{i-1}^{(l)}}{\sum_{i=1}^{n}\;|\epsilon_i^{(l)}|}. \tag{3.5}$$

The higher the SNR value is the more information is contained in the data. If the SNR is small (<1) many measurements are needed. The componentwise analysis has the advantage that high or low SNR values can be assigned to the corresponding components. Another criterion is a measure on the residuals in comparison to the total system's dynamics:

$$NDR(\nu,\theta) \;=\; \Big(NDR(\nu^{(1)},\theta),\ldots,NDR(\nu^{(d)},\theta)\Big) \text{ with} \tag{3.6}$$

$$NDR(\nu^{(l)},\theta) \;=\; \frac{\sum_{i=1}^{n}\;|\epsilon_i^{(l)}|}{\sum_{i=1}^{n}\;|\nu_i^{(l)}-\nu_{i-1}^{(l)}|}. \tag{3.7}$$

If the NDR is small the systems is containing more information than for large NDR. The end of this section will give a remark on the solution of the minimization problem for the Kullback-Leibler divergence which is suggested to the interested reader.

Remark 3.1 (Solution of the minimization problem for the Kullback-Leibler divergence). The Kullback-Leibler divergence $KL(f,g)$ between two probability density functions f and g on some common space Ω is defined as

$$KL(f,g) \;=\; \int_\Omega f(x)\;\log\left(\frac{f(x)}{g(x)}\right)dx.$$

Now let f be the density estimated from the residuals and Ω the support of f, which is assumed to be an interval. If this is not the case the procedure of the density estimation can be adopted or a subset of the support chosen. Of course the supports of g, which is $\mathbb{R}$, and f ,which is in general a subset of $\mathbb{R}$, are not identical. Calculating the KL-divergence on Ω is hence again an approximation, which will be however adopted without change of notation. Then the problem from above is to find a σ which minimizes the Kullback-Leibler divergence between f and a normal distribution with variance σ^2 and density $g(\sigma,x)$:

$$\min_\sigma\; KL(f,g) \;=\; \min_\sigma \int_\Omega f(x)\; log\left(\frac{f(x)}{g(\sigma,x)}\right)\; dx.$$

For the first derivative it holds

$$\begin{aligned}\frac{\partial}{\partial\sigma}\; KL(f,g) \;&=\; \int_\Omega -\frac{f(x)}{g(\sigma,x)}\frac{\partial}{\partial\sigma}g(\sigma,x)\; dx\\ &=\; \int_\Omega f(x)\left(\frac{1}{\sigma^2}-\frac{x^2}{2\sigma^3}\right)\; dx\\ &=\; \underbrace{\int_\Omega x^2 f(x)dx}_{E_f[X^2]}\; -\;\sigma^2.\end{aligned}$$

Note that the order of integration and differentiation may be changed. Thus the optimal σ is the square root of the second moment of the density estimate and therefore easy to calculate.

It is in fact a minimum because

$$\begin{aligned}\left(\frac{\partial}{\partial\sigma}\right)^2 KL(f,g) &= \int_\Omega f(x)\left(\frac{\partial}{\partial\sigma}\right)^2\left(\frac{1}{\sigma^2}-\frac{x^2}{2\sigma^3}\right)\,dx \\ &= \int_\Omega f(x)\frac{3x^2-\sigma^2}{2\sigma^2}\,dx\Big|_{\sigma^2=E_f[X^2]} \\ &> 0.\end{aligned}$$

3.4 Optimization

In case of the fully observed system the number of variables to be optimized is exactly the number of parameters. In case of a partially observed system the number of variables for the optimization is the number of parameters plus the product of the number of unobserved species and the number of time points included in the optimization (the length of K).
As the objective function is completely deterministic – meaning, it is calculated without the use of stochastic simulations – the optimization procedure is the same as in parameter estimation for ODE systems: it is possible to apply derivative-based methods [58] or global methods [41].
Certainly the optimization of equation (3.3) with larger K can be much more challenging than with $K = \{0\}$ due to the increased dimensionality of the optimization problem. However, the focus of the thesis is the formulation of the optimization problem with a suitable objective function. The choice of the numerical optimization method, question of local minima or the question whether derivative-based or global methods are preferable shall not be the focus of this work.

Chapter 4

Applications

Now the different methods will be tested on some biological systems. Units are given in particles, seconds and moles and in the ODE interpretation of rate laws with a compartment of 1 ml. All stochastic simulations in this chapter are performed with the software COPASI [28] using an exact implementation [59] of the Gillespie algorithm (section 1.2.2). In this context exact means a rigorous equivalence to the CME approach.

4.1 Immigration-Death model

The Immigration-Death model contains one specie X and two reactions: an immigration reaction independent of the number of species in the systems and a death reaction proportional to the number of species in the system:

$$\emptyset \xrightarrow{\theta_1} X$$
$$X \xrightarrow{\theta_2} \emptyset$$

with a representation in ODEs

$$\frac{dx}{dt} = \theta_1 - \theta_2 x, \quad x(0) = x_0$$

as already mentioned in section 1.2.3, where some example trajectories are shown in figure 1.1.

4.1.1 Estimation using the MLS function and a Particle Swarm algorithm

Original Particle Swarm

This subsection will present the results using the original Particle Swarm optimization algorithm described in section 2.2. For the evaluation of this procedure 25 data sets are simulated using the Gillespie algorithm implemented in COPASI [28]. For each data set the parameters are estimated using the MLS function $F_g^{(m)}$, equation (2.11), with $m = 1000$ simulations for the evaluation of the transition probabilities and a Particle Swarm program as in figure 2.8 of section 2.4.2 implemented in Mathematica [38] with 100 iterations and 20 particles per iteration on a range of $[0, 10] \times [0, 10]$. This result is compared to an exact result, which is possible for this example using the probability generating function in section 2.3.1 to calculate the transition probabilities for the functional

l, equation (2.7). The optimization is performed with the *FindMinimum*-routine of the software Mathematica and the initial value for the optimization is the true parameter and the constraints are $0 < \theta_k < 5,\ k = 1, 2$. The *FindMinimum*-routine of Mathematica uses generally a quasi-Newton method. If the function is structurally a sum of squares then a Levenberg-Marquardt variant of the Gauss-Newton method is chosen, further details can be found in the Mathematica documentation. The procedure using the functional l will be named exact likelihood (EL) method or estimator in the following. For both estimators the relative errors are calculated with respect to the true parameter and for the estimator using simulations the relative error with respect to the EL estimator is calculated. The mean of the estimators and the relative errors are presented in table 4.1.

Table 4.1: **Estimation with PS and $F_g^{(m)}$.** Averages, standard deviation and relative errors for estimation results for 25 simulated data sets for the Immigration-Death model with $x_0 = 10$, 50 observations, $T = 50$, $\theta^{(0)} = (1, 0.1)$.

	Estimation Results				Relative error		
	EL		PS with $F_g^{(1000)}$		EL	PS with $F_g^{(1000)}$	PS with $F_g^{(1000)}$ versus EL
θ_1	0.99	$\pm$0.19	0.96	$\pm$0.19	15%	16%	7%
θ_2	0.106	$\pm$0.023	0.104	$\pm$0.025	18%	20%	5%

Table 4.1 indicates that an estimation using the MLS function $F_g^{(m)}$ is possible and unbiased. The relative error compared to an exact estimation with the EL method is below 10% and with respect to the true parameter comparable to an EL estimation. Calculating these different relative error values has the following reason: The error of the EL estimation to the true parameter is due to the stochasticity in the data. The error of the estimator using simulations to the EL estimator is due to the stochasticity of both using simulations and a stochastic optimization algorithm. The relative error of the estimator using simulations to the true parameter indicates that these effects cancel out in such a way that the relative errors of both estimators to the true parameter are of the same size.
The number of function evaluations for the parameter estimation of each time series is the following: The number of iterations times the number of particles per iteration, thus 2000. For each function evaluation 1000 stochastic simulations have to be performed. This results in 2 000 000 stochastic simulations. The average of the computing times of the 25 estimations was 12 hours on a personal computer with a 2, 66 GHz Intel Core$^{\text{TM}}$ 2 Duo T9550 processor.

Modified Particle Swarm

This subsection will present the results using the modified Particle Swarm optimization algorithm described in figure 2.9 of section 2.4.2. For the evaluation of this procedure 25 data sets are simulated using the Gillespie algorithm implemented in COPASI [28]. For each data set the parameters are estimated using the MLS function $F_g^{(m)}$ with $m = 1000$ simulations for the estimation of the transition probabilities and the modification of Particle Swarm program implemented in Mathematica [38] with 20 particles per iteration. The termination criterion is checked every 10 iterations and 20 particles are used per check. This result is compared to an exact result with the EL method, which is possible for this example using the probability generating function in section 2.3.1 to calculate the transition probabilities. For both estimators the relative errors are calculated with respect to the true parameter. For the estimator using simulations the relative error is calculated with respect to the EL estimator.

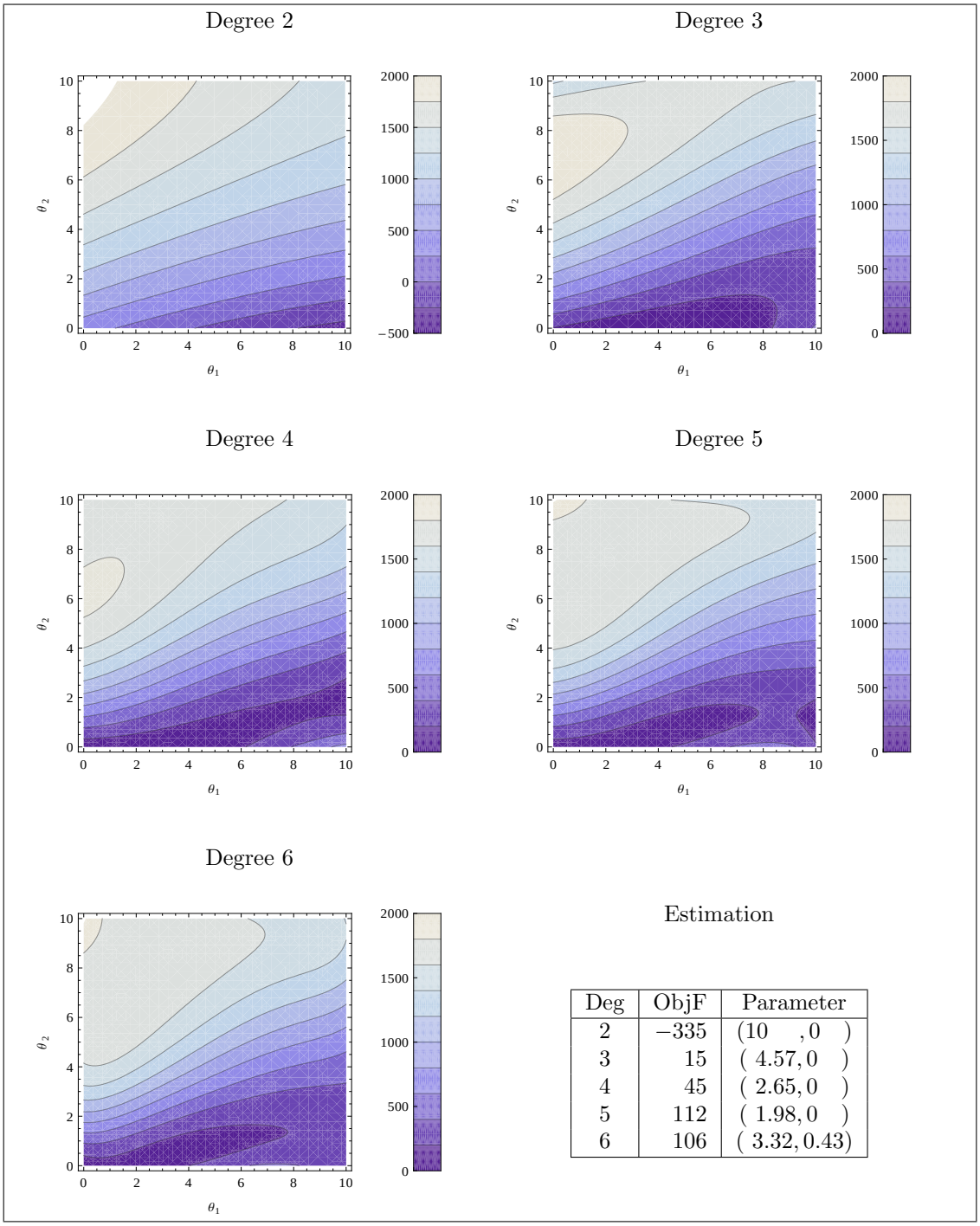
Table 4.2: **Estimation with the modified PS and $F_g^{(m)}$.** Averages, standard deviation and relative errors for estimation results for 25 simulated data sets for the Immigration-Death model with $x_0 = 10$, 50 observations, $T = 50$, $\theta^{(0)} = (1, 0.1)$ and $m = 1000$ simulations for the estimation of the transition probabilities.

	Estimation Results				Relative error		
	EL		mod PS with $F_g^{(1000)}$		EL	modPS, $F_g^{(1000)}$	modPS, $F_g^{(1000)}$ to EL
θ_1	0.99	$\pm$0.19	0.98	$\pm$0.24	18%	15%	10%
θ_2	0.106	$\pm$0.023	0.104	$\pm$0.027	22%	18%	9%

The mean of the estimators and the relative errors are presented in table 4.2. Table 4.2 shows that an estimation using the MLS function $F_g^{(m)}$ is possible and unbiased. The relative error compared to an EL estimation is around 10% and with respect to the true parameter of the same size as an EL estimation. Calculating this different relative error values has the following reason: The error of the EL estimation to the true parameter is due to the stochasticity in the data. The error of the estimator using the MLS function to the EL estimator is due to the stochasticity of both using simulations and a stochastic optimization algorithm. The relative error of the estimator using the MLS function to the true parameter indicates that these effects cancel out in a way such that the relative errors of both estimators to the true parameter are of the same size.
Comparing the modification to the original Particle Swarm (table 4.1), which was designed for the optimization of deterministic functions it has to be stated that the modification terminated in average after 19 iterations. This means a reduction of function evaluations by a factor 5 with respect to the original version, which is an important improvement as function evaluations using simulations are computationally intensive. It would be possible to check the termination criterion after each iteration leading to a termination with even less function evaluations. But this would considerably increase the costs of evaluating the termination criterion. This directly leads to the question of an optimal frequency for the termination criterion check is a point for further research. An idea for a further speed up would be to start with a small number of simulations m. This small number of iterations might be enough for a rough and fast search for a good region of low objective function values. After some iterations the algorithm is restarted on a smaller range with a higher number of simulations for a fine adjustment.
With an average number of 414 function evaluations and 1000 stochastic simulations per evaluation the average of the computing times of the 25 estimations was 133 minutes on the same computer as described in the previous subsection.

4.1.2 Estimation using the MLS function and the polynomial chaos expansion

This paragraph will investigate the performance of the polynomial chaos expansion for estimating the parameters for the Immigration-Death model. It is divided into two parts: The first will start with the same range of the parameter space as for the Particle Swarm method, $[0, 10] \times [0, 10]$. The graphical output for this range is visually investigated and a manual zoom into smaller ranges with low function values is performed. Nevertheless due to the small number of function evaluations for each estimation procedure the total number of function evaluations for the estimation is lower than for both of the Particle Swarm algorithms in the previous subsection. The second part is an approach for an automatization of the manual zoom.



Deg	ObjF	Parameter
2	−335	(10 , 0)
3	15	(4.57, 0)
4	45	(2.65, 0)
5	112	(1.98, 0)
6	106	(3.32, 0.43)

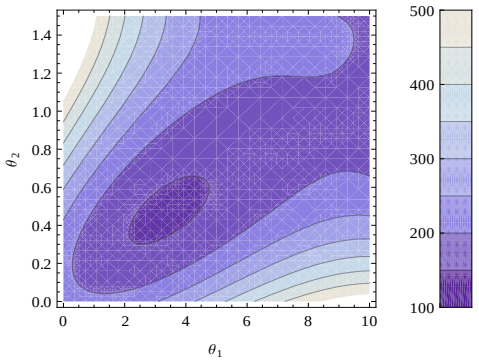
Figure 4.1: **Approximation of the MLS function landscape with polynomial chaos expansion I.** Approximation of the Immigration-Death model MLS function landscape and optimization result for simulated data with polynomial chaos expansion with a grid of 33 data points on $[0, 10] \times [0, 10]$. White stands for higher function values than assigned in the color bar.

Figure 4.2 (A)

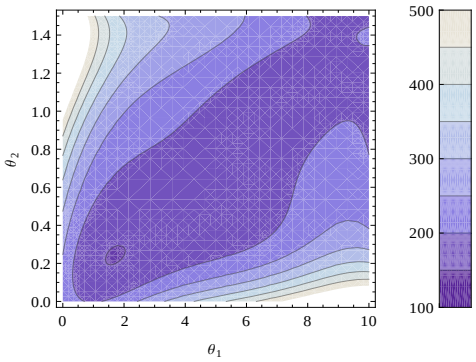
Range: $[0, 10] \times [0, 1.5]$

Estimation

Deg	ObjF	Parameter
2	156	(5.97, 0.81)
3	143	(3.42, 0.45)
4	138	(2.41, 0.31)
5	138	(1.86, 0.25)
6	138	(1.78, 0.26)



Degree 4



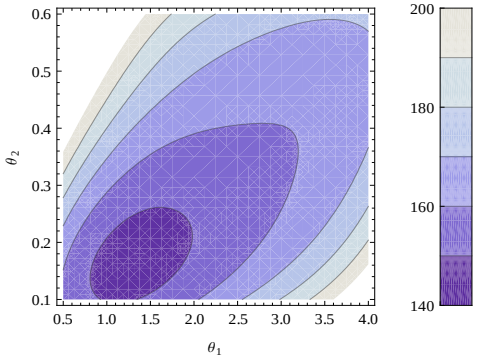
Degree 6

Figure 4.2 (B)

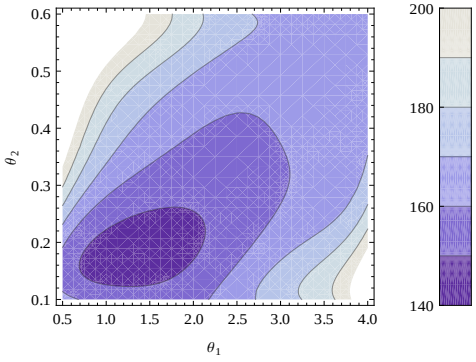
Range: $[0.5, 4] \times [0.1, 0.6]$

Estimation

Deg	ObjF	Parameter
2	148	(1.28, 0.10)
3	146	(1.36, 0.15)
4	146	(1.30, 0.17)
5	145	(1.23, 0.17)
6	145	(1.32, 0.18)



Degree 4



Degree 6

Figure 4.2: **Approximation of the MLS function landscape with polynomial chaos expansion II.** Approximation of the Immigration-Death model MLS function landscape and optimization result for simulated data with polynomial chaos expansion with a grid of 33 data points on different ranges. White stands for higher function values than assigned in the color bar.

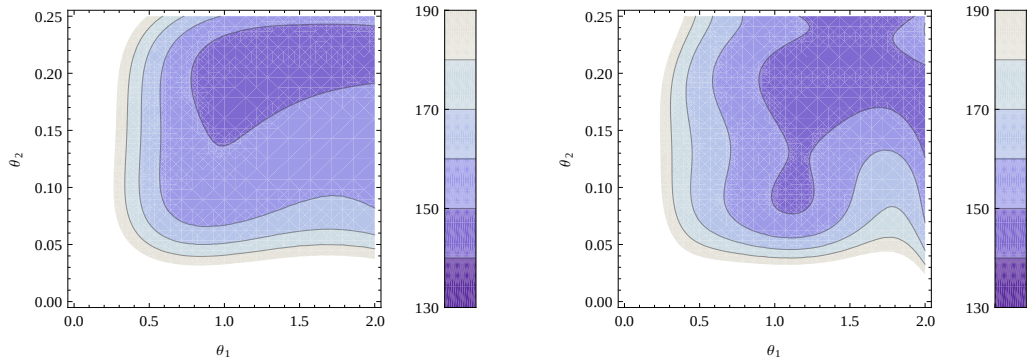
Estimation with manual zoom in:
One data set is simulated stochastically with $\nu_0 = 10$, $\theta^{(0)} = (1, 0.1)$ and 100 observations with $T = 50$. Then the method described in section 2.4.3 is applied with a grid $\Gamma_{l=5}^{d=2}$ of 33 points (section 2.4.3) and the MLS function $F_g^{(m)}$, (2.11), is evaluated with 1000 simulations. The figures 4.1 and 4.2 show the plots of the approximated objective function landscape and the tables give the result for the optimization for different degrees of the polynomial chaos expansion. The graphics in figure 4.1-4.3 are plotted with the *ColorbarPlot* package, which is available in the Wolfram library archive of Mathematica [38]. The optimization is performed with the *FindMinimum*-routine of the software Mathematica. The initial value for the optimization is the mean of the range and the range is as well used as constraints for all cases except 4.1, degree 2, where the lower bound is chosen as initial value because choosing the mean results in a local minimum.

Figure 4.2 (C)

Range: $[0.1, 2] \times [0.01, 0.25]$

Estimation

Deg	ObjF	Parameter
2	138	(1.36, 0.17)
3	145	(1.00, 0.14)
4	145	(1.20, 0.21)
5	145	(1.40, 0.21)
6	148	(1.12, 0.09)



Degree 4 Degree 6

Figure 4.3: **Approximation of the MLS function landscape with polynomial chaos expansion III.** Approximation of the Immigration-Death model MLS function landscape and optimization result for simulated data with polynomial chaos expansion with a grid of 33 data points on different ranges. White stands for higher function values than assigned in the color bar.

All landscapes in figure 4.1 show a valley for small θ_2.Therefore for the first zoom in the range $[0, 10] \times [0, 1.5]$ is chosen. The negative value for degree 2 is due to a bad approximation of the landscape, which might result in negative values. This relatively rough approximation does not allow for a reliable estimate of the parameters, which can be seen by the table of estimation results included in figure 4.1. For the first zoom in the graphics for degrees 4 and 6 are shown representatively in figure 4.2 (A). One point which should be noted is that the landscape seems to have a valley for $\theta_2 \approx 0.1\theta_1$, which is the unidentifiability in the ODE modeling. The figure 4.2 (A) suggests a further zoom in to the range $[0.5, 4] \times [0.1, 0.6]$. The estimates are given in the table included in the figure 4.2 (A). The estimates become better but still are not reliable. The third and fourth zoom in, figure 4.2 (B)-(D), leads to more accurate representations of the landscape

and reliable estimates.
Altogether the $4 \times 33 = 132$ function evaluations took 69 minutes of computing time.
Using the EL method with the probability generating function in section 2.3.1 the minimum is $(\theta_1, \theta_2) = (1.23, 0.16)$. For further research it would be interesting if it is possible to improve the estimation using the distances between the function values of the evaluated grid points and the approximative landscape. Furthermore the minimum function value of the grid could also provide some information.

Estimation results for an automatized zoom in:
The question is now whether it is possible to do the zooming automatically. The procedure is described in the pseudo code in figure 4.4. Now a simulation study is performed as in the Particle

Pseudo code for automatized zoom in

- Initialize range for the first iteration: $r^{(0)} = \prod_{i=1}^{d} [r_{l,i}^{(0)}, r_{u,i}^{(0)}]$.
- For $j = 0, \ldots, J$
  1. Calculate estimates $\theta^{(l,j)}$ for degree $l = 2, ..., N_{PC}$ on a grid Γ_j for a range $r^{(j)}$.
  2. For $i = 1, ..., d$
     - Calculate $Floor\left(\left(\theta_i^{(l,j)}\right)_{(l=2,\ldots,N_{PC})}\right) =: min$ and $Ceiling\left(\left(\theta_i^{(l,j)}\right)_{(l=2,\ldots,N_{PC})}\right) =: max.$
     - If $min < r_{l,i}^{(j)}$, $r_{l,i}^{(j+1)} = r_{l,i}^{(0)}$, else $r_{l,i}^{(j+1)} = min.$
     - If $max > r_{u,i}^{(j)}$, $r_{u,i}^{(j+1)} = r_{u,i}^{(0)}$, else $r_{u,i}^{(j+1)} = max.$
  3. $r^{(j+1)} = \prod_{i=1}^{d} [r_{l,i}^{(j+1)}, r_{u,i}^{(j+1)}]$.
- Output: $\left(\theta^{(l,J)}\right)_{(l=1,\ldots,N_{PC})}$ of Γ_J.

Figure 4.4: **Pseudo code for polynomial chaos expansion with automatized zoom in**

Swarm, section 4.1.1. 50 data sets are simulated with the Gillespie algorithm. For each of the data sets the parameter is estimated using the automatic zoom in with the polynomial chaos expansion described in figure 4.4 with a grid $\Gamma_{l=10}^{d=3}$ of 161 data points (section 2.4.3) and $J = 2$ iterations. 1000 simulations are used for the estimation of the transition probabilities in the MLS function $F_g^{(m)}$, (2.11), see table 4.3. The average computing time for the $3 \times 161 = 483$ function evaluations was 140 minutes, which is the same magnitude as for the modified Particle Swarm. The results display a quite large fluctuation of the estimates. Thus the automatization procedure still has to be improved. Furthermore a stopping criterion should be developed terminating the zoom in when the landscape is "similar enough" to the evaluated points.
What is nevertheless an advantage of the polynomial chaos expansion is the possibility of having

Table 4.3: **Estimation results for the polynomial chaos expansion with automatized zoom in.** Averages, standard deviation and relative errors for estimation results for 50 simulated data sets for the Immigration-Death model with $x_0 = 10$. 50 observations, $\Delta t = 0.5$, $\theta^{(0)} = (1, 0.1)$.

	Estimation Results		Rel err
θ_1	1.58	± 0.91	60%
θ_2	0.14	± 0.10	62%

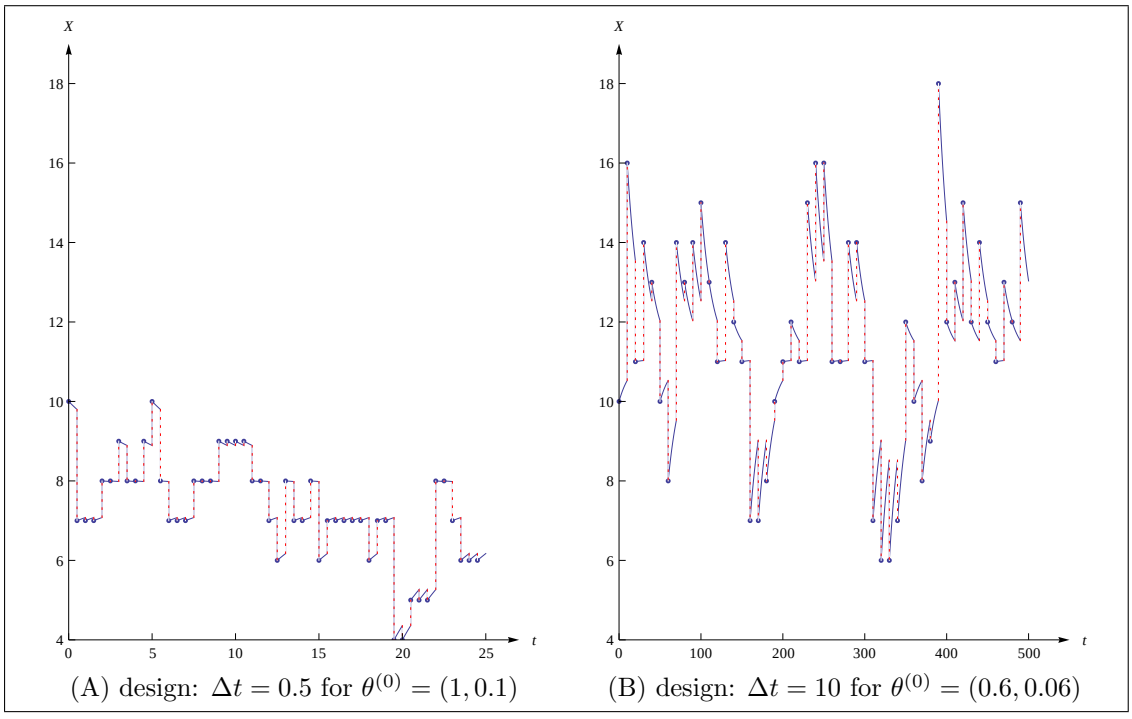
a graphical representation of the landscape. This allows a much better analysis than just a single estimate as given with Particle Swarm. For higher dimensions projections can be used to give information on the parameters of interest. Hence the use of the polynomial chaos expansion seems to be more an expert tool for investigation of the landscape structure of the objective function than a general applicable tool being ready for implementation.

4.1.3 Estimation using the MSS method

On the one hand the Immigration-Death model is an instructive example – the partially observed case even cannot be studied – on the other hand it is highly relevant as it allows a comparison between the performance of the MSS method with the functional of equation (3.2) and the result of the EL method with the exact transition probability, which can only be calculated exactly in very simple cases.
To evaluate the performance of the MSS method it is compared to an EL estimation on 50 data sets obtained from simulations using the Gillespie method [2] with the software COPASI [28]. The initial condition is always the steady state of the system. For each of the data sets the objective function of the MSS method and the objective function of the EL method are optimized using the *FindMinimum*-routine of the software Mathematica [38] with true parameter as initial value for the optimization and constraints $0 < \theta_k < 5$ for both components $k = 1, 2$ of the parameter vector. For remarks on the optimization method used by Mathematica see section 4.1.1. Then the mean and the standard deviation of both estimators as well as the average relative error are calculated. This procedure is done for different parameters and designs. The results are shown in table 4.4. To check the approximation the mean and autocorrelation time of the residuals as well as the SNR are calculated.
The results given in table 4.4 lead to two conclusions: An estimation is possible and unbiased if there are enough measurements, 4.4 (B), (C). If the trajectory is very short 4.4 (A) the estimator might be biased. The reason for that is that for a low SNR more measurements are needed to sum up enough information for the estimation - see figure 4.5, which shows the ODE dynamics in form of the solution of the corresponding initial value problem $h(t_i, \theta, \nu_{i-1}, t_{i-1})$ as well as the residuals (dotted red line) for each interval $[t_{i-1}, t_i]$. One can see that for this situation the system's dynamics are not well represented by the ODE solutions. Nevertheless an estimation in this scenario is possible, for example using methods suggested in [60]. This method uses the ODE steady state information to determine the functional relationship between the two parameters and then uses the stochastic fluctuations to determine their absolute value. The EL method also makes much better use of the intrinsic fluctuations than the MSS method and therefore results in more accurate estimates. But it is only possible in this simple example model. The mean and autocorrelation time of the residuals of the MSS method behave very well with respect to the comments on the residuals given in the methods section. The computing time for one estimation is 0.05 seconds on the same machine as mentioned in section 4.1.1.

Table 4.4: **Estimation results for Immigration-Death model.** Mean, standard deviation, average relative error and test functions of estimation results for 50 simulated data sets for the Immigration-Death model with $\nu_0 = 10$.

	Estimation Results				Rel Err		Test functions (averages)
	EL		MSS		EL	MSS	
(A) 100 observations, $\Delta t = 0.5$, $\theta^{(0)} = (1, 0.1)$							
θ_1	1.00	± 0.19	1.56	± 0.92	15%	71%	$\bar{\epsilon} = 10^{-8}$, $\hat{\sigma} = 1$, KL: 0.14
θ_2	0.101	± 0.02	0.156	± 0.09	16%	72%	$act(\epsilon) = 1.2$, NDR: 1.07, SNR: 0.21
(B) 2000 observations, $\Delta t = 0.5$, $\theta^{(0)} = (1, 0.1)$							
θ_1	1.00	± 0.04	1.00	± 0.15	2%	13%	$\bar{\epsilon} = 9 \cdot 10^{-9}$, $\hat{\sigma} = 1$, KL: 0.18
θ_2	0.101	± 0.00	0.101	± 0.015	2%	13%	$act(\epsilon) = 1.0$, NDR: 1.06, SNR: 0.17
(C) 500 observations, $\Delta t = 5$, $\theta^{(0)} = (1, 0.1)$							
θ_1	1.01	± 0.09	1.02	± 0.11	7%	9%	$\bar{\epsilon} = 4 \cdot 10^{-8}$, $\hat{\sigma} = 2.6$, KL: 0.01
θ_2	0.101	± 0.01	0.102	± 0.01	7%	9%	$act(\epsilon) = 6.7$, NDR: 0.91, SNR: 0.50
(D)100 observations, $\Delta t = 10$, $\theta^{(0)} = (0.6, 0.06)$							
θ_1	0.60	± 0.12	0.65	± 0.17	15%	21%	$\bar{\epsilon} = 10^{-8}$, $\hat{\sigma} = 2.7$, KL: 0.03
θ_2	0.061	± 0.01	0.065	± 0.018	16%	22%	$act(\epsilon) = 13.8$, NDR: 0.88, SNR: 0.55



(A) design: $\Delta t = 0.5$ for $\theta^{(0)} = (1, 0.1)$ (B) design: $\Delta t = 10$ for $\theta^{(0)} = (0.6, 0.06)$

Figure 4.5: **Representation of the system's dynamics with the MSS method.** Blue points: data points, blue curve: system's dynamics with MSS method, red dotted lines: residuals for Immigration-Death model with different designs (A) and (B). For the estimation of the parameters 100 observations are used, only the first 50 are shown in the figure.

This example is a proof of concept example as it demonstrates that an estimation with MSS method is possible even in a case which would be structurally unidentifiable for TSS methods [41].
The results of table 4.4 (D) can be compared to [7], where a stochastic gradient descent method is applied to the same setting. The gradients are evaluated with a reversible jump Markov chain Monte Carlo method. Hence this method does not use any approximation and its results are a comparison how much influence the approximation of the MSS method has on the estimation results. As there is only the result of one estimation procedure given a comparison with respect to unbiasedness and variance is not possible. However 4 out of 5 of the relative errors of table 4.4 (D) are smaller than the relative error of [7]. Table 4.4 (B) - (D) also suggest that using the MSS method models become identifiable which are unidentifiable using TSS methods.

4.1.4 Limiting cases

This section will present some of the limiting properties of the functional l, (2.7). The idea behind this is that for large m the functional $F_g^{(m)}$ will be very close to l. Therefore it is also interesting to know about the behavior of l.

Number of observations

In the following the EL method is used. It is based on the function l, equation (2.7), with the exact transition probabilities calculated with the probability generating function in section 2.3.1.

Table 4.5: **Asymptotic properties of l.** Statistics of the estimates using l for 100 simulated time series in dependence on the number of observations. $\theta^{(0)} = (1, 0.1)$, $\Delta t = 0.5$.

numb obs		minimum	10%-quantile	mean	90%-quantile	maximum
100	θ_1	0.64	0.78	1.01	1.25	1.42
	θ_2	0.054	0.080	0.103	0.132	0.167
250	θ_1	0.75	0.85	1.00	1.15	1.30
	θ_2	0.067	0.085	0.101	0.117	0.131
500	θ_1	0.78	0.92	10.2	1.12	1.25
	θ_2	0.084	0.090	0.101	0.109	0.129
1000	θ_1	0.82	0.94	1.00	1.09	1.14
	θ_2	0.086	0.093	0.101	0.109	0.120
10000	θ_1	0.96	0.97	1.00	1.02	1.05
	θ_2	0.096	0.098	0.100	0.103	0.105
100000	θ_1	0.98	0.99	1.00	1.01	1.01
	θ_2	0.099	0.099	0.100	0.101	0.101

For 100 simulated data sets which are most of the time in the steady state phase of ODE modeling the estimates with the EL method are calculated. Table 4.5 presents the minimum, 10%-quantile, mean, 90%-quantile and maximum in dependence of the number of observations, for a visualization see also figure 4.6. In order to increase the speed of the calculation it is especially important to change the order of the summation

$$l(\nu,\theta) \;=\; \sum_{i=1}^{n} \log\left(p_\theta(\nu_i, t_i|\nu_{i-1}, t_{i-1})\right) \;=\; \sum_{(x,y)\in\mathbb{N}_0\times\mathbb{N}_0} \left(\log\; p_\theta(x, t_1|y, t_0) \sum_i 1_{(\{\nu_i=x,\nu_{i-1}=y\})}\right).$$

The reason is that with increasing number of observations the number of identical jumps is strongly increasing whilst the number of jumps which occur the first time is only slightly increasing.

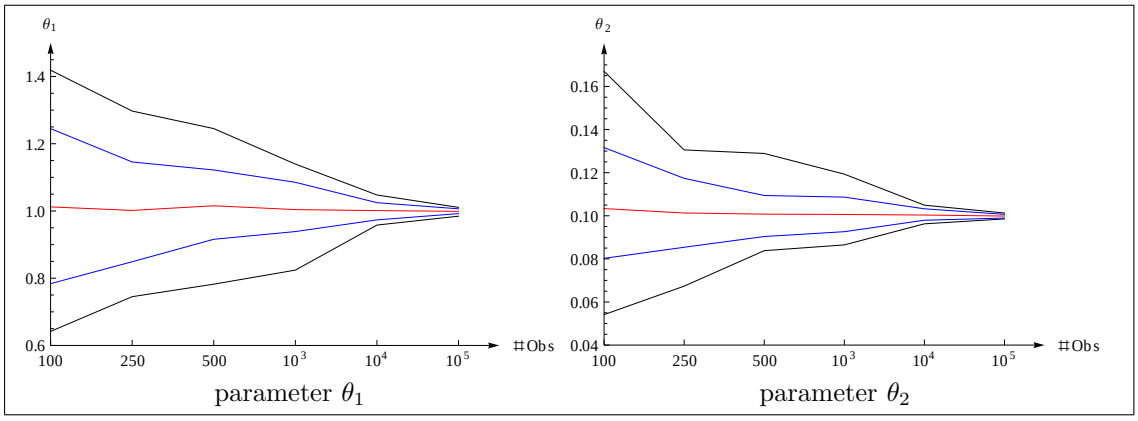


Figure 4.6: **Asymptotic properties of l.** The mean (red line), 10%- and 90%-quantiles (blue lines) and minimum and maximum (black lines) of the 100 estimation from table 4.5.

Number of molecules

In this subsection the EL estimation and the TSS estimation will be compared for different numbers of molecules in the steady state. Therefore different time series are simulated which are not in steady state for the initial value. This is important to make an estimation with the TSS methods possible.
For 100 simulated data sets the estimates using l are calculated and in the table 4.6 the minimum, 10%-quantile, mean, 90%-quantile and maximum are given in dependence of the number of molecules. As with increasing number of molecules the steady state changes, the quotient of the parameters is also increasing. The parameter θ_2 is fixed at 0.1 and the parameter θ_1 is increased in order to achieve the steady state in table 4.6. To compare with a TSS estimation, a TSS estimation is performed using COPASI for the same setting. The estimation results in dependence on the steady state demonstrate that the performance of the TSS estimation is dependent on the number of molecules in the steady state. In contrast to that the estimation with the likelihood function using the probability generating function does not depend on the number of molecules in the steady state. The estimation is performed with the software COPASI [28] using an evolutionary programming algorithm with 2000 iterations and 20 individuals in each generation.
The accuracy of the TSS method increases with increasing number of molecules because the systems behavior can be better and better described with concentrations. For a theoretical argument why this approximation holds see [3], section Langevin Method. The accuracy of the EL method remains constant with respect to an increasing molecule number. Due to computational time the results for the EL method are only calculated up to 500 molecules in the steady state. It would be interesting for further research to see whether the TSS methods performance for high number of molecules is really better (which the table might suggest) than the single molecule-based EL method.

Table 4.6: **Performance of TSS and EL estimation for increasing number of molecules in steady state.** Statistics of 100 estimations with the EL method and TSS method in dependence of the number of molecules in the steady state. 100 observations per simulated data set, $\theta_2 = 0.1$ and $\theta_1 = \frac{x_{steady}}{\theta_2}$. 100 simulated time series of length $T = 50$ and $\Delta t = 0.5$.

x_0	x_{steady}		minimum	10%-quantile	mean	90%-quantile	maximum
5	10	EL	0.064	0.083	0.105	0.128	0.154
		TSS	0	0	0.165	0.342	2.324
10	20	EL	0.074	0.082	0.104	0.125	0.171
		TSS	0	0	0.137	0.301	0.943
25	50	EL	0.072	0.081	0.102	0.122	0.137
		TSS	0	0.025	0.119	0.239	0.470
50	100	EL	0.067	0.081	0.100	0.121	0.147
		TSS	0.010	0.060	0.121	0.198	0.276
100	200	EL	0.075	0.083	0.103	0.121	0.163
		TSS	0.028	0.061	0.107	0.155	0.282
250	500	EL	0.077	0.082	0.099	0.115	0.145
		TSS	0.039	0.076	0.103	0.132	0.186
500	1 000	TSS	0.063	0.078	0.100	0.121	0.146
1 000	2 000	TSS	0.072	0.083	0.099	0.116	0.143
10 000	20 000	TSS	0.092	0.095	0.100	0.105	0.108
100 000	200 000	TSS	0.098	0.098	0.100	0.102	0.103

4.2 Lotka-Volterra model

A second example which is still small but allows an investigation of the behavior in partially observed models is a Lotka-Volterra model:

$$\begin{aligned} X &\xrightarrow{\theta_1} 2\ X \\ X + Y &\xrightarrow{\theta_2} 2\ Y \\ Y &\xrightarrow{\theta_3} \varnothing \end{aligned} \tag{4.1}$$

where X is the prey and Y the predator and $\theta_1, \theta_2, \theta_3$ parameters. The first reaction of equation (4.1) is the prey reproduction, the second the predator reproduction and the third is the predator death. In terms of ODEs this system reads as

$$\begin{aligned} \frac{d[X]}{dt} &= \theta_1[X](t) - \theta_2[X](t)[Y](t) \\ \frac{d[Y]}{dt} &= \theta_2[X](t)[Y](t) - \theta_3[Y](t). \end{aligned}$$

In this example different levels of measurement noise are considered.

4.2.1 Fully observed case with noise

To investigate the behavior of the MSS method with the functional from equation (3.2), 50 data sets are simulated with the Gillespie algorithm using the software COPASI [28]. The true parameter is $\theta^{(0)} = (0.5, 0.0025, 0.3)$ and 40 observations are taken with $\Delta t = 1$. The initial conditions are $(\nu_0^{(1)}, \nu_0^{(2)}) = (71, 79)$. The setting is chosen in a way such that it is identically with [6]. A comparison of the results will be given later in this section. Measurement noise is simulated as follow: For each time point a normally distributed random variable is generated with a given variance. Then it is rounded to the next integer. Hence the variance remains the same. This is done because measurements are assumed to be integer counts. This is not necessary for theoretical reasons or performance of the estimation. Note that due to the added noise negative measurements may occur. As negative molecule counts are impossible they should be corrected to zero. This will be done for $\sigma = 25$, which effectively leads to a measurement error which is not centered around zero. For $\sigma = 10$ they are not corrected to test whether the method is able to handle negative data points to some extent. The results demonstrate that the method is able to handle both situations. For all data sets the objective function is optimized with the software Mathematica [38] with the true parameter as initial value for the optimization and $\theta > 0$ as constraints. The initial value problems are solved with the *NDSolve*-routine. For details of this routine refer to the Mathematica documentation. The average computing time on a personal computer as in section 4.1.1 was approximately 3 minutes. Table 4.7 gives the mean and standard deviation of the 50 estimation results using equation (3.2). For each estimation result the relative error is calculated. The MSS method works very well in this example. The relative error of the estimation is in the range of the relative error of the method proposed in [6], where only one data set is used for estimation. The mean of the residuals is close to zero and the autocorrelation time small compared to the total duration of observations. The signal to noise ratio is much better than in the Immigration-Death case. For $\sigma = 25$ the mean of the residuals is not close to zero but considering the means of the 10% and 90%-quantiles, -90 and 47, one sees that the residuals still are small.

Table 4.7: **Estimation results for Lotka-Volterra model.** Mean, standard deviation, average relative error and test functions of estimation results for 50 simulated data sets with 40 observations with $\Delta t = 1$ and true parameter $\theta^{(0)} = (0.5, 0.0025, 0.3)$ for the Lotka-Volterra model, $\nu_0 = (71, 79)$.

	Estimation Results		Rel Err	Test functions (averages)
exact measurements				
θ_1	0.501	± 0.016	2.5%	$\bar{\epsilon} = (0.16, 0.07)$, $\hat{\sigma}_{KL} = (12, 11)$,
θ_2	0.00250	$\pm 7 * 10^{-5}$	2.2%	KL: $(0.2, 0.1)$, $act(\epsilon) = (1.5, 1.8)$,
θ_2	0.301	± 0.011	3.1%	NDR: $(0.24, 0.18)$, SNR: $(4.3, 5.8)$
noise: $\sigma = 10$				
θ_1	0.490	± 0.019	3.2%	$\bar{\epsilon} = (0.89, 0.62)$, $\hat{\sigma}_{KL} = (19, 18)$,
θ_2	0.00248	$\pm 9 * 10^{-5}$	2.9%	KL: $(0.1, 0.1)$, $act(\epsilon) = (1, 1)$,
θ_2	0.302	± 0.012	3.4%	NDR: $(0.42, 0.34)$, SNR: $(2.2, 2.9)$
noise: $\sigma = 25$				
θ_1	0.454	± 0.031	9.7%	$\bar{\epsilon} = (3.48, 2.52)$, $\hat{\sigma}_{KL} = (40, 40)$,
θ_2	0.00243	$\pm 15 * 10^{-5}$	5.2%	KL: $(0.04, 0.04)$, $act(\epsilon) = (1, 1)$,
θ_2	0.301	± 0.021	5.6%	NDR: $(0.68, 0.60)$, SNR: $(1.1, 1.3)$

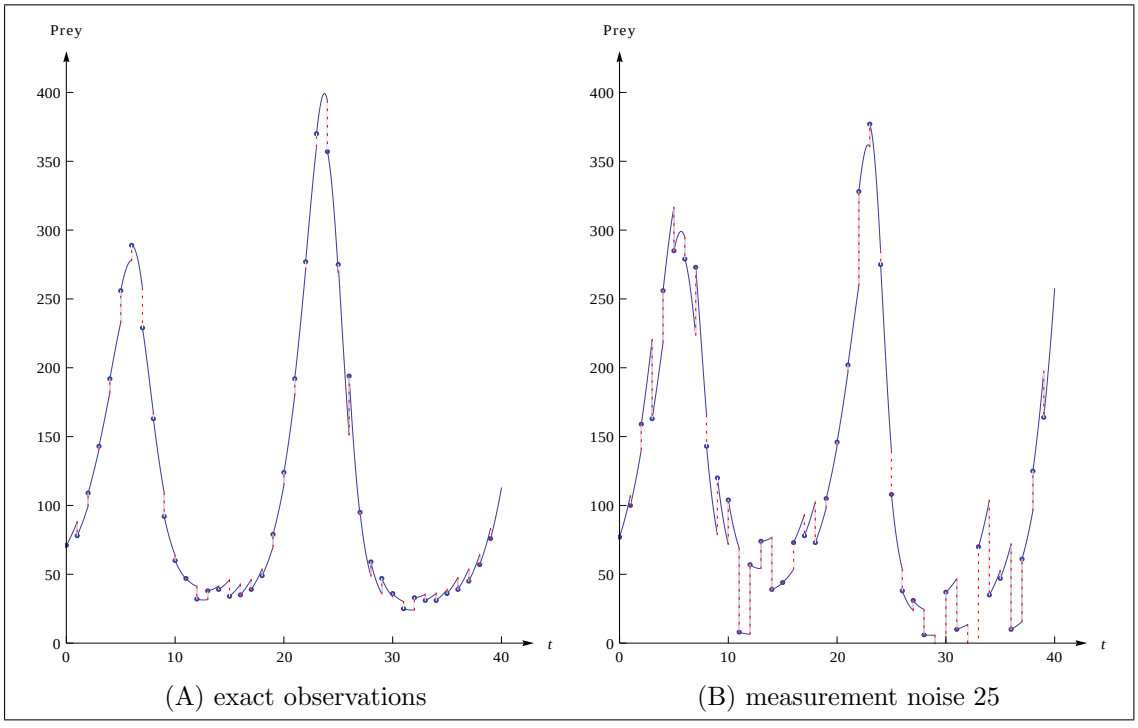
4.2.2 Partially observed Lotka-Volterra model with noise

Now assume that only prey can be observed. As in the completely observed case simulated data with true parameter $\theta^{(0)} = (0.5, 0.0025, 0.3)$ and 40 observations with $\Delta t = 1$ is used. Initial condition is again $(\nu_0^{(1)}, \nu_0^{(2)}) = (71, 79)$ but as only prey can be observed only $\nu_0^{(1)} = 71$ will be used as data point for the parameter estimation. $\nu_0^{(2)}$ will be a variable for the optimization, hence there are four variables for the optimization now: $\left(\theta_1, \theta_2, \theta_3, \nu_0^{(2)}\right)$. Noise is simulated as in the previous subsection. The optimization is performed with a particle swarm program (figure 2.8) implemented in Mathematica with 100 iteration with 25 particles on a range of $[0.3, 0.7] \times [0.001, 0.005] \times [0.2, 0.4] \times [25, 150]$. The average computing time on a personal computer as in section 4.1.1 was approximately 3 minutes. Table 4.8 gives the mean and standard deviation of the 50 estimation results using (3.3) with $K = \{0\}$. For each estimation result the relative error is calculated. The estimation performs still quite well even if only one species is observed. The residuals are still approximately normally distributed as in the fully observed case. Negative measurements due to the added measurement noise are treated as in the fully observed case, which leads to the data points with value zero in figure 4.7. Instead of $\nu_0^{(1)} = 71$ the noisy value for $\nu_0^{(1)}$ is utilized.

The same model is used for parameter estimation in stochastic models by Boys et al. [6], where a Bayesian approach is used in combination with a reversible jump method for the evaluation of the likelihood function. As previously mentioned the special choice of true parameters and initial conditions in this chapter is made to allow for comparison with these results. In two out of three cases the relative error with the MSS method lies below the smallest relative error of the methods suggested by Boys et al. [6], where one estimation result per method is given.

Table 4.8: **Estimation results for partially observed Lotka-Volterra model, prey observed**: Mean, standard deviation, average relative error and test functions of estimation results for 50 simulated data sets with 40 observations with $T = 40$ and true parameter $\theta^{(0)} = (0.5, 0.0025, 0.3)$, only prey can be observed, $\nu_0 = (71, 79)$.

	Estimation Results		Rel Err	Test functions (averages)
exact measurements				
θ_1	0.501	± 0.054	8.8%	$\bar{\epsilon} = -0.1$, $\hat{\sigma}_{KL} = 12$,
θ_2	0.0026	$\pm 3 * 10^{-4}$	11.1%	KL: 0.1 , $act(\epsilon) = 1.5$,
θ_2	0.312	± 0.048	13.4%	NDR: 0.2, SNR: 4.4
noise: $\sigma = 10$				
θ_1	0.478	± 0.050	9.3%	$\bar{\epsilon} = 0.7$, $\hat{\sigma}_{KL} = 20$,
θ_2	0.0026	$\pm 3 * 10^{-4}$	10.6%	KL: 0.04 , $act(\epsilon) = 1$,
θ_2	0.318	± 0.045	13.4%	NDR: 0.4, SNR: 2.3
noise: $\sigma = 25$				
θ_1	0.430	± 0.070	16.6%	$\bar{\epsilon} = 3.1$, $\hat{\sigma}_{KL} = 19$,
θ_2	0.0028	$\pm 4 * 10^{-4}$	16.8%	KL: 0.04 , $act(\epsilon) = 1$,
θ_2	0.336	± 0.048	16.5%	NDR: 0.65, SNR: 1.1



(A) exact observations (B) measurement noise 25

Figure 4.7: **Representation of the system's dynamics with the MSS method for Lotka-Volterra model.** Blue points: data points, blue curve: system's dynamics with MSS method, red dotted lines: residuals for partially observed Lotka-Volterra model without (A) and with (B) measurement noise.

4.3 Calcium oscillation model

Third model is a Calcium oscillation model [22]:

$$\begin{aligned}
\frac{d\ [g]}{dt} &= \theta_1 + \theta_2[g](t) - \frac{\theta_5\ [g](t)\ [plc](t)}{[g](t)+\theta_6} - \frac{\theta_7\ [g](t)\ [plc](t)}{[g](t)+\theta_8} \\
\frac{d\ [plc]}{dt} &= \theta_3[g](t) - \frac{\theta_9\ [plc](t)}{[plc](t)+\theta_{10}} \\
\frac{d\ [ca]}{dt} &= \theta_4[g](t) - \frac{\theta_{11}\ [ca](t)}{[ca](t)+\theta_{12}}.
\end{aligned} \tag{4.2}$$

where ca stands for the cytosolic Calcium, g for the active subunit of the G-protein and plc for the activated form of PLC [22].
The behavior of this model differs qualitatively between stochastic and ODE modeling for small particle numbers as presented in [22] and in figure 1.2. In this model the estimation of the transition probabilities is a challenge due to the large state space.

4.3.1 Fully observed model

To investigate the behavior of the MSS method with objective functional from equation (3.2), 50 data sets are simulated with the Gillespie algorithm using the software COPASI [28]. The true parameter and initial condition with the units ml, s and particles (#) and with a compartment volume of 1 ml are:

$$\begin{aligned}
\theta^{(0)} &= (212, 2.95, 1.52, 190, 4.88, 1180, 1.24, 32240, 29090, 13.58, 153000, 160) \\
(ca_0, g_0, plc_0) &= (10, 10, 10).
\end{aligned}$$

Note that for this set of parameters the system oscillates irregularly with large amplitudes modeled stochastically but oscillates regularly with small amplitudes modeled with ODEs. 100 observations are taken with $\Delta t = 0.5$, which cover about 4 oscillation cycles. For all data sets the objective function (3.2) is optimized using a Particle Swarm program (see figure 2.8) implemented in the software Mathematica [38] with 500 iteration with 100 particles on a range of

$$\begin{aligned}
&[150, 250] \times [2, 4] \times [0.5, 2.5] \times [100, 250] \times [0, 10] \times [500, 3000] \times [0.5, 2.5] \\
&\times [20000, 40000] \times [20000, 40000] \times [10, 20] \times [100000, 200000] \times [100, 200].
\end{aligned}$$

This relatively small range is chosen to focus on the objective function and not on optimization issues such as local minima. The initial value problems are solved with the *NDSolve*-routine of Mathematica as in section 4.2.1. The average computation time was 3.5 hours. It should be stated that this can be reduced drastically using efficient methods as mentioned at the end of section 3.1. Using stochastic simulations each evaluation of $F_g^{(m)}$, equation (2.11), with $m = 3$ would last approximately 35 seconds. Considering that $m = 3$ only gives a very rough estimate of the transition probabilities and that several thousand functional evaluations are necessary this is just too slow.
Table 4.9 gives mean and standard deviation of the 50 estimation results. For each estimation result the relative error is calculated. The estimation performs quite successfully. The parameter θ_2, which determines the oscillatory behavior of the system, has very small relative error.

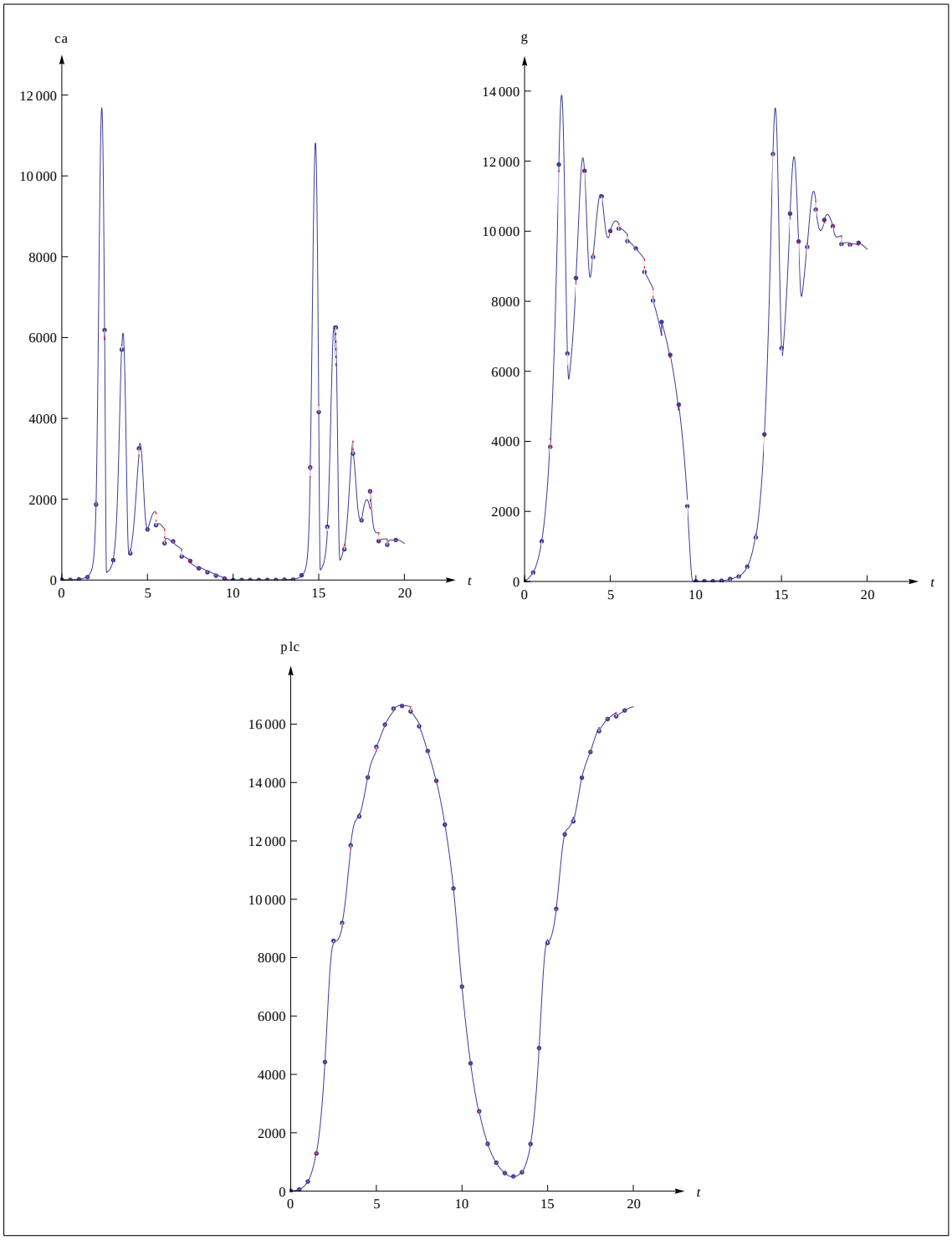


Figure 4.8: **Representation of the system's dynamics with the MSS method for the fully observed Calcium oscillation model.** Blue points: data points, blue curve: system's dynamics with the MSS method, red dotted lines: residuals for fully observed Calcium oscillation model. 100 observations are used for the estimation of the parameters, only the first 40 are shown in the figure. The fact that the red dotted lines can hardly be seen underlines that the systems is very well represented.

Table 4.9: **Estimation results for fully observed Calcium oscillation model**. Mean, standard deviation and average relative error of estimation results for 50 simulated data sets of 100 observations with $T = 50$ for the Calcium oscillation model, $(ca_0, g_0, plc_0) = (10, 10, 10)$.

	True param	Estimation results		Rel err
θ_1	212	219	±32	13.4%
θ_2	2.95	2.94	±0.03	0.74%
θ_3	1.52	1.52	±0.02	0.87%
θ_4	190	192	±52	24.1%
θ_5	4.88	4.93	±0.30	5.02%
θ_6	1180	1381	±747	52.5%
θ_7	1.24	1.24	±0.01	0.57%
θ_8	32240	32390	±2013	4.08%
θ_9	29090	29335	±2766	5.95%
θ_{10}	13.58	13.63	±0.39	2.30%
θ_{11}	153000	153587	±4606	2.40%
θ_{12}	160	162.6	±7.0	3.46%

To investigate how well the method covers the stochastic dynamics the test functions are evaluated with the optimal parameters for all data sets, the averages are given in the following. Calculating the residuals yields for 10%-quantile, mean and 90%-quantile: *ca*, $\{-156, 0.09, 150\}$; *g*, $\{-246, -0.8, 226\}$; *plc*, $\{-123, -0.3, 117\}$. The estimated variances are $\hat{\sigma}_{KL} = (154, 195, 97)$ with a KL divergence value of $(0.49, 0.09, 0.05)$. The autocorrelation times, $ca: 0.85$, $g: 1.01$, $plc: 0.95$, are much smaller than the total observation time. The NDR are $(0.1, 0.2, 0.1)$, SNR are $(9.7, 8.8, 14.2)$, which states that the system's dynamic is well represented, see also figure 4.8, which shows for Calcium that the residuals (red dotted lines) are small compared to the ODE system's dynamics (blue line).

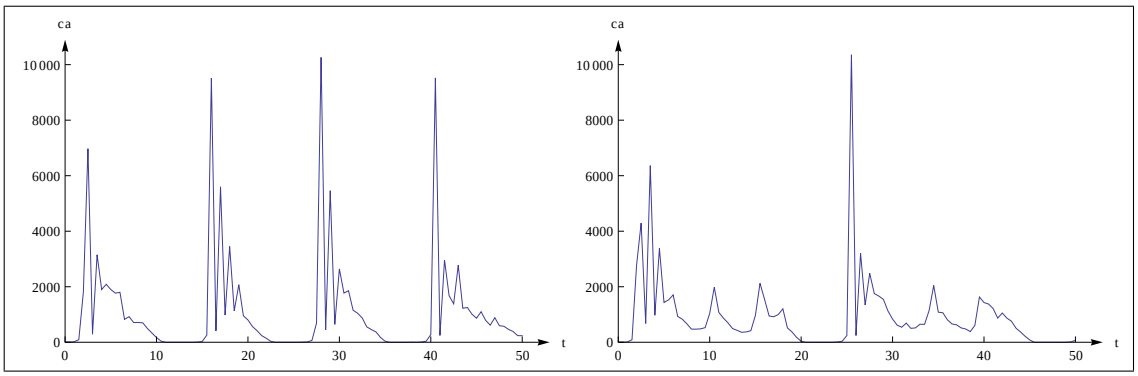


Figure 4.9: **Widely varying amplitudes of two recorded simulated data in *ca*-oscillations** due to the intrinsic stochasticity and due to the fact that the peak of the oscillation is not always recorded because of the length of the time interval between two successive points of measurements.

To emphasize the strong influence of the stochasticity figure 4.9 shows two different *ca* time courses. The amplitude of the recorded simulated data varies widely, which is due to the fact that the oscillations have different amplitudes because of the stochasticity and because of the fact that the

time interval between two successive measurements is so large that the peak of the oscillation is not always recorded.

4.3.2 Partially observed Calcium oscillation model

Time course data for g

At first assume that only g can be observed. As in the completely observed case simulated data with true parameter $\theta^{(0)}$ and 100 observations with $\Delta t = 0.5$ is used. Initial condition is again $(ca_0, g_0, plc_0) = (10, 10, 10)$ but as only g can be observed only $g_0 = 10$ will be used as data input for the parameter estimation. The ODE system is structurally unidentifiable in this case. An increase in ca or plc can be "compensated" with a change of the parameter values and lead to exactly the same systems behavior. To illustrate this assume that instead of an amount $plc(t)$ in the system there is a multiple of this amount present: $\widetilde{plc}(t) = \alpha\ plc(t)$ with $\alpha > 0$. Adjust θ_7 to $\frac{\theta_7}{\alpha}$. Next consider the second ODE of (4.2), which is now in terms of $\widetilde{plc}$:

$$\begin{aligned}
\frac{d\ [\widetilde{plc}]}{dt} &= \theta_3[g](t) - \frac{\theta_9\ [\widetilde{plc}](t)}{[\widetilde{plc}](t) + \theta_{10}} \\
\frac{d\ (\alpha\ [plc])}{dt} &= \theta_3[g](t) - \frac{\theta_9\ \alpha\ [plc](t)}{\alpha\ [plc](t) + \theta_{10}} \\
\frac{d\ [plc]}{dt} &= \frac{\theta_3}{\alpha}[g](t) - \frac{\theta_9\ [plc](t)}{\alpha\ [plc](t) + \theta_{10}} \\
\frac{d\ [plc]}{dt} &= \frac{\theta_3}{\alpha}[g](t) - \frac{\frac{\theta_9}{\alpha}\ [plc](t)}{[plc](t) + \frac{\theta_{10}}{\alpha}},
\end{aligned}$$

which shows that the system can be traced back to the old system with adjusted parameter values. This can be done in a similar way for the amount of ca. The result is that the solution g of the system has the same behavior for a set of combinations of initial conditions and parameters. The manifold $\{\rho(\theta, ca_0, plc_0), ca_0, plc_0 | ca_0, plc_0 > 0\}$ is two-dimensional and described with the function ρ:

$$\begin{aligned}
&\rho(\theta, ca_0, plc_0) \\
&\quad = (\theta_1,\ \theta_2,\ \frac{\theta_3}{plc_0},\ \theta_4,\ \frac{\theta_5}{ca_0},\ \theta_6,\ \theta_7 plc_0,\ \theta_8 plc_0,\ \theta_9 plc_0,\ \theta_{10} ca_0,\ \theta_{11} ca_0,\ \theta_{12} ca_0).
\end{aligned}$$

Therefore (ca_0, plc_0) can be fixed at an arbitrary value for the optimization leading to a single result $\hat{\theta}$, which spans the manifold. Due to that the optimization problem remains 12-dimensional although the model is only partially observed.
The optimization is again performed with a Particle Swarm (figure 2.8) implemented with Mathematica using 250 iterations with 50 particles per iteration on a range of

$$\begin{aligned}
&[100, 500] \times [2, 4] \times [1, 3] \times [1, 1000] \times [0.1, 10] \times [100, 10000] \times [0.1, 10] \\
&\times [10000, 100000] \times [10000, 100000] \times [10, 20] \times [100000, 200000] \times [10, 500].
\end{aligned}$$

The small range is again chosen to focus on the objective function and not on optimization issues such as local minima. The average computing time was 0.75 hours, which is less than in the fully observed case as the number of iterations and particles used in the Particle Swarm is smaller. Determining an optimal number of particles is an issue for further research. Determining an optimal number of iterations means the implementation of a suitable stopping criterion. As in the previous

subsection it should be noted that the use of efficient numerical optimization methods as in [19] would be most time saving.
Table 4.10 gives mean and standard deviation of the 50 estimation results using (3.3) with $(ca_0, plc_0) = (1, 1)$. To calculate the relative errors with respect to the true parameter the estimate parameter is transformed with respect to the true initial conditions: $\rho(\hat{\theta}, 10, 10)$. This transformation along the manifold does not change the value of the objective function. The results depict a quite good estimation. Especially the parameter θ_2, which determines whether the system oscillates, is estimated with very low relative error. Some other parameters still can not be estimated. Further research has to be carried out to determine the underlying reasons.
To investigate how well the method covers the stochastic dynamics the test functions are evaluated with the optimal parameters for all data sets, the averages are given in the following. Calculating the averages of the residuals yields the following numbers for 10%-quantile, mean and 90%-quantile: g, $\{-285, -2.7, 270\}$. The estimated variance is $\hat{\sigma}_{KL} = 232$ with a KL divergence value of 0.07. The autocorrelation time 0.5 is much shorter than the total observation time. Figure 4.10 shows that the system's dynamic is well represented, SNR is 7.1 and NDR is 0.14, which demonstrates that the estimation is possible even with only one observed species.

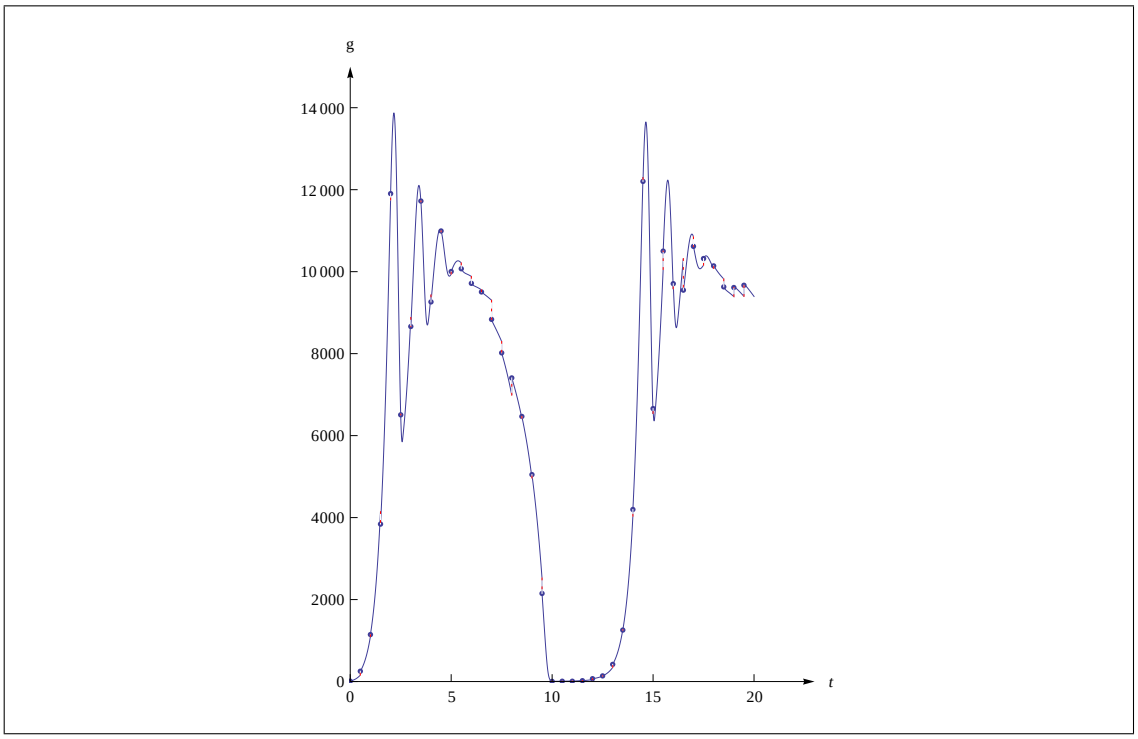


Figure 4.10: **Representation of the system's dynamics with the MSS method for the partially observed (g) Calcium oscillation system.** Blue points: data points, blue curve: system's dynamics with MSS method, red dotted lines: residuals for partially observed Calcium oscillation model observing g. 100 observations are used for the estimation of the parameters, only the first 40 are shown in the figure.

Table 4.10: **Estimation results for partially observed Calcium oscillation model: g.** Mean, standard deviation and average relative error of estimation results for 50 simulated data sets of 100 observations with $T = 50$ for the partially observed Calcium oscillation model: only g observed.

	True param	Estimation results		Rel err
θ_1	212	293	$\pm$100	52%
θ_2	2.95	2.97	$\pm$0.05	1.8%
θ_3	1.52	1.88	$\pm$0.6	36%
θ_4	190	410.6	$\pm$178	123%
θ_5	4.88	4.86	$\pm$0.67	10.7%
θ_6	1180	1610	$\pm$1332	86%
θ_7	1.24	1.10	$\pm$0.39	28.4%
θ_8	32240	48216	$\pm$11676	51%
θ_9	29090	56510	$\pm$17299	94%
θ_{10}	13.58	14.26	$\pm$1.67	10.3%
θ_{11}	153000	160625	$\pm$18698	10.3%
θ_{12}	160	167.0	$\pm$29.7	15.4%

Time course data for Calcium

Observing Calcium in time course data is the realistic case from biological point of view. The system is again structurally unidentifiable in this case. The reason is the same as described in the previous subsection. Instead of ca the amount of g can now vary. The manifold $\{(\rho(\theta, g_0, plc_0), g_0, plc_0) \mid g_0, plc_0 > 0\}$ is two-dimensional and described with the function ρ:

$$\rho(\theta, g_0, plc_0) \;=\; (\theta_1 g_0,\ \theta_2,\ \frac{\theta_3 g_0}{plc_0},\ \theta_4 g_0,\ \theta_5 g_0,\ \theta_6 g_0,\ \frac{\theta_7 plc_0}{g_0},\ \theta_8 plc_0,\ \theta_9 plc_0,\ \frac{\theta_{10}}{g_0},\ \theta_{11},\ \theta_{12}).$$

Therefore for the optimization (g_0, plc_0) can be fixed leading to a single result $\hat{\theta}$, which spans the manifold. Due to that the optimization problem remains 12-dimensional although the model is only partially observed. But this situation leads to difficulties. An estimation is now performed with the functional (3.3) with $(g_0, plc_0) = (1, 1)$ as it is done for case where g is observed. After a transformation to $\rho(\hat{\theta}, 10, 10)$ due to the true initial conditions as in the previous subsection yields for one example to an estimate of

$$\hat{\theta} = (45.8, 2.46, 1.46, 501, 7.26, 71.7, 1.89, 54833, 10862, 23.6, 106135, 1.0),$$

which is far away from the true parameter. Again the Particle Swarm (figure 2.8) was used with 250 iterations and 100 particles on a range of

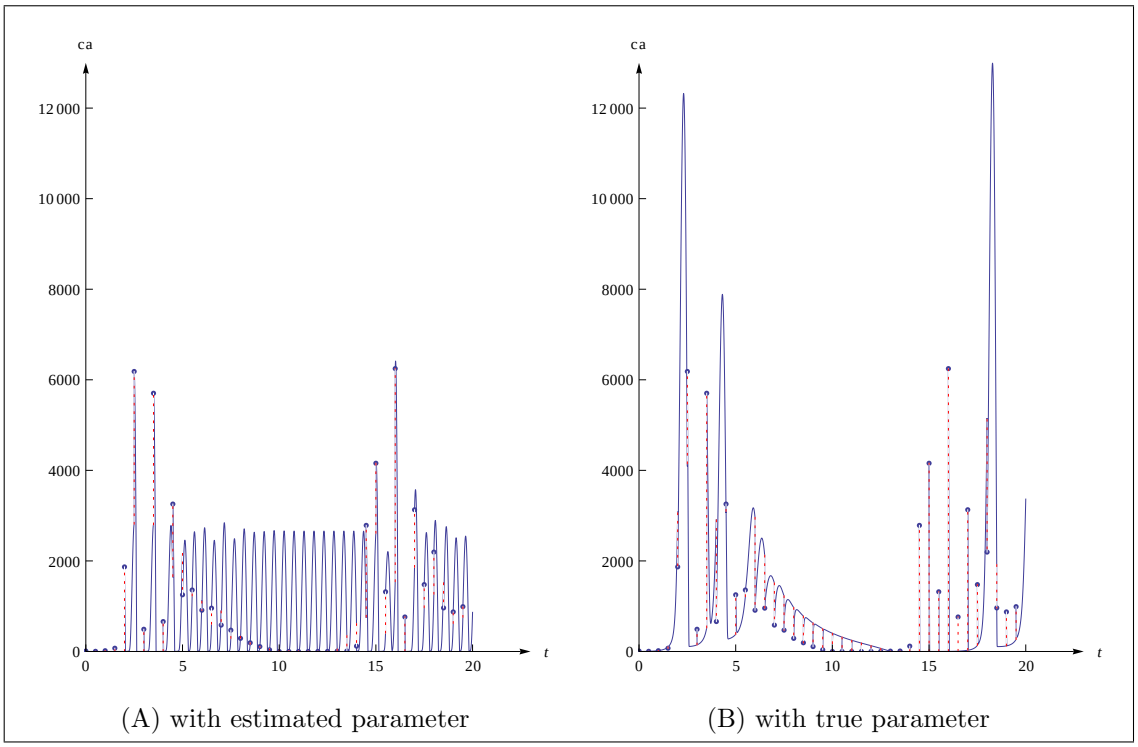
$$[10, 1000] \times [2, 4] \times [1, 2] \times [10, 1000] \times [1, 100] \times [10, 10000] \times [0.1, 2]$$
$$\times [10000, 100000] \times [10000, 35000] \times [1, 100] \times [10000, 200000] \times [1, 200].$$

To investigate how well the method covers the stochastic dynamics the test functions are evaluated with the optimal parameter. The residuals yield as above with 10%-quantile, mean and 90%-quantile: ca, $\{-469, 376, 1621\}$, which seems to question the hypothesis of centered normally distributed residuals. The estimated variance is $\hat{\sigma}_{KL} = 1203$ with a KL divergence value of 1.02. The autocorrelation time is 3, $SNR(\nu_{(ca)}, \hat{\theta}) = 1.05$ and $NDR(\nu_{(ca)}, \hat{\theta}) = 0.65$ and the system's

dynamics are not at all represented, see also figure 4.11 (A). With the knowledge of the true parameter one can identify the problem: For a short time interval the system's dynamics are well represented but then due to the development of the unobserved species it is not well represented any longer, see figure 4.11 (B). Hence in this case it is better to enlarge the optimization vector and use functional (3.3) with $K = \{0, 5, 10, \ldots, 45\}$, which means that also unobserved states at other time points than zero are included in the optimization vector. Again $(g_0, plc_0) = (1, 1)$ is fixed for t_0 and the results transformed to $\rho(\hat{\theta}, 10, 10)$.
The optimization is performed again with the Particle Swarm (figure 2.8) with 250 iterations and 250 particles on a range of

$$[10, 1000] \times [2, 4] \times [1, 2] \times [10, 1000] \times [1, 100] \times [10, 10000] \times [1, 2]$$
$$\times [10000, 100000] \times [10000, 35000] \times [1, 100] \times [100000, 200000] \times [10, 200].$$

For the unobserved states included in the optimization (g_{t_j}, plc_{t_j}), $0 \neq t_j \in K$, the range for the optimization is $[0.9g_{t_j}, 1.1g_{t_j}] \times [0.9plc_{t_j}, 1.1plc_{t_j}]$. This is done to focus on the objective function and not on optimization issues such as local minima. The average computing time was approximately 4 hours, again the remark stated in the fully observable case applies here.
The results demonstrate that an estimation is possible with this functional, see table 4.11.



(A) with estimated parameter (B) with true parameter

Figure 4.11: **Representation of the system's dynamics with the MSS method for the partially observed (*ca*) Calcium oscillation system I.** Blue points: data points, blue curve: system's dynamics with MSS method, red dotted lines: residuals for partially observed Calcium oscillation model observing *ca*. (A) with an estimated parameter and (B) with the true parameter. 100 observations are used for the estimation of the parameters, the first 40 are shown in the figure. Around $t = 15$ it seems that the blue lines do not start at the blue points. This is only due to graphical resolution.

Table 4.11: **Estimation results for partially observed Calcium oscillation model:** *ca*. Mean, standard deviation and average relative error of estimation results for 50 simulated data sets of 100 observations with $T = 50$ for the partially observed Calcium oscillation model, only *ca* observed.

	True param	Estimation results		Rel err
θ_1	212	145.4	±115.1	56.2%
θ_2	2.95	3.06	±0.39	9.9%
θ_3	1.52	1.49	±0.22	11.5%
θ_4	190	212.3	±200.5	76.5%
θ_5	4.88	7.85	±5.58	70.4%
θ_6	1180	1944	±2405	144.4%
θ_7	1.24	1.48	±0.20	21.7%
θ_8	32240	33860	±8578	20.9%
θ_9	29090	23670	±6278	23.7%
θ_{10}	13.58	15.12	±5.55	24.3%
θ_{11}	153000	161206	±22965	13.0%
θ_{12}	160	101	±56.9	43.0%

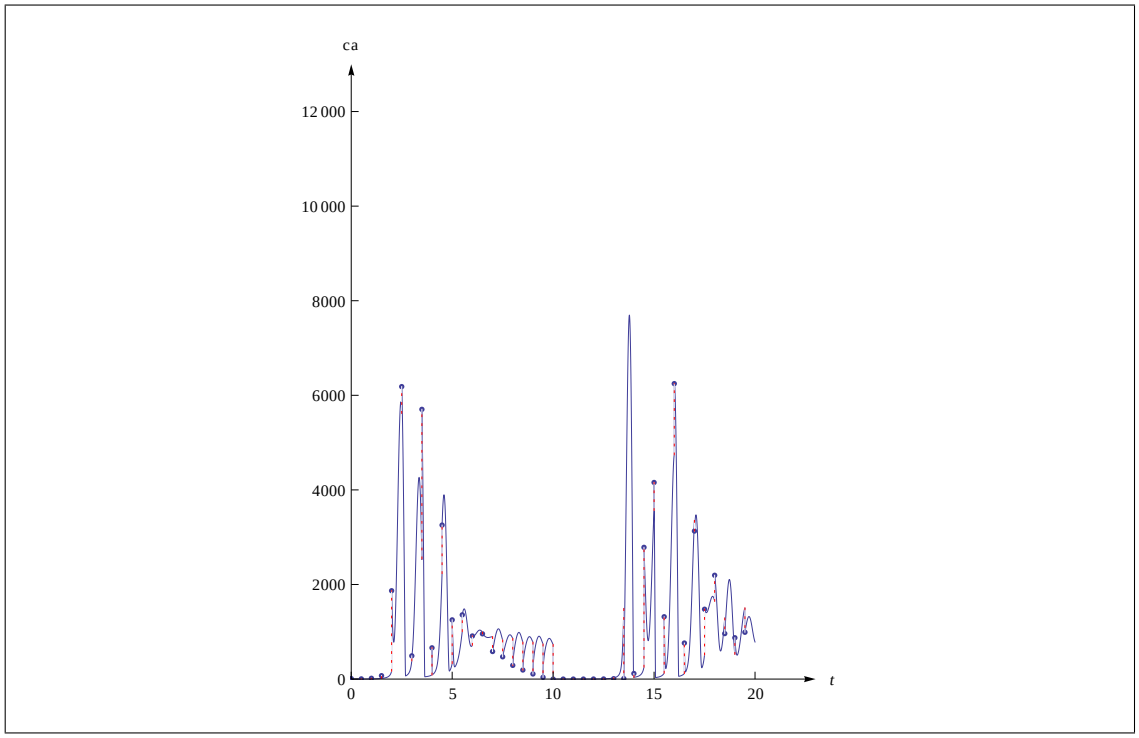


Figure 4.12: **Representation of the system's dynamics with the MSS method for the partially observed (*ca*) Calcium oscillation system II.** Blue points: data points, blue curve: system's dynamics with MSS method, red dotted lines: residuals for partially observed Calcium oscillation model observing *ca* with the objective function (3.3). 100 observations are used for the estimation of the parameters, only the first 40 are shown in the figure.

Now the situation has improved. To investigate how well the method covers the stochastic dynamics the test functions are evaluated with the optimal parameters for all data sets, the averages are given in the following: 10%-quantile, mean and 90%-quantile of the residuals are $(-1997, 162, 1214)$. The estimated variance is $\hat{\sigma}_{Kl} = 1823$ with a KL divergence value of 0.67. The autocorrelation time is 1.16 and the SNR is 1.7 and NDR 5.2. The relatively high value of the NDR is due to the fact that it is small on most intervals of the form $[t_j, t_j + 5]$ with $t_j \in K$ and very high on very few intervals. This means that the other intervals give enough information for the estimation but the NDR value is influenced by this single high value because the average is not resistant to outliers. Figure 4.12 shows that the system's dynamic is well represented.
The unidentifiability is caused by an identical system's dynamics of the ODE system on a set of parameters and initial conditions. There is a method for tackling identifiability problems caused by this situation [60] using the intrinsic fluctuations. It is suggested for further research to investigate its impact on the Calcium identifiability problem.
This example is highly important as it demonstrates that the short time ODE integration method can still be used if the system's behavior is very different in ODE modeling or stochastic modeling. The method is even applicable to partially observed models.

Discussion and conclusion

The thesis investigates methods for parameter estimation for stochastic models. In particular it presents a fast method (MSS method) motivated by multiple shooting for the ODE integration. The MSS method is able to estimate parameters from partially observed stochastic time course data with measurement noise even in models which behave qualitatively different in stochastic modeling and in ODE modeling. The method using the MLS function with simulations to estimate the transition probabilities of the likelihood function is much more cost-intensive from computational point of view. Therefore it can be seen as an addition for situations where the test functions advise that the MSS method does not cover the stochasticity very well. Amongst the investigated examples this happened only in the Immigration-Death example under certain designs where only very few reactions occurred.

Before presenting the MSS method the thesis also investigates methods based on stochastic simulations. The first simulation-based method uses a least squares functional, well known from ODE parameter estimation, evaluated with stochastic simulations. Although being clear from a statistical point of view that the Gaussian distribution assumption does not hold for stochastic models it was not clear how strong this would affect the estimation. To get a result not based on simulations and stochastic simulated data sets the concept of the probability generating function is used to derive a solution of the CME for an Immigration-Death model. This is done by building a power series of the probabilities for the states and verifying that the power series satisfies a partial differential equation. This can be solved analytically in this specific example. This leads to the solution of the CME. Applied to the objective function this leads to the conclusion that the average landscape of the objective function is biased.

Therefore the likelihood function in which the transition probabilities are estimated using simulations (LS function) is more appropriate for the estimation of model parameters in stochastic systems. Again the expectation value of the landscape of the objective function is calculated using the concept of the probability generating function. The behavior is unbiased. For the estimation of the transition probabilities also other more sophisticated methods could be applied [4, 6], nevertheless large state spaces remain a challenge. Especially for this situation a modification is suggested with the MLS function to cope with cases where estimates of the transition probabilities could be zero due to simulation effects.

The optimization of the resulting stochastic landscape can be done using black box optimizers such as Particle Swarm, for which a stopping criterion that takes into account the stochasticity is suggested. This stopping criterion could be also useful for other global optimization algorithms applied to stochastic problems. Another approach is a transformation to a deterministic landscape with the polynomial chaos expansion. This is normally used for deterministic functions and here extended to stochastic functions.

As the MSS method did not represent the stochasticity very well for certain designs in the Immigration-Death model and therefore led to biased estimates, the MLS function is tested in this scenario and compared to exact results with the EL method gained by the application of the probability generating function. Using a Particle Swarm algorithms (section 2.4.2) demonstrates that the parameters can be estimated with a similar relative error to the true parameter as with the EL method. Applying the stopping criterion proposed in figure 2.9 leads to a significant reduction in the number of function evaluations. This is an important improvement as the function evaluations using stochastic simulation are time intensive from a computational point of view. With the stopping criterion the computing time was 133 minutes with averagely 414 function evaluations (see section 4.1.1 for details). Using the polynomial chaos expansion the number of function evaluations can be further reduced to 132 function evaluations but to this end a zoom in procedure has to be carried out manually (section 4.1.2). An automatization is not yet successful (table 4.3). Therefore the application of the polynomial chaos seems to be more an expert tool for gaining information concerning the structure of the landscape than a tool being ready for implementation. Further investigating the properties of the likelihood function – to which the MLS function $F_g^{(m)}$ (2.11) should be close if enough simulations are performed – demonstrates that the relative error decreases with increasing numbers of observations. For increasing number of molecules the performance of the TSS method improves while the performance of the EL method remains constant.

For larger models with larger state spaces all simulation-based methods become very time-consuming. Therefore chapter 3.1 suggests the MSS method calculating the residuals between the ODE dynamics of the system and the stochastic data points on short time intervals. The MSS method is able to estimate parameters even in models which have a qualitatively different behavior in stochastic modeling than in ODE modeling [22]. The advantage of the MSS method is that it does not need stochastic simulations nor a solution of a high-dimensional CME, which increases its speed. Hence it is well applicable in larger realistic size models, which would be very time-consuming in simulation-based methods.
The approximation with an ODE model on a short time interval does not pose a problem as the test functions demonstrate that the approximation works even in models with qualitatively different behavior in stochastic modeling than in ODE modeling. For this it is important that the approximation is only done on a relatively short time interval. In models with very few reactions per time interval such as the Immigration-Death model the signal to noise ratio is bad so that many observations are necessary or the inter sample distance has to be increased (table 4.4 (C)), which underlines the importance of experimental design. But in many realistic models it is not possible to measure fast enough to capture every single reaction so this fact does not reduce the applicability of the method much. Especially in oscillatory systems such as Lotka-Volterra or the Calcium oscillation model, in which a larger state space makes other approaches more time-consuming, the MSS method proposed in this thesis performs very well.
Therefore the use of a moment-closure approach [11] as mentioned in section 3.1 with only the first moment does not seem to be necessary. Whether a moment-closure with higher moments would be advantageous in cases with poor test function values is up to further research.

The results for the Immigration-Death model demonstrate that the MSS method yields an unbiased result compared to an exact analytical estimation with the EL method if there are enough measurements or the measurements are placed considering the concept of experimental design. An analysis how an optimum experimental design could be calculated for stochastic models is suggested for further research. The standard deviation of the MSS method seems to be slightly higher than the standard deviation of the EL method. Compared to a stochastic approach by Wang et al. [7], the suggested MSS method performed well: the results for sample data sets were more accurate in

80% of the 50 sample data sets, which still might be due to the stochastic effects as [7] provided a single data set. The big advantage of the MSS method is the fast computation, which took in average 0.05 seconds per sample data set. The case of table 4.4 (A), in which the MSS method is biased, is a situation when only very few reactions happen per time interval between two points of measurements as can be detected with the test functions. In reality this is generally not the case. If the system which is under investigation shows that behavior it is suggested to use CME-based methods, which will work fast in that case due to the very small state space. An alternative is a method using the ODE steady state information in combination with the stochastic fluctuations. This work has also been part of the parameter estimation for stochastic models project and its manuscript is in preparation [60].
The results of the Lotka-Volterra model demonstrate that the MSS method is well able to provide estimates for fully and partially observed models within the accuracy of methods using stochastic simulations [6]. Further simulated data sets – not shown here – containing time courses in which the species prey dies out indicate that the MSS method handles even such cases successfully.
The Calcium oscillation model was a very important test case and the MSS method showed good performance in estimating the parameters although the system's behavior is completely different in stochastic and deterministic modeling. Figure 4.9 shows that even the amplitudes of the oscillations for a single time course vary widely. This is due to two reasons: the stochasticity causes different amplitudes and the distance between two succeeding measurements leads to the effect that the peak of the oscillation is not always recorded. Computing times are in the order of a few hours and can be reduced by using efficient numerical optimization methods. Tackling this model with methods using stochastic simulations seems by far too slow from computational point of view. For the partially observed case the amount of information depends on the fact which species is observed. Observing g the estimation is possible. Observing only Ca the objective function of the MSS method has to be modified to equation (3.3) with $K = \{0, 5, \ldots, 45\}$.
The examples demonstrate that although the maximum likelihood property is theoretically lost for the MSS method the estimation is still quite precise because the violations of the maximum likelihood conditions are not strong. The proposed test functions work fine as they identify those situations which lead to a bias and "accept" the others. The two measures SNR and NDR perform inversely proportionally so it would be enough to calculate only one of them. The MSS method also allows for the extension to the case of measurement noise, in which even data points with resulting negative molecule counts can be used. An example with measurement noise is given for the Lotka-Volterra example.

The investigation of methods for parameter estimation for stochastic models results in the fast and efficient MSS method based on multiple shooting and the method based on the MLS function for cases where the MSS method does not cover the stochasticity well. Test functions are used to support the decision on the appropriate use of the methods. As the MSS method with multiple shooting is based on ODEs it allows from computational point of view to tackle systems from point of state space as large as systems tackled in deterministic modeling.

Outlook for future work:

An interesting point for future research would be an in-depth analysis of the question which method is appropriate for which class of models and a criterion to classify models a priori into the classes. Furthermore research on the statistical properties of the estimates, e.g. confidence intervals for the parameters, would be important for the quality assessment of the estimates. Another point is the investigation of optimum experimental design, which helps in saving experimental cost whilst increasing the estimation accuracy, which is important, e.g. in table 4.4.

Acknowledgements

I would like to express my thanks to my first supervisor Prof. Dr. Dr. h. c. Hans Georg Bock for providing me the opportunity to write my PhD thesis on this very interesting and developing field. His guidance and the chance of working together with his group were important for the development of my thesis.

Furthermore I would like to thank my second supervisor Prof. Dr. Ursula Kummer. She gave me important guidance and let me be part of her department. The agreeable atmosphere in her group will let me remember the three years as a very pleasant time. Being part of her department gave me the chance of hearing on a day-to-day basis of the biological necessities for the applicability of mathematical methods, which had major influence on my work.

Thanks as well to my mentors:
Special thanks to Dr. Sven Sahle, my group leader, who initiated the project. During the three years of writing my thesis, he has always been available and open to my questions. Discussions with him always gave me valuable advice. I'm very grateful for his support. Joint work with him was submitted for publication and is an important result of this thesis.
Furthermore I would like to thank Dr. Johannes Schlöder, who was my contact person from the numerical optimization group. He gave me important feedback and I was impressed by the level of detail of his far-reaching comments, which had an important impact on my work. Moreover he gave me the chance of presenting my results in the SimOpt group meeting, which led to stimulating discussions and ideas.
In addition, I would like to thank Dr. Ralph Gauges, who was my contact person for all my technical questions concerning implementation of programming code and general system settings. Ralph always took the time for my questions, to which he rapidly had solutions, which I found impressive. I am very happy about this essential support.

The ViroQuant/HGS MathComp supported me with a full stipend for three years, which allowed me to spend all my efforts on my PhD project. Without this I would not have been able to accomplish my thesis so fast. The HGS MathComp and the Modeling of Biological Processes group supported me with travel funds allowing me to participate in several conferences gaining new ideas and insights as well as first chances to present my own research results.

The graduate school HGS MathComp supported me with an interdisciplinary research environment, which made it possible for me to consider requirements from both the applications and the methodology. In addition, they encouraged me to take responsibility in organizing events and participating in administrative meetings as one of the speakers of the fellows. Thanks especially to Dr. Michael Winckler for guiding this process.

Thanks as well to the administrative people, who are never in the spotlight but so essential for success: Jocelyn Faberman, Oktavia Klassen, Ria Hillenbrand-Lynott, Margret Rothfuss and Sarah Steinbach.

Thanks to Simon Lenz, Dr. Mario Mommer and Andreas Sommer, all people from the SimOpt group, the modeling of biological processes group and all students and friends with whom I had interesting and fruitful discussions.

Finally thanks to my parents for bringing me on a way which I could continue up to this PhD project these days.

List of Symbols

This list of symbols contains the frequently used symbols and whenever necessary the equation and page of the first mention.

Objective functions I

$F_d(\nu, \theta)$	based on ODE single shooting	see eq (2.1), page 19
$F_{s,q}^{(m)}(\nu, \theta)$	based on stochastic single shooting, average on squared differences, L^q-norm	see eq (2.2), page 20
$F_{\tilde{tr},q}^{(m)}(\nu, \theta)$	based on stochastic single shooting, average on trajectories	see eq (2.3), page 20
$F_{\tilde{s},q}^{(m)}(\nu, \theta)$	based on stochastic single shooting, average on squared differences, L^q-norm with σ_i	see eq (2.4), page 20
$L(\nu, \theta)$	Likelihood function giving the probability to get the data ν given the parameter θ	see eq (2.6), page 21
$l(\nu, \theta)$	log likelihood function: $log\ L(\nu, \theta)$	see eq (2.7), page 21
$F_L^{(m)}(\nu, \theta)$	based on likelihood function factorized in transition probabilities, LS function	see eq (2.9), page 21
$F_g^{(m)}(\nu, \theta)$	based on modification of the transition probabilities, MLS function	see eq (2.11), page 22
$F_{s,2}^{E}(\theta)$	expectation of the stochastic single shooting function $F_{s,2}^{(m)}$	see eq (2.21), page 26
$F_{\tilde{s},2}^{E}(\theta)$	expectation of the stochastic single shooting function $F_{\tilde{s},2}^{(m)}$ with σ_i	see eq (2.24), page 27

Objective functions II

$F_L^E(\theta)$	expectation of the likelihood-based functional F_L^E	see page 31
$F_h(\nu, \theta)$	MSS objective function	see eq (3.2), page 48
$F_K(\nu, \theta, \nu_K)$	MSS objective function for partially observed models	see eq (3.3), page 48

Notation for transition probabilities

$h(t, \theta, x_0, t_0)$	ODE solution at time $t - t_0$ of the initial value problem with parameter θ and initial value x_0	see eq (2.1), page 19
$H^{(j)}(t, \theta, x_0, t_0)$	Stochastic simulation with parameter θ and initial value x_0 at time t_0	see eq (2.2), page 20
$p_\theta(\nu_i, t_i \vert \nu_{i-1}, t_{i-1})$	Transition probability for a transition from state ν_{i-1} at time t_{i-1} to state ν_i at time t_i	see eq (2.6), page 21
$\hat{p}_\theta^{(m)}(\nu_i, t_i \vert \nu_{i-1}, t_{i-1})$	Estimation of p_θ with the relative frequency from m simulations	see eq (2.8), page 21
$g_\theta^{(m)}(\nu_i, t_i \vert \nu_{i-1}, t_{i-1})$	Modification of $\hat{p}_\theta^{(m)}$ with m simulations	see eq (2.10), page 22

Functions

$SNR(\nu, \theta)$	signal to noise ratio	see eq (3.4), page 50
$NDR(\nu, \theta)$	noise to dynamics ratio	see eq (3.6), page 50
$E[\,]$	expectation value	page 26

Further notation

n	number of measurements	see eq (2.1), page 19
X	substance or species, in reaction equation also on entity of the species	page 13
x	number of particles X in the system	page 15
$[X]$	concentration of X	see eq (1.3), page 13
ν	$\nu = (\nu_1, \ldots, \nu_n)$ measurements at time points $t_1, \ldots, t_n$, exact: $\nu_i \in \mathbb{N}_0^D$, noisy: $\nu_i \in \mathbb{Z}^D$	page 19
D	dimension of ν	see page 48
d	observed dimensions of ν	see page 48
m	number of simulations used to evaluate transition probabilities	see eq (2.2), page 20
θ	parameter	page 15
$\theta^{(0)}$	true parameter	page 20
ϵ	$\epsilon = (\epsilon_1, \ldots, \epsilon_n)$, residuals	see eq (3.1), page 47

Abbreviations

CME	Chemical Master Equation	page 9
ODE	Ordinary Differential Equation	page 5
TSS	Traditional single shooting method	see eq (2.1), page 19
LS	Likelihood simulation function $F_L^{(m)}(\nu, \theta)$	see eq (2.9), page 21
MLS	Modified likelihood simulation function $F_g^{(m)}(\nu, \theta)$	see eq (2.11), page 22
MSS	Multiple shooting for stochastic systems	page 47
KL	Kullback-Leibler divergence	page 49
EL	exact likelihood method, using $l(\nu, \theta)$ with the exact transition probabilities from the probability generating function	page 54

List of Figures

List of Tables

Bibliography

[1] A. Raj and A. van Oudenaarden. Single-molecule approaches to stochastic gene expression. *Annu. Rev. Biophys.*, 38:255–270, 2009.

[2] D.T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22 (4):403–434, 1976.

[3] J. Pahle. Biochemical simulations: stochastic, approximate stochastic and hybrid approaches. *Briefings in Bioinformatics*, 10 (1):53–64, 2009.

[4] T. Tian, S. Xu, J. Gao and K. Burrage. Simulated maximum likelihood method for estimating kinetic rates in gene expression. *Bioinformatics*, 23 (1):84–91, 2007.

[5] S.K. Poovathingal and R. Gunawan. Global parameter estimation methods for stochastic biochemical systems. *BMC Bioinformatics*, 11:414, 2010.

[6] R.J. Boys, D.J. Wilkinson and T.B.L. Kirkwood. Bayesian inference for a discretely observed stochastic kinetic model. *StatComput*, 18:125–135, 2008.

[7] Y. Wang, S. Christley, E. Mjolsness and X. Xie. Parameter inference for discretely observed stochastic kinetic models using stochastic gradient descent. *BMC Systems Biology*, 4:99, 2010.

[8] D.A. Henderson, R.J. Boys, C.J. Proctor and D.J. Wilkinson. Linking systems biology models to data: a stochastic kinetic model of p53 oscillations. *The Oxford Handbook of Applied Bayesian Analysis*, pages 155–187, 2010.

[9] S. Reinker, R.M. Altman and J. Timmer. Parameter estimation in stochastic biochemical reactions. *IEE Proceedings Systems Biology*, 153 (4):168–178, 2006.

[10] A. Andreychenko, L. Mikeev, D. Spieler and V. Wolf. Parameter identification for markov models of biochemical reactions. *arXiv*, 1102.2819v1 [q-bio.QM]:52–58, 2011.

[11] C.S. Gillespie. Moment-closure approximations for mass-action models. *IET Systems Biology*, 3 (1):52–58, 2009.

[12] C.S. Gillespie and A. Golightly. Bayesian inference for generalized stochastic population growth models with application to aphids. *Applied Statistics*, 59 (2):341–357, 2009.

[13] P. Deuflhard, W. Huisinga, T. Jahnke and M. Wulkow. Adaptive discrete galerkin methods applied to the chemical master equation. *SIAM J. on Scientific Computing*, 30 (6):2990–3011, 2008.

[14] S. Engblom. A discrete spectral method for the chemical master equation. *Technical Report, Uppsala University*, 2006-036, 2008.

[15] B. Munsky and M. Khammash. Identification from stochastic cell-to-cell variation: a genetic switch case study. *IET Systems Biology*, 4 (6):356–366, 2010.

[16] C. Zimmer and S. Sahle. Parameter estimation for stochastic models of biochemical reactions. *Submitted to PloS ONE*, 2012.

[17] Y. Marzouk and D. Xiu. A stochastic collocation approach to bayesian inference in inverse problems. *Communications in Computational Physics*, 6 (4):826–847, 2009.

[18] H.G. Bock. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, volume 183 of *Bonner Mathematische Schriften*. Universität Bonn, Bonn, 1987.

[19] J.P. Schlöder and H.G. Bock. Identification of Rate Constants in Bistable Chemical Reactions. In Deuflhard and Hairer, editors, *Numerical Treatment of Inverse Problems in Differential and Integral Equations, Progress in Scientific Computing*, pages 27–47, Basel Boston Berlin, 1983. Birkhäuser.

[20] E. Baake, M. Baake, H.G. Bock and K.M. Briggs. Fitting ordinary differential equations to chaotic data. *Physical Review A*, 45 (8):5524–5529, 1992.

[21] J. Kallrath, J.P. Schlöder and H.G. Bock. Least squares estimation in chaotic differential equations. *Celestial Mechanics and Dynamical Astronomy*, 56:353–371, 1993.

[22] U. Kummer, B. Krajnc, J. Pahle, A.K. Green C.J. Dixon and M. Marhl. Transition from stochastic to deterministic behavior in calcium oscillations. *Biophysical Journal*, 89:1603–1611, 2005.

[23] M.J. Berridge, M.D. Bootman and P. Lipp. Calcium - a life and death signal. *Nature, news and views feature*, 395 (6703):645–648, 1998.

[24] D.J. Wilkinson. *Stochastic Modelling for Systems Biology*. Chapman & Hall/CRC, Mathematical and Computational Biology Series, Boca Raton, 2006.

[25] E. Klipp, W. Liebermeister, C. Wierling, A. Kowald, H. Lehrach and R. Herwig. *Systems biology*. Wiley-Blackwell, Weinheim, 2009.

[26] T.G. Kurtz. The relationship between stochastic and deterministic models for chemical reactions. *The Journal of Chemical Physics*, 57(7):2976–2978, 1972.

[27] S. Wu, J. Fu, Y. Cao and L. Petzold. Michaelis-menten speeds up tau-leaping under a wide range of conditions. *The Journal of Chemical Physics*, 134 (13):134112, 2011.

[28] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus et al. COPASI - a COmplex PAthway SImulator. *Bioinformatics*, 22 (24):3067–3074, 2006.

[29] D.T. Gillespie, M. Roh and L.R. Petzold. Refining the weighted stochastic simulation algorithm. *The Journal of Chemical Physics*, 130:174103, 2009.

[30] H. Kuwahara and I. Mura. An efficient and exact stochastic simulation method to analyze rare events in biochemical systems. *The Journal of Chemical Physics*, 129:165101, 2008.

[31] T. Toni, D. Welch, N. Strelkowa, A. Ipsen and M.P.H. Stumpf. Approximate bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society*, 6:187–202, 2008.

[32] D.R. Cox and H.D. Miller. *The Theory of Stochastic Processes.* Methuen & Co LTD, London, 1965.

[33] H.T.H. Piaggio. *An elementary treatise on differential equations and their applications.* Bell & Hyman, London, 1982.

[34] I.N. Bronstein, K.A. Semendjajew, G. Musiol and H. Mühlig. *Taschenbuch der Mathematik, in german language.* Verlag Harri Deutsch, 2001.

[35] A. Kurtz. Continious time markov chain models for chemical reactions. In Koeppl, H.; Densmore, D.; Setti, G.; di Bernardo, M. (Eds.), editor, *Design and analysis of biomolecular circuits*, pages 1–43, New York, 2011. Springer.

[36] C.C. Heyde and E. Heyde. A stochastic approach to a one substrate, one product enzyme reaction in the initial velocity phase. *J. Theoret. Biol.*, 25:159–172, 1969.

[37] D.J. Wilkinson. Stochastic modeling for quantitative description of heterogeneous biological systems. *Nature Reviews Genetics*, 10 (2):122–133, 2009.

[38] Mathematica, version 8.0. *Wolfram Research, Inc.*, Champaign, IL, 2010.

[39] A. Prékopa. *Stochastic Programming.* Kluwer Academic Publishers, Dordrecht, 1995.

[40] J.J. Schneider and S. Kirkpatrick. *Stochastic Optimization.* Springer, Berlin, 2006.

[41] C.G. Moles, P. Mendes and J.R. Banga. Parameter estimation in biochemical pathways: A comparision of global optimization methods. *Genome Research*, 13:2467–2474, 2003.

[42] J. Kennedy and R. Eberhart. Particle swarm optimization. *Proceedings of the Fourth IEEE International conference on Neural Networks, Perth, Australia*, 4 (6):1942–1948, 1995.

[43] K. Zielinski and R. Laur. Stopping criteria for a constrained single-objective particle swarm optimization algorithm. *Informatica*, 31:51–59, 2007.

[44] D.P. Morton. Stopping rules for a class of sampling-based stochastic programming algorithms. *Operation Research*, 46 (5):710–718, 1998.

[45] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60 (4):897–936, 1938.

[46] G. Walz. *Lexikon der Mathematik.* Spektrum Akademischer Verlag, Heidelberg, Berlin, 2002.

[47] M. Friedman and A. Kandel. *Introduction to pattern recognition: Statistical, structural and fuzzy logic approaches.* World Scientific, Singapore, 1999.

[48] U. Krengel. *Einführung in die Wahrscheinlichkeitstheorie und Statistik, in german language.* Vieweg, Wiesbaden, 2005.

[49] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.

[50] K. Petras. Smolyak cubature of given polynomial degree with few nodes for increasing dimension. *Numerische Mathematik*, 93:729–753, 2003.

[51] T.N.L. Patterson. The optimum addition of points to quadrature formulae. *Mathematics of Computation*, 22 (104):847–856 + s21–s31, 1968.

[52] F. Heiss and V. Winschel. Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics*, 144:62–80, 2008.

[53] M. Komorowski, B. Finkenstädt, C.V. Harper and D.A. Rand. Bayesian inference of biochemical kinetic parameters using the linear noise approximation. *BMC Bioinformatics*, 10:343, 2009.

[54] H.G. Bock. Numerical treatment of inverse problems in chemical reaction kinetics. In K.H. Ebert, P. Deuflhard and W. Jäger, editors, *Modelling of Chemical Reaction Systems*, volume 18 of *Springer Series in Chemical Physics*, pages 102–125. Springer, Heidelberg, 1981.

[55] M.E. Meyer and D.V. Gokhale. Kullback-leibler information measure for studying convergence rates of densities and distributions. *IEEE Transactions on information theory*, 39 (4):1401–1404, 1993.

[56] M.D. Reid and R.C. Williamson. Generalised pinsker inequalities. *Proceedings 22nd Annual Conference on Learning Theory (Colt)*, Montreal, 2009.

[57] A. Klenke. *Wahrscheinlichkeitstheorie*. Springer, Berlin, 2006.

[58] H.G. Bock, E. Kostina and J.P. Schlöder. Numerical methods for parameter estimation in nonlinear differential algebraic equations. *GAMM Mitteilungen*, 30 (2):376–408, 2007.

[59] M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *Journal of Physical Chemistry*, 104:1876–1889, 2000.

[60] C. Zimmer, S. Sahle and J. Pahle. Exploring intrinsic fluctuations to identify model parameters. *Manuscript in preparation*, 2012.