INFERENCE IN STOCHASTIC SYSTEMS WITH TEMPORALLY AGGREGATED DATA

A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy
in the Faculty of Biology, Medicine and Heal

2017

By
Maria Myrto Folia
School of Health Sciences

Contents

List of Tables

List of Figures

To my parents

Acknowledgements

I would like to thank my supervisor Magnus Rattray for his guidance and support. The microscopy data for chapter 5 of this thesis were kindly provided by the lab group of Mike White who has been my co-supervisor together with Pawel Paszek who I would also like to thank.

I would like to thank my examiners Mark Muldoon and Simon Rogers for their comments and the interesting discussion.

I also want to thank all the members of my group for the time we shared together. Special thanks to Mudassar for going through my MCMC code. Furthermore, I want to thank my greek friends either around England or around the world who were there to listen to me!

Thank you to my mum and dad for supporting me throughout my PhD years.

Finally, the greatest of my appreciation goes to Nico. His support has been invaluable for finishing this journey. Thank you!

Abstract

Inference in Stochastic Systems with Temporally Aggregated Data

Maria Myrto Folia

A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy, 2017

The stochasticity of cellular processes and the small number of molecules in a cell make deterministic models inappropriate for modelling chemical reactions at the single cell level. The Chemical Master Equation (CME) is widely used to describe the evolution of biochemical reactions inside cells stochastically but is computationally expensive. The Linear Noise Approximation (LNA) is a popular method for approximating the CME in order to carry out inference and parameter estimation in stochastic models.

Data from stochastic systems is often aggregated over time. One such example is in luminescence bioimaging, where a luciferase reporter gene allows us to quantify the activity of proteins inside a cell. The luminescence intensity emitted from the luciferase experiments is collected from single cells and is integrated over a time period (usually 15 to 30 minutes), which is then collected as a single data point.

In this work we consider stochastic systems that we approximate using the Linear Noise Approximation (LNA). According to the LNA, the state of the system follows $X_t = \phi_t + \xi_t$, where ϕ_t is the deterministic part of the system and ξ_t the stochastic part representing the fluctuations around ϕ_t. ξ_t is described by a linear stochastic differential equation. Therefore, its solution is a Gaussian process with mean m_t and covariance matrix S_t and so, $X_t \sim N(\phi_t + m_t, S_t)$.

For the problem of aggregation most existing approaches treat the data as being proportional to the actual quantities. We instead generalize the LNA to aggregated data by taking the integral of the state process X_t over the period of aggregation: $\int_{t_0}^{t} X(u)du = \int_{t_0}^{t} \phi_u du + \int_{t_0}^{t} \xi_u du = I(t) + Q(t)$. The function Q_t will also follow a Gaussian process, as it is the time integral of ξ_t. Hence, the data follow a multivariate Gaussian distribution and we can perform inference with a

continuous-discrete-time Kalman Filter in order to compute the data likelihood and subsequently to infer the model parameters.

We demonstrate our method by learning the parameters of three different models from which aggregated data was simulated, an Ornstein-Uhlenbeck model, a Lotka-Voltera model and a gene transcription model. We have additionally compared our approach to the existing approach and find that our method is outperforming the existing one. Finally, we apply our method in microscopy data from a translation inhibition experiment.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

Chapter 1

Introduction

1.1 Computational Systems Biology

In recent years, major advances in experimental techniques in Biology resulted in an unprecedented increase of available data and new insights into the workings of biological systems. The vast amount of data made the need for novel computational and mathematical tools in Biology apparent, leading to the emergence of new scientific fields. Computational Systems Biology is concerned with the development of computational methods that, combined with biological data, can describe a biological system of interest [49].

Biochemical networks are used to represent biochemical reactions. Mathematical modelling of such networks is crucial in Systems Biology. Together with biological data, mathematical models can be used to infer quantities that cannot be directly measured by biologists or estimate parameters, e.g. reaction rates, of the biochemical networks.

The most common approach for modelling biochemical networks assumes that their time evolution is described by a set of coupled Ordinary Differential Equations (ODEs). By making this assumption, we have a continuous, deterministic model. Various methods for inference and parameter estimation in ODE models are available in the literature. Available methods emerged from fields such as optimisation [56], Bayesian (nonparametric) statistics [50] and control theory [51].

ODEs provide a reasonable representation of data describing an average over millions of cells, as for example in tissues or cell cultures. Nevertheless, there are cases where an average over cells will not be appropriate for studying a system. For example, the NF-κB transcription factors show out-of-phase oscillations between the nucleus and the cytoplasm (N-C) with a typical frequency of 100 minutes [78, 82]. These oscillations have been found to control the dynamics of gene expression [58], and are therefore of great importance. However, if we consider a whole population of cells, this oscillatory behaviour is not apparent due to

population averaging. This is illustrated in Figure 1.1 [78], where the coloured lines correspond to out-of-phase oscillations in different single cells and the black line indicates the population average. This indicates the need for studying and modelling biological processes at the single-cell level.

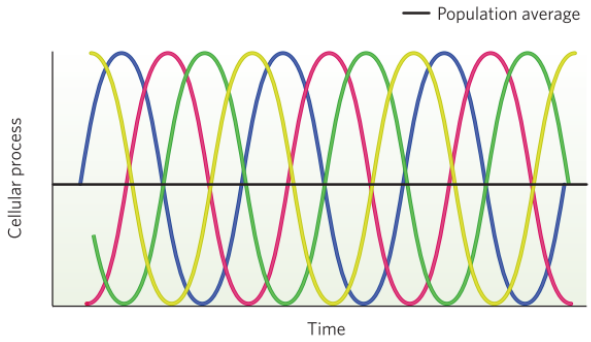


Figure 1.1: Out of phase oscillations at single cells (coloured lines) and population average (black line), adapted from [78].

Microscopy data at the single cell level, such as fluorescence and luminescence data, have shown that biochemical processes inside a cell are intrinsically and extrinsically stochastic [16]. Intrinsic noise is due to the inherent randomness of biochemical processes such as transcription, while extrinsic stochasticity is due to cell-to-cell variation, e.g. in molecular abundance. Furthermore, at the level of single cells, molecule numbers are discrete rather than continuous. Therefore, instead of an ODE approach, a discrete stochastic approach is needed in order to capture their nature correctly.

One of the first obstacles to a stochastic approach in Systems Biology is the exact simulation of biochemical processes. This was solved in 1977 by Daniel T. Gillespie with his famous stochastic simulation algorithm [25]. However, as in the case of ODE models, the parameters governing the system are rarely known completely. Therefore, another important problem is to perform parameter estimation given single cell data. A stochastic model consists of an ensemble of all the possible paths the system can take, making parameter estimation much more difficult than in deterministic models. Studies in this direction have already been performed [47, 18, 89] but parameter estimation in stochastic kinetic models remains an active research area.

1.2 Stochastic vs. deterministic models

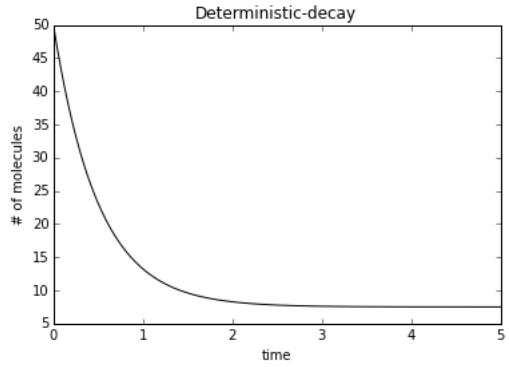
Is a stochastic modelling approach really important? In order to answer this question, we will look at some examples.

Firstly, consider the dynamics of a fairly simple system such as that of a molecular decay. We assume the same initial condition for the number of molecules before decaying, while there is a constant production rate that will keep the molecule numbers from disappearing completely. We simulate different paths using both a deterministic and a stochastic approach. Clearly, the deterministic approach will always give us the same result since we use the same initial conditions. As we can see from Figures 1.2(a) and (b), the deterministic approach has ignored the fluctuations that are apparent in the stochastic model. Although the mean behaviour, in this case, seems equivalent for both approaches, the stochastic model gives us the extra information about the volatility in the system. This extra information can become very important for parameter estimation. In Figures 1.2(c) and (d), we see the same system again, but now it is initialised to a higher number of molecules leading to more resemblance between the deterministic and the stochastic approach.

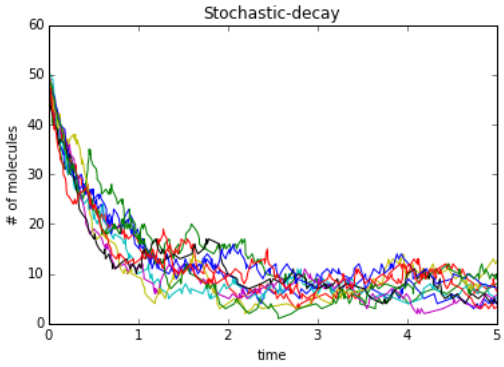
As a second example, consider a Lotka-Volterra model. The Lotka-Volterra model has been used to model biochemical reactions when there are competitive interactions between chemical species (e.g. autoregulatory networks). In Figure 1.3, we see the time evolution of the prey and predator numbers using a deterministic and a stochastic approach. Here, we see that the differences in the behaviour of the system between the stochastic and deterministic model are much more obvious than before. We first remark that in the stochastic case the oscillations possess a different amplitude, while in the deterministic case, they are repeated with the same amplitude and period. The next observation we can make is that the stochastic case allows for the extinction of the species, allowing us to investigate the distribution of the time to extinction—something that is impossible using the deterministic model.

1.3 Thesis motivation - Aggregated data

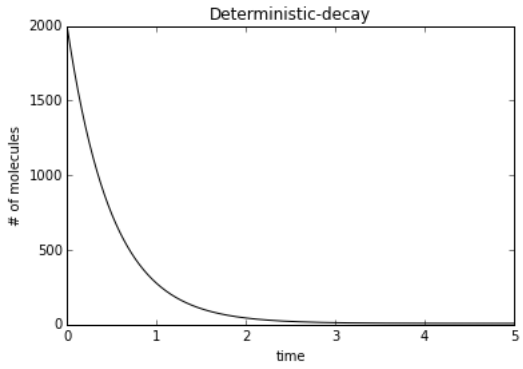
The stochasticity of cellular processes and the small number of molecules in a cell is the reason why stochastic modelling is needed to describe biological processes



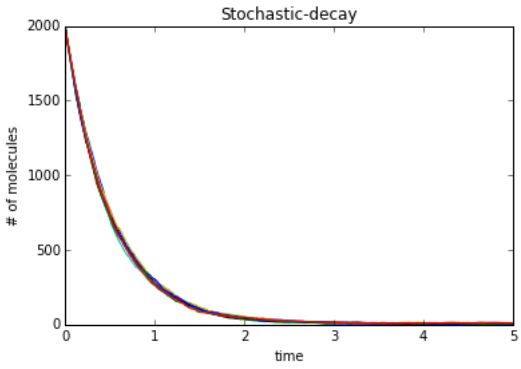
(a) Deterministic model (50 initial molecules).



(b) Stochastic model (50 initial molecules).



(c) Deterministic model (2000 initial molecules).

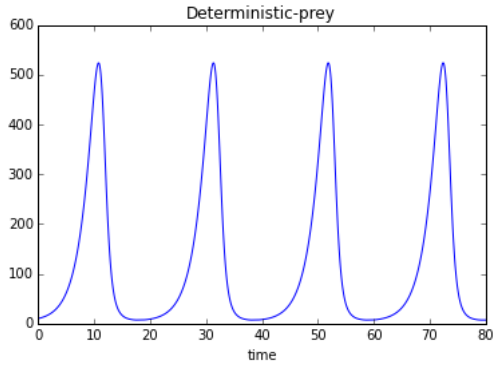


(d) Stochastic model (2000 initial molecules).

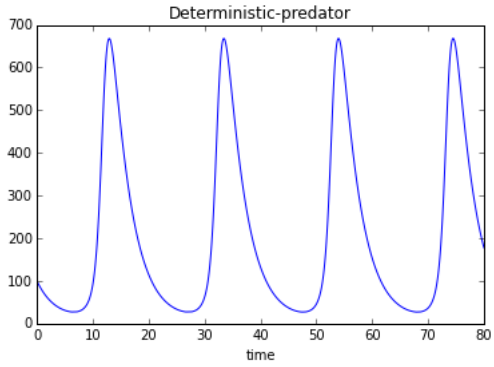
Figure 1.2: Time evolution of a deterministic and stochastic model for molecular decay with different initial states.

at the single cell level. The work for this thesis has been motivated by the need for stochastic models for different kinds of single cell data.

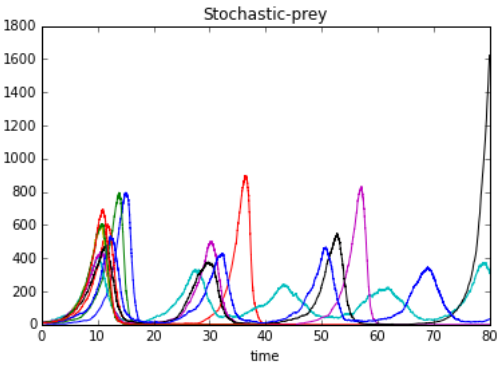
Most of the current work on Stochastic Systems Biology using microscopy data [89] has been successfully applied to fluorescence data. However, another interesting technology for capturing information at the single cell level is luminescence bioimaging. In luminescence bioimaging, a luciferase reporter gene allows us to quantify the activity of proteins inside a cell. The luminescence intensity emitted from the luciferase experiments is collected from single cells and is integrated over a time period, which is then collected as a single data point. The length of the integration period depends on the type of experiment and the strength of the luminescence signal, and can in certain cases be up to 30 minutes. Luciferase is a highly sensitive and non-toxic reporter making it more appropriate for long-term studies than other technologies such as fluorescent reporters [87].



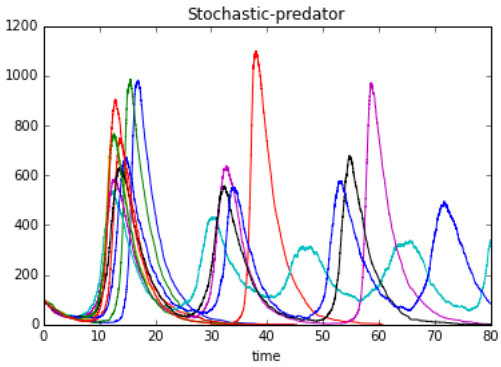
(a) Prey (deterministic model).



(b) Predator (deterministic model).



(c) Prey (stochastic model).

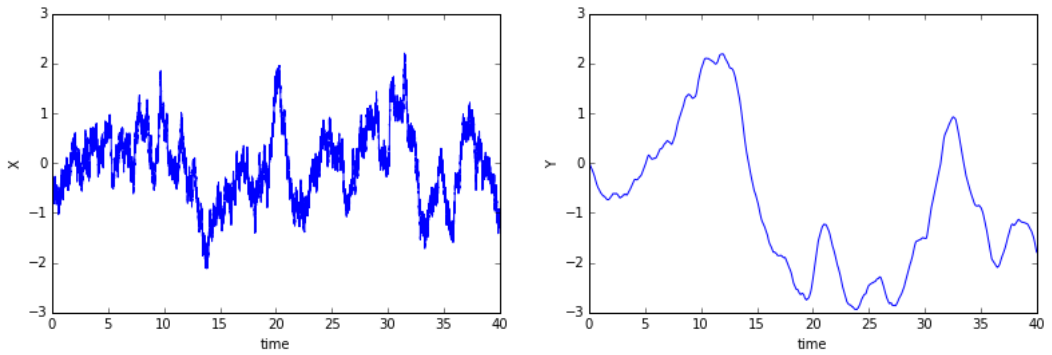


(d) Predator (stochastic model).

Figure 1.3: Time evolution of a deterministic versus stochastic Lotka-Volterra model.

To our knowledge, the available literature does not take into account both the stochastic and aggregated nature of luciferase data. However, aggregation can include extra information of a system and ignoring it can result in loss of accuracy in the results. Thus, we believe it is important to use the appropriate models when working with luciferase data. In this thesis, we focus on Bayesian inference and parameter estimation for stochastic systems that we observe through aggregated time series, such as luciferase data. We have built on the work of [47, 18] and extended it for the case of aggregated data.

Although the examples in this thesis will be inspired by biological applications, our method is not restricted to these applications and could be used in other domains as well. Stochastic modelling has been also used extensively in domains other than biology, such as finance or physics. The idea of temporal aggregation is found in these domains too. For example, in a physics application, we might be interested in studying the position component of a particle when we know



(a) Velocity of a particle. (b) Position of a particle.

Figure 1.4: Time evolution of an Ornstein-Uhlenbeck process (velocity) and its integral (position).

that its velocity component is described by a known process, i.e. an Ornstein-Uhlenbeck process [31]. On the other hand in finance temporal aggregation plays an important role for stochastic volatility models (e.g. [5]).

In Figure 1.4, we show plots of the time evolution of the velocity and position of a particle. The process describing the position of the particle can be seen as the time integral of the velocity process which is described by an Ornstein-Uhlenbeck process. As we can see in Figure 1.4, the position process looks much smoother than the velocity process. In general, aggregation tends to reduce fluctuations and therefore the stochasticity of the original non-aggregated process may be underestimated.

1.4 Overview of the thesis

In **Chapter 2**, we introduce the mathematical background needed for this thesis. We start with an introduction to stochastic processes and move on to their use in biochemical reaction networks. We present a proof from [27] for the derivation of the Chemical Master Equation (CME) and present the Linear Noise Approximation as an approximation of the CME. We finish the chapter with an introduction to Bayesian statistics, which provides us with a framework for inference.

In **Chapter 3**, we move to inference and parameter estimation for stochastic systems. We discuss how we can use the Kalman Filter methodology for systems described by Stochastic Differential Equations and how we can combine it with

the Linear Noise Approximation. Finally, we present our novel method for incorporating temporally aggregated observations, which is the main result of this thesis.

In **Chapters 4** and **5**, we present inference and parameter estimation results on synthetic and real datasets respectively. The purpose of these chapters is to investigate the effect of aggregation in stochastic systems and to assess the performance of our method. In all cases we investigated our method works well and outperforms existing methods that do not model aggregation explicitly.

In **Chapter 6**, we summarise the concluding remarks and contributions of this thesis and propose directions for future research.

Chapter 2

Mathematical Background

In this chapter, we cover the mathematical background needed for the remainder of the thesis. We discuss some basic concepts around Markov processes in discrete/continuous time and discrete/continuous state. We continue by deriving the Chemical Master Equation (CME) and presenting the Gillespie algorithm. An approximation of the CME, known as the Linear Noise Approximation, is presented, and we discuss some of its limitations. Finally, we introduce some notions of Bayesian statistics and inference.

2.1 Introduction to stochastic processes

In the following, we will introduce the mathematical ideas related to systems that evolve probabilistically over time. We will focus on interpreting these ideas, rather than providing a strict mathematical formulation.

Since we are adopting a probabilistic view, we will commonly refer to variables that take different values with a specific probability, i.e. **random variables**. We start by giving the definition of a stochastic process and continue with some special categories of stochastic processes and their properties.

A **stochastic process** $\{X_t, t \in T\}$ is a collection of random variables indexed by t, where t usually represents time. For a specific t, X_t is a random variable. If the index set T is discrete, we have a discrete-time stochastic process, while if it is continuous, we have a continuous-time stochastic process [70].

If $X_t = x$, we say that X_t is at state x at time t. The set of possible values X_t can take is called the state space of the stochastic process $\{X_t, t \in T\}$. As with time, we can have discrete-state and continuous-state stochastic processes. Given that $X_{t_1} = x_1, X_{t_2} = x_2, ...$, we can describe a stochastic process by its joint probability density $p(x_1, t_1; x_2, t_2; ...)$. Note that for continuous time or continuous states, we will have infinitesimal increments, i.e. $[t_i, t_i + dt_i)$, $[x_i, x_i + dx_i)$. A particular realisation of a stochastic process for all t, e.g. $(x_1, x_2, x_3, ...)$, is called

a sample path of the process.

An important category of stochastic processes are the so-called Markov processes. **Markov processes** are stochastic processes that possess the Markov property. The **Markov property** tells us that the future state of a stochastic process, given its past and its present state, is only depended on the present [70]. The theory of Markov processes is well developed and this makes Markov processes popular modelling tools in many domains including biology. In what follows, we present a brief introduction in key aspects of the Markov process theory.

2.1.1 Markov chains

Although there is no universal agreement, with the term **Markov chain** we will refer here to a discrete-time, discrete-state[1] Markov process. The Markov property for a Markov chain can be written in probabilistic terms as $p(x_t, t|x_{t-1}, t-1; ...; x_1, t_1; x_0, t_0) = p(x_t, t|x_{t-1}, t-1)$, which also defines its transition probability.

If the transition probability does not depend on time, then we can simply denote it by $p(i, j)$, where $p(i, j) = p(X_{t+1} = j|X_t = i)$ expresses the probability of the process transitioning to state j after being at state i. Markov chains with this property are said to be time homogeneous. In this thesis, unless otherwise stated, we will assume that a Markov chain is time-homogeneous.

The one-step transition matrix $\boldsymbol{P}$ of a Markov chain is given by the transition probabilities $p(i, j)$:

$$\boldsymbol{P} = \begin{bmatrix} p(0,0) & p(0,1) & p(0,2) & ... \\ p(1,0) & p(1,1) & p(1,2) & ... \\ \vdots & \vdots & \vdots & \\ & & & \\ p(i,0) & p(i,1) & p(i,2) & ... \\ \vdots & \vdots & \vdots & \end{bmatrix}. \tag{1}$$

The matrix $\boldsymbol{P}$ is a stochastic matrix, i.e. its elements are non-negative and its rows sum up to 1.

The n-step transition probabilities $p^n(i, j) = p(X_{t+n} = j|X_t = i)$ are expressing the probability of transitioning from state i to state j in n steps. They can be computed by the **Chapman-Kolmogorov** (C-K) equation:

[1] finite or countable numbers

$$p^{m+n}(i,j) = \sum_{k=0}^{\infty} p^n(i,k)p^m(k,j), \forall n,m \geq 0\,. \tag{2}$$

What these equations actually tell us is that, in order to go from state i to state j in $m+n$ steps, we have to sum over all the possible paths that can take the process from state i to state k in n steps and then from state k to state j in m steps [70]. Proving the C-K equations for a Markov chain is straightforward using the law of total probability.

Equation (2) can be written in matrix notation as $\boldsymbol{P^{(n+m)}} = \boldsymbol{P^{(n)}}\boldsymbol{P^{(m)}}$, where $\boldsymbol{P^{(n)}}$ refers to the n-step transition matrix with elements $p^n(i,j)$. It is induced that $\boldsymbol{P^{(n)}} = \boldsymbol{P^n}$, where $\boldsymbol{P^n}$ indicates the multiplication of $\boldsymbol{P}$ with itself n times.

For a Markov chain with transition matrix P, the **stationary distribution** π of the process can be found by solving $\pi = \pi P$.

2.1.2 Discrete-time continuous-state Markov process

Another category of stochastic processes corresponds to a stochastic process where the index set of time takes only discrete values, as in a Markov chain, but the states correspond to continuous random variables. If we further assume that such a stochastic process possesses the Markov property, then we are dealing with a **discrete-time, continuous-state Markov process**.

We can deduce the usual rules of the Markov chains for the case of the discrete-time, continuous-state Markov processes by just replacing sums with integrals. Thus, the C-K equation becomes:

$$p^{m+n}(i,j) = \int p^n(i,k)p^m(k,j)dk, \forall n,m \geq 0\,. \tag{3}$$

The most common discrete-time, continuous-state process is the Autoregressive (AR) Process. The AR process takes its name from the fact that if X_t is an AR process, then it can be inferred by its previous states through a regression equation [64]. AR processes have been studied extensively in the time series literature [7]. As an example, Equation (4) describes a first order AR(1) process.

$$X_{t+1} = aX_t + w_t\,, \tag{4}$$

where $w_t \sim N(0, \sigma^2)$ and w_t is independent of $w_{t'}$ for all $t \neq t'$. The Markov property can be established easily for an AR(1) process since its future state at $t+1$ depends only on the present state at time t. The form of w_t leads to X_t being also a Gaussian process.[2] An AR(1) is a discrete-time Gaussian Markov process and its extension to continuous time leads to stochastic differential equations that will be studied later [64].

2.1.3 Continuous-time Markov chains

A **continuous-time Markov chain** is a Markov process with continuous time and discrete states and can be interpreted as the continuous analogue of a Markov chain. The Markov property for this case can be written as [70]:

$$p(X_{t+dt} = j | X_t = i, X_u = x_u, 0 \leq u < t) = p(X_{t+dt} = j | X_t = i)\,. \tag{5}$$

A continuous-time Markov chain is a process that jumps from one state to another state in accordance with a Markov chain, but the amount of time spent at a state before jumping to the next is exponentially distributed [70].

In analogy with the discrete case, the process is **time homogeneous** if (5) is independent of t. In that case we can write its transition probability just as $p(i, j, \tau)$, where τ is the time spent at state i before it jumps to state j. Once again, analogously to the discrete case, we can define the **C-K** equation as:

$$p(i, j, t + \tau) = \sum_{k=0}^{\infty} p(i, k, t) p(k, j, \tau), \forall t, \tau \geq 0\,. \tag{6}$$

For continuous-time Markov chains we will need to introduce two more quantities: q_{ij}, which denotes the **transition rate** from state i to j and v_i, the **rate** at which the process makes a transition from state i, such that $v_i = \sum_i q_{ij}$. These two quantities are defined by:

$$q_{ij} = \lim_{h \to 0} \frac{p(i, j, h)}{h}, i \neq j \tag{7}$$

$$v_i = \lim_{h \to 0} \frac{1 - p(i, i, h)}{h} \tag{8}$$

[2] A Gaussian process is a stochastic process that consists of random variables any finite number of which have a joint Gaussian distribution [64]

We can now find how the process varies over time by:

$$\frac{dp(i,j,t)}{dt} = \sum_{k \neq j} q_{kj} p(i,k,t) - v_j p(i,j,t)\,. \tag{9}$$

Equation (9) is known as the **forward Kolmogorov** equation for a continuous-time Markov chain. Using matrix notation we can define the **transition rate matrix** $\boldsymbol{R}$ with elements:

$$R_{ij} = \begin{cases} q_{ij}, & i \neq j \\ -v_i, & i = j \end{cases}. \tag{10}$$

The transition rate matrix has rows that sum to 0 and non-negative off-diagonal elements. Using $\boldsymbol{R}$ we can deduce the matrix form of the forward Kolmogorov equation:

$$\frac{d\boldsymbol{P(t)}}{dt} = \boldsymbol{P(t)R}\,, \tag{11}$$

where $\boldsymbol{P(t)}$ is the transition matrix with elements $p(i,j,t)$ and has solution $\boldsymbol{P(t)} = e^{\boldsymbol{R}}$. For a continuous-time Markov chain we have that π is a stationary distribution iff $\pi\boldsymbol{R} = 0$ [13].

An important class of continuous-time Markov chains are the so-called Birth-Death processes. A **Birth-Death process** is a continuous-time Markov chain with non-negative discrete states (0,1,2,...), where all jumps are of length 1. We can think of a Birth-Death process as a system where we have n people initially and new individuals arrive at the system at rate λ_n and leave at rate μ_n. Here, λ_n and μ_n are the birth and death rates respectively [70].

A special case of a Birth-Death process is a Poisson process. A Poisson process is a pure birth process with constant birth rate, i.e. $\mu_n = 0$ and $\lambda_n = \lambda$, so we can define its transition rate matrix as:

$$R_{ij} = \begin{cases} \lambda, & j = i+1 \\ -\lambda, & i = j \\ 0, & \text{otherwise} \end{cases}. \tag{12}$$

The forward Kolmogorov equation for a Poisson process can be written as: $\frac{dp(i,j,t)}{dt} = \lambda p(i,j-1,t) - \lambda p(i,j,t)$.

A sample path from a Poisson process is shown in Figure 2.1. The key to

simulating the Poisson process is that we have a new jump that will update the size of the system by +1 at independent exponential times [26].

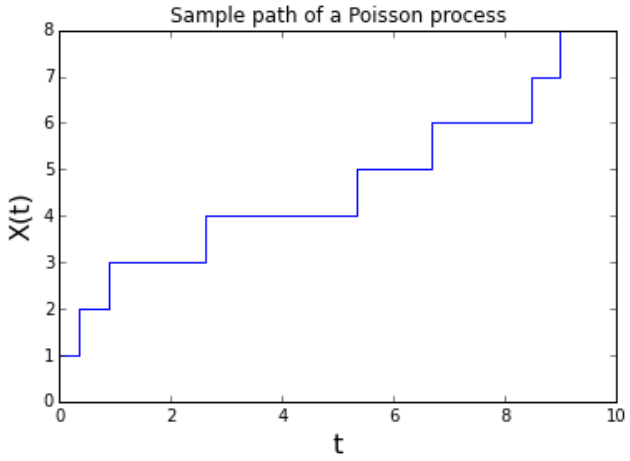


Figure 2.1: Sample path of a Poisson process with rate = 0.5.

2.1.4 Diffusions

A **diffusion process** is a continuous-time, continuous-state Markov process. The most well-known diffusion process is Brownian motion. Brownian motion describes the motion of tiny particles that are immersed in a fluid (liquid or gas). It owes its name to the botanist Robert Brown who in 1827 observed the random motion of particles of pollen grains in water through his microscope [9]. Einstein in 1905 [14] was the first to model this motion in a probabilistic way and Wiener finally established a mathematical analysis of this motion in a series of papers starting in 1918 [70]. Thus, the Brownian motion is also called a Wiener process and is denoted with W_t.

The Wiener process W_t has the following properties:

1. W_t is continuous.

2. $W_0 = 0$.

3. It has independent increments. $W_{t_1}, W_{t_2} - W_{t_1}, ..., W_{t_\kappa} - W_{t_{\kappa-1}}$ are independent for all $t_\kappa > ... > t_1 \geq 0$.

4. It has stationary increments. $W_t - W_s \sim N(0, t-s)$ $\forall t > s \geq 0$.

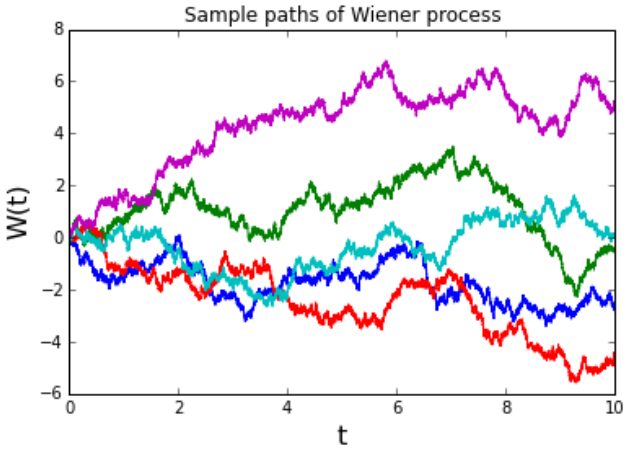


Figure 2.2: Five sample paths of a one dimensional Wiener process.

From the definition, we can see that $W_t \sim N(0, t)$ is a Gaussian process. Five different sample paths of W_t are shown in Figure 2.2.

Diffusion processes appear as solutions to stochastic differential equations. Therefore, stochastic differential equations need to be defined before we can move on to diffusion processes.

Stochastic differential equations (SDEs)

Ordinary differential equations (ODEs) are used to describe the evolution of a system in which no randomness takes place. Therefore, the result will always be determined for specified initial conditions. If we want to take into account the randomness or uncertainty in a model, we are naturally lead to stochastic differential equations (SDEs).

Assume a deterministic model for population growth with growth rate $a(t)$ that is described by the following ODE:

$$\frac{dN_t}{dt} = a_t N_t \,, N_0 = A \,. \tag{13}$$

In many real life situations, we will not be able to deterministically model population growth due to uncertainty in the rate parameter, i.e. $a_t = r_t + noise$ [60], or due to some general noise term inserted in the system such as $a_t N_t + noise$, as a result of noisy measurements, for example. Randomness can also be inserted in the initial conditions or other coefficients of an ODE. In these cases, we consider

to model the system using an SDE given by the general form:

$$dX_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t\,, \tag{14}$$

where for a d-dimensional process X_t, $\mu(t, X_t)$ is a d-vector, $\sigma(t, X_t)\sigma(t, X_t)^T$ is a $(d \times d)$ matrix and W_t stands for the m-dimensional Wiener process with components $W_t^1, ..., W_t^m$ and represents the noise term. Functions $\mu(t, X_t)$ and $\sigma(t, X_t)$ are referred to as the drift and diffusion term of the SDE respectively.

As an example, the univariate Wiener process is described by an SDE with drift equal to 0 and diffusion equal to 1. Returning to the population growth model (13) in the case of $a_t = r_t + noise$ we can formulate its stochastic counterpart given by the SDE: $dN_t = r_t N_t dt + N_t dW_t$ [60].

The general solution X_t to (14) is a diffusion process and is given by:

$$X_t = X_0 + \int_{t_0}^{t} \mu(t', X_{t'})\, dt' + \int_{t_0}^{t} \sigma(t', X_{t'})\, dW_{t'}\,, \tag{15}$$

where the last integral is a stochastic integral in the Ito sense [3]. We can observe that if the diffusion term is zero then we are lead to a purely deterministic process, also known as a Liouville process.

Existence and uniqueness theorems for the solutions of SDEs exist, and the interested reader is referred to [60]. For this thesis, we will assume that $\mu(t, X_t)$ and $\sigma(t, X_t)$ are known smooth, non-anticipating[4] functions of t.

Ito stochastic integrals can be calculated using Ito's formula (see Appendix A.1) and three important rules:

1. $dW_t dW_t = dt$,
2. $dW_t dt = 0$,
3. $dtdt = 0$.

In case we cannot solve a stochastic integral analytically, we can rely on numerical methods [46]. The **Euler-Maruyama approximation** is the equivalent of the Euler approximation for ODEs. The approximation is based on a time discretisation such as $0 < t_1 < ...t_{n-1} < T$, where $\Delta t = T/n$ for equidistant time

[3] The Ito integral $I(f) = \int_{t_0}^{t} f(t', X_{t'})\, dW_{t'}$ of a random function $f(t, X(t))$ given a partition $0 < t_1 < ... < t_{k-1} < t$ is defined as the limit $I(f) = \lim_{k\to\infty} \sum_{j=0}^{k-1} f(t_{j-1}, X_{t_{j-1}})(W_{t_j} - W_{t_{j-1}})$

[4] A function $f(t)$ is non-anticipating if it is independent of $W_s - W_t$ for $t \leq s$ [21].

steps. Setting $X_0 = x_0$ and $\Delta W_n = W_{t_{n+1}} - W_{t_n}$, the scheme proceeds iteratively as follows:

$$X_{n+1} = X_n + \mu(t_n, X_n)\Delta t + \sigma(t_n, X_n)\Delta W_n \,. \tag{16}$$

We will be interested in a particular class of SDEs, where the drift and diffusion terms in Equation (14) have a specific form. A **linear SDE in the narrow sense** is one that has a linear drift term in terms of X_t, $\mu(t, X_t) = a_1(t)X_t + a_2(t)$ and a diffusion term independent of X_t $\sigma(t) = b_1(t)$, such that the noise appears only additively [1]. A linear SDE in the narrow sense can be obtained by a linear ODE plus a noise term and has the following form:

$$dX_t = (a_1(t)X_t + a_2(t))dt + b_2(t)dW_t \,. \tag{17}$$

In order to calculate the solution of (17), we use the fundamental solution Φ_t which can be obtained from the homogeneous ODE: $\frac{d\Phi_t}{dt} = a_1(t)\Phi_t$. Using the transformation $\Phi_t^{-1}X_t$ and the Ito formula, the solution of Equation (17) is given by [1]:

$$X_t = \Phi_t(X_{t_0} + \int_{t_0}^{t} a_2(s)\Phi_s^{-1}ds + \int_{t_0}^{t} b_2(t)\Phi_s^{-1}dW_s) \,. \tag{18}$$

The solution of (17) is a Gaussian process iff the initial value X_{t_0} is a constant or normally distributed [1]. This result agrees with the form of (18), as X_t is a linear combination of Gaussian random variables. Note that the stochastic integral $\int_{t_0}^{t} \sigma(t', X_{t'})\, dW_t'$ is Gaussian, as it refers to the Ito integral of a non-random function and X_{t_0} is assumed to be non-random or Gaussian.

The evolution of the first two moments of the solution can be defined by two ODEs. For the first moment $m(t) = \mathbb{E}[X_t]$, we can easily deduce that [1]:

$$\frac{dm(t)}{dt} = a_1(t)m(t) + a_2(t) \,. \tag{19}$$

For the second moment $P(t) = \mathbb{E}[X_t X_t^T]$ we get [1]:

$$\frac{dP(t)}{dt} = P(t)a_1(t)^T + a_1(t)P(t)^T + m(t)a_2^T + a_2 m(t)^T + b_2 b_2^T \,. \tag{20}$$

Equations (19) and (20) are obtained by taking the expectation on the integral solution (18) and making use of the Ito's formula. We provide an alternative proof for Equations (19) and (20) in Appendix A.2.

It follows that the ODE of the variance $K(t) = P(t) - m(t)m(t)^T$ is given by:

$$\begin{aligned} dK(t) &= dP(t) - m(t)dm(t)^T - dm(t)m(t)^T \\ \frac{dK(t)}{dt} &= K(t)a_1(t)^T + a_1(t)K(t)^T + b_2 b_2^T \,. \end{aligned} \tag{21}$$

So far, we have examined diffusion processes as solutions to SDEs. However, as diffusion processes are Markov processes, we would be interested to look at the time evolution of their transition densities. So, we want the equivalent of a forward Kolmogorov equation for a continuous-time, continuous-state Markov process. This is given by the **Fokker-Planck** equation, which, for a d-dimensional diffusion process as described by Equation (14), is [1]:

$$\frac{\partial p(X_t,t)}{\partial t} = -\sum_{i=1}^{d} \frac{\partial}{\partial t}[\mu_i(X_t,t)p(X_t,dt)] + \frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d} \frac{\partial^2}{\partial X_{ti}\partial X_{tj}}[\Sigma_{ij}(X_t,t)p(X_t,t)]\,, \tag{22}$$

where $\Sigma_{ij}(X_t,t) = \sigma(t,X_t)\sigma(t,X_t)^T$ and is symmetric and positive definite.

The Fokker-Planck equation can be used for obtaining the stationary distribution of the underlying process by just setting Equation (22) to zero. Fokker-Planck equations are also used to approximate continuous-time Markov chains. One particular approximation will be studied in Section 2.4.

2.2 The Chemical Master Equation (CME)

Here, we attempt a stochastic description of biochemical systems with the help of Markov processes as presented in the literature [26, 84, 21, 89]. The presence of intrinsic noise in the way biochemical reactions occur at the single cell level makes a stochastic approach necessary.

2.2.1 Biochemical reaction networks

Molecules inside the cell can collide and, provided there is enough energy, they will react. A biochemical reaction where a molecule of the A species is converted to a molecule of the B species is represented by [83]:

$$A \to B. \tag{23}$$

Following this notation, an A molecule could react with a B molecule and produce a C molecule, i.e.

$$A + B \rightarrow C, \tag{24}$$

This reaction could also be reversed so that a C molecule decomposes back to an A and B molecule:

$$C \rightarrow A + B. \tag{25}$$

Degradation of an A molecule is denoted by

$$A \rightarrow \emptyset, \tag{26}$$

while the production of a molecule of species A can be denoted by

$$\emptyset \rightarrow A. \tag{27}$$

Note here that $\emptyset$ does not correspond to 'nothing', but represents a species that we are not interested in including into the description of the system. Species on the left side of $\rightarrow$ are called reactant species and on the right side product species. Reaction (27) is a zeroth-order reaction, Reactions (23), (25) and (26) are of first-order and Reaction (24) is of second-order. Higher-order reactions are less common [84] and can usually be decomposed into pairs of lower order reactions [89].

Assume a biochemical reaction network consisting of u chemical species $\mathcal{X}_1$, ..., $\mathcal{X}_u$ (usually mRNA or protein) and v reactions R_1, ..., R_v (usually transcription, mRNA degradation, translation, protein degradation) as shown in (28) [89].

$$\begin{aligned} R_1 &: p_{11}\mathcal{X}_1 + p_{12}\mathcal{X}_2 + ... + p_{1u}\mathcal{X}_u \rightarrow q_{11}\mathcal{X}_1 + q_{12}\mathcal{X}_2 + ... + q_{1u}\mathcal{X}_u \\ R_2 &: p_{21}\mathcal{X}_1 + p_{22}\mathcal{X}_2 + ... + p_{2u}\mathcal{X}_u \rightarrow q_{21}\mathcal{X}_1 + q_{22}\mathcal{X}_2 + ... + q_{2u}\mathcal{X}_u \\ &\cdots \\ R_v &: p_{v1}\mathcal{X}_1 + p_{v2}\mathcal{X}_2 + ... + p_{vu}\mathcal{X}_u \rightarrow q_{v1}\mathcal{X}_1 + q_{v2}\mathcal{X}_2 + ... + q_{vu}\mathcal{X}_u \,. \end{aligned} \tag{28}$$

$\boldsymbol{X_t} = (X_1, ..., X_u)^T$ represents the state of the system at time t, i.e. X_i corresponds to the molecules of chemical species $\mathcal{X}_i$ at time t.

We denote with P the $v \times u$ matrix whose elements are given by p_{ij}, and Q

the $v \times u$ matrix given by q_{ij}. Now, we can define the *stoichiometry matrix* S:

$$S = (Q - P)^T. \tag{29}$$

S is a $u \times v$ matrix whose columns represent the effect of individual transitions (reactions) on the state of the network. Each S_{ij} describes the change in the number of molecules of type i from n_i to $n_i + S_{ij}$ due to reaction j [89].

Assume further that the system described by the biochemical network (28) is kept well stirred in a container of constant volume Ω and in thermal equilibrium at a constant temperature T. By keeping the system well stirred, we make sure that the position of any molecule is assumed to be uniformly distributed in the container [27].

2.2.2 Derivation of the CME

We are interested in determining the probability of the system being at different states over time. We provide here a very intuitive derivation of the equation that describes the time evolution of $\boldsymbol{X_t}$, which is known as the Chemical Master Equation (CME). The specific derivation was first presented in [27].

Before we describe the derivation, we need to define some quantites of interest. Each reaction R_i is associated with a scalar called the **stochastic rate constant** and denoted by c_i [89]. The probability that a randomly selected combination of the reactant molecules from R_i reaction will react in the infinitesimal interval $[t, t + dt)$ is given by $c_i dt$. Note that c_i is independent of dt.

We denote by $w_i(X_1, ..., X_u)$ the number of distinct combinations of reactant molecules from R_i reaction at the current state of the system. The form of w_i, where we have dropped the dependency of w_i on $X_1, ..., X_u$, will vary according to the order of the reaction [89].

Consider a first order reaction, for example, Reaction (23), and assume that there are n_A molecules of the reactant species A. In order for Reaction (23) to occur, we need one particular molecule of A to react, but there are n_A molecules of A, such that $w = n_A$. If we look at the second order Reaction (24) and assume that there are n_A molecules of species A and n_B molecules of species B, we would need one particular pair of molecules A and B to react, according to the multiplication principle $w = n_A n_B$. Now, if we had a second order reaction where both reactant species were the same such as $A + A \rightarrow C$ we would need

two distinct molecules of A to react. Given that we have n_A molecules of species A, there are $n_A(n_A-1)/2$ pairs of molecules A, so $w = n_A(n_A-1)/2$. For a zeroth-order reaction of the form (27), we essentially have $w = 1$.

The derivation of the CME in [27] is based on the following three theorems regarding a well stirred thermally equilibrated system such as System (28).

Theorem 1. If $\boldsymbol{X_t} = \boldsymbol{X}$, then the probability that exactly one R_μ reaction occurs in $[t, t+dt)$ is given by $h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)$, where $h_\mu(\boldsymbol{X}, c_\mu) = c_\mu w_\mu$ and $o(dt)/dt \to 0$ as $dt \to 0$.

Proof. Since we are assuming a well stirred system, we can randomly choose any one of the w_μ distinct combinations of the R_μ reactant molecules. Each of these combinations has a probability $c_\mu dt$ of reacting and $1 - c_\mu dt$ of not reacting according to R_μ in $[t, t+dt)$. The probability of a specific one from the w_μ combinations reacting, while the other $w_\mu - 1$ combinations do not react, is given by the multiplication law as $c_\mu dt(1-c_\mu dt)^{w_\mu - 1} = c_\mu dt(1 + (w_\mu - 1)(-c_\mu dt) + \frac{(w_\mu-1)(w_\mu-2)}{2}{c_\mu}^2 dt^2 + ...) = c_\mu dt + o(dt)$. Since we have w_μ distinct combinations, the probability of having any of the w_μ combinations reacting alone is given by p(1st combination only reacting)+p(2nd combination only reacting)+...+p(w_μth combination only reacting)$= w_\mu(c_\mu dt + o(dt)) = w_\mu c_\mu dt + o(dt) = h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)$, as we refer to mutually exclusive events. The function $h_\mu(\boldsymbol{X}, c_\mu)$ is usually called the **hazard function** of reaction R_μ [89]. □

Theorem 2. If $\boldsymbol{X_t} = \boldsymbol{X}$, then the probability that no reaction will occur in the system in the time interval $[t, t+dt)$ is given by $1 - \sum_{\mu=1}^{v} h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)$

Proof. For each of the w_μ distinct combinations of the R_μ reactant molecules at time t, there is a $1 - c_\mu dt$ probability of not reacting according to R_μ in $[t, t+dt)$. So, using the multiplication law, the probability of all w_μ combinations not reacting is given by: $(1-c_\mu dt)^{w_\mu} = 1 + w_\mu(-c_\mu dt) + \frac{w_\mu(w_\mu-1)}{2}{c_\mu}^2 dt^2 + ... = 1 - w_\mu c_\mu dt + o(dt) = 1 - h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)$. Now, we are interested in the probability that none of the v reactions of System (28) will react, i.e. p((no R_1) AND (no R_2) AND ... AND (no R_v)). The probability of the intersection of independent events is given by their product; thus, the probability of no reaction occurring in the system in $[t, t+dt)$ is given by $\prod_{\mu=1}^{v}(1 - h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)) = 1 - \sum_{\mu=1}^{v} h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)$. □

Theorem 3. The probability of more than one reaction occurring in the system in the time interval $[t, t+dt)$ is $o(dt)$.

Proof. Using the multiplication law and the fact that a distinct combination of R_μ reactant molecules has a probability of $c_\mu dt$ to react in $[t, t+dt)$, we conclude that the probability of k reactions to occur in $[t, t+dt)$ is proportional to dt^k and thus $o(dt)$ for $k > 1$. □

At this point, we are able to calculate the probability of the system being at state $\boldsymbol{X}$ at time t, given its state $\boldsymbol{X_0}$ at time t_0, i.e. $p(\boldsymbol{X_t} = \boldsymbol{X} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})$.

We start by expressing the probability $p(\boldsymbol{X_{t+dt}} = \boldsymbol{X} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})$. For this, we have to consider three possible routes to reach $\boldsymbol{X}$ at $t + dt$ from $\boldsymbol{X_0}$ at t_0.

The first route concerns the probability of no reaction occurring in $[t, t + dt)$. In order to be at state $\boldsymbol{X}$ at time $t + dt$, we need to have reached state X at time t from state X_0. According to the multiplication law and ***Theorem 2***, the probability of the first route is given by $p(\boldsymbol{X_t} = \boldsymbol{X} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})(1 - \sum_{\mu=1}^{v} h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt))$.

For the second route we are interested in the probability of having exactly one of the v reactions of System (28) occurring in $[t, t+dt)$. Assuming that the reaction occurring is R_μ, this means that at time t the state of the system $\boldsymbol{X_t} = \boldsymbol{n_1}$ will be updated according to the μth column of the stoichiometry matrix (29): $\boldsymbol{n_1} + S^{(\mu)}$. In order for the system to be at state X at time $t + dt$ it should have reached state $\boldsymbol{n_1} = \boldsymbol{X} - S^{(\mu)}$ at time t. Using ***Theorem 1*** and the multiplication law, the probability of reaching state $\boldsymbol{X} - S^{(\mu)}$ at time t from state X_0 and R_μ taking place in $[t, t + dt)$ is given by $p(\boldsymbol{X_t} = \boldsymbol{X} - S^{(\mu)} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})(h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt))$. However, there are v different reactions that could happen in $[t, t + dt)$; thus, the probability of the second route is given by $\sum_{\mu=1}^{v} p(\boldsymbol{X_t} = \boldsymbol{X} - S^{(\mu)} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})(h_\mu(\boldsymbol{X} - S^{(\mu)}, c_\mu)dt + o(dt)$.

The third route concerns the probability of having more than one reaction occurring at $[t, t + dt)$. According to ***Theorem 3***, the probability of this route is of order $o(dt)$.

All of these routes refer to mutually exclusive events; therefore, we can add

their probabilities and conclude that:

$$\begin{aligned} & p(\boldsymbol{X_{t+dt}} = \boldsymbol{X} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0}) = \\ & = p(\boldsymbol{X_t} = \boldsymbol{X} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})(1 - \sum_{\mu=1}^{v} h_\mu(\boldsymbol{X}, c_\mu)dt + o(dt)) + \\ & + \sum_{\mu=1}^{v} p(\boldsymbol{X_t} = \boldsymbol{X} - S^{(\mu)} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})(h_\mu(\boldsymbol{X - S^{(\mu)}}, c_\mu)dt + o(dt)) + \\ & + o(dt)\,. \end{aligned} \tag{30}$$

If we subtract $p(\boldsymbol{X_t} = \boldsymbol{X} | \boldsymbol{X_{t_0}} = \boldsymbol{X_0})$ from both sides of Equation (30) and divide by dt, we can take the limit as $dt \to 0$, vanishing all $o(dt)/dt$ terms, and consequently have:

$$\begin{aligned} & \frac{dp(\boldsymbol{X}, t)}{dt} = \\ & = \sum_{\mu=1}^{v} [p(\boldsymbol{X} - S^{(\mu)}, t) h_\mu(\boldsymbol{X - S^{(\mu)}}, c_\mu) - p(\boldsymbol{X}, t) h_\mu(\boldsymbol{X}, c_\mu)]\,, \end{aligned} \tag{31}$$

where we have dropped the dependence on the initial state n_0 at t_0. By $p(\boldsymbol{X}, t)$, we refer to the probability of being at state $\boldsymbol{X}$ at time t. This way we have derived the CME which refers to the Equation (31).

It is useful, in particular when Taylor expansions are considered, to write the CME in a more compact form using a step operator $E^{-S_{ij}}$ such as $E^{-S_{ij}} g(..., X_i, ...) = g(..., X_i - S_{ij}, ...)$ and $E^{-S_{ij}} E^{-S_{kj}} g(..., X_i, ..., X_k, ...) = g(..., X_i - S_{ij}, ..., X_k - S_{kj}, ...)$. The CME can be written then as:

$$\frac{dp(\boldsymbol{X}, t)}{dt} = \sum_{j=1}^{v} (\prod_{i=1}^{N} E^{-S_{ij}} - 1) h_j(\boldsymbol{X}, c_j) p(\boldsymbol{X}, t)\,. \tag{32}$$

We can regard the time evolution of System (28) as a continuous-time Markov chain. The Markov property is obvious since the future state (i.e. number of molecules) of the system depends only on the present state. The system evolves at a continuous time, but its state is discrete, as molecule numbers can only take discrete values. In fact, each reaction R_μ occurs according to a Poisson process with rate $h_\mu(\boldsymbol{X}, c_\mu)$. The CME is nothing more than the forward Kolmogorov equation of the Markov process describing System (28).

2.3 The Gillespie algorithm

It is of great interest to be able to simulate realisations of the CME. In [25] we are provided with a stochastic simulation algorithm that allows us to simulate sample paths from the CME. This algorithm, known as the *Gillespie algorithm*, is exact in the sense that it can be derived from the same principles as the CME, which makes them logically equivalent to each other [27].

The Gillespie algorithm refers to systems of the form of System (28) where the three theorems discussed in the previous section hold. The rationale behind the algorithm depends on answering two questions. Given that the system is at state n at time t a) When will the next reaction occur? and b) Which one of the v reactions R_1, ... ,R_v will it be?

Before we answer these two questions, consider the probability that there will be a jump in the time interval $[t, t+dt)$. We already know from ***Theorem 1*** that the hazard of reaction R_μ occurring in $[t, t+dt)$ is given by the hazard function $h_\mu(\boldsymbol{X_t}, c_\mu)$. Thus, the combined hazard expressing the probability of a reaction occurring in $[t, t+dt)$ is given by $h_0(\boldsymbol{X_t}, c_\mu) = \sum_{\mu=1}^{v} h_\mu(\boldsymbol{X_t}, c_\mu)$.

Now, we can answer questions a) and b). The time to the next event can be sampled from $Exp(h_0(\boldsymbol{X_t}, c_\mu))$ and the type of the reaction R_i can be sampled as a discrete random variable with probabilities $h_i(\boldsymbol{X_t}, c_\mu)/h_0(\boldsymbol{X_t}, c_\mu)$.

A pseudocode for the algorithm is given below [89].

1. Initialize the state $\boldsymbol{X_t} = \boldsymbol{n}$ of the system.

2. Calculate the hazard function $h_i(\boldsymbol{X_t}, c_\mu)$ for each reaction of the system.

3. Calculate the combined hazard $h_0(\boldsymbol{X_t}, c_\mu)$.

4. Find the time t_1 for the next reaction by sampling from $Exp(h_0(\boldsymbol{X_t}, c_\mu))$.

5. Update time: $t = t + t_1$

6. Sample the type i of the next reaction according to the probabilities $h_i(\boldsymbol{X_t}, c_\mu)/h_0(\boldsymbol{X_t}, c_\mu)$.

7. Update the state of the system: $\boldsymbol{n} = \boldsymbol{n} + S_i$.

8. While $t < Tmax$, go to step 2.

The Gillespie algorithm becomes very slow as the size of the system increases since it simulates every single reaction occurring. Methods to accelerate the algorithm have been proposed, such as the τ-leap method [30, 11]. The main idea behind the τ-leap method is to calculate all reactions that happen in a time interval of length τ and then update the hazard function. Of course, the method is not exact anymore, leading to an, albeit small, cost in accuracy.

2.4 The LNA as an approximation to the CME

The CME is computationally expensive and, only in rare cases, analytically solvable [55]. Consequently, approximation methods are necessary. We will present the Linear Noise Approximation (LNA), also known as van Kampen's system size expansion [84], as an approximation to the CME. The LNA has been recently successfully applied in Systems Biology applications [18, 47].

The idea behind the LNA is an expansion of the CME in powers of a small parameter that regulates the size of fluctuations in the system. Following [84], we denote with Ω a parameter that for large values of it the fluctuations become relatively small. One such parameter is the size of the system; in the case of System (28), Ω is taken to be the volume of the system. In that case, a small parameter is taken to be the $1/\sqrt{\Omega}$.

Our next step in the expansion of the CME is the inclusion of Ω in Equation (32). We denote with $\boldsymbol{x} = \frac{\boldsymbol{X}}{\Omega}$ the concentration of molecules at the current state of the system and rescale the hazard function $h_\mu(\boldsymbol{X}, c_\mu)$ to include Ω as $\Omega\tilde{f}_j(\boldsymbol{x}, \Omega)$. The CME now becomes:

$$\frac{dp(\boldsymbol{X}, t)}{dt} = \Omega \sum_{j=1}^{v} (\prod_{i=1}^{N} E^{-S_{ij}} - 1)\tilde{f}_j(\boldsymbol{x}, \Omega)p(\boldsymbol{X}, t)\,. \tag{33}$$

It is expected that $p(\boldsymbol{X}, t)$ will have a peak around a macroscopic value of order Ω and width of order $\Omega^{1/2}$ [84]. This would lead us to decompose the system into a deterministic part denoted by $\boldsymbol{\phi}$ and a stochastic part $\boldsymbol{\xi}$:

$$\boldsymbol{X} = \Omega\boldsymbol{\phi} + \Omega^{1/2}\boldsymbol{\xi}\,. \tag{34}$$

We will illustrate the system size expansion of [84] using a simple one dimensional example. Consider the following reaction network, where we simply have

production and degradation of a species X:

$$\emptyset \xrightarrow{\gamma} X \tag{35a}$$

$$X \xrightarrow{\kappa} \emptyset \tag{35b}$$

Reaction 1 happens with a stochastic rate constant γ and reaction 2 happens with a stochastic rate constant κ; $X/\Omega = x$ denotes the concentration of X.

The above equations result in the following stoichiometry matrix:

$$S = \begin{bmatrix} 1 & -1 \end{bmatrix}, \tag{36}$$

and reaction rates:

$$\tilde{f}(\boldsymbol{x}, \Omega) = \begin{bmatrix} \gamma \\ \kappa x \end{bmatrix}, \tag{37}$$

The CME (33) for this reaction can be written using $\Omega\tilde{f}(\boldsymbol{x}, \Omega)$ as:

$$\begin{aligned} &\frac{dp(X,t)}{dt} = \\ &= \Omega\gamma p(X-1,t) - \Omega\gamma p(X,t) + \Omega\kappa\frac{X+1}{\Omega}p(X+1,t) - \Omega\kappa\frac{X}{\Omega}p(X,t) \\ &= \Omega\gamma(E^{-1}-1)p(X,t) + \kappa(E-1)Xp(X,t)\,. \end{aligned} \tag{38}$$

Using now Equation (34), we get that $X = \Omega\phi + \Omega^{1/2}\xi$, and we can express $p(X,t) = p(\Omega\phi + \Omega^{1/2}\xi, t) = \Pi(\xi, t)$. Assuming constant X for the time derivatives, we get that $\frac{dX}{dt} = 0$, which gives us:

$$\frac{d\xi}{dt} = -\Omega^{1/2}\frac{d\phi}{dt}\,. \tag{39}$$

With the help of (39), we can express the left hand side of (38) as:

$$\frac{dp(X,t)}{dt} = \frac{\partial\Pi(\xi,t)}{\partial t} + \frac{\partial\Pi(\xi,t)}{\partial\xi}\frac{d\xi}{dt} = \frac{\partial\Pi(\xi,t)}{\partial t} - \Omega^{1/2}\frac{d\phi}{dt}\frac{\partial\Pi(\xi,t)}{\partial\xi}\,. \tag{40}$$

For the right hand side of (38), care should be taken about the step operator E. We know that the operator changes the molecules of X to $X+1$: $Eg(X) = g(X+1)$. Observe that $X + 1 = \Omega\boldsymbol{\phi} + \Omega^{1/2}\xi + 1 = \Omega\boldsymbol{\phi} + \Omega^{1/2}(\xi + \Omega^{(-1/2)})$, thus E changes ξ to $\xi + \Omega^{-1/2}$. So, $Eg(\xi) = g(\xi + \Omega^{-1/2})$ and this leads us to the

following result:

$$Eg = (1 + \Omega^{-1/2}\frac{\partial}{\partial\xi} + 1/2\Omega^{-1}\frac{\partial^2}{\partial^2\xi})g\,. \tag{41}$$

Equivalently, for E^{-1}, we can find that

$$E^{-1}g = (1 - \Omega^{-1/2}\frac{\partial}{\partial\xi} + 1/2\Omega^{-1}\frac{\partial^2}{\partial^2\xi})g\,. \tag{42}$$

We will now express the right hand side of (38) using (34), (41) and (42).

$$\begin{aligned}&\Omega\gamma(-\Omega^{-1/2}\frac{\partial}{\partial\xi} + \frac{1}{2}\Omega^{-1}\frac{\partial^2}{\partial^2\xi})\Pi(\xi,t)+\\ &+\kappa(\Omega^{-1/2}\frac{\partial}{\partial\xi} + \frac{1}{2}\Omega^{-1}\frac{\partial^2}{\partial^2\xi})(\Omega\phi + \Omega^{1/2}\xi)\Pi(\xi,t)\,.\end{aligned} \tag{43}$$

By looking at (40), we notice that terms of order $\Omega^{1/2}$ will involve the deterministic part ϕ of the state, while terms of order Ω^0 involve the stochastic part. Terms of higher order will not be included in the analysis.

By collecting terms of order $\Omega^{1/2}$ in (43), we get:

$$-\Omega^{1/2}\frac{d\phi}{dt}\frac{\partial\Pi(\xi,t)}{\partial\xi} = -\gamma\Omega^{1/2}\frac{\partial\Pi(\xi,t)}{\partial\xi} + \kappa\Omega^{1/2}\phi\frac{\partial\Pi(\xi,t)}{\partial\xi}\,. \tag{44}$$

If $\frac{\partial\Pi(\xi,t)}{\partial\xi}$ cancel out, we are lead to the deterministic equation:

$$\frac{d\phi}{dt} = \gamma - \kappa\phi\,. \tag{45}$$

Cancelling out $\frac{\partial\Pi(\xi,t)}{\partial\xi}$ is valid since we have made the hypothesis that $X = \Omega\phi + \Omega^{1/2}\xi$, where ϕ corresponds to the solution of the deterministic equation (45).

Collecting now terms of order Ω^0 we get:

$$\begin{aligned}&\Omega^0\frac{\partial\Pi(\xi,t)}{\partial\xi} =\\ &= \Omega\gamma\frac{1}{2}\Omega^{-1}\frac{\partial^2\Pi(\xi,t)}{\partial^2\xi} + \kappa\frac{1}{2}\Omega^{-1}\Omega\phi\frac{\partial^2\Pi(\xi,t)}{\partial^2\xi} + \kappa\Omega^{-1/2}\Omega^{1/2}\frac{\partial}{\partial\xi}\Pi(\xi,t)\xi\,.\end{aligned} \tag{46}$$

which leads to the following Fokker-Planck equation:

$$\frac{\partial\Pi(\xi,t)}{\partial\xi} = \frac{\partial}{\partial\xi}\kappa\xi\Pi(\xi,t) + \frac{1}{2}(\gamma + \kappa\phi)\frac{\partial^2}{\partial\xi^2}\Pi(\xi,t)\,. \tag{47}$$

Equation (47) corresponds to the following SDE:

$$d\xi = -\kappa\xi dt + \sqrt{\kappa + \gamma\phi}dW_t\,. \tag{48}$$

As discussed in Section 2.1, this is a linear SDE in the narrow sense, and its solution is a Gaussian process. Depending on the author, LNA either refers just to this SDE, or to the expansion method of van Kampen in general; we will use the latter definition.

In summary, the LNA decomposes the state of the system into a deterministic part and a stochastic part that represents the fluctuations around the deterministic one as seen in (34). The correctness of this decomposition is supported by the fact that (47) is independent of Ω, indicating that fluctuations were assumed to be of the correct order.

This example has been used to illustrate exactly the method used in [84] for approximating the CME. A detailed derivation of a general multidimensional system can be found in [15]. Fortunately, we can just use the general result of this method in order to approximate the CMEs of more complex networks.

Given a biochemical reaction network with stoichiometry matrix $\boldsymbol{S}$ and the rescaled hazard function defined by $\tilde{f}(\boldsymbol{x}, \Omega)$, we can calculate its deterministic solution by:

$$\frac{d\phi_i}{dt} = S_i\tilde{f}(\phi_t)\,, \tag{49}$$

where i stands for the i-th species. For the linear SDE that characterizes the fluctuations around $\boldsymbol{\phi}$, we need to first calculate the following matrices:

$$A_t = SF_t, \tag{50}$$

where the elements of the matrix F are given by:

$$F_{ij} = \frac{\partial\tilde{f}_i(\phi_t)}{\partial\phi_j(t)}\,, \tag{51}$$

Again, subscript i corresponds to the i-th reaction and j to the j-th species.

$$EE_t{}^T = S\mathrm{diag}(\tilde{f}(\phi_t))S^T\,, \tag{52}$$

where $E_t = S\sqrt{\mathrm{diag}(\tilde{f}(\phi_t))}$. Using (50) and (52), we get the following linear SDE in the narrow sense:

$$d\xi_t = A_t \xi_t dt + E_t dW_t \,. \tag{53}$$

2.4.1 Limitations of the LNA

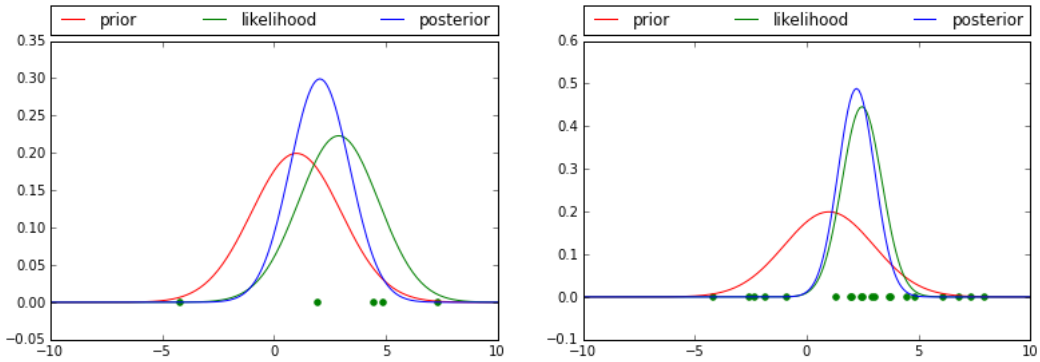
At this point, we would like to discuss some limitations of the LNA and how they can be overcome. Omitting discretisation can become problematic for low molecule numbers. Thus, LNA does not work well when the number of molecules is very low. Therefore, hybrid models have been proposed [71, 76] that partition the system into fast and slow reactions. Fast reactions are simulated using the LNA, and slow reactions are simulated using a discrete scheme such as Gillespie algorithm.

An assumption that is implied by the derivation of the LNA is that the macroscopic state has one stable stationary solution. This assumption keeps the fluctuations around the macroscopic state bounded [84]. However, as discussed in [15] the LNA seems to work well even when there are large fluctuations (e.g. weakly attracting stable state). In addition, the authors suggested a variable transformation to improve LNA in such cases. For bistable systems, the LNA can give misleading results due to its dependence on the deterministic solution, but if the system is very close to its thermodynamic limit, then, for a limited time, the LNA can describe the system quite accurately [86].

The LNA involves a Taylor expansion of up to second order. In [80] it has been shown that in certain cases, where there is a low number of molecules and non-linear reactions, including higher order terms can improve the accuracy of the LNA. As stated in [84], higher order terms can be seen as a non-Gaussian correction to the LNA, but, in most applications, it will not be of practical importance. Although we are not dealing with the limitations of LNA in this thesis, some of the suggested approaches can be used to extend our method when LNA fails to give an accurate estimation. Diagnostic tools for assessing the suitability of the LNA for a specific system have been recently proposed in [24].

2.5 Bayesian Inference

We will begin with a brief introduction to Bayesian inference, which has found many applications in Computational Biology [49]. The Bayesian approach to statistics involves three steps. First, we form a probability distribution to link



(a) Triplot using 5 data points.

(b) Triplot using 20 data points.

Figure 2.3: Triplots of the likelihood, prior and posterior distribution. Data points (denoted by dots) were sampled from N(2,16) and prior on θ is N(1,4).

the model parameters θ with the data X, $p(X|\theta)$, which is called the **likelihood**. Then we incorporate our prior beliefs about the parameters by defining the **prior** distribution $p(\theta)$. This immediately means that in Bayesian statistics, parameters are treated as random variables. Finally, we combine the likelihood and the prior through Bayes' rule to get the **posterior** distribution, which forms an updated distribution for the parameters:

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)}. \tag{54}$$

In Figure 2.3, we can see a plot of the prior, the likelihood and the posterior distribution of a hypothetical model. In Figure 2.3a, we sampled five points from N(2,16), while in Figure 2.3b, we sampled twenty points from N(2,16); in both cases, the prior on θ was set to N(1,4). The posterior distribution combines information of both the data and the prior and lies somewhere between the prior and likelihood. If the prior is not informative, then the posterior will be closer to the likelihood. A narrower curve for the posterior indicates that we have stronger information [59]. As we are getting more and more data, we will move further from the prior and become increasingly confident.

Many problems in Bayesian statistics involve intractable integrals. One example is the marginal likelihood (or evidence) $p(X)$, i.e. the denominator in (54). The marginal likelihood plays an important role in model comparison, but it is usually not possible to compute it analytically. It is computed by marginalising

(integrating) out the parameters of the model:

$$p(X) = \int p(X|\theta)p(\theta)\, d\theta. \tag{55}$$

In the biological applications that we are going to investigate in Chapters 4 and 5, we use Bayesian statistics to infer parameter values of our models for synthetic and real data. This means that we have some prior belief about the parameters, which can be less or more informative, and we want to compute a posterior over the parameters of interest. The posterior probability of one parameter θ_1, regardless of the rest $\theta_2, ..., \theta_n$, is computed by the following, typically intractable, integral:

$$p(\theta_1|X) = \int p(\theta|X)d\theta_2...d\theta_n. \tag{56}$$

In the following, we discuss calculating posterior distributions, which can be difficult due to the denominator in (54). However, we can sample from the posterior using a Markov Chain Monte Carlo (MCMC) algorithm. MCMC is a Monte Carlo method, i.e. a statistical method to compute samples from quantities we cannot compute analytically, such as the integrals in (55) or (56) [48]. We have already seen an example of a Monte Carlo algorithm in Section 2.3, the Gillespie algorithm. In an MCMC algorithm, samples are taken from a Markov chain. The distribution of interest, such as the posterior, is, in that setting, called the **target distribution**. In order to achieve sampling from the target distribution, the Markov chain must be constructed in such a way that it has the target distribution as its unique stationary distribution.

The Metropolis-Hastings (MH) algorithm [41] is one of the most common MCMC algorithms. Given an initial state θ_0, we sample from a proposal transition distribution $q(\theta^*|\theta_{t-1})$ and accept samples according to an acceptance function α. If we want to sample from the posterior, the acceptance function becomes $\alpha(\theta^*, \theta_{t-1}) = \min\left(1, \frac{p(\theta^*|X)q(\theta_{t-1}|\theta^*)}{p(\theta_{t-1}|X)q(\theta^*|\theta_{t-1})}\right) = \min\left(1, \frac{p(X|\theta^*)p(\theta^*)q(\theta_{t-1}|\theta^*)}{p(X|\theta_{t-1})p(\theta_{t-1})q(\theta^*|\theta_{t-1})}\right)$. As we can see, the denominator of (54) cancels out so that the acceptance function can be computed given the likelihood and prior. A common choice for the proposal distribution is a symmetric distribution such as the Gaussian distribution centered at the current state $N(\theta_t, \lambda\Sigma)$. The parameter λ is called the scale factor of the proposal, as it defines the size of the proposal's variance. An MH algorithm with a symmetric distribution is also called a random-walk Metropolis. Below, we

provide pseudocode for the general MH algorithm:

(1) Initialize the state at θ_0 and set $t = 1$.

(2) Sample a new state θ^* from $q(\theta^*|\theta_{t-1})$.

(3) Set θ_t according to $\alpha(\theta^*, \theta_{t-1})$.

   (a) Sample u from $U(0, 1)$.

   (b) If $u < \alpha$: accept θ^* and set $\theta_t = \theta^*$

   (c) Else reject and set $\theta_t = \theta_{t-1}$.

(4) Update time: $t = t + 1$

(5) While $t < Tmax$ go to step 2.

Note that for a symmetric proposal, the acceptance function reduces to just $\alpha(\theta^*, \theta_{t-1}) = \min\left(1, \frac{p(\theta^*|X)}{p(\theta_{t-1}|X)}\right)$.

The resulting sequence of samples forms a time-reversible Markov chain[5] with $p(\theta)$ as its stationary distribution. If we denote with $G(j, i) = G(\theta_t = i|\theta_{t-1} = j)$ the transition matrix of the resulting Markov chain,[6] it can be proved that the following condition holds for the MH algorithm:

$$G(j, i)p(i) = G(i, j)p(j) \tag{57}$$

Condition (57) is called *detailed balance* and it is a sufficient but not a necessary condition for the existence of $p(\theta)$ as the stationary distribution of the Markov chain. Although the construction of the time reversible Markov chain in the MH algorithm guarantees convergence to the target distribution, we cannot know in how many iterations this convergence will be reached. In many cases, convergence can be very slow, and techniques to improve the proposal distribution and accelerate convergence have been studied. We discuss some of these techniques in the following paragraphs.

A lot of effort has been put in finding an optimal acceptance rate for the MH algorithm. In [66] the asymptotically optimal rate for a random walk Metropolis is found to be around 0.234 under general conditions. However, in [67] an

[5]The reversed chain $\theta_T, \theta_{T-1}, ..., \theta_0$ is also a Markov chain

[6]The elements of the transition matrix are given by $g(\theta_t|\theta_{t-1}) = q(\theta_t|\theta_{t-1})\alpha(\theta_t|\theta_{t-1})$ for $\theta_t \neq \theta_{t-1}$ and $g(\theta_{t-1}|\theta_{t-1}) = 1 - \sum_{\theta_t \neq \theta_{t-1}}$

acceptance rate between 0.1 and 0.4 was about optimal for the random walk Metropolis. Convergence can be improved by tuning the scale factor λ to give a reasonable acceptance rate, while recommendations for the value of λ also exist [22]. Depending on the complexity of the problem tuning of the scale factor might not be enough for rapid convergence.

One simple way to improve the performance of the MH is to tune the variance of the proposal distribution so that it resembles the shape of the covariance of the target distribution. By letting the chain run for a sufficient number of iterations, we can calculate an estimate of the covariance from the sample covariance of the posterior samples [22]. The scale factor λ can be tuned afterwards for getting a desirable acceptance rate. Again, this is not an optimal method, especially if we have sampled the covariance from a region of low interest.

An alternative approach would be to tune the MH continuously at runtime. This leads to the concept of adaptive MCMC. However, the adaptation can lead to violations of the assumptions that guarantee convergence to the target distribution. In [68], two conditions that guarantee convergence are presented. The first one is diminishing adaptation, which means that changes to the proposal tend to vanish as the number of iterations tends to infinity. The second one is containment, which implies that the convergence time is bounded.

An example of an adaptive MH algorithm where diminishing adaptation and containment hold was developed in [69, 75]. According to the specific adaptive MH, the new state θ^* is sampled from a mixture of Gaussians:

$$\theta^* = \begin{cases} N(\theta_t, \Sigma_0), & \text{w.p. } \delta \\ N(\theta_t, \lambda\Sigma_t), & \text{w.p. } 1-\delta \end{cases} \tag{58}$$

Σ_t corresponds to the sampled variance up to iteration t and is estimated after enough samples have been accepted. The parameter $\delta \in (0, 1)$ and is defined by the user. The scaling factor λ can either be fixed [69] or be tuned [75, 18]. This algorithm targets an acceptance rate of ≈ 0.3. In Chapters 4 and 5, the MH algorithm, and when necessary the adaptive MH algorithm, will be used for parameter estimation.

2.6 Summary

In this chapter, we presented the mathematical background for studying stochastic systems. We reviewed basic concepts of Markov processes in both continuous and discrete time. We continued by introducing the notation and definitions of biochemical reaction networks and presented a probabilistic proof of the Chemical Master Equation (CME). We showed how we can simulate sample paths of the CME exactly using the Gillespie algorithm. We introduced the Linear Noise Approximation (LNA) as an approximation to the CME and followed van Kampen's proof of the LNA using a simple example. We further discussed limitations of this approximation method. We closed the chapter by a brief introduction to Bayesian inference, which forms the basis of the next chapter, where we are concerned with inference in stochastic systems.

Chapter 3

Kalman Filter inference in stochastic systems

This chapter begins with a discussion of existing methods for inference in stochastic systems that are described by continuous-time Markov chains and observed at discrete times. We introduce the Kalman Filter methodology and show how it can be used for inference in systems described by SDEs. We continue showing how we can incorporate the LNA in a continuous-time Kalman Filter. Finally, we present our method for making inference in biochemical networks given aggregate data that is based, again, on the Kalman Filter and the LNA.

3.1 Existing methods

High quality microscopy data that allow for measurement of molecular processes at the single cell level have recently become available [2, 10, 78]. As discussed in Chapter 1 of this thesis, single cell data are highly stochastic, requiring new modelling approaches to replace ODE models which are appropriate for smoothly varying trajectories observed in data from large cell populations. In many cases, it is not possible to measure all quantities of interest in a system directly due to technical difficulties or high cost. In addition, measurements are usually corrupted by technical noise. It is, therefore, useful to develop the tools for inferring both the state of the system and any unknown parameters from single cell data. Inference in models fitted to single cell data has become an increasingly popular area of research over the last decade. In the Systems Biology literature, more emphasis is put on parameter estimation; this, however, usually requires inference of the latent states of the model. In this section, by *inference* we refer to both state and parameter estimation.

The standard way of describing molecular reactions stochastically was presented in Section 2.2.2. Consequently, we are interested in performing inference

for a continuous-time Markov chain observed at discrete times. In the ideal scenario, where we observe a system completely, i.e. we know all the times and types of reactions occurring at a time interval, we can directly compute and work with the exact likelihood of the system [8]. In a realistic scenario, we will only be able to observe the level of (some) species of a system at discrete times. Even in that case, we are still able to adhere to the exact underlying structure of the stochastic model, i.e. discrete states and continuous time. A Bayesian approach to the problem of inference given discrete data is presented in [8]. There, a sophisticated MCMC algorithm with two blocks is developed, where one block is used to infer the exact process given the data and the parameters, and the second one is used to infer the parameters given the process. In [65], parameter estimation is performed within a Hidden Markov Model (HMM) setting, where hidden states represent molecular numbers and transitions between the states represent different reactions. An approximate likelihood of the HMM is maximised, imposing a restriction on the maximum number of reaction events between the observations and making the method applicable only to frequently sampled time series. In a different work [81], a simulated maximum likelihood approach was adopted. There, a frequency distribution of simulated realisations of the process is used for estimating the transition distribution of the Markov chain and constructing the likelihood.

The methods described so far are computationally expensive since the state space of the Markov chain can become very high dimensional. Therefore, other means of making inference have been further considered in the literature; for example, a variational inference scheme is considered in [61] that has a lower computational cost. However, a more common approach for inference in stochastic biochemical networks is based on approximating the stochastic model itself. This way we can carry out inference on a simplified model. A first thought is to neglect any fluctuations and make a deterministic approximation. Consequently, the system would be described by an ODE instead of a continuous-time Markov chain. However, such an approximation assumes that the fluctuations in the system are negligible. Therefore, this approximation can become particularly problematic in the case of a low molecular number of species in a system, leading to inaccurate results as shown in [81].

Alternatively, a diffusion process can be used to approximate the continuous-time Markov chain. The Chemical Langevin equation (CLE) [29] is a non-linear

SDE that describes the diffusion process that approximates the system. Although using a diffusion approximation simplifies inference in contrast to the exact model, it is still not straightforward. The problem of inference for discretely observed diffusion processes has been studied more thoroughly in the mathematical finance literature, where various techniques have been proposed [77]; it remains an active area of research. Examples of using the CLE for inference in biochemical reaction networks can be found in [36] and [37]. In [36], latent data are augmented between the observations and an MCMC is employed for inference, while in [37], the authors demonstrate how a particle MCMC can be used for inference with the CLE.

In Section 2.4, we discussed the Linear Noise Approximation (LNA) as an alternative approximation of the CME. The LNA decomposes the system into a macroscopic part, given as a solution to a deterministic system, and the fluctuations around it, described by a linear (multivariate) SDE in the narrow sense. Different authors have argued in favor of the LNA [84] or the CLE [29, 80], while in [86] the LNA is viewed as an approximation to the CLE. Note that both diffusion approximations are valid for systems that are close to the thermodynamic limit (i.e. large system size) [86]. In our work, the tractability of the SDE arising from the LNA gives the LNA an advantage over the CLE.

Applications of the LNA for inference in biochemical reaction networks can be found in [47, 18, 79]. In all these cases, parameter estimation has been achieved with an MCMC algorithm ranging from a simple Metropolis-Hastings [47] to a Riemannian-Manifold MCMC [79] that leads to an improved mixing of the chain.

We are interested in using the LNA for making inference given noisy, discrete and aggregated (partial) observations of the system. Fearnhead et al. [18] developed a continuous-discrete Kalman filter to deal with noisy and discrete (partial) observations of a system. We will extend this work and develop a Kalman filter framework for aggregated data. In the following sections, we provide the theory for optimal filtering and present our work for the aggregated case.

3.2 Optimal filtering

Here, we focus on the problem of inferring the state x_k of a system using noisy, indirect or even partial observations $y_{1:k}$ up to time k. This problem is known in

the literature as filtering [45] and has its roots in the Wiener filter [88] for stationary signals. In 1960, a recursive algorithm based on least squares was proposed for solving the filtering problem in linear systems known as the Kalman Filter (KF) [45]. The KF became popular due to its simplicity and found immediate applications in engineering. The Bayesian community has also been interested in the problem of filtering [44, 43], and extensions for non-linear systems have been developed as well [72, 73].

The KF aims at the computation of the marginal posterior distribution of the state $p(x_k|y_{1:k})$. The reason for computing the marginal posterior instead of the joint posterior of all states is the reduced computational complexity of the marginal posterior since the posterior needs to be updated for every observation [73]. The marginal posterior $p(x_k|y_{1:k})$ can be calculated recursively given the following quantities:

1. The prior distribution $p(x_0)$ that initialises the recursion.
2. The predictive distribution $p(x_k|y_{1:k-1}) = \int p(x_k, x_{k-1}|y_{1:k-1})dx_{k-1}$.

At each recursion, the posterior of the previous step is assumed to be the prior distribution for the current step. The KF can also be used for inferring unknown parameters of a system. For that task, we would need to compute the likelihood function of the parameters of the system given the observations, as further discussed in Sections 3.3 and 3.4. In the following, we introduce the discrete KF, where both the state and the observations of the system are discrete-time stochastic processes before we move on to the continuous-discrete case that deals with discrete observations from a system described by a continuous-time process.

3.2.1 The discrete Kalman Filter

We will represent the state x_t and the observation process y_t of the system in a state-space model such as:[1]

$$x_t = A_{t-1}x_{t-1} + \zeta_{t-1}\,, \tag{1a}$$

$$y_t = P_t x_t + \epsilon_t\,. \tag{1b}$$

[1]In the literature, it is common to have a control vector u_t in the state process such as $x_t = A_{t-1}x_{t-1} + W_t u_t + \zeta_{t-1}$, but that is not required in our applications.

where $\zeta_t \sim N(0, Q_t)$ and $\epsilon_t \sim N(0, R_t)$ represent Gaussian noise. The state process has the form of an AR(1) process. P_t is the observability matrix that deals with partial and indirect observations. In case of a fully observed system, P_t will be of full rank.

The following assumptions are made for the state-space model:

1. The states x_t form a Markov chain, so $p(x_t|x_{1:t-1}, y_{1:t-1}) = p(x_t|x_{t-1})$.

2. The measurements y_t are conditionally independent of past measurements and states, $p(y_t|x_{1:t}, y_{1:t-1}) = p(y_t|x_t)$.

We can represent a state-space model by a directed acyclic graph [6]. The nodes of the graph represent the random variables of the system (discrete or continuous) and the arrows correspond to dependencies between the random variables. The graphical model represents the joint distribution of the system. The state process of the discrete KF is a discrete-time, continuous-state Markov process which is observed through noisy measurements at discrete times. A graphical representation of the state-space model (1) is shown in Figure 3.1, from where we can deduce that the joint distribution can be decomposed according to $p(x_1, x_2, x_3, ..., y_1, y_2, y_3, ...) = p(x_1)p(x_2|x_1)p(x_3|x_2)....p(y_1|x_1)p(y_2|x_2)p(y_3|x_3)...$.

Since we have assumed Gaussian noise, Equation (1) can be written in probabilistic terms as:

$$p(x_t|x_{t-1}) = N(x_t|A_{t-1}x_{t-1}, Q_{t-1})\,, \tag{2a}$$

$$p(y_t|x_t) = N(y_t|P_t x_t, R_t)\,. \tag{2b}$$

We are now ready to write the recursive equations that are needed to compute the marginal posterior $p(x_k|y_{1:k})$. Different derivations of the KF equations have been suggested in the literature. Here, we show a purely probabilistic derivation as in [72, 43, 44].

We start by computing the predictive distribution $p(x_t|y_{1:t-1})$ at time t. Using the Markov property of the states we can write:[2]

$$\begin{aligned} p(x_t, x_{t-1}|y_{1:t-1}) &= p(x_t|x_{t-1}, y_{1:t-1})p(x_{t-1}|y_{1:t-1})\,, \\ &= p(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})\,. \end{aligned} \tag{3}$$

[2] $P(A, B|C) = \frac{P(A,B,C)}{P(C)} = \frac{P(A|B,C)P(B,C)}{P(C)} = P(A|B, C)P(B|C)$

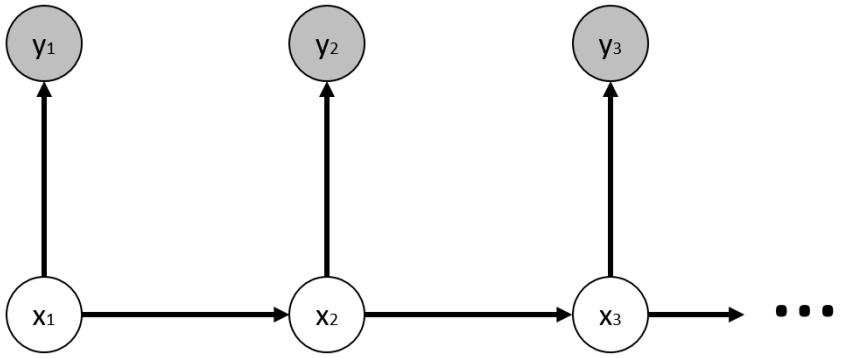


Figure 3.1: Graphical representation of a discrete Kalman Filter. The shaded circles correspond to the observations and the unshaded to the latent states.

We already know that $p(x_t|x_{t-1}) = N(x_t|A_{t-1}x_{t-1}, Q_{t-1})$, and we further assume that the posterior at the previous time step is known and follows a Gaussian distribution[3] $p(x_{t-1}|y_{1:t-1}) = N(m_{t-1}, \Sigma_{t-1})$. Using Lemma 2 from Appendix A.3 we conclude:

$$\begin{bmatrix} x_{t-1} \\ x_t \end{bmatrix} |y_{1:t-1} \sim N\left(\begin{bmatrix} m_{t-1} \\ A_{t-1}m_{t-1} \end{bmatrix}, \begin{bmatrix} \Sigma_{t-1} & \Sigma_{t-1}A_{t-1}^T \\ A_{t-1}\Sigma_{t-1} & A_{t-1}\Sigma_{t-1}A_{t-1}^T + Q_{t-1} \end{bmatrix} \right) \quad (4)$$

In order to get $p(x_t|y_{1:t-1})$ from (4) we need to use Lemma 1 (Appendix A.3), which gives:

$$p(x_t|y_{1:t-1}) = N(A_{t-1}m_{t-1}, A_{t-1}\Sigma_{t-1}A_{t-1}^T + Q_{t-1}) = N(m_t^-, \Sigma_t^-)\,. \quad (5)$$

In the same way, we can now derive the marginal posterior distribution $p(x_t|y_{1:t})$ at time t. We start by computing the joint distribution:[4] $p(x_t, y_t|y_{1:t-1}) = p(y_t|x_t, y_{1:t-1})p(x_t|y_{1:t-1}) = p(y_t|x_t)p(x_t|y_{1:t-1})$. We have knowledge of both distributions $p(y_t|x_t) = N(y_t|P_tx_t, R_t)$ and $p(x_t|y_{1:t-1}) = N(m_t^-, \Sigma_t^-)$, so we can use again Lemma 2 (Appendix A.3) and conclude that:

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} |y_{1:t-1} \sim N\left(\begin{bmatrix} m_t^- \\ P_tm_t^- \end{bmatrix}, \begin{bmatrix} \Sigma_t^- & \Sigma_t^-P_t^T \\ P_t\Sigma_t^- & P_t\Sigma_t^-P_t^T + R_t \end{bmatrix} \right). \quad (6)$$

[3]The joint distribution of a directed graph of linear Gaussian units is Gaussian, leading to all marginals and conditional distributions being Gaussian too [6].

[4]We make use of the conditional independence of measurements $p(y_t|x_t, y_{1:t-1}) = p(y_t|x_t)$.

Finally, we can compute $p(x_t|y_{1:t})$ by noting that $p(x_t|y_t, y_{1:t}) = p(x_t|y_{1:t})$. By applying Lemma 1 (Appendix A.3) on the joint distribution (6), we can find the conditional distribution $p(x_t|y_t, y_{1:t})$:

$$x_t|y_{1:t} \sim N(m_t^- + K_t[y_t - P_t m_t^-], \Sigma_t^- - K_t P_t \Sigma_t^-)\,, \tag{7}$$

where $K_t = \Sigma_t^- P_T^T [P_t \Sigma_t^- P_t^T + R_t]^{-1}$ is the Kalman gain.

In summary, we have derived the following results:

- The predictive distribution is $p(x_t|y_{1:t-1}) = N(x_t|m_t^-, \Sigma_t^-)$.
- The posterior distribution is $p(x_t|y_{1:t}) = N(x_t|m_t, \Sigma_t)$,

where m_t^-, Σ_t^- and m_t, Σ_t are given by:

$$m_t^- = A_{t-1} m_{t-1}\,, \tag{8}$$

$$\Sigma_t^- = A_{t-1} \Sigma_{t-1} A_{t-1}^T + Q_{t-1}\,, \tag{9}$$

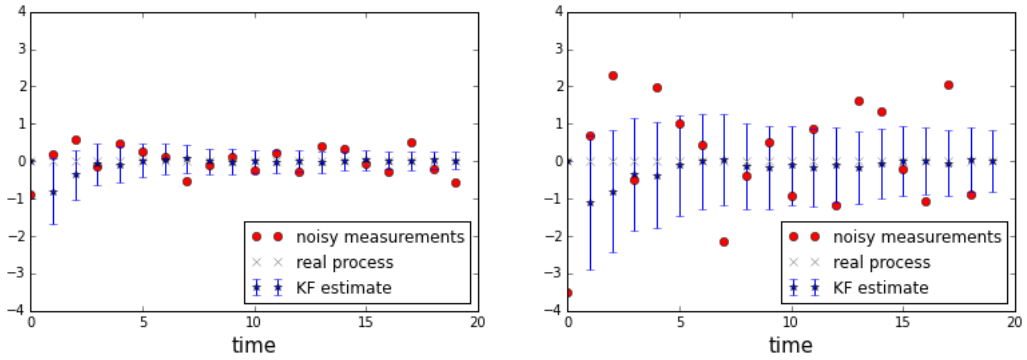
$$K_t = \Sigma_t^- P_t^T [P_t \Sigma_t^- P_t^T + R_t]^{-1}\,, \tag{10}$$

$$m_t = m_t^- + K_t[y_t - P_t m_t^-]\,, \tag{11}$$

$$\Sigma_t = \Sigma_t^- - K_t P_t \Sigma_t^-\,. \tag{12}$$

The KF recursions (8), (9), (11), (12) tell us that at each time point, we make a prediction about the state (Equations (8) and (9)) which is subsequently corrected when we make a new observation (Equations (11) and (12)). The Kalman gain (10) tells us how much we can trust the observations. A high Kalman gain will give more weight in the observations. Eventually, the KF is making predictions of the actual state of the system that are corrected every time we have an observation iteratively . The KF recursions we have derived give the best state estimate for a Gaussian linear system and coincide with the least square solution developed in [45].

In Figure 3.2, we provide filtering results from a discrete KF. The discrete true



(a) $R_t = 0.5$ (b) $R_t = 2.0$

Figure 3.2: Filtering results from a discrete KF from observations at different noise levels, $R_t = 0.5$ (a) and $R_t = 2.0$ (b).

state process [5] is shown with grey crosses, and noisy observations are shown with red dots. The posterior mean of the KF is shown as a blue star along with one standard deviation. In Figure 3.2(a), we have generated data with observation noise set to 0.5, while in Figure 3.2(b), observation noise was set to 2.0. As we can see from the larger error bars, adding more observation noise will increase uncertainty.

3.2.2 The continuous-discrete Kalman Filter

The continuous-discrete KF [44, 72] refers to a system with continuous-time states that we observe discretely over time. The state process is now described by an SDE, while the observation process has the same form as in the discrete KF. The state and observation process are now described by the following state-space model:

$$dx_t = \mu(t, x_t)dt + \sigma(t, x_t)dW_t\,, \tag{13a}$$

$$y_t = P_t x_t + \epsilon_t\,. \tag{13b}$$

As in the discrete case, we need to be able to calculate the transition probability of the states $p(x_t|x_{t-1})$, which is not analytically tractable for general SDEs. However, in the case of linear systems that evolve according to a linear SDE

[5] corresponding to $A_t = 1$ and $Q_t = 0.00001$

(in the narrow sense), which we studied in Chapter 2, there exists an analytical solution. We will focus on linear filtering and, henceforth, will only refer to the linear case of the discrete-continuous KF which is represented by the following state-space model:

$$dx_t = F_t x_t dt + G_t dW_t \,, \tag{14a}$$

$$y_t = P_t x_t + \epsilon_t \,. \tag{14b}$$

The graphical representation of the continuous-discrete KF is shown in Figure 3.3. In the continuous-discrete KF, we assume that the states are continuous—in contrast with the discrete KF, where the system is assumed to have discrete states. The graphical model of Figure 3.3 extends the graphical model of Figure 3.1 by assuming an infinite number of states between the observation times.

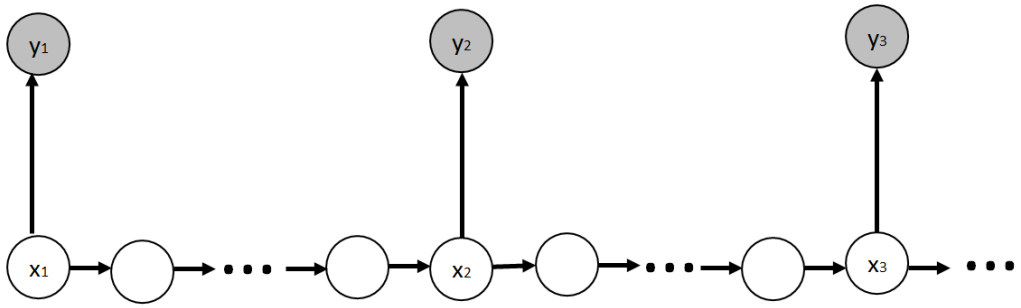


Figure 3.3: Graphical representation of a continuous-discrete Kalman Filter. The shaded circles correspond to the observations and the unshaded circles to the continuous states, there are infinitely many states between the observation points.

The transition probabilities of a linear SDE in the narrow sense have been shown to follow a Gaussian distribution with mean and variance given by Equations (19) and (21), see Chapter 2. We can use this result to write the state-space model of the continuous-discrete KF in probabilistic terms such as:

$$p(x_t|x_{t-1}) = N(x_t|A_{t-1}x_{t-1}, Q_{t-1}) \,, \tag{15a}$$

$$p(y_t|x_t) = N(y_t|P_t x_t, R_t) \,, \tag{15b}$$

where the matrices A_{t-1} and Q_{t-1} can be calculated by the following ODEs:

$$\frac{dA_t}{dt} = F_t A_t \,, \tag{16a}$$

$$\frac{dQ_t}{dt} = F_t Q_t + Q_t F_t^T + G_t G_t^T \,. \tag{16b}$$

In order to calculate the predicted mean and variance we move on by solving forward the ODEs for the mean and variance of the state process initialised at their previous posterior values [4]. The prediction and update steps of the continuous-discrete KF are summarized in the following paragraph.

- The mean and variance of the predictive distribution are found by solving the ODEs:

$$\frac{dm_t^-}{dt} = F_t m_t^- \,, \tag{17a}$$

$$\frac{d\Sigma_t^-}{dt} = F_t \Sigma_t^- + \Sigma_t^- F_t^T + G_t G_t^T \,, \tag{17b}$$

  initialised at the previous posterior mean m_{t-1} and variance Σ_{t-1}.

- The posterior mean m_t and variance Σ_t are given by:

$$K_t = \Sigma_t^- P_t^T [P_t \Sigma_t^- P_t^T + R_t]^{-1} \,, \tag{18}$$

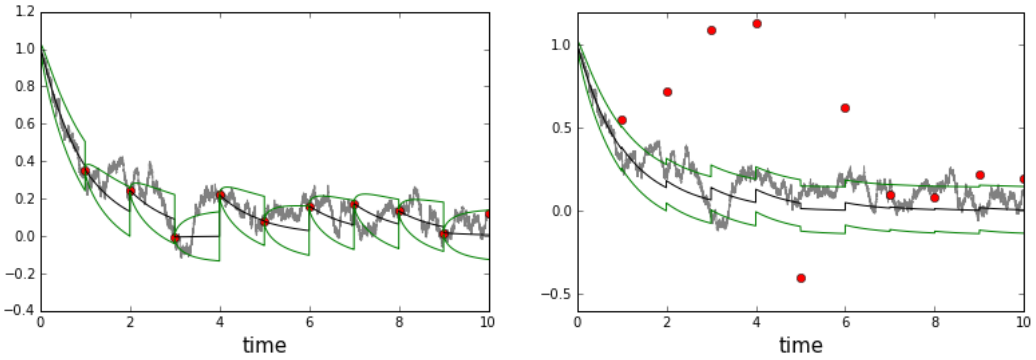
$$m_t = m_t^- + K_t[y_t - P_t m_t^-], \tag{19}$$

$$\Sigma_t = \Sigma_t^- - K_t P_t \Sigma_t^- \,. \tag{20}$$

Note that, since we have discrete observations, the observation process remains the same as in the discrete case, and so do the updated mean and variance of the posterior $p(x_t|y_{1:t}) = N(x_t|m_t, \Sigma_t)$.

In Figure 3.4, we present estimate results from a continuous-discrete KF, where we have assumed discrete observations from a linear SDE.[6] In panel (a) we have noisy ($R_t = 0.01$) discrete observations sampled every 1 minute, whereas in

[6]We have used an Ornstein-Uhlenbeck process which will be studied extensively in Section 4.1.



(a) $R_t = 0.01$ (b) $R_t = 0.5$

Figure 3.4: Filtering results from a continuous-discrete KF from observations at different noise levels, $R_t = 0.01$ (a) and $R_t = 0.5$ (b). The grey trace represents the SDE driving the state process and red dots represent noisy observations. Blue lines correspond to the posterior mean estimate and green lines to 1 s.d. .

panel (b) we have increased the noise ($R_t = 0.5$). The grey trace corresponds to the underlying SDE, red circles correspond to the noisy observations, while the black and green lines represent the posterior mean estimate of the KF and one standard deviation respectively. As we can see, the estimate between the observations is smooth, while, at observation time, there are discontinuities and jumps towards the observations. This is a characteristic behaviour of the continuous-discrete KF since the observations are discrete but the states are described by a continuous-time process. By comparing the two panels, we can observe the effect of observation noise in our estimations. In panel (a) we are, due to the low noise, very confident at observation time, while in panel (b) the jumps towards the observations are less extreme, since the increased noise makes us less certain about our observations, indicating a lower Kalman gain.

3.3 Kalman Filter for the LNA

The KF can be used in conjunction with the LNA for making inference in stochastic systems that are observed discretely over time and whose dynamics can be approximated by the LNA. In Chapter 2, we have seen that the LNA decomposes the system $\boldsymbol{X_t}$ into a deterministic part $\boldsymbol{\phi_t}$ and a stochastic part $\boldsymbol{\xi_t}$:

$$\boldsymbol{X_t} = \Omega\boldsymbol{\phi_t} + \Omega^{1/2}\boldsymbol{\xi_t}\,. \tag{21}$$

The solution to the SDE driving the stochastic part $\boldsymbol{\xi_t}$ was shown to be a Gaussian process with a known mean and variance that can be calculated by a set of ODEs. The initial distribution of the state X_0 can be either set to a constant or follow a Gaussian distribution $X_0 \sim N(\mu_0, \Sigma_0)$. The ODEs that give the deterministic part along with the mean and variance of the stochastic part are repeated here for convenience:

$$\frac{d\phi_t}{dt} = S_t \tilde{f}(\phi_t)\,, \tag{22}$$

$$\frac{dm_t}{dt} = A_t m_t\,, \tag{23}$$

$$\frac{dV_t}{dt} = V_t {A_t}^T + A_t V_t + E {E_t}^T\,. \tag{24}$$

The system is approximated by a Gaussian process as it is the sum of a deterministic term and a Gaussian term. By taking the expectation and variance in Equation (21), we conclude that:

$$\boldsymbol{X_t}|\boldsymbol{X_0} \sim N(\Omega \boldsymbol{\phi_t} + \Omega^{1/2}\boldsymbol{m_t}, \Omega V_t) = N(\mu_t^1, \Sigma_t^1)\,. \tag{25}$$

Assuming that the observation process is given, again, by $y_t = P_t X_t + \epsilon_t$ with $\epsilon_t \sim N(0, R_t)$, the continuous-discrete KF can be used to carry on inference.

- The predictive distribution $p(X_t|y_{1:t-1}) = N(\mu_t^{1^-}, \Sigma_t^{1^-})$ is calculated by solving ODEs (22), (23) and (24). There are two ways to proceed for solving these ODEs.

  1. ODE (22) concerning the deterministic part ϕ_t is solved subject to the initial observation until time t. At each observation point, the initial condition for the ODE of the stochastic mean m_t is updated according to the posterior mean μ_t^1 and Equation (21) [47].
  2. The deterministic solution is restarted at each observation point according to the posterior mean μ_t^1. This implies that the stochastic mean m_t is reset to zero at each observation point [18].

  We will refer to the first method as the Non-Restarting LNA and to the second one as the Restarting LNA in accordance with [23].

- The mean and variance of the posterior distribution $p(X_t|y_{1:t}) = N(\mu_t^1, \Sigma_t^1)$ are given by Equations (19) and (20), so for both the Restarting and the Non-Restarting LNA we have:

$$K_t = \Sigma_t^{1^-} P_t^T [P_t \Sigma_t^{1^-} P_t^T + R_t]^{-1}\,, \tag{26}$$

$$\mu_t^1 = \mu_t^{1^-} + K_t[y_t - P_t \mu_t{}^-]\,, \tag{27}$$

$$\Sigma_t^1 = \Sigma_t^{1^-} - K_t P_t \Sigma_t^{1^-}\,. \tag{28}$$

The Restarting and Non-Restarting LNA have been studied and compared in [23] and [18]. By restarting the deterministic part at each observation point, we get an improved approximation of the system dynamics, since the LNA depends on the deterministic solution which can become inaccurate over long periods of time, especially in non-linear systems. A computational advantage of the restarting method is that Equation (23) does not need to be solved if it is initialised to zero, since it will always be zero. For these reasons, the Restarting LNA will be the preferred method in this thesis.

A major challenge in Systems Biology is parameter inference. The setting that we have assumed so far can also be used for inferring the unknown parameters of the system. We will denote the set of unknown parameters by the vector $\theta = (\theta_1, ..., \theta_n)$; in biological models, θ usually corresponds to reaction rate constants. For this task, we will need to obtain the likelihood $L(\theta)$ of the system:

$$L(\theta) = p(y_1, ..., y_t|\theta) = p(y_1|\theta) \prod_{i=1}^{t} p(y_i|y_{1:i-1}, \theta)\,. \tag{29}$$

According to the observation process $y_t = P_t X_t + \epsilon_t$ and the predictive distribution $p(X_t|y_{1:t-1}) = N(\mu_t^{1^-}, \Sigma_t^{1^-})$, the conditional distribution of the observations will be:

$$p(y_t|y_{1:t-1}) = N(P_t \mu_t^{1^-}, P_t \Sigma_t^{1^-} P_t^T + R_t)\,. \tag{30}$$

The likelihood can then be used for obtaining estimates of the parameter vector θ. If a frequentist approach is considered, the likelihood can be directly maximised, either analytically or via a numerical optimisation algorithm such

as Nelder-Mead [57]. In the case of a Bayesian approach, priors should be placed on the parameters which, together with the likelihood, will be used to form the posterior distribution that can be computed with an MCMC algorithm (e.g. Metropolis-Hastings). A review of MCMC methods for state-space models can be found in [17]. The algorithms for calculating the likelihood of the system using both the Restarting and the Non-Restarting LNA are presented below.

Algorithm 3.3.1: Kalman Filter for Non Restarting LNA

procedure LIKELIHOOD($y_{1:T}, \theta$)
  Initialisation ($t = 0$) *Set prior for* $X_0 \sim N(\mu_0^{1^-}, \Sigma_0^{1^-})$.
  Set initial conditions for the ODEs (22), (23), (24): $\phi_0 = \frac{\mu_0^1}{\Omega}$, $m_0 = 0$ *and* $V_0 = \frac{\Sigma_0^1}{\Omega}$.
  $prod \leftarrow 1$
loop:
  Solve the ODEs (22), (23), (24) of ϕ_t m_t *and* V_t *s.t. the initial conditions for* $[t-1, t]$ *to obtain* $\mu_t^{1^-}$ *and* $\Sigma_t^{1^-}$.
  Calculate $p(y_t|y_{1:t-1}, \theta)$, μ_t^1 *and* Σ_t^1 *according to (29), (27) and (28).*
  $prod \leftarrow prod * p(y_t|y_{1:t-1}, \theta)$.
  Reset initial conditions: $m_t = \frac{\mu_t^1 - \Omega\phi_t}{\sqrt{\Omega}}$ *and* $V_t = \frac{\Sigma_t^1}{\Omega}$.
  Set $t = t + 1$
  if $t < T$ **goto** *loop* .
  Return prod
end procedure

3.4 Kalman Filter Aggregate LNA

In this section, we present our method for making inference in stochastic systems with aggregated data, i.e. the observations are aggregated over a period of time and then collected as a single point. The tricky part of making inference with integrated diffusions is that, as we will see later, they do not possess the Markov property. In our approach, we are using a KF for the bivariate process consisting of the integrated and the original process.

Inference in integrated diffusions has been considered in particular in finance, and various estimators of unknown parameters have been suggested. In [33] parameter estimation of the integrated Ornstein-Uhlenbeck process is considered using an estimator that is asymptotically equivalent to the maximum likelihood

Algorithm 3.3.2: Kalman Filter for Restarting LNA

procedure LIKELIHOOD($y_{1:T}, \theta$)
  Initialisation ($t = 0$) *Set prior for* $X_0 \sim N(\mu_0^{1^-}, \Sigma_0^{1^-})$.
  Set initial conditions for the ODEs (22) and (24): $\phi_0 = \frac{\mu_0^1}{\Omega}$ *and* $V_0 = \frac{\Sigma_0^{1^-}}{\Omega}$.
  $prod \leftarrow 1$
loop:
  Solve the ODEs (22), (24) of ϕ_t *and* V_t *s.t. the initial conditions for* $[t-1, t]$ *to obtain* $\mu_t^{1^-}$ *and* $\Sigma_t^{1^-}$.
  Calculate $p(y_t|y_{1:t-1}, \theta)$, μ_t^1 *and* $\Sigma_t^{1^-}$ *according to (29), (27) and (28).*
  $prod \leftarrow prod * p(y_t|y_{1:t-1}, \theta)$.
  Reset initial conditions: $\phi_t = \frac{\mu_t^1}{\Omega}$ *and* $V_t = \frac{\Sigma_t^1}{\Omega}$.
  Set $t = t + 1$
  if $t < T$ **goto** *loop* .
  Return prod
end procedure

solution. A non-parametric estimator for discretely observed integrated diffusions is suggested in [12]. Additionally, a simulated EM algorithm is proposed in [3].

Here, we assume that the dynamics of the system can be approximated using the LNA, so the system is decomposed according to (21). Integration can be regarded as an infinite summation. Therefore, we will estimate the aggregated process of the system X_t by its integral. Assuming that we have aggregated observations over the period $[t_0, t]$, the corresponding aggregated process is given by:

$$H_t = \int_{t_0}^{t} X_u du = \Omega \int_{t_0}^{t} \phi_u du + \Omega^{1/2} \int_{t_0}^{t} \xi_u du = \Omega I_t + \Omega^{1/2} Q_t \,. \tag{31}$$

So, the deterministic part of the aggregated process is given by I_t, and the stochastic part is given by Q_t. Subsequently, we have the following ODEs:

$$\frac{dI_t}{dt} = \frac{d}{dt} \int_{t_0}^{t} \phi_u du = \phi_t \,, \tag{32}$$

$$\frac{dQ_t}{dt} = \xi_t \,. \tag{33}$$

Q_t will also follow a Gaussian process, as it is the integral of a Gaussian process and we need to compute its mean and variance. We will use the Restarting LNA;

as a result, the mean m_t of the stochastic part ξ_t will be zero.

We start by computing $\mathbb{E}[Q_t]$, i.e. the **mean** of Q_t. So far we have that:

$$d\xi_t = A_t\xi_t dt + E_t dW\,, \tag{34}$$

$$dQ_t = \xi_t dt \Leftrightarrow Q_{t+dt} = Q_t + \xi_t dt\,. \tag{35}$$

- Averaging Equation (35), dividing by dt and letting $dt \to 0$, gives us:

$$\begin{aligned} \mathbb{E}[Q_{t+dt}] &= \mathbb{E}[Q_t] + \mathbb{E}[\xi_t]dt \\ \mathbb{E}[Q_{t+dt}] - \mathbb{E}[Q_t] &= \mathbb{E}[\xi_t]dt \\ \frac{dE[Q_t]}{dt} &= \mathbb{E}[\xi_t] = m_t = 0 \end{aligned} \tag{36}$$

  The mean of Q_t is set to zero, as we have chosen to use the Restarting LNA.

We now need to compute the **covariance** between Q_t and ξ_t. Again $\mathbb{E}[Q_t] = 0$ and $\mathbb{E}[\xi_t] = 0$ since we are using the Restarting LNA and thus, the covariance is given by $C_t = \mathbb{E}[Q_t\xi_t^T]$. For our derivation, we need to use:

$$\xi_{t+dt}^T = \xi_t^T + \xi_t^T A_t^T dt + E_t^T dW_t\,. \tag{37}$$

- By multiplying Equations (35) and (37) we get:

$$\begin{aligned} Q_{t+dt}{\xi_{t+dt}}^T &= (Q_t + \xi_t dt)(\xi_t^T + \xi_t^T A^T dt + E_t^T dW_t) \\ &= Q_t\xi_t^T + Q_t\xi_t^T A_t^T dt + Q_t E_t^T dW_t + \\ &+ \xi_t\xi_t^T dt + \xi_t\xi_t^T A_t^T dtdt + \xi_t E_t^T dtdW_t\,. \end{aligned} \tag{38}$$

  Averaging the result (38), retaining terms up to first order in dt, dividing by dt and letting $dt \to 0$, we get:

$$\begin{aligned} \mathbb{E}[Q_{t+dt}{\xi_{t+dt}}^T] &= \mathbb{E}[Q_t\xi_t^T] + \mathbb{E}[Q_t\xi_t^T]A_t^T dt + \mathbb{E}[Q_t dW_t]E_t^T + \mathbb{E}[\xi_t\xi_t^T]dt\,, \\ \frac{d\mathbb{E}[Q_t\xi_t^T]}{dt} &= \mathbb{E}[Q_t\xi_t^T]A(t)^T + \mathbb{E}[\xi_t\xi_t^T]\,, \\ \frac{dC_t}{dt} &= C_t A(t)^T + V_t \end{aligned} \tag{39}$$

- The **variance** of Q_t is given by $G_t = \mathbb{E}[Q_t Q_t^T]$ since $\mathbb{E}[Q_t] = 0$. We have that,

$$\begin{aligned} Q_{t+dt}{Q_{t+dt}}^T &= (Q_t + \xi_t dt)(Q_t + \xi_t dt)^T \,, \\ Q_{t+dt}{Q_{t+dt}}^T &= Q_t Q_t^T + Q_t \xi_t^T dt + \xi_t Q_t^T dt + \xi_t \xi_t^T dt dt \,. \end{aligned} \tag{40}$$

By averaging (40), retaining terms up to first order in dt, dividing by dt and letting $dt \to 0$, we get:

$$\begin{aligned} \mathbb{E}[Q_{t+dt}{Q_{t+dt}}^T] &= \mathbb{E}[Q_t Q_t^T] + \mathbb{E}[Q_t \xi_t^T] dt + \mathbb{E}[\xi_t Q_t^T] dt \,, \\ \mathbb{E}[Q_{t+dt}{Q_{t+dt}}^T] - \mathbb{E}[Q_t Q_t^T] &= \mathbb{E}[Q_t \xi_t^T] dt + \mathbb{E}[\xi_t Q_t^T] dt \,, \\ \frac{dG_t}{dt} &= \mathbb{E}[Q_t \xi_t^T] + \mathbb{E}[\xi_t Q_t^T] \,, \\ \frac{dG_t}{dt} &= C_t + {C_t}^T \,. \end{aligned} \tag{41}$$

If instead we were using the Non-Restarting LNA, additional ODEs would have to be solved, since $\mathbb{E}[Q_t] \neq 0$ and $\mathbb{E}[\xi_t] \neq 0$, so the variance and covariance terms would not be given by G_t and C_t anymore. It is straightforward to extend to the Non-Restarting case, and the relevant ODEs can be found in Appendix A.5.

We are now at a position to construct a Kalman Filter framework to carry out inference. So far, we have been dealing with Markovian processes. The aggregated process Q_t, however, is clearly not Markovian, since knowing the state at a present time t is not enough for determining the state at a future time $t + dt$. However, if we also knew the state of ξ_t at time t, we would be able to determine Q_{t+dt}. As a result, Q_t and ξ_t jointly form a bivariate Markov process that is characterised by the following linear SDE in the narrow sense:

$$d \begin{bmatrix} \xi_t \\ Q_t \end{bmatrix} = \begin{bmatrix} A_t & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \xi_t \\ Q_t \end{bmatrix} dt + \begin{bmatrix} E_t \\ 0 \end{bmatrix} dW_t, \begin{bmatrix} \xi_0 \\ Q_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} . \tag{42}$$

We know that the solution to a linear SDE in the narrow sense is a Gaussian process. From Equation (42) we have that Q_t and ξ_t are jointly Gaussian and consequently their marginals are also Gaussians. Equation (31) suggests that H_t will follow a Gaussian process, since it is the sum of a Gaussian random variable with a deterministic variable:

$$H_t | H_0, X_0 \sim N(\mu_t^2, \Sigma_t^2), \tag{43}$$

where $\mu_t^2 = \Omega I_t + \Omega^{(1/2)} \mathbb{E}[Q_t]$ and $\Sigma_t^2 = \Omega Var[Q_t] = \Omega G_t$. Since we are working

with the Restarting LNA, μ_t^2 simplifies to just $\mu_t^2 = \Omega I_t$.

We have assumed that we have a system X_t with continuous states, and that we have noisy, partial observations from the aggregated process H_t. A graphical representation of this system is shown in Figure 3.5, where we can see that H_t is not a Markov process, since it depends on the past of X_t too.

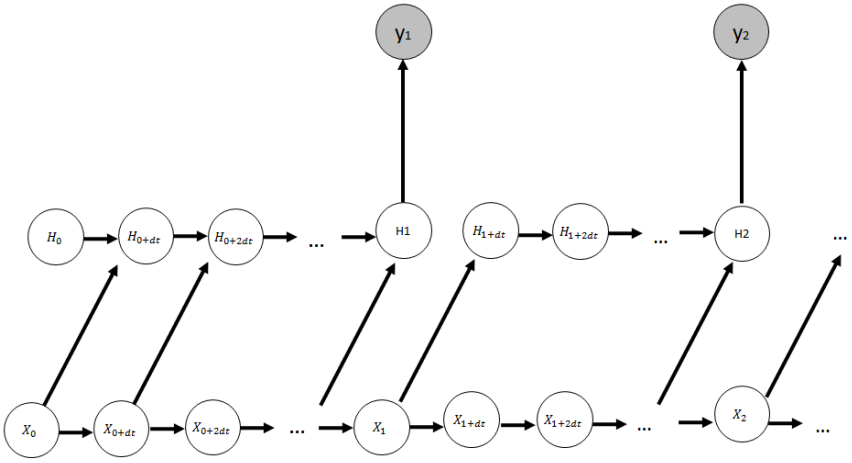


Figure 3.5: Graphical representation of the aggregated state-space model. The shaded circles correspond to noisy observations of the aggregated process, the circles of the second layer correspond to the aggregated process H_t, and the bottom layer corresponds to the underlying process X_t.

Since we are using the Restarting LNA, the predictive distribution of our system $p(\begin{bmatrix} X_t \\ H_t \end{bmatrix} |y_{1:t-1}) = N\left(\begin{bmatrix} \mu_t^{1-} \\ \mu_t^{2-} \end{bmatrix}, \begin{bmatrix} \Sigma_t^{1-} & C_t^{-T} \\ C_t^{-} & \Sigma_t^{2-} \end{bmatrix} \right)$ will be given by solving the ODEs (22), (24), (32), (41) and (39) with the appropriate initial conditions. Note that the integrated process H_t needs to be restarted at each observation point in order to capture correctly the "area under graph" of the underlying process X_t.

In order to compute the posterior distribution $p(X_t|y_{1:t})$, consider the joint distribution of (H_t, X_t, y_t) conditioned on $y_{1:t-1}$:

$$\begin{bmatrix} X_t \\ H_t \\ y_t \end{bmatrix} |y_{1:(t-1)} \sim N\left(\begin{bmatrix} \mu_t^{1-} \\ \mu_t^{2-} \\ P_t\mu_t^{2-} \end{bmatrix}, \begin{bmatrix} \Sigma_t^{1-} & C_t^{-T} & C_t^{-T}P_t^T \\ C_t^{-} & \Sigma_t^{2-} & \Sigma_t^{2-}P_t^T \\ P_tC_t^{-} & P_t\Sigma_t^{2-} & P_t\Sigma_t^{2-}P_t^T + R_t \end{bmatrix} \right). \quad (44)$$

More details about the terms of (44) are found in Appendix A.4. By using Lemma 1 (Appendix A.2), we can now calculate the posterior mean and variance of $p(X_t|y_{1:t})$ by using the corresponding blocks of the joint distribution (44):

$$\begin{aligned}\mu_t^1 &= \mu_t^{1^-} + C_t^{-^T} P_t^T (P_t \Sigma_t^{2^-} P_t^T + R_t)^{-1} (y_t - P_t \mu_t^{2^-}),\\ \Sigma_t^1 &= \Sigma_t^{1^-} - C_t^{-^T} P_t^T (P_t \Sigma_t^{2^-} P_t^T + R_t)^{-1} P_t C_t^-.\end{aligned} \tag{45}$$

If we are interested in parameter inference, we will need to compute the likelihood $L(\theta)$ of the system. As in the case of non-aggregate data, $L(\theta)$ is given by:

$$L(\theta) = p(y_1|\theta) \prod_{i=1}^{t} p(y_i|y_{1:i-1}, \theta)\,, \tag{46}$$

where $p(y_t|y_{1:t-1}) = N(P_t \mu_t^{2^-}, P_t \Sigma_t^{2^-} P_t^T + R_t)$. The algorithm that we have used for calculating the likelihood of a system observed through its aggregate process is given in Algorithm 3.4.1.

Algorithm 3.4.1: Kalman Filter for aggregate LNA

procedure LIKELIHOOD($y_{1:T}, \theta$)
  Initialisation ($t = 0$) *Set prior for* $X_0 \sim N(\mu_0^{1^-}, \Sigma_0^{1^-})$.
  Set initial conditions for the ODEs (22), (24), (32), (39), (41): $\phi_0 = \frac{\mu_0^1}{\Omega}$, $V_0 = \frac{\Sigma_0^{1^-}}{\Omega}$, $I_0 = 0$, $C_0 = 0$, $S_0 = 0$.
  $prod \leftarrow 1$
loop:
  Solve the ODEs (22), (24), (32), (39), (41) s.t. the initial conditions for $[t-1, t]$ *to obtain* mx_t^-, Sx_t^-, μ_t^-, C_t^-, S_t^- .
  Calculate $p(y_t|y_{1:t-1}, \theta)$, mx_t *and* Sx_t *according to (46) and (45).*
  $prod \leftarrow prod * p(y_t|y_{1:t-1}, \theta)$.
  Reset initial conditions: $\phi_t = \frac{\mu_t^1}{\Omega}$, $V_t = \frac{\Sigma_t^1}{\Omega}$, $I_t = 0$, $C_t = 0$, $S_t = 0$.
  Set $t = t + 1$
  if $t < T$ **goto** *loop* .
  Return prod
end procedure

3.5 Summary

In this chapter, we studied inference in stochastic systems that evolve continuously over time but are observed at discrete time points. We briefly reviewed various existing methods for inference, before focussing on the Kalman Filter (KF) methodology. We first introduced the discrete KF, where both the state of the system and the observations are discrete-time stochastic processes. We continued

with the variation of the KF for continuous time stochastic systems described by SDEs and discussed how the LNA could be incorporated in the KF methodology when needed. Finally, we studied inference with temporally aggregated data. We proved the integral form of the LNA and developed a KF framework for temporally aggregated data from continuous-time processes approximated by the LNA. In the following chapters, we are going to apply our method to different systems and assess the effect of temporal aggregation on inference.

Chapter 4

Results on synthetic datasets

In this chapter, we present results of the Kalman Filter (KF) methodology developed in Chapter 3. We study stochastic systems with aggregated observations and compare inference between two cases: treating the aggregate data as (1) coming directly from the system itself or (2) coming from the integral of the studied system. In the first case, we will be using a standard continuous-discrete KF (Algorithm 3.4.1) referred to as KF1, and in the second case, we will be using our newly proposed aggregate KF (Algorithm 3.3.2), referred to as KF2. We first study the Ornstein-Uhlenbeck process as an example of an exactly tractable system. We continue with two systems that are approximated by the LNA, the Lotka-Volterra model and a single gene expression model. The datasets in this chapter are synthetic; we will consider a real world application in Chapter 5. All experiments in this thesis were carried out on a cluster of 64bit Ubuntu machines with an i5-3470 CPU @ 3.20 GHz x 4 processor and 8 GB RAM. All scripts were run in Spyder (Anaconda 2.5.0, Python 2.7.11, Numpy 1.10.4).

4.1 The Ornstein-Uhlenbeck process and its integral

In this section, we investigate the effect of aggregation on parameter estimation in systems modelled by a linear SDE[1] using different observation intervals. We assume that we have aggregated data from the integral of a linear SDE and we compare the accuracy of the inferred parameters with and without ignoring the aggregated nature of the observations. As an example, we are working with the one-dimensional Ornstein-Uhlenbeck (OU) process. A zero mean OU process satisfies the following linear SDE:

[1] We refer to linear SDEs in the narrow sense.

$$dX_t = -\alpha X_t dt + \sigma dW_t, \tag{1}$$

where α is the drift or decay rate of the process, σ is the diffusion constant, both being time-invariant. The OU process has the property of reverting towards its long term mean which, for Equation (1), is zero. The rate at which it reverts to the zero mean depends on the drift α. The OU process has found applications in finance, where it is also known as the Vasicek model, as well as in physics and biology. It also appears as the solution to the stochastic part of the LNA when the deterministic state is at a stationary point [84].

We denote by Y_t the integral of X_t such that

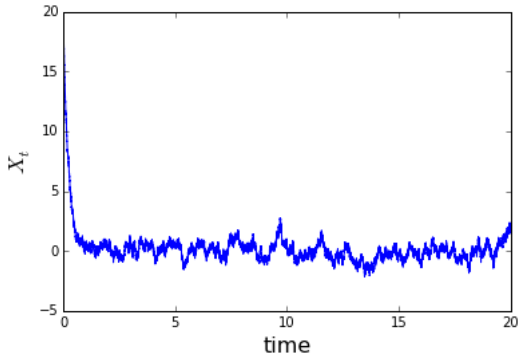
$$dY_t = X_t dt\,. \tag{2}$$

The Euler-Maruyama algorithm can be used to simulate the OU process and its integral according to Equations (1) and (2). In Figures 4.1 (a) and (b), the trace of a realisation of X_t and the corresponding Y_t is shown. We have set the drift of the process equal to $\alpha = 4$, and the diffusion constant equal to $\sigma = 2$, with X_t initialised to 20. However, in our applications, the aggregated data are not collected directly from Y_t, as depicted in Figure 4.1 (b), but instead at each observation point Y_t is restarted. This process is shown in Figure 4.1 (c), and we will refer to it as the aggregated process. For the aggregated process in Figure 4.1 (c), we assume observations every 2 minutes. We observe that the OU process has reverted towards its zero mean from its initial state, which was set to 20, and that the traces of both the integrated and aggregated process appear to be much smoother than the corresponding OU process.

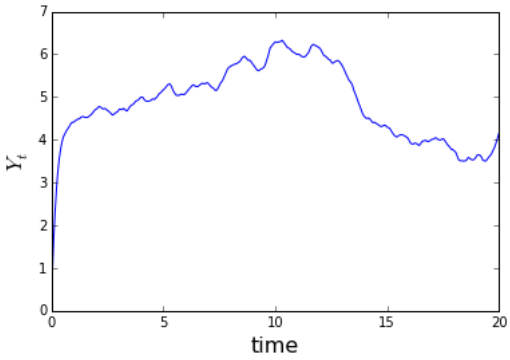
Analytical solutions of both the OU and its integral are available [28] and derivations can be found in Appendix A.6. As a solution to a linear SDE in the narrow sense, the OU is a Gaussian Markov process characterised by its mean and variance. The mean m_t and variance V_t of X_t satisfying (1) are given below for $\Delta = t - t_0$

$$m_t = m_0 e^{-\alpha\Delta}, \tag{3a}$$

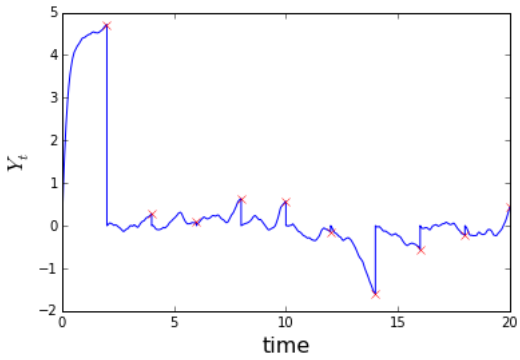
$$V_t = e^{-2\alpha\Delta} V_0 + \tfrac{\sigma^2}{2\alpha}(1 - e^{-2\alpha\Delta}). \tag{3b}$$



(a) OU process.



(b) Integrated process.



(c) Aggregated process.

Figure 4.1: Simulated trajectories from an OU process with $\alpha = 4$ and $\sigma = 2$, along with its integrated and aggregated process. For the aggregated process, we assumed observations every 2 minutes, which are indicated by red crosses.

Since Y_t is the integral of a Gaussian process, it will also follow a Gaussian process. By setting the initial condition of Y_t to zero, its mean, variance and covariance are given below for $\Delta = t - t_0$:

$$\mathbb{E}[y_t] = \tfrac{m_0}{\alpha}(1 - e^{-\alpha\Delta}), \tag{4a}$$

$$Cov(X_t, Y_t) = \frac{\sigma^2}{2\alpha^2} + (-\frac{\sigma^2}{\alpha^2} + \frac{V_0}{\alpha})e^{-\alpha\Delta} + (\frac{\sigma^2}{2\alpha^2} - \frac{V_0}{\alpha})e^{-2\alpha\Delta}, \tag{4b}$$

$$\mathrm{Var}[y_t] = \frac{\sigma^2}{\alpha^2}\Delta + (\frac{\sigma^2}{2\alpha^3} - \frac{V_0}{\alpha^2})(1 - e^{-2\alpha\Delta}) + 2(-\frac{\sigma^2}{\alpha^3} + \frac{V_0}{\alpha^2})(1 - e^{-\alpha\Delta}). \tag{4c}$$

We are interested in inferring the parameters α and σ given observations from Y_t at discrete times, where the interval Δ between two observations is constant. We want to compare two approaches. In the first approach, we will use the standard continuous-time KF ignoring the aggregated nature of the observations. According to the current methodology [47, 18], aggregated observations of single cell data are being treated with the use of a multiplication constant in the observation matrix P. For the first approach, we will, therefore, use the normalised observations by dividing the observations with Δ. In the second approach, we will use an aggregated KF as described in Section 3.4. As there is no need for using the LNA in this example, the mean and variance of the aggregated process are given solely by the statistics of Y_t and the underlying process X_t. We will refer to the first approach as the standard KF (KF1) and to the second as the aggregate KF (KF2).

By taking $\Delta \to \infty$ in Equation (3), it can be shown that X_t will reach its stationary distribution after a time of order $\frac{1}{\alpha}$, which is given by $N(0, \frac{\sigma^2}{2\alpha})$. This means that for a time sufficiently greater than $\frac{1}{\alpha}$, the estimate of X_t will be the same, although X_t will keep changing over time [26]. However, the integrated process Y_t is non-stationary since $\mathrm{Var}[y_t] \to \infty$ in Equations (4), when $\Delta \to \infty$. This already shows us that the two processes behave differently.

Since we are going to use the normalised observations from Y_t in the first scenario, we will take a look at the averaged process $Z_t = \frac{1}{\Delta} Y_t$:

$$\mathbb{E}[z_t] = \mathbb{E}[\tfrac{1}{\Delta} Y_t] = \tfrac{1}{\Delta}\mathbb{E}[y_t] = \tfrac{m_0}{\alpha\Delta}(1 - e^{-\alpha\Delta}) \tag{5a}$$

$$\begin{aligned} \mathrm{Var}[z_t] = \mathrm{Var}[\tfrac{1}{\Delta} Y_t] = \tfrac{1}{\Delta^2}\mathrm{Var}[y_t] = \\ \tfrac{\sigma^2}{\alpha^2\Delta} + \tfrac{1}{\Delta^2}(\tfrac{\sigma^2}{2\alpha^3} - \tfrac{V_0}{\alpha^2})(1 - e^{-2\alpha\Delta}) + \\ + \tfrac{2}{\Delta^2}(-\tfrac{\sigma^2}{\alpha^3} + \tfrac{V_0}{\alpha^2})(1 - e^{-\alpha\Delta}) \end{aligned} \tag{5b}$$

By taking the limit as $\Delta \to \infty$ in Equation (5) and using L'Hospital's rule, we can we can show that $\mathbb{E}[z_t] \to 0$ and $\mathrm{Var}[z_t] \to 0$. So the averaged process does not approach the stationary distribution of X_t and the variance will tend to be lower.

We briefly describe here the procedure we followed for simulating the aggregated data. This procedure is followed in the other two case studies of this chapter. To simulate data from Y_t, we need first to be able to simulate data from X_t. This can be done in general by discretising the process and using the Euler-Maruyama algorithm. However, in the case of the OU process, we can also use an exact updating formula, see Appendix A.7. The aggregated data can then be collected using the discretised form of Equation (2) or a numerical integration method such as the trapezoidal rule.

For the example studied in this section, we have simulated data from the aggregated process of an OU process with $\alpha = 4$ and $\sigma = 2$. We have compared the two approaches, standard KF (KF1) and aggregate KF (KF2), using different time intervals Δ between the observations. For this example, we have assumed no observation noise. Parameter estimation was carried out using a random walk MH algorithm with a Gaussian proposal and improper uniform priors on the log parameters $log(\alpha)$ and $log(\sigma)$. The initial states of the parameters were sampled from a uniform distribution U(0,10) and the MH was run for 50K iterations 30K of which were discarded as burn-in. The case of inferring the parameters of an OU process using non-aggregate data with an MCMC algorithm has already been studied in [54]. We also checked parameter estimation using a numerical optimisation algorithm. The Nelder-Mead algorithm was chosen among the available scipy optimisation[2] methods and was initialised to 0.1 for both parameters.

Results of parameter estimation using the random walk MH algorithm and the Nelder-Mead algorithm are presented in Tables 4.1 and 4.5 respectively. Different time intervals Δ have been tested using each time a single dataset. To verify the validity of the results, we have run nine more datasets, separately each time, and an average over all ten datasets is presented in Tables 4.1 and 4.5 for MH and Nelder-Mead respectively. Both the MH and the Nelder-Mead have given similar results. However, the Nelder-Mead provides us with a point estimate in contrast to the MH that converges to a distribution. As expected, the estimates for KF1 deteriorate for a larger Δ, since the aggregated process diverges from the OU process as Δ increases. Estimates remain good for KF2 even when Δ is large. However, they become more uncertain, as shown by the increased standard deviations.

In the first two columns of Figure 4.3, we can see the traces of the posterior

[2] `https://www.scipy.org/`

Δ	α KF1	σ KF1	α KF2	σ KF2
0.1	3.020±0.235	1.891±0.135	4.015±0.292	2.107±0.158
0.5	1.905±0.141	1.256±0.095	4.085±0.335	2.301±0.204
1.0	1.420±0.102	0.868±0.068	3.865±0.368	2.234±0.240
2.0	1.022±0.074	0.539±0.044	3.703±0.530	2.082±0.321

Table 4.1: Mean posterior ± 1 s.d. for α and σ using a Metropolis-Hastings algorithm. Data were simulated from an OU process with $\alpha = 4$ and $\sigma = 2$.

Δ	α KF1	σ KF1	α KF2	σ KF2
0.1	3.035	1.874	4.029	2.083
0.5	1.898	1.239	4.061	2.270
1.0	1.414	0.856	3.799	2.170
2.0	1.014	0.532	3.560	1.984

Table 4.2: Nelder-Mead estimates for α and σ. Data were simulated from an OU process with $\alpha = 4$ and $\sigma = 2$.

estimate of α using both KF1 and KF2 that correspond to the results presented in Table 4.1. In the third column, we provide the histograms of the posterior estimate of α for both KF1 and KF2. As we can see, the chains appear to mix well. All chains were started off at random initial states and were run for 50000 iterations from which 30000 were regarded as burn-in, with the acceptance rate varying between 0.1 and 0.2. The equivalent plots for parameter σ are provided in Appendix B.1.

It is of interest to investigate the inferred stationary variance of the OU process using KF1 and KF2. We show the inferred stationary variances obtained by the MH for both KF1 and KF2 in Figure 4.2. The boxplots are obtained by the average of the 10 different datasets and correspond, again, to an OU process with $\alpha = 4$ and $\sigma = 2$, thus giving rise to a stationary variance of $\frac{\sigma^2}{2\alpha} = 0.5$. As we can see, by using the normalised aggregate data directly with a KF, we infer the wrong stationary variance of the underlying OU process, which tends to zero as Δ becomes larger, something that we had already verified by the theoretical results from Equation 5. For a sufficiently small Δ, parameter estimation results of the two different Kalman Filters will tend to agree.

In this section, we have looked at an example of inferring the parameters of an SDE using aggregated data and we have found that to obtain accurate results, we need to explicitly model the aggregated process. As the observation intervals become larger, we showed that there is a greater mismatch between KF1 and

Δ	α KF1	σ KF1	α KF2	σ KF2
0.1	2.985±0.233	1.703±0.198	3.979±0.319	1.917±0.202
0.5	1.924±0.152	1.199±0.124	3.999±0.353	2.028±0.255
1.0	1.475±0.106	0.796±0.087	4.027±0.420	2.045±0.274
2.0	1.053±0.086	0.483±0.046	4.105±0.735	2.044±0.362

Table 4.3: Average of mean posterior ± 1 s.d. over 10 different datasets for α and σ using a Metropolis-Hastings algorithm. Data were simulated from an OU process with $\alpha = 4$ and $\sigma = 2$.

Δ	α KF1 Median[LB,UB]	σ KF1 Median[LB,UB]
0.1	3.078 [2.984,3.232]	1.637 [1.552,1.768]
0.5	1.958 [1.909,1.978]	1.190 [1.150,1.237]
1.0	1.486 [1.435,1.538]	0.782 [0.762,0.828]
2.0	1.041 [1.016,1.093]	0.478 [0.457,0.484]
Δ	α KF2 Median[LB,UB]	σ KF2 Median[LB,UB]
0.1	4.146 [3.978,4.463]	1.951 [1.909,2.058]
0.5	4.157 [3.913,4.265]	2.001 [1.879,2.206]
1.0	4.070 [3.943,4.212]	2.0357 [1.908,2.162]
2.0	3.885 [3.709,4.453]	1.935 [1.862,2.108]

Table 4.4: Median values of the Nelder-Mead estimates over 10 different datasets for α and σ along with lower and upper bounds for KF1 and KF2. Data were simulated from an OU process with $\alpha = 4$ and $\sigma = 2$.

KF2. In the next sections, we will look at examples of stochastic systems that can be approximated by the LNA and compare, again, inference results for KF1 and KF2.

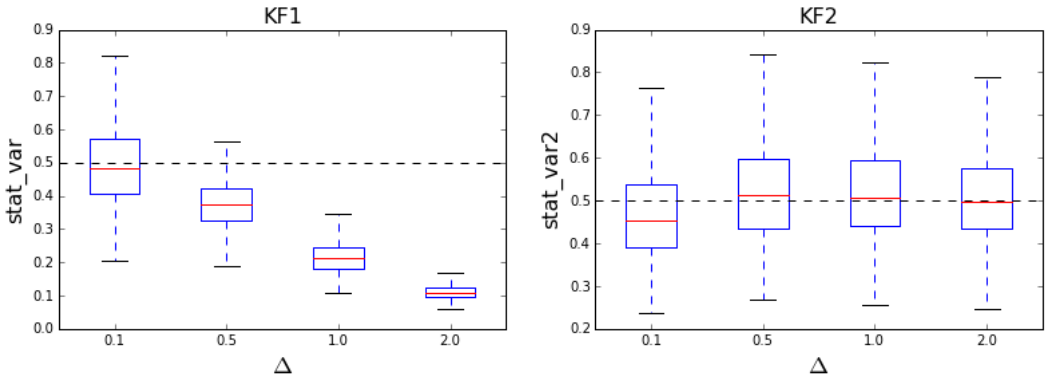
4.2 The Lotka-Volterra model

We are now going to look at a system of two species that interact with each other according to three reactions:

$$X_1 \xrightarrow{\theta_1} 2X_1 \ \textit{(prey production)} \tag{6a}$$

$$X_1 + X_2 \xrightarrow{\theta_2} 2X_2 \ \textit{(predator production)} \tag{6b}$$

$$X_2 \xrightarrow{\theta_3} \oslash \ \textit{(predator death)} \tag{6c}$$



(a) KF1 (b) KF2

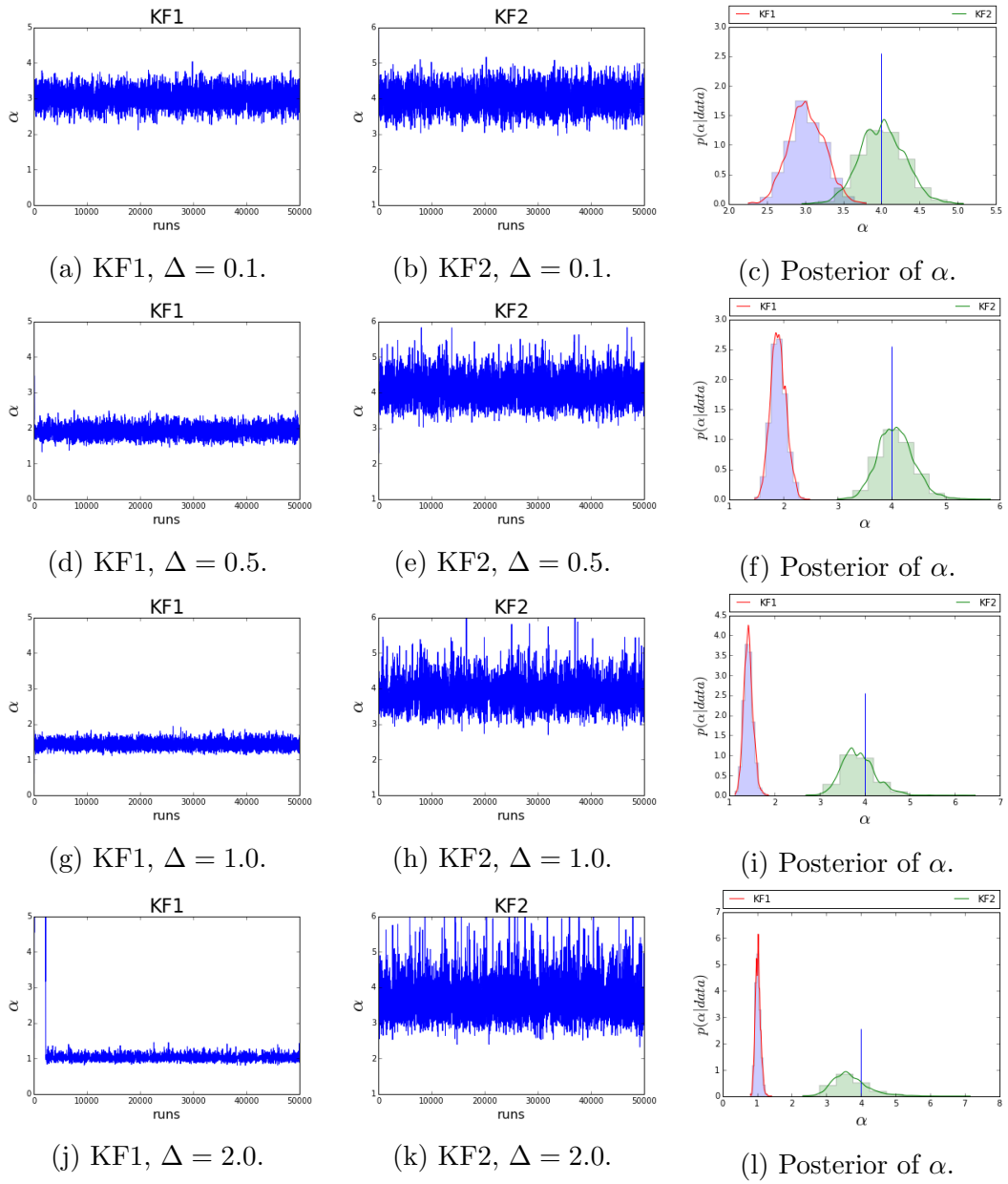
Figure 4.2: Boxplots of inferred stationary variance of the OU process for different Δ. The simulated OU process has $\alpha = 4$ and $\sigma = 2$ corresponding to a stationary variance of 0.5, as indicated by the dotted horizontal line. The inferred stationary variance using KF1 tends to zero as Δ grows, but the stationary variance from KF2 is inferred correctly at all Δ.

This model was initially developed by Lotka [52] to explain oscillatory behaviour in autocatalytic chemical reactions and was later applied in predator-prey interactions [53]. Volterra [85] came up with the same model to explain the interactions between voracious fishes (selachians) and eaten fishes in the Adriatic sea. The model is named after both Lotka and Volterra. In the biochemical reaction network (6) X_1 represents the prey species and X_2 the predator species. Modifications of the Lotka-Volterra model have also found applications in economics, where the Lotka-Volterra model is commonly referred to as the Goodwin model [39].

We assume that $\Omega = 1$ and move forward to constructing the LNA representation of (6). Following the LNA methodology, we decompose the system into a macroscopic part ϕ_t and a stochastic part ξ_t. In order to compute the macroscopic part, we need to define the stoichiometry matrix S and the hazard function $\tilde{f}(X)$ of (6),

$$S = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}, \tag{7}$$

$$\tilde{f}(X) = \begin{bmatrix} \theta_1 X_1 \\ \theta_2 X_1 X_2 \\ \theta_3 X_2 \end{bmatrix} \quad . \tag{8}$$



(a) KF1, $\Delta = 0.1$. (b) KF2, $\Delta = 0.1$. (c) Posterior of α.

(d) KF1, $\Delta = 0.5$. (e) KF2, $\Delta = 0.5$. (f) Posterior of α.

(g) KF1, $\Delta = 1.0$. (h) KF2, $\Delta = 1.0$. (i) Posterior of α.

(j) KF1, $\Delta = 2.0$. (k) KF2, $\Delta = 2.0$. (l) Posterior of α.

Figure 4.3: MCMC traces and histograms of the posterior of α using a MH for both KF1 and KF2. Ground truth for $\alpha = 4$, indicated by the vertical blue line on the histogram plots.

The following matrices are needed in order to compute the stochastic part ξ_t such as $d\xi_t = A_t\xi_t dt + E_t dW_t$,

$$F = \begin{bmatrix} \theta_1 & 0 \\ \theta_2\phi_2 & \theta_2\phi_1 \\ 0 & \theta_3 \end{bmatrix}, \tag{9}$$

$$SF^T = A = \begin{bmatrix} \theta_1 - \theta_2\phi_2 & -\theta_2\phi_1 \\ \theta_2\phi_2 & \theta_2\phi_1 - \theta_3 \end{bmatrix}, \tag{10}$$

$$S\text{diag}(\tilde{f}(\phi_t))S^T = EE^T = \begin{bmatrix} \theta_1\phi_1 + \theta_2\phi_1\phi_2 & -\theta_2\phi_1\phi_2 \\ -\theta_2\phi_1\phi_2 & \theta_2\phi_1\phi_2 + \theta_3\phi_2 \end{bmatrix}. \tag{11}$$

We can now define the ODEs that correspond to the macroscopic part by using the formula $d\phi_i/dt = S_i\tilde{\boldsymbol{f}}(\phi_t, c)$,

$$\frac{d\phi_1}{dt} = \theta_1\phi_1 - \theta_2\phi_1\phi_2\,, \tag{12}$$

$$\frac{d\phi_2}{dt} = \theta_2\phi_1\phi_2 - \theta_3\phi_2\,. \tag{13}$$

As we are using the Restarting method (see Chapter 3), we only need to compute the elements of the covariance matrix of the solution to the stochastic part where $p(\xi_t|\xi_0) = N(m_t, V_t)$,

$$\frac{dV_{11}}{dt} = 2V_{11}(\theta_1 - \theta_2\phi_2) - 2V_{12}\theta_2\phi_1 + \theta_2\phi_1\phi_2 + \theta_1\phi_1\,, \tag{14}$$

$$\frac{dV_{12}}{dt} = V_{12}(\theta_2\phi_1 - \theta_3 + \theta_1 - \theta_2\phi_2) + V_{11}\theta_2\phi_2 - \theta_2\phi_1 V_{22} - \theta_2\phi_1\phi_2\,, \tag{15}$$

$$\frac{dV_{22}}{dt} = 2V_{22}(\theta_2\phi_1 - \theta_3) + 2V_{12}\theta_2\phi_2 + \theta_2\phi_1\phi_2 + \theta_3\phi_2\,. \tag{16}$$

We can simulate the stochastic dynamics of the Lotka-Volterra model using either the Gillespie algorithm or the LNA. In Figures 4.4 (a), (b), (d) and (e), we present 10 different simulated trajectories of the prey and predator populations using the Gillespie algorithm and the LNA. As we can see, there is a different behaviour between the two simulators. The reason for this mismatch is that the LNA simulator assumes that the expected value of the process corresponds to the

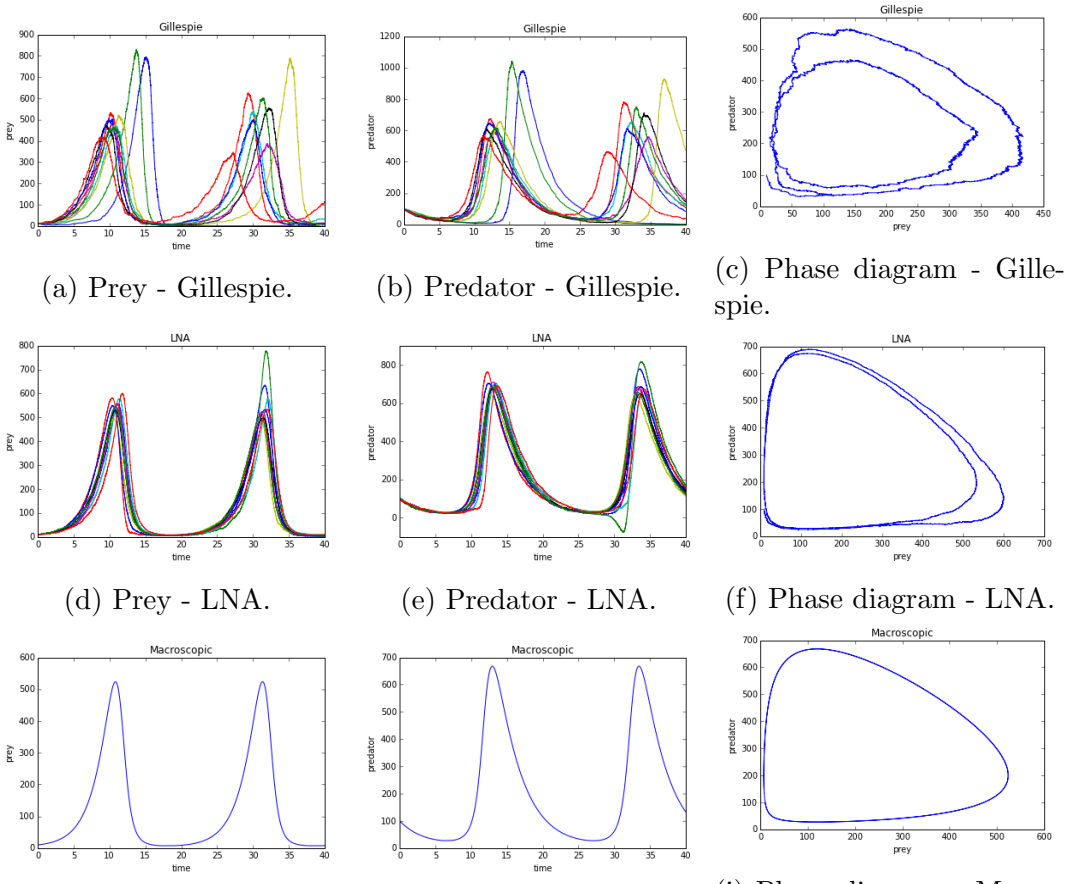
macroscopic solution. However, this is not true for systems including second order reactions [89]. By using the Gillespie algorithm, we can see that the resulting oscillations are more noisy compared to the LNA simulator. Additionally, some of the predator trajectories have reached extinction by the time of the second oscillation. Extinction cannot be achieved by using the LNA simulator, where predators can reach negative values, but they will continue oscillating according to the repeated oscillations of the macroscopic solution as shown in Figures 4.4 (g) and (h). We can also observe the mismatch in the phase diagrams in Figures 4.4 (c), (f) and (i), produced by each method.

Our LNA simulator uses the Euler and Euler-Maruyama algorithms for simulating the macroscopic part ϕ_t and the stochastic part ξ_t respectively. In [34] the Restarting method was used for simulating the Lotka-Volterra model. According to [34], simulation of the stochastic part of the LNA simulator is achieved via its solution. Consequently, an ODE solver is used for simulating forward the ODEs (12) - (16) at specified intervals $(t, t + dt)$. At the end of each interval, the ODE solver is restarted by sampling from $N(\phi_{t+dt}, V_{t+dt})$, which corresponds to the approximate solution of the Restarting LNA. This simulator matches the Gillespie output more closely, but still cannot account for the extinction of species.

Although the LNA does not seem to be a very good simulator for the Lotka-Volterra system, it does not mean that it will not be appropriate for inference, since the observations will drive the approximation to the correct mean when using the Restarting method. In the following, we present parameter estimation results comparing, again, KF1 and KF2. Results for non-aggregate data using KF1 have also been previously presented in [18].

We have collected aggregated data from a Lotka-Volterra model using the Gillespie algorithm. We assumed a known initial population of 10 preys and 100 predators. The parameters of the system used for producing the synthetic data were set to $(\theta_1, \theta_2, \theta_3) = (0.5, 0.0025, 0.3)$, following the setting of [8]. We have added Gaussian noise with a zero mean and a variance of 3. We have assumed that initial populations and the noise level were known during inference. Our goal was to infer the three parameters $(\theta_1, \theta_2, \theta_3)$ of the system using only observations of the predator population.

The Gillespie algorithm was run for 20 minutes, corresponding to one period of the system as indicated by Figure 4.4, and data were aggregated and collected every 2 minutes resulting in 10 observations per sample. To infer the parameters,



(a) Prey - Gillespie. (b) Predator - Gillespie. (c) Phase diagram - Gillespie.

(d) Prey - LNA. (e) Predator - LNA. (f) Phase diagram - LNA.

(g) Prey - Macroscopic. (h) Predator - Macroscopic. (i) Phase diagram - Macroscopic.

Figure 4.4: Simulated trajectories from the Lotka-Volterra model using the Gillespie algorithm, the LNA and the macroscopic solution. The Gillespie algorithm leads to more noisy oscillations in contrast to the LNA and extinction of species. The macroscopic solution leads to repeated oscillations of equal peak and phase. The phase diagrams corresponding to one trajectory are shown in the third column.

we assumed that we had 40 independent samples available. Since we assumed independence between the samples, we worked with the product of their likelihoods. Gamma(2,10) priors were placed on all three parameters, and we worked in the log space since we know that we want all parameters to be positive. The adaptive MCMC with the Gaussian mixture proposal (58) described in Chapter 2 was used. It was run for 30000 iterations, and a burn-in of 10000 samples was assumed. The MCMC was initialised to random values sampled from uniform distributions U(0,1). Parameter estimation results for all three parameters using the adaptive MCMC are shown in Table 4.5, while Figure 4.5 shows histograms of their posterior densities. As we can see, KF1 leads to an inaccurate estimate of the parameters, underestimating all three parameters.

In order to further assess the consistency of our results, we used a numerical optimisation algorithm (Nelder-Mead) on 100 datasets, each consisting of 40 independent samples. As the optimiser was failing to converge when initialised at random values, we initialised it at the ground truth, since we were only interested in comparing it to the MCMC results. In Table 4.6, we present the results of the optimised parameter values across all 100 datasets. Again, we see that KF1 underestimates the parameter values, thus verifying our previous results using the MCMC algorithm.

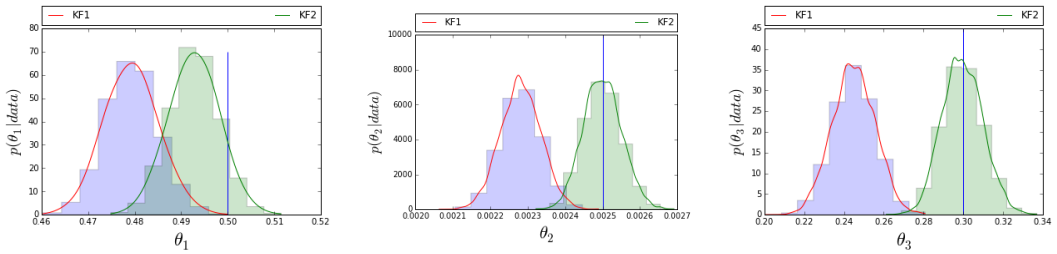
θ	Ground truth	KF1	KF2
θ_1	0.5	0.4793±0.0057	0.4928±0.0053
θ_2	0.0025	$0.0023\pm5\cdot10^{-5}$	$0.0025\pm5\cdot10^{-5}$
θ_3	0.3	0.2450±0.0108	0.2997±0.0102

Table 4.5: Mean posterior ± 1 s.d. for $\theta_1, \theta_2, \theta_3$ using an adaptive MCMC. Data were simulated from a Lotka-Volterra model according to the ground truth values.

θ	Ground truth	KF1 Median[LB,UB]	KF2 Median[LB,UB]
θ_1	0.5	0.48160 [0.47770,0.48651]	0.49746 [0.49278,0.50122]
θ_2	0.0025	0.00227 [0.00222,0.00232]	0.00248[0.00244,0.00254]
θ_3	0.3	0.24773 [0.23927,0.25797]	0.30047[0.29320,0.31061]

Table 4.6: Nelder-Mead results for $\theta_1, \theta_2, \theta_3$. The median values across 100 datasets are shown in the third and fourth column for KF1 and KF2 respectively. Lower and upper bounds are shown in brackets.

At this point, we want to emphasise that the strong correlations among the parameters made the tuning of the MCMC hard; we therefore used an adaptive



(a) Posterior density of θ_1. (b) Posterior density of θ_2. (c) Posterior density of θ_3.

Figure 4.5: Posterior densities of $\theta_1, \theta_2, \theta_3$ from aggregate data using KF1 (red histogram) and KF2 (green histogram).

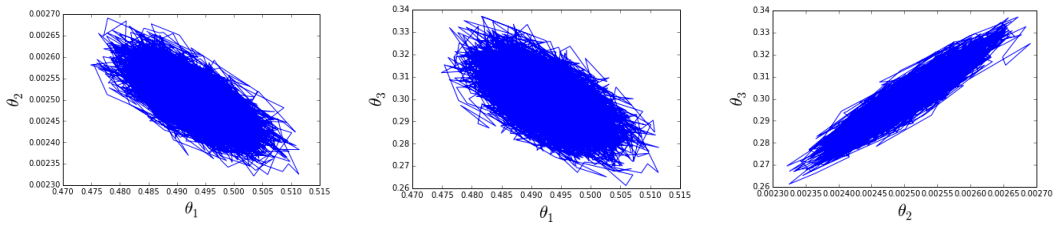


Figure 4.6: Correlations between the MCMC samples of the three parameters $\theta_1, \theta_2, \theta_3$.

MCMC. In Figure 4.6, we can observe the correlations between the parameters by looking at their MCMC samples. The parameters θ_2 and θ_3 are particularly highly correlated with a Pearson correlation coefficient of 0.93. We have attempted to run a random walk MH with a Gaussian proposal for the same dataset, where we only tuned the step size of the random walk for the first 10000 runs to keep the acceptance rate between 0.2 and 0.4. Again, the initial states of the parameters were sampled from uniform distributions U(0,1). The mixing of the chain was considerably slower. Trace plots of the parameters using the adaptive MCMC and the random walk Metropolis with KF2 are shown in Figure 4.7, while trace plots corresponding to KF1 can be found in Appendix B.2.

As we have already discussed, the LNA is not a good simulator of the Lotka-Volterra model. However, we see that it leads to accurate estimates of the parameters if we work with the Restarting method, since the observations help the model revert to the correct solution. In Figure 4.8, we can see filtering plots using the Restarting and Non-Restarting method for the prey population using KF1 and KF2. Filtering plots for the predator population can be found in Appendix B.3. For the purposes of this example, we have used aggregate observations only for

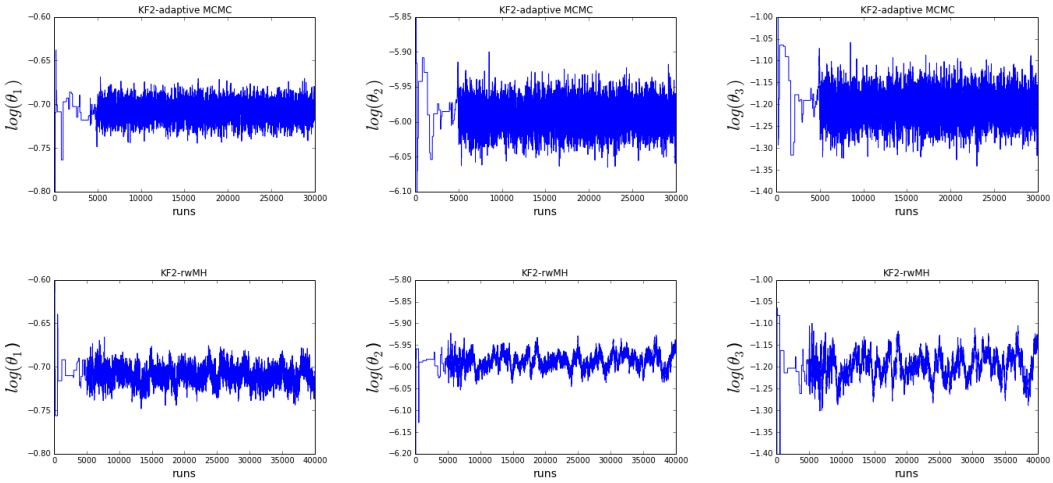


Figure 4.7: Trace plots of the Lotka-Volterra parameters using KF2 with an adaptive MCMC (first row) and a random walk MH (second row).

KF2 while in both KF1 and KF2, we use the actual values of the parameters $(\theta_1 = 0.5, \theta_2 = 0.0025, \theta_3 = 0.3)$. As we can observe, the Restarting method proves to be beneficial when a system does not follow its deterministic solution.

4.3 Single gene expression (SGE) model

In the following, we consider a simple model for describing single gene expression stochastically. The model was presented in [47] and was used in conjunction with the LNA for inference using non-aggregate data.[3] Furthermore, it has been used as a reference model by different authors when evaluating their methods in systems approximated by the LNA [79, 35].

The model can be described by a biochemical reaction network involving three species X =(DNA, mRNA, protein) and four reactions,

$$DNA \xrightarrow{k_R(t)} DNA + R \ \textit{(transcription)} \tag{17a}$$

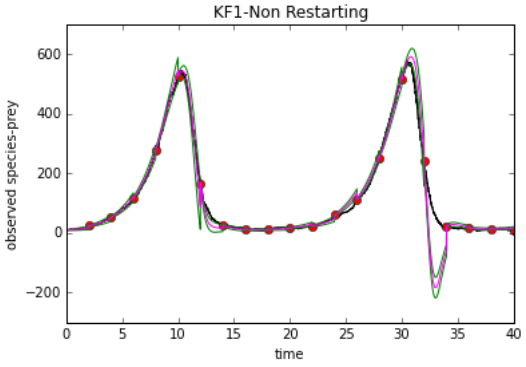
$$R \xrightarrow{\gamma_R} \oslash \ \textit{(mRNA degradation)} \tag{17b}$$

$$R \xrightarrow{\kappa_P} R + P \ \textit{(translation)} \tag{17c}$$

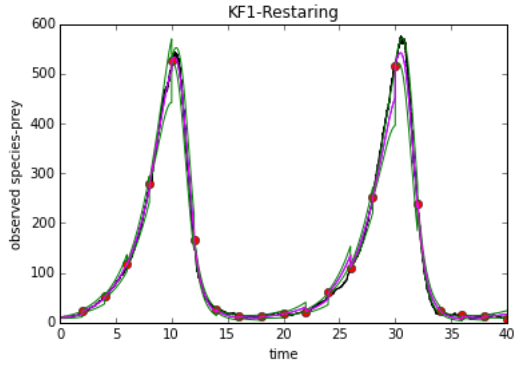
$$P \xrightarrow{\gamma_P} \oslash \ \textit{(protein degradation)} \tag{17d}$$

where $c = (k_R(t), \gamma_R, \kappa_P, \gamma_P)$ correspond to the vector of stochastic rate constants

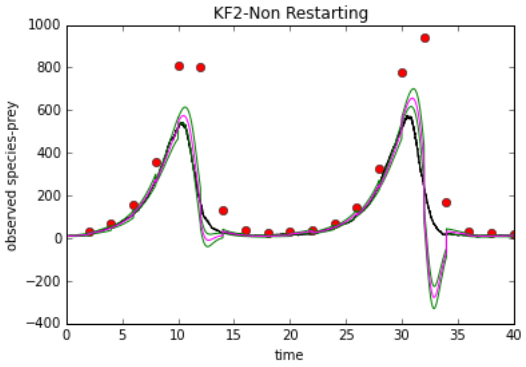
[3] Equivalent to what we refer to as the KF1.



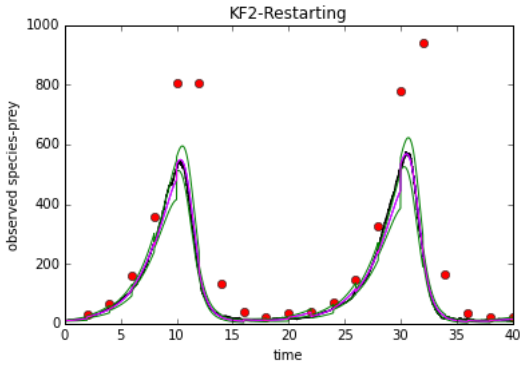
(a) Non-Restarting KF1 with non-aggregate data from the prey population.



(b) Restarting KF1 with non-aggregate data from the prey population.



(c) Non-Restarting KF2 with aggregate data from the prey population.



(d) Restarting KF2 with aggregate data from the prey population.

Figure 4.8: Filtering plots for the prey population with (KF2) and without (KF1) aggregate data. The Non-Restarting method is shown in the first column and the Restarting on the second column. Red dots correspond to the observation data available; the black line represents the actual process. Purple lines represent the mean estimate, and green lines 1 standard deviation.

of each reaction. Given the form of the biochemical reaction network (17) we can form its LNA representation by forming the appropriate matrix from Section 2.4. In the following, we will assume that the volume $\Omega = 1$ leading to the molecule numbers being equal to the molecular concentrations.

In order to compute the macroscopic part ϕ_t of the LNA, we need the stoichiometry matrix S and the hazard function $\tilde{f}(X)$ of the system,

$$S = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}, \tag{18}$$

$$\tilde{f}(X) = \begin{bmatrix} k_R(t) \\ \gamma_R R \\ k_P R \\ \gamma_P P \end{bmatrix} \quad . \tag{19}$$

The macroscopic part is then given by $d\phi_i/dt = S_i \tilde{\boldsymbol{f}}(\phi_t)$. For the stochastic part ξ_t, we need to compute the matrices,

$$F_{ij} = \frac{\partial \tilde{f}_i(\phi_t, c_i)}{\partial \phi_j(t)} = \begin{bmatrix} 0 & 0 \\ \gamma_R & 0 \\ k_P & 0 \\ 0 & \gamma_P \end{bmatrix}, \tag{20}$$

$$A_t = SF_t = \begin{bmatrix} -\gamma_R & 0 \\ k_P & -\gamma_P \end{bmatrix}, \tag{21}$$

$$S\text{diag}(\tilde{f}(\phi_t, c))S^T = EE^T = \begin{bmatrix} k_R(t) + \gamma_R r & 0 \\ 0 & k_P r + \gamma_P p \end{bmatrix}, \tag{22}$$

The stochastic part is then given by $d\xi_t = A_t \xi_t dt + E_t dW_t$, with mean and variance given by the following equations:

$$\frac{dm}{dt} = Am, \tag{23}$$

$$\frac{dV}{dt} = VA^T + EE^T + AV. \tag{24}$$

Therefore, the system of ODEs that gives the LNA solution using the Restarting method is given by:

$$\frac{d\phi_r}{dt} = k_R(t) - \gamma_R r \tag{25}$$

$$\frac{d\phi_p}{dt} = k_P r - \gamma_P p \tag{26}$$

$$\frac{dV_{11}}{dt} = -2\gamma_R V_{11} + k_R(t) + \gamma_R r \tag{27}$$

$$\frac{dV_{12}}{dt} = k_P V_{11} - (\gamma_R + \gamma_P) V_{12} \tag{28}$$

$$\frac{dV_{22}}{dt} = 2(k_P V_{12} - \gamma_P V_{22}) + k_P r + \gamma_P p \tag{29}$$

Note that $V_{12} = V_{21}$, as V_t corresponds to a covariance matrix. In this example, the transition rate $k_R(t)$ is a function of time and not a constant. It is used to describe an experiment where transcription switches from an on- to an off-period. The specific chosen form of $k_R(t)$, according to [47], is given by:[4]

$$k_R(t) = b_0 \exp(-b_1(t - b_2)^2) + b_3\,. \tag{30}$$

Such a function is increasing for $t \leq b_2$ and decreasing towards b_3 for $t > b_2$. A plot of $k_R(t)$ for specific values of its parameters is shown in Figure 4.9.

It is now straightforward to approximate data from System (17) using the LNA. Alternatively, we can use an exact simulation algorithm such as the Gillespie algorithm described in Section 2.3. In Figure 4.10, we have simulated 40 trajectories of the protein molecules using the Gillespie and the LNA. We can see that both the Gillespie and the LNA give us equivalent results, and we can conclude that for this example, the LNA can be regarded as a valid approximation of the system. Figures of the mRNA trajectories are provided in Appendix B.4, which also verify the validity of the LNA.

We move now to parameter estimation using temporally aggregated data. We assume that only the protein levels are observed, and we sample from the protein time series generated by the Gillespie algorithm. The data are aggregated over a period of 3 hours and corrupted by Gaussian noise with a mean of zero and a variance of 1. Additionally, we assume that the observations are proportional to

[4]Note that in the original paper, b_1 is allowed to change after b_3. However, in the actual application of the model, b_1 is kept constant.

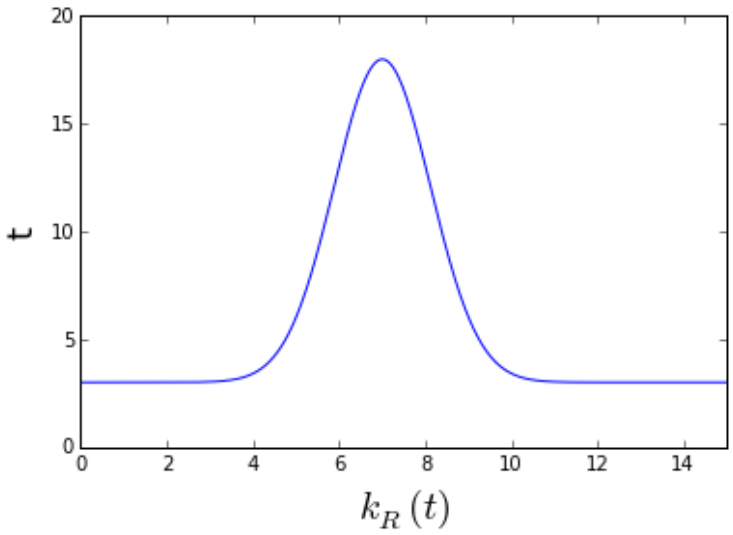
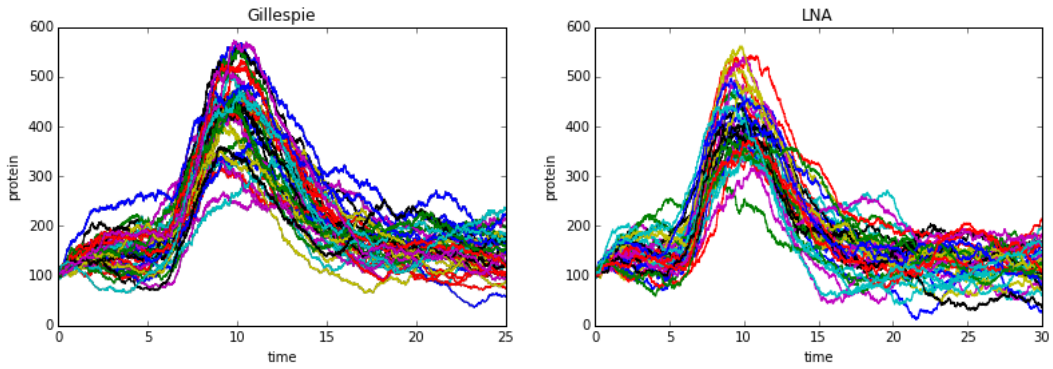


Figure 4.9: Plot of $k_R(t)$ with $b_0 = 15.0$, $b_1 = 0.4$, $b_2 = 7.0$, $b3 = 3.0$.

the actual quantities and we denote with k the constant of proportionality. We used 30 time series and sampled 10 data points from each. The noise level as well as the initial protein and mRNA levels are assumed to be known, and we want to infer the parameters included in the set $\theta = (\gamma_R, k_P, \gamma_P, k, b_0, b_1, b_2, b_3)$.

Again, the adaptive MCMC from Chapter 2 with proposal (58) was used due to high correlations among the parameters. Informative Gamma priors were placed on the degradation rates of both the protein and mRNA according to [35]; otherwise, the system was unidentifiable due to non-observability of the mRNA. For the rest of the parameters, we used weakly informative exponential priors. The full set of priors can be found in Appendix A.8. As all parameters are positive, we preferred to work with their log values. The initial states of the parameters were sampled from Uniform distributions that were covering their ground truth values and the MCMC was run for 80K iterations, from which 40K were discarded as burn-in. Trace plots from KF2 are found in Figure 4.12, where convergence seems to have been reached. Equivalent trace plots for KF1 can be found in Appendix B.5.

Results of parameter estimation using both KF1 and KF2 are presented in Table 4.7, where we have chosen to report the median and the interquartile range due to the skewness of the posterior distributions. A better overview of the results can be found by looking at the posterior histograms of the parameters in Figure 4.11. As we can see, all parameters but b_2 were identified by both



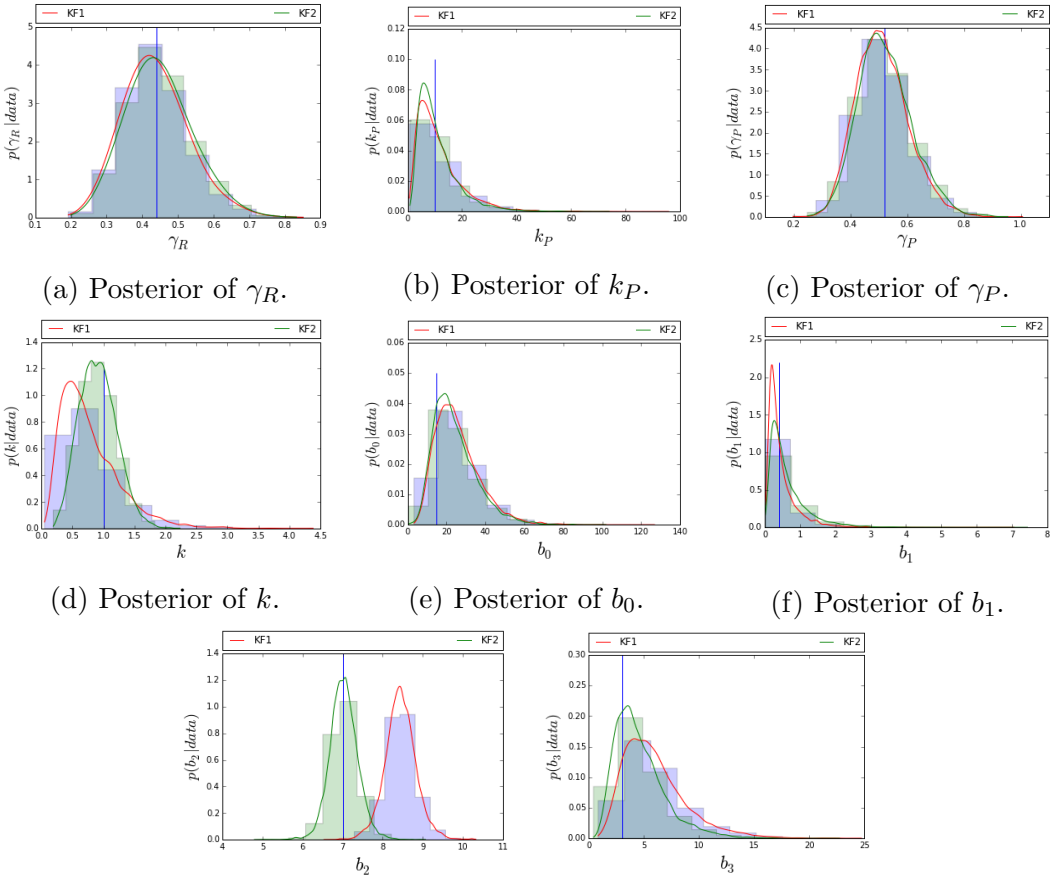
(a) Gillespie (b) LNA

Figure 4.10: Simulated trajectories of protein using the Gillespie algorithm (a) and the LNA (b).

algorithms. However, b_2 determines the switch time of the system and remains a very important parameter which would be inaccurately inferred if aggregation was not taken into consideration. We believe that the prior assumptions we made, as well as knowledge of the initial molecular abundance and noise level, restricted the parameter space so that KF1 could give accurate results for the rest of the parameters. We note that as the aggregation period becomes smaller, b_2 becomes identifiable by KF1 as well.

θ	Ground Truth	KF1 Median[IQR]	KF2 Median[IQR]
γ_R	0.44	0.4303 [0.1192]	0.4406[0.1204]
k_P	10.0	9.3078 [9.2720]	9.0071[8.1740]
γ_P	0.52	0.5067[0.1237]	0.5149[0.1251]
k	1.0	0.6645 [0.5925]	0.8988[0.4142]
b_0	15.0	23.5970[14.3870]	22.1950[13.9552]
b_1	0.4	0.3248[0.3794]	0.4760[0.5636]
b_2	7.0	8.4379[0.4755]	7.0199[0.4397]
b_3	3.0	5.4389[3.5004]	4.1378[2.7951]

Table 4.7: Posterior medians and interquartile ranges for $\theta = (\gamma_R, k_P, \gamma_P, k, b_0, b_1, b_2, b_3)$ using an adaptive MCMC. Data were simulated from the SGE model according to the ground truth values.

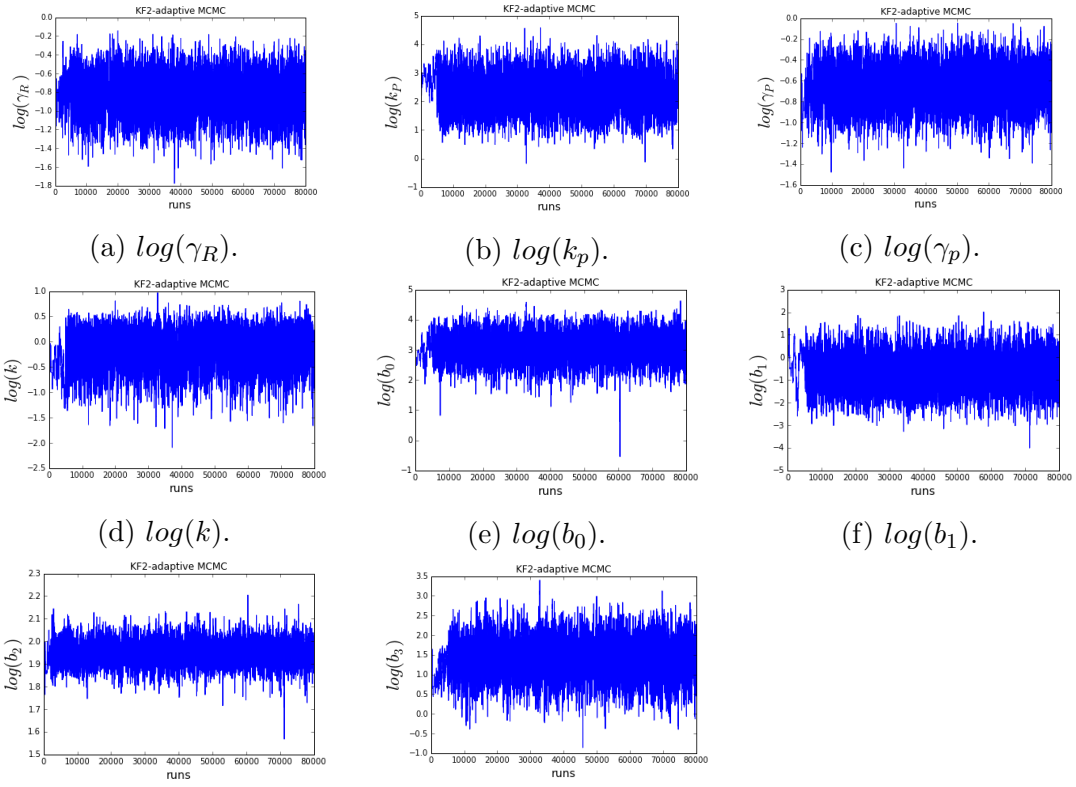


(a) Posterior of γ_R. (b) Posterior of k_P. (c) Posterior of γ_P.

(d) Posterior of k. (e) Posterior of b_0. (f) Posterior of b_1.

(g) Posterior of b_2. (h) Posterior of b_3.

Figure 4.11: Posterior histograms of the parameter set $\theta = (\gamma_R, k_P, \gamma_P, k, b_0, b_1, b_2, b_3)$ using the adaptive MCMC for both KF1 (red) and KF2 (green). Ground truth is indicated in each case by a vertical blue line.



(a) $log(\gamma_R)$. (b) $log(k_p)$. (c) $log(\gamma_p)$.

(d) $log(k)$. (e) $log(b_0)$. (f) $log(b_1)$.

(g) $log(b_2)$. (h) $log(b_3)$.

Figure 4.12: Adaptive MCMC traces for the log parameters of the SGE model using KF2 with aggregated data.

4.4 Summary

In this chapter, we studied three different stochastic systems using synthetic data that were aggregated over time. We focused on parameter estimation with synthetic aggregated data using the algorithms developed in Chapter 3 and compared our results with and without ignoring the aggregated nature of the observations, i.e. KF1 against KF2.

We began our analysis by studying an Ornstein-Uhlenbeck (OU) process, where analytical expressions for its distribution and its integral are available. We showed that the OU process possesses different statistical properties than its integral, such as stationary variance. We studied inference using synthetic data with different aggregation intervals and compared the results between KF1 and KF2. The accuracy of KF1 decreased with increasing aggregation intervals while KF2 retained a high accuracy at all aggregation intervals tested.

We further studied a non-linear system, the Lotka-Volterra (LV) model, using the LNA to approximate its dynamics. Simulations of the LV model using the Gillespie algorithm and the LNA showed us that the LNA would not be an appropriate simulator of the system. However, since we simulated observations of the system, the LNA was still able to give accurate results for inference. We compared parameter estimation results with aggregated observations over a period of 2 minutes where KF2 outperformed KF1.

Finally, we studied a single gene expression (SGE) model. Again, its dynamics were approximated by the LNA, which also seemed to be a reasonable simulator of the system. Informative priors were needed in order to ensure identifiability of the system. The effect of integration was apparent in the inferred switch times of the system only after a period of 3 hours. All studies in this chapter were concerned with synthetic data. In the next chapter, we are going to study microscopy data with an inherently aggregated nature.

Chapter 5

Results on real data from single cell experiments

In this chapter, we apply our method to real data from a translation inhibition experiment. By inhibiting protein translation, it is possible to estimate the protein degradation rate. We first discuss the case where all cells have the same initial protein abundance. We further study the case of different initial conditions for each cell, which accounts for extrinsic noise but increases the size of the parameter space. We have tested both cases using synthetic data before applying them on the experimental data. We conclude that extrinsic noise, as well as aggregation, should be modelled explicitly to obtain accurate results.

5.1 Translation inhibition model

In Section 4.3, we studied a stochastic gene expression model and performed inference using synthetic data corresponding to aggregated observations of the protein molecules. In order to perform inference, we assumed that the degradation rates of both the protein and mRNA were known. Here, we focus on inferring the protein degradation rate from a translation inhibition experiment. The purpose of a translation inhibition experiment is to decrease the protein production to a basal level allowing us to measure the rate at which the initial protein abundance degrades. As we are only interested in inferring the protein degradation rate we choose to use a simple model that does not consider mRNA degradation. We assume the following translation inhibition model involving two molecular species $X = (R, P)$, where R and P stand for gene mRNA and protein molecules respectively:

$$R \xrightarrow{c_P} R + P \ \textit{(translation)} \tag{1a}$$

$$P \xrightarrow{d_P P/\Omega} \oslash \ \textit{(protein degradation)} \tag{1b}$$

The vector of the stochastic rate constants is denoted by $c = (c_p, d_p)$, where c_P corresponds to the protein basal rate and d_p to the protein degradation rate. The stoichiometry matrix corresponding to the biochemical network (1) is given by

$$S = \begin{bmatrix} 1 & -1 \end{bmatrix}, \tag{2}$$

while the hazard function is given by

$$\tilde{f}(X) = \begin{bmatrix} c_P \\ d_P P \end{bmatrix}. \tag{3}$$

The macroscopic part of the system can then be formulated according to $d\phi_i \backslash dt = S_i \tilde{\boldsymbol{f}}(\phi_t)$, giving rise to a linear ODE,

$$\frac{d\phi_p}{dt} = c_P - d_P \phi_p. \tag{4}$$

The stochastic part of the system is easily obtained using the following matrices:

$$F = \begin{bmatrix} 0 & d_P \end{bmatrix}, \tag{5}$$

$$SF^T = A = \begin{bmatrix} -d_P \end{bmatrix}, \tag{6}$$

$$S\text{diag}(\tilde{f}(\phi_t))S^T = EE^T = \begin{bmatrix} c_P + d_P \phi_p \end{bmatrix}, \tag{7}$$

such that the variance of the stochastic part V_p is given by

$$\frac{dV_p}{dt} = -2d_P V_p + c_P + d_P \phi_p. \tag{8}$$

For the deterministic part of the integrated process I_p we simply have:

$$\frac{dI_p}{dt} = \phi_p, \tag{9}$$

Finally, the variance of the integrated process S_p and the covariance of the integrated process with the unintegrated process C_p are given by:

$$\frac{dC_p}{dt} = -d_P C_p + V_p \tag{10}$$

$$\frac{dS_p}{dt} = 2C_p \tag{11}$$

Since (1) is a low dimensional linear system, it is also possible to work with the analytical solution of the system of ODE equations (4), (8), (9), (10), (11).

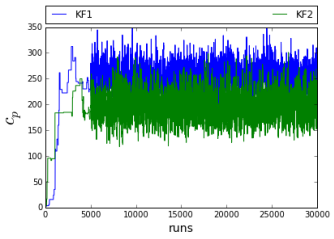
To test the performance of our method, we have simulated synthetic data according to (1) using the Gillespie algorithm. We simulated 30 time series (corresponding to 30 different cells), assuming the following values as the ground truth for the kinetic parameters: $c_P = 200$ and $d_P = 0.97$. We further set the initial protein abundance of m_0 to 400 molecules. We have scaled the data by a factor $k = 0.03$ so that they are proportional to the original synthetic data and added Gaussian noise with a variance of $s = 0.1$. For the purposes of this study, we have assumed that data were integrated over 30 minutes.

Parameter estimation results from running the adaptive MCMC are summarised in Table 5.1. Non-informative exponential priors with mean 10^4 were placed on all parameters. We have adopted the parametarisation used in [47, 20] such as $\widetilde{c}_P = k \cdot c_p$ and $\widetilde{m}_0 = k \cdot m_0$ and worked in the log parameter space. MCMC traces and posterior histograms are presented in Figure 5.1. As we can see the degradation rates are successfully inferred by both KF1 and KF2, however, KF1 cannot infer the initial condition and underestimates the noise level.

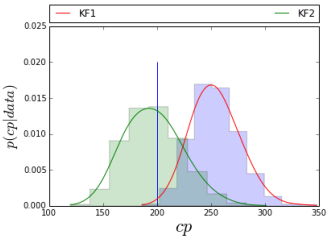
c	Ground Truth1	KF1	KF2
c_p	200	254.0284 ± 22.2252	196.6355 ± 26.4500
d_p	0.97	0.9819 ± 0.0350	0.9949 ± 0.0431
s	0.1	0.0351 ± 0.0244	0.0998 ± 0.0097
k	0.03	0.0236 ± 0.0016	0.0312 ± 0.0040
m_0	400	589.0252 ± 42.6292	393.2417 ± 50.4983

Table 5.1: Mean posterior ± 1 s.d. for (c_P, d_P, s, k, m_0) using an adaptive MCMC. Data were simulated from a translation inhibition model according to the ground truth values.

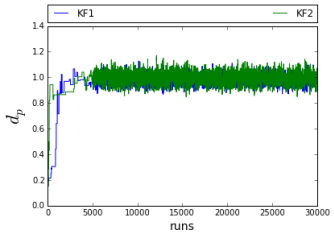
Additionally, we have used the Nelder-Mead algorithm to assess further the validity of our results using different datasets. For that reason, we generated 10 different datasets, each consisting of 30 time series. Due to convergence issues with the Nelder-Mead algorithm, initialisation was done at the ground truth values. These results are only used as an indication of the maximum likelihood regions using KF1 and KF2. Indeed, the tendency to underestimate the noise variance and overestimate the initial molecule numbers by KF1 is confirmed by



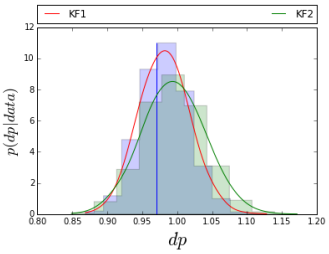
(a) MCMC traces for c_P.



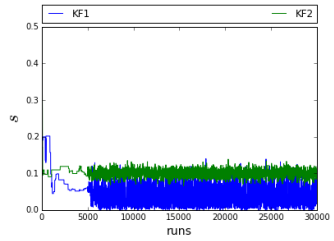
(b) Posterior of c_P.



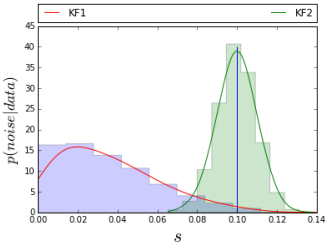
(c) MCMC traces for d_P.



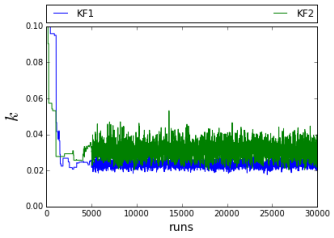
(d) Posterior of d_P.



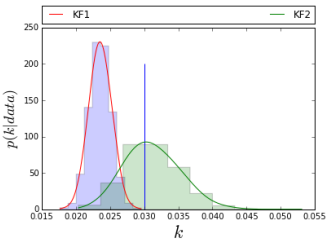
(e) MCMC traces for s.



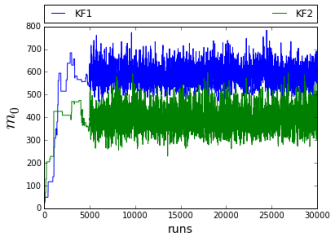
(f) Posterior of s.



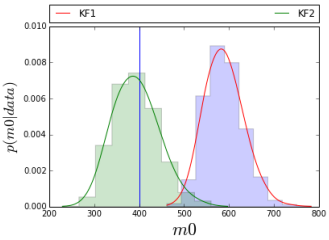
(g) MCMC traces for k.



(h) Posterior of k.



(i) MCMC traces for m_0.



(j) Posterior of m_0.

Figure 5.1: MCMC traces and histograms of the posterior of the parameters (c_P, d_P, s, k, m_0) using an adaptive MCMC for both KF1 and KF2. Ground truth for the parameters and is indicated by the vertical red line on the histogram plots.

the results presented in Table 5.2. These results can be intuitively explained, as data is smoothed by averaging, such that low noise and higher molecular abundance agrees with a macroscopic view of the system that underestimates its inherent stochasticity.

c	Ground Truth	KF1 Median[LB,UB]	KF2 Median[LB,UB]
c_p	200	251.2680 [232.9793,270.3414]	213.6844[191.0145,240.4731]
d_p	0.97	0.9748 [0.9526,0.9873]	0.9891 [0.9694,1.0074]
s	0.1	$3.6{\cdot}10^{-8}$ [$2.6{\cdot}10^{-8}$, $4.1{\cdot}10^{-8}$],	0.1018 [0.0993, 0.1059]
k	0.03	0.0236 [0.0226,0.0239]	0.0274 [0.0255,0.0321]
m_0	400	579.4208 [577.3605,611.2480]	441.0402 [378.6169],475.2906]

Table 5.2: Nelder-Mead results for (c_P, d_P, s, k, m_0) across 10 different datasets. Median values are shown on the third and fourth column for KF1 and KF2 respectively, while lower and upper bounds are shown in brackets.

5.2 Data with similar initial conditions

We are now ready to use our model for fitting real data from single cell experiments. We have data from a translation inhibition experiment using rat pituitary GH3 cells [40] expressing luciferase under the control of the human prolactin promoter. Cycloheximide (CHX) was chosen as an inhibitor of protein synthesis in the cells. At first, the cells were stimulated with 5μM of Forskolin and 0.5μM of BayK-8644 for some hours (3-6). 10μg/ml of CHX were then added and were subsequently imaged for some more hours.

The cells were seeded in 35 mm glass coverslip-based dishes for 20 hours before imaging. 1 mM of luciferin was then added to the cells, and they were transferred to a Zeiss Axiovert 200 with an XL incubator where they were maintained at standard cell incubator conditions (37°C, 5% CO2). Luminescence images were taken using an air objective (Fluar x20, 0.75 NA Zeiss) and a photocounting charge coupled device camera. The integration period of the luminescence signal was 30 minutes. The Kinetic Imaging software AQM6 was used for analysing sequential images of the cells. In Figure 5.2, we can see an image of the luminescence signal in GH3 cells from the translation inhibition experiment. The 37 circled cells are the ones that were considered for the final analysis.

In the following, we study a subset of the 37 cells that are depicted in Figure 5.2. The time series corresponding to the subset of cells after the addition

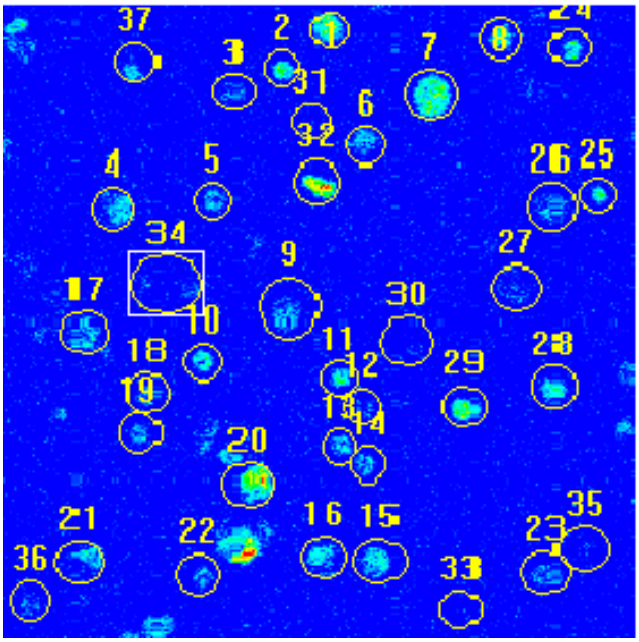


Figure 5.2: Luminescence signal in GH3 cells from the translation inhibition experiment. Circled cells were chosen to be analysed.

of CHX is shown in Figure 5.3. Equivalent datasets have been studied in [40]. However, a deterministic approach was used to estimate the degradation rate of the luciferase protein. We have tested fitting an exponential decay curve to an average over all cells in the subset by using the curvefit function of the scipy module. This resulted in a degradation rate of 1.01811877. A Bayesian approach can also be used in a deterministic setting, as presented in [19].

We used, again, an adaptive MCMC algorithm with exponential priors in order to estimate the unknown parameters. We first tried to fit the time series from each cell separately. However, this led to very flat posteriors for c_p and m_0, indicating high uncertainty for our results and the need for additional data. Results from a single cell are shown in Table 5.4.

c	KF1	KF2
c_p	3951.9602 ± 2894.6981	3739.0568 ± 3357.08006
d_p	1.2444 ± 0.1229	1.2603 ± 0.1289
s	0.0119 ± 0.0090	0.0033 ± 0.0023
k	0.0023 ± 0.0017	0.0028 ± 0.0025
m_0	7938.0594 ±5846.8587	6184.9648 ± 5475.5964

Table 5.3: Mean posterior ± 1 s.d. for (c_P, d_P, s, k, m_0) from an adaptive MCMC using only one cell.

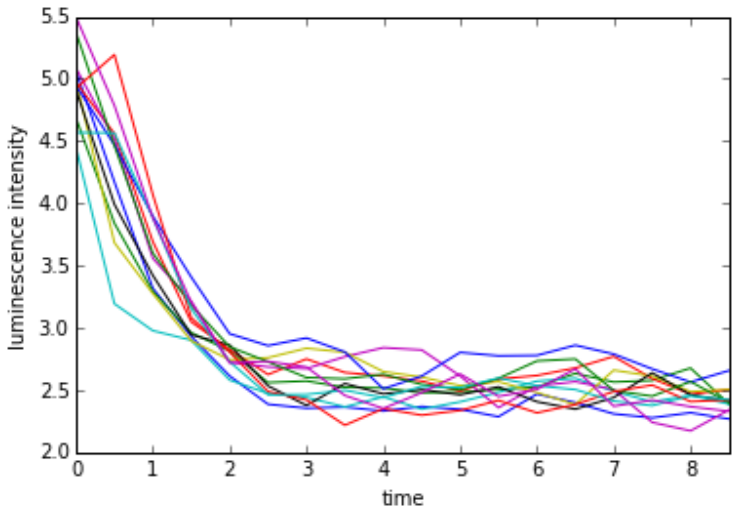
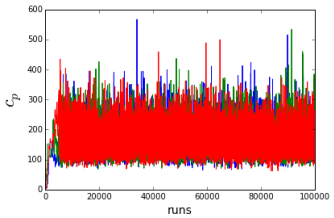


Figure 5.3: Time series of 13 cells from the translation inhibition experiment.

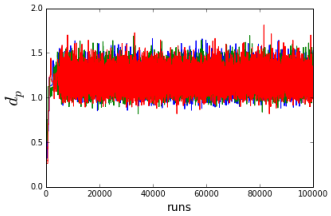
To get better estimates, we then fit all cells together assuming independence between their time series. Parameter estimation results using KF1 and KF2 on all 13 cells are shown in Table 5.4. In accordance with the synthetic data, the estimated initial condition is higher with KF1 than with KF2. To assess convergence with real data, we ran two additional chains with different starting values to calculate the Gelman-Rubin convergence diagnostic [22]. MCMC traces for 3 chains using KF2 are shown in Figure 5.4, and traces for KF1 can be found in Appendix B.6. The main parameter of interest in this experiment is the degradation rate. In Figure 5.5 we can see that the degradation rates inferred by KF1 and KF2 are close to each other, and both include the deterministic value of 1.01811877 within their credible regions. This is expected due to the simple form of this experiment, as the data follow their deterministic mean. We have also tried fitting the data using improper uniform priors on all parameters, which showed no difference in the results.

c	KF1	KF2
c_p	217.7365 ± 32.3792	167.8247 ± 42.2919
d_p	1.1041 ± 0.0762	1.2082 ± 0.1070
s	0.0026 ± 0.0025	0.0079 ± 0.0038
k	0.0255 ± 0.0028	0.0379 ± 0.0089
m_0	450.2443 ± 52.3521	273.7025 ± 68.3398

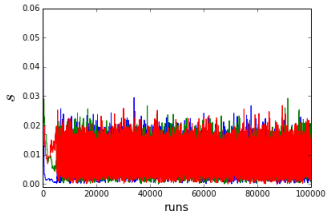
Table 5.4: Mean posterior ± 1 s.d. for (c_P, d_P, s, k, m_0) using an adaptive MCMC with real data.



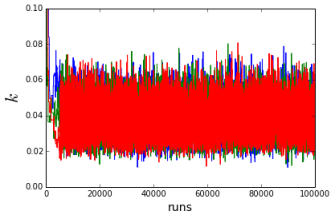
(a) MCMC trace of c_P.



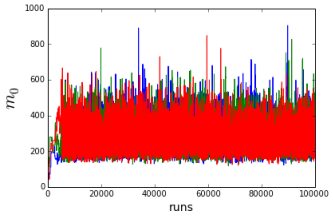
(b) MCMC trace of d_P.



(c) MCMC trace of s.



(d) MCMC trace of k.



(e) MCMC trace of m_0.

Figure 5.4: Adaptive MCMC traces of the translation inhibition model parameters using KF2 with real data.

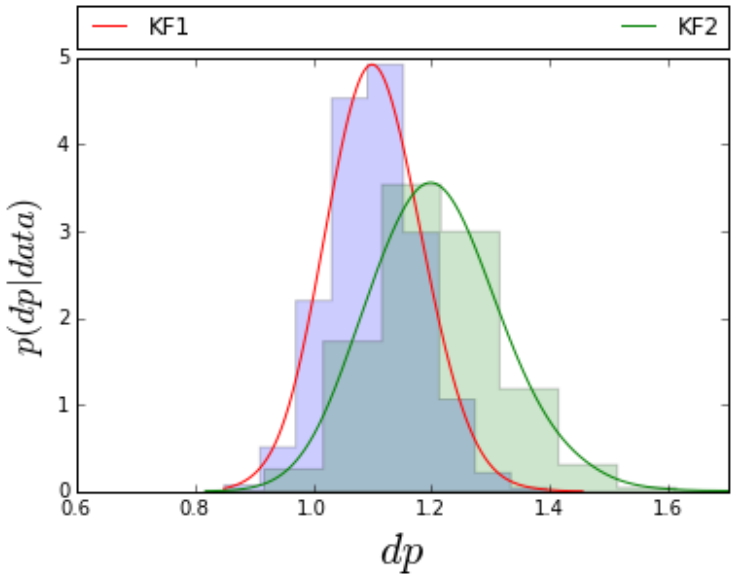


Figure 5.5: Posterior histograms of degradation rate using KF1 and KF2.

It is necessary to mention here that in [47], fluorescence data from a translation inhibition experiment of the same biological group were studied using the Non-Restarting KF1. As fluorescence data do not have the aggregated nature of luminescence data, there is no need for using KF2.

5.3 Data with heterogeneous initial conditions

If we now take a look at the whole dataset of the 37 cells in Figure 5.6, we see that there is a lot of variability in the initial conditions between the different cells. The variability in the initial conditions can be caused by different external factors and can be regarded as extrinsic noise. In this section, we consider different initial conditions for each cell, something that will significantly increase the parameter space of our model. Before we move on fitting the translation inhibition model

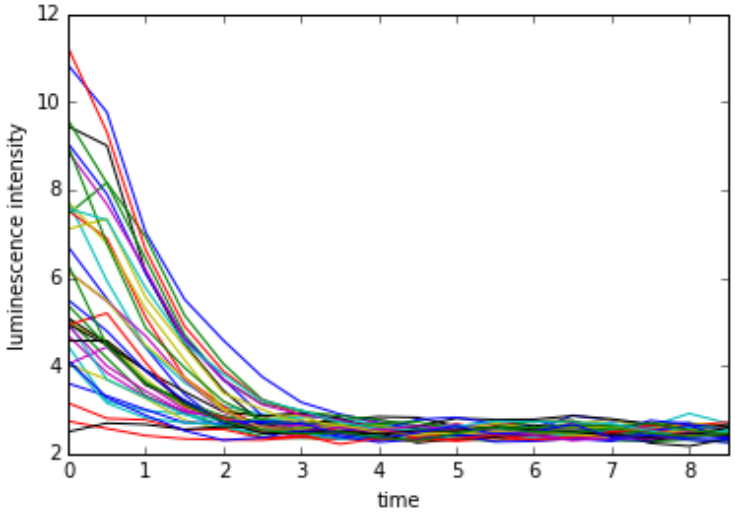


Figure 5.6: Time series of all 37 cells from the translation inhibition experiment.

with the different initial conditions of each cell to real data, we will work with a synthetic dataset of 30 cells as before, where half of them come from an initial population of 400 molecules and the other half from an initial population of 300 molecules; the rest of the parameters follow our previous setting.

In the experiment, we have prior information on the protein levels just before adding the CHX, which can be used as a prior for the initial protein abundance $\widetilde{m}_{i0}$ of each cell i. Consequently, we will use an informative prior for the $log(\widetilde{m}_{i0})$ in the synthetic dataset as well. We further place a $N(0, 2)$ prior on the log variance, so that it does not reach extreme values. Improper uniform priors were set on the rest of the parameters. Inference results using KF2 with the adaptive MCMC for 150K iterations are shown in Figure 5.7. We can clearly see in Figure 5.7 (j) that the two modes of m_0 have been correctly inferred, as well as the rest of the parameters. MCMC results using KF1 are presented in Figure 5.8, where we can see that it was not possible to converge to a value for s even

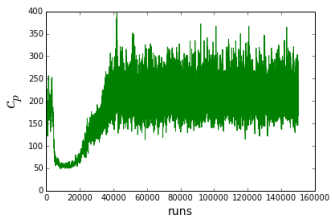
after 200K iterations. We have also tested inference by placing improper uniform priors on all parameters but inference with KF2 was not affected.

We proceed by fitting the reduced dataset of the 13 cells of Figure 5.3, considering a different initial condition for each one of them. We will use, again, normal priors for $log(\widetilde{m}_{i0})$ and $log(s)$. However, we will only show results using KF2, as convergence with KF1 would, as already indicated by the experiment with synthetic data, take a longer time. To ensure convergence of the MCMC, we ran 3 chains and calculated the Gelman-Rubin statistic [22], which was close to 1 for all parameters. In Figure 5.9, we see the MCMC traces of the 3 chains which indicate convergence after the first 20K iterations. Again, the deterministic value of the degradation rate is included in the posterior histogram of d_p.

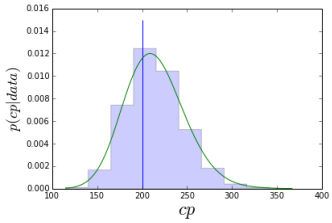
Finally, we fit the whole dataset of the 37 cells considering, again, different initial conditions for each cell. We adopt the same priors as in the case of the reduced dataset and run three MCMC chains for calculating the Gelman-Rubin statistic [22]. In Figure 5.10 we see the results of fitting the experimental data. Convergence was slower than fitting the reduced dataset but the posterior histograms are narrower indicating more confidence in our results. Notice that the histograms of the initial conditions m_0 are now more spread as the full dataset includes highly heterogeneous cells.

As we cannot know the real values of the parameters, we have generated data using the inferred values from the MCMC as shown in Figure 5.10 and compared them to the time series of the experimental data of Figure 5.6. In Figure 5.11, we have plotted sampled values (red dots) from synthetic data against the experimental data. This way we have verified that the inferred values can reproduce time series equivalent to the experimental data.

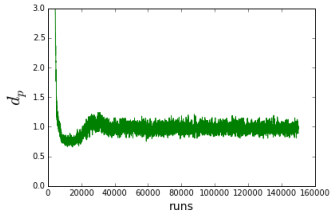
In [20], the authors developed a hierarchical model following their work in [47] to deal with the extrinsic noise related to the heterogeneity between the different cells. They have concluded that protein degradation rates show little cell to cell variability, while the basal rates show more cell to cell variability as they are also affected by the initial protein abundance in each cell. Although considering different initial condition for each cell captures a lot of their variability, we believe that a hierarchical approach could benefit our method as well.



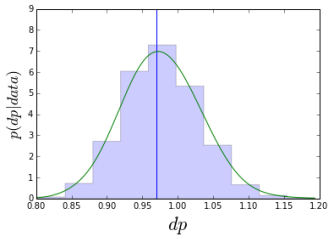
(a) KF2 - MCMC trace c_P.



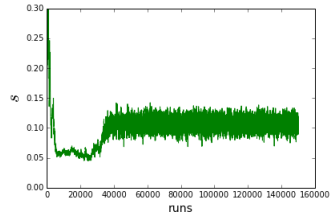
(b) KF2 - Posterior c_P.



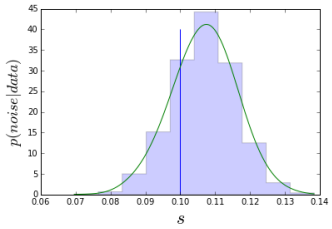
(c) KF2 - MCMC trace d_P.



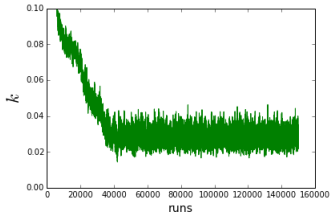
(d) KF2 - Posterior d_P.



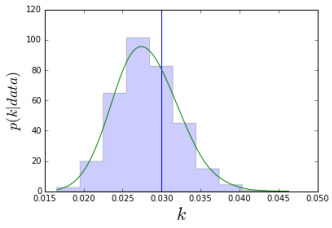
(e) KF2 - MCMC trace s.



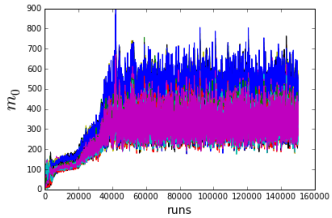
(f) KF2 - Posterior s.



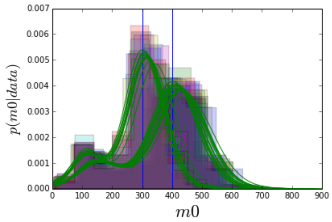
(g) KF2 - MCMC trace k.



(h) KF2 - Posterior k.

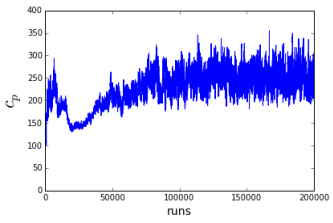


(i) KF2 - MCMC trace of all m_0.

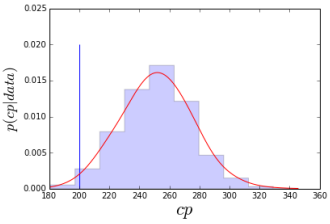


(j) KF2 - Posterior of all m_0.

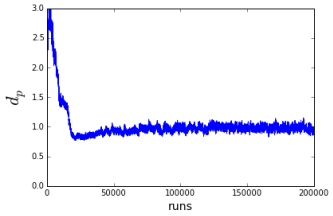
Figure 5.7: Adaptive MCMC traces and histograms of the parameters of the translation inhibition model for different initial conditions using KF2. Ground truth for the parameters is indicated by a vertical blue line on the histogram plots.



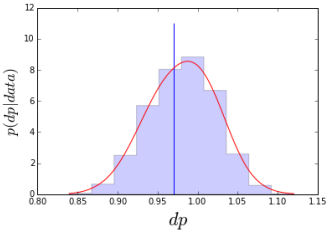
(a) KF1 - MCMC trace c_P.



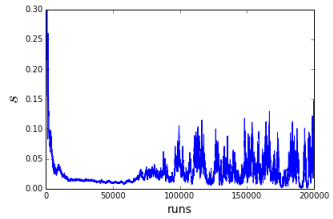
(b) KF1 - Posterior c_P.



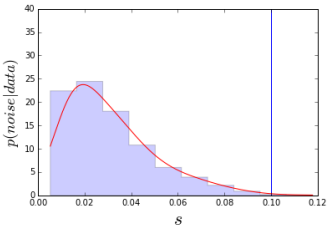
(c) KF1 - MCMC trace d_P.



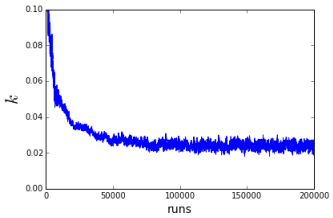
(d) KF1 - Posterior d_P.



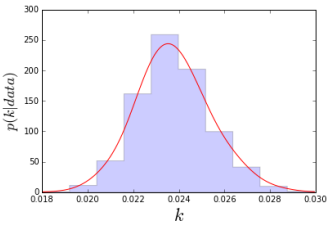
(e) KF1 - MCMC trace s.



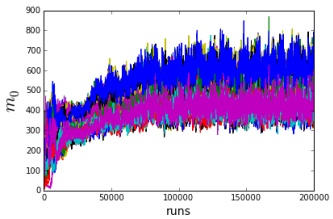
(f) KF1 - Posterior s.



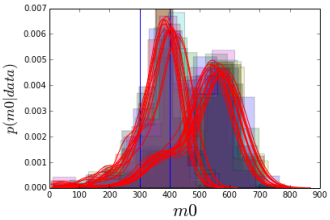
(g) KF1 - MCMC trace k.



(h) KF1 - Posterior k.

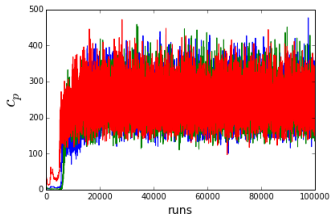


(i) KF1 - MCMC trace of all m_0.

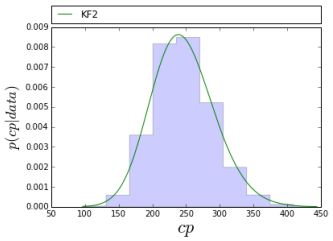


(j) KF1 - Posterior of all m_0.

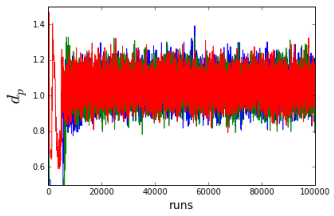
Figure 5.8: Adaptive MCMC traces and histograms of the parameters of the translation inhibition model for different initial conditions using KF1. Ground truth for the parameters is indicated by a vertical blue line on the histogram plots.



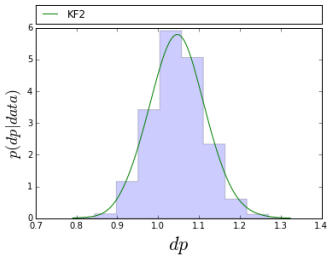
(a) KF2 - MCMC trace c_P.



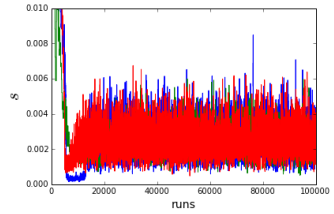
(b) KF2 - Posterior c_P.



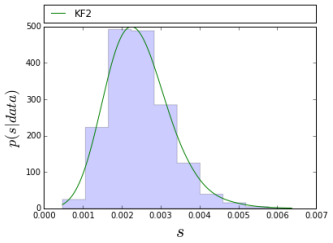
(c) KF2 - MCMC trace d_P.



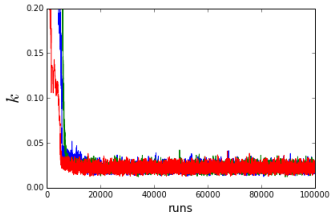
(d) KF2 - Posterior d_P



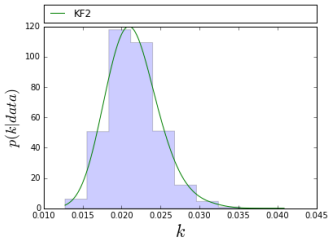
(e) KF2 - MCMC trace s.



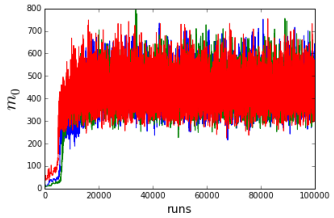
(f) KF2 - Posterior s.



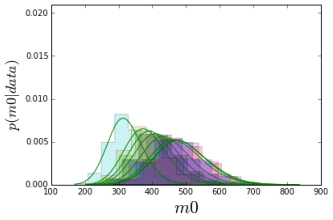
(g) KF2 - MCMC trace k.



(h) KF2 - Posterior k.

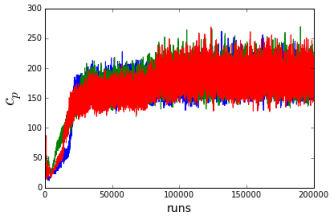


(i) KF2 - MCMC trace m_0 from cell1.

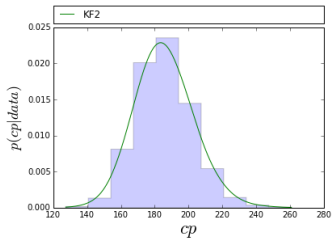


(j) KF2 - Posterior m_0 from cell1.

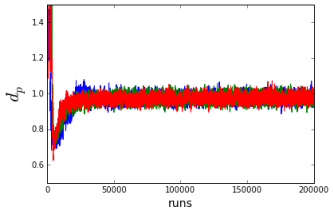
Figure 5.9: Adaptive MCMC traces and histograms of the parameters of the translation inhibition model for different initial conditions using the reduced dataset with KF2.



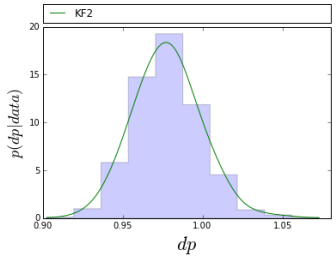
(a) KF2 - MCMC trace c_p.



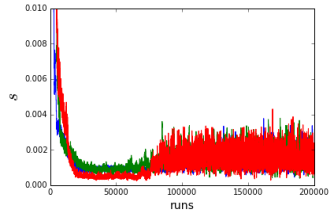
(b) KF2 - Posterior c_P.



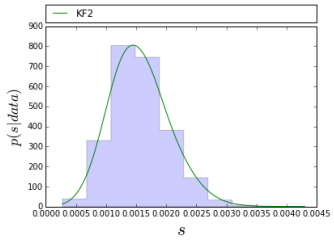
(c) KF2 - MCMC trace d_p.



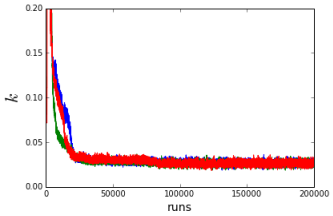
(d) KF2 - Posterior d_P.



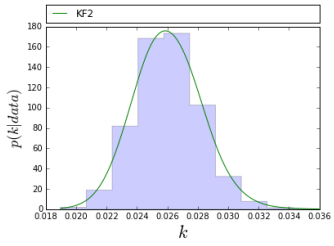
(e) KF2 - MCMC trace s.



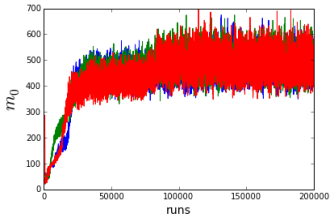
(f) KF2 - Posterior s.



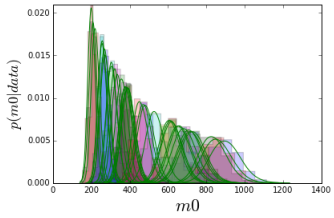
(g) KF2 - MCMC trace k.



(h) KF2 - Posterior k.



(i) KF2 - MCMC trace m_0 from cell1.



(j) KF2 - Posterior m_0 from cell1.

Figure 5.10: Adaptive MCMC traces and histograms of the parameters of the translation inhibition model for different initial conditions using the full real dataset with KF2.

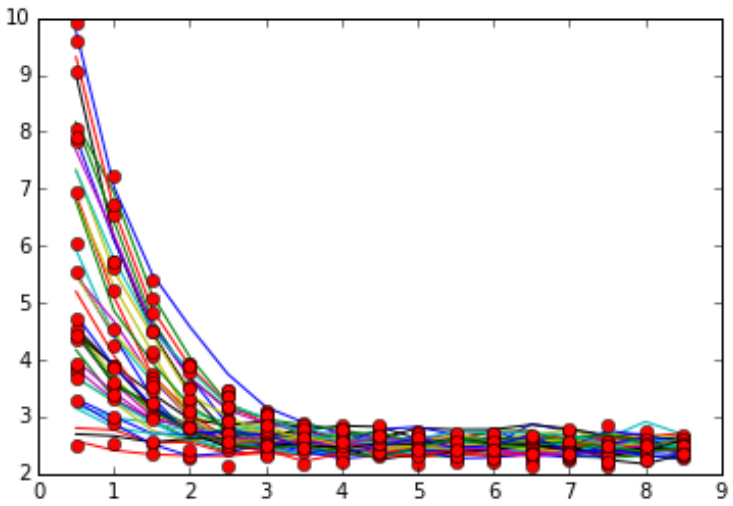


Figure 5.11: Experimental data against synthetic data. The colored traces correspond to the experimental time series from Figure 5.6 from 0.5 to 8.5 hours. The red dots correspond to aggregated synthetic data simulated from the translation inhibition system with different initial conditions using the inferred parameter values corresponding to Figure 5.10.

5.4 Summary

In this chapter, we studied a basic system corresponding to a translation inhibition experiment. We performed our analysis using luciferase data with an integration period of 30 minutes. At first, we attempted to fit the data assuming that the different cells in the dataset had the same initial protein abundance. We first verified our results using synthetic data, where ignoring the aggregated nature of the data (KF1) resulted in the wrong estimation of noise level and initial protein abundance. For this case, we fitted only a subset of the experimental data that had similar initial protein levels.

We further studied the case were the initial protein abundance of each cell is a different parameter in our system. This lead to an increase in the parameter space, which made inference more difficult. Again, parameter estimation with synthetic data was tested before using the real data. When ignoring the aggregated nature of the data (KF1), it was not possible to reach convergence for all parameters at a reasonable time, so real data were tested only in conjunction with KF2.

Chapter 6

Conclusion

6.1 Conclusions and contributions to research

In this thesis, we studied inference in stochastic systems given temporally aggregated data. We were motivated by recent advances in single cell imaging data and were particularly interested in modelling luciferase data, due to their aggregated nature. To our knowledge, luciferase data are typically treated in much the same way as fluorescence data, which comprises observations of single time points rather than aggregated signals. We based our work on [47] and [18], which consider inference in biochemical reaction networks stochastically using the Linear Noise Approximation (LNA). We extended the work of [47] and [18] to take aggregated data into account when carrying out inference of state trajectories and model parameters.

We decided to work with the LNA, as it was used in [47, 18], which provided us with a tractable Gaussian likelihood for inference. The LNA was presented in detail in Chapter 2 and derived for a simple system (translation inhibition model) following the proof of van Kampen.

The Kalman Filter (KF) framework provides us with a straightforward way to update the state of a system given noisy measurements. When the state of the system is described by an SDE but is observed at discrete time points, we refer to a continuous-discrete KF. The LNA results in an approximation of the state dynamics by a linear SDE (in the narrow sense) that can represent the state process in a continuous-discrete KF. As we were interested in aggregated observations, we needed to consider the integral of the LNA. The solution of a linear SDE is a Gaussian process; thus, the integral of the LNA is another Gaussian process. The solution of the LNA X_t and its integral H_t are characterised by their two means, variances and their covariance. The mean and variance of X_t were already known, but we were able to provide proofs for the other three quantities corresponding to the mean and variance of H_t, as well as the covariance of X_t with H_t. Together

X_t and H_t form a Markov process that is used as the state process in the KF. This way, we have developed a novel method for treating temporally aggregated data, which can be applied to any stochastic system described by a linear SDE.

We have presented inference results from different stochastic systems with aggregated observations using our method. By aggregating over time, we smooth out the original process and reduce the fluctuations. Therefore, the stochastic contribution of the observed process may be underestimated if aggregation is not taken into account. We, therefore, expected that aggregation would influence the inferred process. To assess its impact, we compared two approaches for fitting temporally aggregated data. In the first approach, we assumed that the data were coming directly from the system without aggregation (KF1), and in the second one we used our method for aggregated data (KF2). We followed a Bayesian approach in parameter estimation so, in all cases, we have used an (adaptive) Metropolis-Hastings algorithm. We provided, however, whenever possible, results using an optimisation algorithm. The Nelder-Mead algorithm was chosen among other optimisation algorithms in the scipy library, as it was able to converge in more cases. The success of the Nelder-Mead algorithm is attributed to its affine-invariant property [38]. Since it was not always possible for the Nelder-Mead to converge to the global optimum, it had to be initialised, in most cases, very close to the ground truth values.

The study of an analytically tractable system such as the Ornstein-Uhlenbeck (OU) model was important for understanding, theoretically, how aggregation changes the properties of the original process. In the case of the OU process, for example, we saw that by taking its integral, we no longer have a stationary process. So, the aggregated and the original process do not have the same statistical properties. As confirmed by our study of the OU process, the effect of aggregation becomes more apparent as we increase the aggregation period Δt. For large Δt ignoring aggregation led to an underestimation of the process variance and inaccurate estimation of the parameters. The OU model, therefore, helped us to gain insight on the importance of aggregation in modelling and parameter estimation.

The Lotka-Volterra (LV) model corresponds to a non-linear, two-dimensional system. Surprisingly, although the LNA did not appear to be a suitable simulator for the LV model, it was found to be appropriate for inference using the Restarting method [23, 18]. Aggregation over two minutes in our simulations led

to inaccuracies in the inferred values of the unknown model parameters when using KF1 instead of KF2. Our method (KF2) was able to give accurate parameter estimation results using aggregate measurements from only one of the species. The adaptive MCMC was needed for fast convergence due to high correlations between the unknown parameters, which lead to slow convergence when using a standard Metropolis-Hastings (MH) sampler.

A Single Gene Expression (SGE) model [47] has also been studied, since our initial motivation for this thesis originated from Systems Biology. This model assumes that there is a switch in the level of transcription at a specific point in time that needs to be inferred. Synthetic data corresponding to aggregated protein levels were sampled with the help of the Gillespie algorithm, with no knowledge of the mRNA levels being assumed apart from the initial abundance. This model is identifiable only if we assume that the degradation rates of both the protein and the mRNA are known. Knowledge of the degradation rates and initial molecular levels seem to restrict the parameter space enough so that both KF1 and KF2 would give good estimates for short aggregation periods. As the aggregation period becomes larger, e.g. 3 hours, we showed that only KF2 was able to recover the correct switch time of the model. Using a different setting, where for example the mRNA half-life is shorter, we can expect that the effect of aggregation could become apparent in a shorter aggregation interval than 3 hours.

Finally, we applied our method to Luciferase reporter data from single cells from a translation inhibition experiment. The purpose of the translation inhibition experiment is to infer the protein degradation rate that is used as prior information in the SGE model. Luciferase data are treated as aggregated data with added Gaussian noise. We first assumed that the initial protein abundance in each cell was the same, and fitted a subset of luciferase data whose time series were starting around the same values. We compared parameter estimation results using KF1 and KF2 using both the luciferase data and synthetic data and observed that KF1 would overestimate the initial protein abundance and underestimate the noise level. Although the degradation rate posteriors were close to each other and both included the deterministic estimate, the abundance of molecules is also an important parameter of the system as it can suggest the appropriate modelling technique. We further studied the case where all cell time series had different initial protein levels in an attempt to include extrinsic noise.

The increased parameter space makes inference more challenging and a longer burn-in period is needed for the MCMC sampler to converge. The MCMC sampler was not able to converge using KF1 when tested with synthetic data, and real data were only modelled using KF2.

In summary, we have developed and evaluated a novel method for inference in stochastic systems using aggregated measurements. Our examples have shown that aggregation needs to be taken into consideration for the task of inference for stochastic systems and our method provides a way to successfully deal with it.

6.2 Discussion and future work

In this thesis, we have focused on developing a methodology for treating aggregated data in stochastic systems. In the following, we will briefly describe some of the directions we want to pursue in the future.

6.2.1 Hierarchical structure

Extrinsic noise is an important source of heterogeneity among cells. Heterogeneity plays an important role in biological processes, as, for example, it can control population robustness [62]. Consequently, it should be taken into account when developing computational models. When we fitted the translation inhibition model to the luciferase data, we only accounted for extrinsic noise by assuming different initial protein levels among the cells. However, we would like to adopt a more elegant approach such as a hierarchical model [20] that has the additional advantage of treating more parameters as heterogeneous among the cell population. The hierarchical model in [20] is an extension of [47] and our method could be applied in the specific framework immediately.

6.2.2 Applications

The translation inhibition model that we studied was a simple system; the main parameter of interest (degradation rate) would not necessarily require a stochastic approach. Consequently, we would be interested in fitting more complex models to luciferase data. The SGE model presented in Chapter 4, or the more general switch model in [42] that allows for multiple switch points, have a more complex

structure than the translation inhibition model. In the future, we would be interested in using our method to model experimental data from these systems.

An example of a system where stochasticity is crucial for understanding its dynamics is presented in [63]. There, a Hes1/miR-9 oscillator is studied, and the LNA was successfully used as a simulator of the system when the system size was sufficiently large. It would be of interest to study luciferase data corresponding to this model, as it is a system that necessitates a stochastic approach. Additionally, it is suggested that molecular abundance is an important parameter of this system, and we have already shown in our translation inhibition model that, by ignoring temporal aggregation, we are lead to inaccurate estimates of the initial molecular abundance.

We would be particularly interested in applying our method in different domains such as finance. Integrated stochastic processes are studied in finance in the concept of integrated volatility [5]. Additionally, temporal aggregation appears in economic time series, for example in income data [74]. Therefore, we believe that there are potential applications of our method in that domain as well.

6.2.3 Inference and computational efficiency

Most of the systems we studied exhibited a strong correlation between the parameters, and an adaptive MCMC was needed for convergence at a reasonable time. Studying more complex systems would lead to an even harder inference problem; therefore, it would be advisable to study alternative MCMC schemes. In [32], the authors have exploited the geometric structure of the parameter space in order to develop an efficient MCMC algorithm which was shown to outperform an adaptive MH. This algorithm has also been applied to the SGE model studied in Section 4.3 [79]. Therefore, we believe it would lead to a more efficient MCMC for the systems we studied. Strong correlations can also arise between the state and parameters which can make an MCMC inefficient. A way to tackle this problem is to jointly update the state and parameters [17].

The KF methodology is very intuitive and easy to implement algorithmically. However, it can be computationally expensive, due to the matrix operations involved in its steps. Additionally, in our specific applications, it was necessary to solve ODE systems for the state process which can become particularly costly

when dealing with stiff systems. Since we are using Bayesian inference, the likelihood function from the KF needs to be computed for thousands of iterations, which slows down the implementation of our algorithm considerably. We have implemented our method using Python 2.7 and Numpy 1.10.4. We believe that the runtime could be significantly reduced by using a more efficient, compiled language such as C/C++. An alternative would be the Cython language, which allows for the integration of (efficient) C functions with regular Python scripts.

Bibliography

[1] L. Arnold. *Stochastic differential equations theory and applications*. [S.l.], 1974. (Cited on pages 32 and 33.)

[2] O. G. Bahcall. Single cell resolution in regulation of gene expression. *Molecular Systems Biology*, 1, 2005. (Cited on page 50.)

[3] F. Baltazar-Larios and M. Sørensen. Maximum Likelihood Estimation for Integrated Diffusion Processes. In C. Chiarella and A. Novikov, editors, *Contemporary Quantitative Finance: Essays in Honour of Eckhard Platen*, pages 407–423. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. (Cited on page 64.)

[4] Y. Bar-Shalom, X. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, New York, NY, USA, 2001. (Cited on page 59.)

[5] O. E. Barndorff-Nielsen and N. Shephard. Aggregation and model construction for volatility models. Technical report, Department of Mathematical Sciences, Aarhus University, Denmark, 1998. (Cited on pages 22 and 112.)

[6] C. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2007. (Cited on pages 54, 55, and 125.)

[7] G. Box, G. M. Jenkins, and G. Reinsel. *Time Series Analysis: Forecasting & Control (3rd Edition)*. Prentice Hall, 3rd edition, 1994. (Cited on page 26.)

[8] R. Boys, D. Wilkinson, and T. Kirkwood. Bayesian inference for a discretely observed stochastic kinetic model. *Statistics and Computing*, 18(2):125–135, 2008. (Cited on pages 51 and 80.)

[9] R. Brown. A brief account of microscopical observations made in the months of June, July and August, 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies. 1828. (Cited on page 29.)

[10] L. Cai, N. Friedman, and X. S. Xie. Stochastic protein expression in individual cells at the single molecule level. *Nature*, 440(7082):358–362, 2006. (Cited on page 50.)

[11] Y. Cao, D. T. Gillespie, and L. R. Petzold. Efficient step size selection for the tau-leaping simulation method. *Journal of Chemical Physics*, 124(4), 2006. (Cited on page 40.)

[12] F. Comte, V. Genon-Catalot, and Y. Rozenholc. Nonparametric estimation of a discretely observed integrated diffusion model. In *MAP 5, Mathématiques Appliquées - Paris 5, UMR CNRS 8145*, 2006. (Cited on page 64.)

[13] R. Durrett. *Essentials of Stochastic Processes*. Springer, 1999. (Cited on page 28.)

[14] A. Einstein. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der Physik*, 322(8):549–560, 1905. (Cited on page 29.)

[15] J. Elf and M. Ehrenberg. Fast evaluation of fluctuations in biochemical networks with the linear noise approximation. *Genome Research*, 13(11):2475–2484, 2003. (Cited on pages 43 and 44.)

[16] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain. Stochastic gene expression in a single cell. *Science (New York, N.Y.)*, 297(5584):1183–1186, 2002. (Cited on page 18.)

[17] P. Fearnhead. MCMC for state-space models. In S. Brooks, A. Gelman, G. Jones, and X. Meng, editors, *Handbook of Markov Chain Monte Carlo*, Chapman & Hall/CRC Handbooks of Modern Statistical Methods, pages 513–529. Chapman and Hall, 2011. (Cited on pages 63 and 112.)

[18] P. Fearnhead, V. Giagos, and C. Sherlock. Inference for reaction networks using the linear noise approximation. *Biometrics*, 70(2):457–466, 2014. (Cited on pages 18, 21, 40, 48, 52, 61, 62, 73, 80, 108, and 109.)

[19] B. Finkenstädt, E. A. Heron, M. Komorowski, K. Edwards, S. Tang, C. V. Harper, J. R. E. Davis, M. R. H. White, A. J. Millar, and D. A. Rand.

Reconstruction of transcriptional dynamics from gene reporter data using differential equations. *Bioinformatics*, 24:2901–7, 2008. (Cited on page 98.)

[20] B. Finkenstädt, D. J. Woodcock, M. Komorowski, C. V. Harper, J. R. E. Davis, M. R. H. White, and D. A. Rand. Quantifying intrinsic and extrinsic noise in gene transcription using the linear noise approximation: An application to single cell data. *The Annals of Applied Statistics*, 7(4):1960–1982, 2013. (Cited on pages 95, 102, and 111.)

[21] C. Gardiner. *Handbook of Stochastic Methods: for Physics, Chemistry and the Natural Sciences (Springer Series in Synergetics)*. Springer, 3rd edition, 2004. (Cited on pages 31 and 33.)

[22] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis, Second Edition (Chapman & Hall/CRC Texts in Statistical Science)*. Chapman and Hall/CRC, 2 edition, 2003. Published: Hardcover. (Cited on pages 48, 99, and 102.)

[23] V. Giagos. *Inference for auto-regulatory genetic networks using diffusion process approximations*. PhD thesis, Lancaster University, 2010. (Cited on pages 61, 62, and 109.)

[24] C. S. Gillespie and A. Golightly. Diagnostics for assessing the linear noise and moment closure approximations. *Statistical Applications in Genetics and Molecular Biology*, 15(5):363–379, 2016. (Cited on page 44.)

[25] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977. (Cited on pages 18 and 39.)

[26] D. T. Gillespie. *Markov processes : an introduction for physical scientists*. Academic Press, Boston, San Diego, New York, 1992. (Cited on pages 29, 33, 73, and 130.)

[27] D. T. Gillespie. A rigorous derivation of the chemical master equation. *Physica A: Statistical Mechanics and its Applications*, 188, 1992. (Cited on pages 22, 35, 36, and 39.)

[28] D. T. Gillespie. Exact numerical simulation of the Ornstein-Uhlenbeck process and its integral. *Physical Review E*, 54:2084–2091, 1996. (Cited on page 71.)

[29] D. T. Gillespie. The chemical Langevin equation. *The Journal of Chemical Physics*, 113(1):297–306, 2000. (Cited on pages 51 and 52.)

[30] D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *Journal of Chemical Physics*, 115(4):1716–1733, 2001. (Cited on page 40.)

[31] D. T. Gillespie and E. Seitaridou. *Simple Brownian Diffusion: An Introduction to the Standard Theoretical Models*. 2012. (Cited on page 22.)

[32] M. Girolami and B. Calderhead. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):123–214, 2011. (Cited on page 112.)

[33] A. Gloter. Parameter estimation for a discrete sampling of an intergrated Ornstein-Uhlenbeck process. *Statistics*, 35(3):225–243, 2001. (Cited on page 63.)

[34] A. Golightly and C. Gillespie. Simulation of stochastic kinetic models. In *In-silico Systems Biology: A systems-based approach to understanding biological processes.*, pages 169–187. Humana Press, 2013. (Cited on page 80.)

[35] A. Golightly, D. Henderson, and C. Sherlock. Efficient particle MCMC for exact inference in stochastic biochemical network models through approximation of expensive likelihoods. *Statistics and Computing*, 25(5):1039–1055, 2015. (Cited on pages 84 and 88.)

[36] A. Golightly and D. J. Wilkinson. Bayesian Inference for Stochastic Kinetic Models Using a Diffusion Approximation. *Biometrics*, 61(3):781–788, 2005. (Cited on page 52.)

[37] A. Golightly and D. J. Wilkinson. Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo. *Interface Focus*, 2011. (Cited on page 52.)

[38] J. Goodman and J. Weare. Ensemble samplers with affine invariance. *Communications in Applied Mathematics and Computational Science*, 5(1):65–80, 2010. (Cited on page 109.)

[39] R. M. Goodwin. A growth cycle. *Socialism, capitalism and economic growth*, pages 54–58, 1967. (Cited on page 77.)

[40] C. V. Harper, B. Finkenstädt, D. J. Woodcock, S. Friedrichsen, S. Semprini, L. Ashall, D. G. Spiller, J. J. Mullins, D. A. Rand, J. R. E. Davis, and M. R. H. White. Dynamic analysis of stochastic transcription cycles. *PLoS biology*, 9(4), 2011. (Cited on pages 97 and 98.)

[41] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970. (Cited on page 46.)

[42] K. L. Hey, H. Momiji, K. Featherstone, J. R. E. Davis, M. R. H. White, D. A. Rand, and B. Finkenstädt. A stochastic transcriptional switch model for single cell imaging data. *Biostatistics*, 16(4):655–669, 2015. (Cited on page 111.)

[43] Y. Ho and R. Lee. A Bayesian approach to problems in stochastic estimation and control. *Automatic Control, IEEE Transactions on*, 9(4):333–339, 1964. (Cited on pages 53 and 54.)

[44] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, 1970. (Cited on pages 53, 54, and 57.)

[45] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, 1960. (Cited on pages 53 and 56.)

[46] P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer, New York, corrected edition, 2011. (Cited on page 31.)

[47] M. Komorowski, B. Finkenstädt, Claire V. Harper, and David A. Rand. Bayesian inference of biochemical kinetic parameters using the linear noise approximation. *BMC Bioinformatics*, 10(1):1–10, 2009. (Cited on pages 18, 21, 40, 52, 61, 73, 84, 87, 95, 100, 102, 108, 110, and 111.)

[48] W. Krauth. *Statistical Mechanics: Algorithms and Computations (Oxford Master Series in Physics)*. Oxford University Press, 2006. Published: Paperback. (Cited on page 46.)

[49] N. D. Lawrence, G. Sanguinetti, M. Girolami, and M. Rattray. *Learning and Inference in Computational Systems Biology*. The MIT Press, 1st edition, 2009. (Cited on pages 17 and 44.)

[50] N. D. Lawrence, G. Sanguinetti, and M. Rattray. Modelling transcriptional regulation using Gaussian Processes. In *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, pages 785–792, 2006. (Cited on page 17.)

[51] X. Liu and M. Niranjan. State and parameter estimation of the heat shock response system using Kalman and particle filters. *Bioinformatics*, 28(11):1501–1507, 2012. (Cited on page 17.)

[52] A. J. Lotka. Undamped oscillations derived from the law of mass action. *Journal of the American Chemical Society*, 42(8):1595–1599, 1920. (Cited on page 77.)

[53] A. J. Lotka. *Elements of physical biology,*. Williams & Wilkins company, 1925. (Cited on page 77.)

[54] I. S. Mbalawata, S. Särkkä, and H. Haario. Parameter estimation in stochastic differential equations with Markov chain Monte Carlo and non-linear Kalman filtering. *Computational Statistics*, 28(3):1195–1223, 2013. (Cited on page 74.)

[55] D. A. McQuarrie. Stochastic Approach to Chemical Kinetics. *Journal of Applied Probability*, 4(3):413–478, 1967. (Cited on page 40.)

[56] C. G. Moles, P. Mendes, and J. R. Banga. Parameter estimation in biochemical pathways: a comparison of global optimization methods. *Genome research*, 13(11):2467–2474, 2003. (Cited on page 17.)

[57] J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *The Computer Journal*, 7(4):308–313, 1965. (Cited on page 63.)

[58] D. E. Nelson, A. E. C. Ihekwaba, M. Elliott, J. R. Johnson, C. A. Gibney, B. E. Foreman, G. Nelson, V. See, C. A. Horton, D. G. Spiller, S. W. Edwards, H. P. McDowell, J. F. Unitt, E. Sullivan, R. Grimley, N. Benson, D. Broomhead, D. B. Kell, and M. R. H. White. Oscillations in NF-κB

Signaling Control the Dynamics of Gene Expression. *Science*, 306(5696):704–708, 2004. (Cited on page 17.)

[59] A. Ohagan. Bayesian statistics: principles and benefits. In M. A. J. S. van Boekel, A. Stein, and A. H. C. van Bruggen, editors, *Bayesian Statistics and Quality Modelling in the Agro-Food Production Chain*, pages 31–45. Oxford University Press, Springer, 2004. (Cited on page 45.)

[60] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications (Universitext)*. Springer, 6th edition, 2014. Published: Paperback. (Cited on pages 30, 31, and 124.)

[61] M. Opper and G. Sanguinetti. Variational inference for Markov jump processes. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, pages 1105–1112. 2008. (Cited on page 51.)

[62] P. Paszek, S. Ryan, L. Ashall, K. Sillitoe, C. V. Harper, D. G. Spiller, D. A. Rand, and M. R. H. White. Population robustness arising from cellular heterogeneity. *Proceedings of the National Academy of Sciences*, 107(25):11644–11649, 2010. (Cited on page 111.)

[63] N. E. Phillips, C. S. Manning, T. Pettini, V. Biga, E. Marinopoulou, P. Stanley, J. Boyd, J. Bagnall, P. Paszek, D. G. Spiller, M. R. H. White, M. Goodfellow, T. Galla, M. Rattray, and N. Papalopulu. Stochasticity in the miR-9/Hes1 oscillatory network can account for clonal heterogeneity in the timing of differentiation. *eLife*, 5:e16118, 2016. (Cited on page 112.)

[64] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. (Cited on pages 26 and 27.)

[65] S. Reinker, R. M. Altman, and J. Timmer. Parameter estimation in stochastic biochemical reactions. *Systems Biology*, 153(4):168–178, 2006. (Cited on page 51.)

[66] G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *Annals of Applied Probability*, 7:110–120, 1997. (Cited on page 47.)

[67] G. O. Roberts and J. S. Rosenthal. Optimal Scaling for Various Metropolis-Hastings Algorithms. *Statistical Science*, 16(4):351–367, 2001. (Cited on page 47.)

[68] G. O. Roberts and J. S. Rosenthal. Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of Applied Probability*, 44(2):458–475, 2007. (Cited on page 48.)

[69] G. O. Roberts and J. S. Rosenthal. Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18(2):349–367, 2009. (Cited on page 48.)

[70] S. M. Ross. *Stochastic Processes (Wiley Series in Probability and Statistics)*. Wiley, 2 edition, 1995. Published: Hardcover. (Cited on pages 24, 25, 26, 27, 28, and 29.)

[71] H. Salis and Y. Kaznessis. Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *The Journal of Chemical Physics*, 122(5), 2005. (Cited on page 44.)

[72] S. Särkkä. *Recursive Bayesian Inference on Stochastic Differential Equations*. PhD thesis, Helsinki University of Technology, 2006. (Cited on pages 53, 54, and 57.)

[73] S. Särkkä. *Bayesian filtering and smoothing*. Cambridge University Press, 2013. (Cited on page 53.)

[74] C. Schluter and M. Trede. Estimating Continuous-Time Income Models. CQE Working Paper 1811, Center for Quantitative Economics (CQE), University of Muenster, 2011. (Cited on page 112.)

[75] C. Sherlock, P. Fearnhead, and G. O. Roberts. The Random Walk Metropolis: Linking Theory and Practice Through a Case Study. *Statistical Science*, 25(2):172–190, 2010. (Cited on page 48.)

[76] C. Sherlock, A. Golightly, and C. Gillespie. Bayesian inference for hybrid discrete-continuous stochastic kinetic models. *Inverse Problems*, 30(11), 2014. (Cited on page 44.)

[77] H. Sorensen. Parametric Inference for Diffusion Processes Observed at Discrete Points in Time: a Survey. *International Statistical Review*, 72(3):337–354, 2004. (Cited on page 52.)

[78] D. G. Spiller, C. D. Wood, D. A. Rand, and M. R. H. White. Measurement of single-cell dynamics. *Nature*, 465(7299):736–745, 2010. (Cited on pages 7, 17, 18, and 50.)

[79] V. Stathopoulos and M. A. Girolami. Markov chain Monte Carlo inference for Markov jump processes via the linear noise approximation. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 371(1984), 2012. (Cited on pages 52, 84, and 112.)

[80] P. Thomas, H. Matuschek, and R. Grima. How reliable is the linear noise approximation of gene regulatory networks? *BMC Genomics*, 14(4):1–15, 2013. (Cited on pages 44 and 52.)

[81] 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. (Cited on page 51.)

[82] D. A. Turner, P. Paszek, D. J. Woodcock, D. E. Nelson, C. A. Horton, Y. Wang, D. G. Spiller, D. A. Rand, M. R. H. White, and C. V. Harper. Physiological levels of TNFα stimulation induce stochastic dynamics of NF-κB responses in single living cells. *Journal of Cell Science*, 123(16):2834–2843, 2010. (Cited on page 17.)

[83] M. Ullah and O. Wolkenhauer. Stochastic approaches in systems biology. *Wiley interdisciplinary reviews. Systems biology and medicine*, 2(4):385–397, 2010. (Cited on page 33.)

[84] N. van Kampen. *Stochastic Processes in Physics and Chemistry*. Elsevier, 3rd edition, 2007. (Cited on pages 33, 34, 40, 43, 44, 52, and 71.)

[85] V. Volterra. Fluctuations in the abundance of a species considered mathematically. *Nature*, 118:558–560, 1926. (Cited on page 77.)

[86] E. W. J. Wallace, D. T. Gillespie, K. R. Sanft, and L. R. Petzold. Linear noise approximation is valid over limited times for any chemical system that is

sufficiently large. *IET Systems Biology*, 6(4):102+, 2012. (Cited on pages 44 and 52.)

[87] D. K. Welsh and S. A. Kay. Bioluminescence imaging in living organisms. *Current Opinion in Biotechnology*, 16(1):73 – 78, 2005. Analytical biotechnology. (Cited on page 20.)

[88] N. Wiener. Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications. 1950. (Cited on page 53.)

[89] D. J. Wilkinson. *Stochastic Modelling for Systems Biology, Second Edition*. Chapman & Hall/CRC Mathematical and Computational Biology. Taylor & Francis, 2011. (Cited on pages 18, 20, 33, 34, 35, 36, 39, and 80.)

Appendix A

Supporting materials

A.1 Ito's formula

Let,

$$dX_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t, \tag{1}$$

be a d-dimensional process and define $Y_t = f(t, X_t)$ for a twice differentiable function $f(\cdot)$. Then Y_t is given by [60]:

$$dY_k = \frac{\partial f_k}{\partial t}(t, X_t)dt + \sum_i \frac{\partial f_k}{\partial x_i}(t, X_t)dX_i + \frac{1}{2}\sum_{i,j} \frac{\partial^2 f_k}{\partial x_i x_j}(t, X_t)dX_i dX_j, \tag{2}$$

A.2 Moments of a linear SDE in the narrow sense

$$\begin{aligned} dX_t =& (a_1(t)X_t + a_2(t))dt + b_2(t)dW_t, \\ \Leftrightarrow& \\ X_{t+dt} =& X_t + (a_1(t)X_t + a_2(t))dt + b_2(t)dW_t \end{aligned} \tag{3}$$

Averaging equation (35) and noting that $\mathbb{E}[dW_t] = 0$ results in:

$$\begin{aligned} \mathbb{E}[X_{t+dt}] &= \mathbb{E}[X_t] + a_1(t)\mathbb{E}[X_t]dt + a_2 dt \\ \mathbb{E}[X_{t+dt}] - \mathbb{E}[X_t] &= a_1(t)\mathbb{E}[X_t]dt + a_2 dt \\ \frac{d\mathbb{E}[X_t]}{dt} &= \frac{dm(t)}{dt} = a_1(t)m(t) + a_2 \end{aligned} \tag{4}$$

For the second moment $P(t) = \mathbb{E}[X_t X_t^T]$ we get from (35):

$$
\begin{aligned}
{X_{t+dt}X_{t+dt}}^T &= (X_t + (a_1(t)X_t + a_2(t))dt \\
&\quad + b_2(t)dW_t)(X_t + (a_1(t)X_t + a_2(t))dt + b_2(t)dW_t)^T \\
{X_{t+dt}X_{t+dt}}^T &= X_tX_t^T + X_t[(a_1(t)X_t + a_2(t))dt + b_2(t)dW_t]^T + \\
&\quad + (a_1(t)X_t + a_2(t))dt + b_2(t)dW_t)X_t^T + \\
&\quad + (a_1(t)X_t + a_2(t))dt + b_2(t)dW_t)(a_1(t)X_t + a_2(t))dt + b_2(t)dW_t)^T \\
{X_{t+dt}X_{t+dt}}^T &= X_tX_t^T + X_tX_t^Ta_1(t)^Tdt + X_ta_2(t)^Tdt + X_tb_2(t)^TdW_t^T + \\
&\quad + a_1(t)X_tX_t^Tdt + a_2(t)X_t^Tdt + b_2(t)dW_tX_t^T + b_2b_2^Tdt
\end{aligned}
\tag{5}
$$

By averaging we get:

$$
\begin{aligned}
\mathbb{E}[{X_{t+dt}X_{t+dt}}^T] &= \\
&\quad \mathbb{E}[X_tX_t^T] + \mathbb{E}[X_tX_t^T]a_1(t)^Tdt + a_1(t)\mathbb{E}[X_tX_t^T]^Tdt \\
&\quad + \mathbb{E}[X_t]a_2(t)^Tdt + a_2(t)\mathbb{E}[X_t]^Tdt \\
\mathbb{E}[{X_{t+dt}X_{t+dt}}^T] - \mathbb{E}[X_tX_t^T] &= \\
&\quad Pa_1(t)^Tdt + a_1(t)P^Tdt + m(t)a_2^Tdt + a_2m(t)^Tdt + b_2b_2^Tdt \\
\frac{dP(t)}{dt} &= Pa_1(t)^T + a_1(t)P^T + m(t)a_2^T + a_2m(t)^T + b_2b_2^T
\end{aligned}
\tag{6}
$$

where terms have been canceled out according to the three rules from (3).

A.3 Gaussian variables

The following Lemmas for distributions of Gaussian variables hold. Proofs of these lemmas can be found in the literature [6].

Lemma1 Partitioned Gaussians:

Let x_a and x_b be jointly Gaussian random vectors:

$$
\begin{bmatrix} x_a \\ x_b \end{bmatrix} \sim N(\begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \begin{bmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{bmatrix}) \tag{7}
$$

Then the marginal and conditional distributions of x_a (equivalently for x_b) are

respectively:

$$x_a \sim N(\mu_a, \Sigma_{aa}) \tag{8a}$$
$$x_b \sim N(\mu_b, \Sigma_{bb}) \tag{8b}$$
$$x_a|x_b \sim N(\mu_a + \Sigma_{ab}\Sigma_{bb}^{-1}(x_b - \mu_b), \Sigma_{aa} - \Sigma_{ab}\Sigma_{bb}^{-1}\Sigma_{ba}) \tag{8c}$$
$$x_b|x_a \sim N(\mu_b + \Sigma_{ba}\Sigma_{aa}^{-1}(x_b - \mu_a), \Sigma_{bb} - \Sigma_{ba}\Sigma_{aa}^{-1}\Sigma_{ab}) \tag{8d}$$

Lemma2 Joint and Marginal Gaussians:

Assume that the distribution of x and the conditional distribution of x given y are normally distributed:

$$x \sim N(\mu, S) \tag{9}$$

$$y|x \sim N(Ax + b, R) \tag{10}$$

Then the following joint and marginal densities can be computed:

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N(\begin{bmatrix} \mu \\ A\mu + b \end{bmatrix}, \begin{bmatrix} S & SA^T \\ AS & ASA^T + R \end{bmatrix}) \tag{11}$$

$$y \sim N(A\mu + b, ASA^T + R). \tag{12}$$

A.4 Terms of joint distribution

For the covariance terms of X_t and y_t and keeping in mind that X_t and ϵ_t are independent we have that :

$$\begin{aligned} cov(X_t, y_t) &= \mathbb{E}[(X_t - \mu_t^1)(y_t - P_t\mu_t^2)^T] \\ &= \mathbb{E}[(X_t - \mu_t^1)(P_tH_t + \epsilon_t - P_t\mu_t^2)^T] \\ &= \mathbb{E}[(X_t - \mu_t^1)(P_t(H_t - \mu_t^2))^T] + \mathbb{E}[(X_t - \mu_t^1)(\epsilon_t - 0)^T] \\ &= \mathbb{E}[(X_t - \mu_t^1)((H_t - \mu_t^2))]P^T \\ &= cov(X_t, H_t)P^T \end{aligned} \tag{13}$$

Making use of Lemma2 it is straightforward to calculate the rest of the terms.

A.5 Variance and covariance for integrated LNA with the Non-Restarting method

For the Non-Restarting case the variance will be given by $S_t = Var[Q_t] = G_t - \mathbb{E}[Q_t]\mathbb{E}[Q_t]^T$, by setting $\mathbb{E}[Q_t] = mq_t$ we get the following:

$$\begin{aligned} dS_t &= dG_t - mq_t d(mq_t)^T - d(mq_t)mq_t^T \\ dS_t &= \mathbb{E}[Q_t\xi_t^T]dt + \mathbb{E}[\xi_t Q_t^T]dt - mq_t m_t^T dt - m_t mq_t^T dt \\ \frac{dS_t}{dt} &= \mathbb{E}[Q_t\xi_t^T] + \mathbb{E}[\xi_t Q_t^T] - mq_t^T m_t - m_t mq_t^T \end{aligned} \tag{14}$$

The cross covariance is calculated by $K_t = Var[Q_t\xi_t] = \mathbb{E}[Q_t\xi_t^T] - mq_t m_t^T$ where $\mathbb{E}[Q_t] = mq_t$ and $\mathbb{E}[\xi_t] = m_t$:

$$\begin{aligned} dK_t &= d\mathbb{E}[Q_t\xi_t^T] - mq_t d(m_t)^T - d(mq_t)m_t^T \\ dK_t &= \mathbb{E}[Q_t\xi_t^T]A(t)^T dt + \mathbb{E}[\xi_t\xi_t^T]dt - mq_t m_t^T A^T dt - m_t m_t^T dt \\ \frac{dK_t}{dt} &= \mathbb{E}[Q_t\xi_t^T]A(t)^T + \mathbb{E}[\xi_t\xi_t^T] - mq_t m_t^T A^T - m_t m_t^T \end{aligned} \tag{15}$$

A.6 Analytical solutions for the OU process and its integral

Given an OU process of the following form:

$$dX_t = -\alpha X_t dt + \sigma dW_t \tag{16}$$

We can derive its solution according to the general theory for linear SDEs in the narrow sense. Since the solution is a Gaussian process we will only need to define its mean and variance. All the ODEs in this case are first order linear ODEs with constant coefficients, so using for example an integrating factor we can derive the following solution for an ODE of the form $\frac{dx}{dt} + ax = g(t), \quad x(t=0) = x_0$:

$$x_t = e^{-a(t-t_0)}x_0 + \int_{t_0}^{t} e^{-a(t-\tau)}g(\tau)d\tau. \tag{17}$$

For the mean we have:

$$\begin{aligned} &\frac{dm_t}{dt} = -\alpha m_t, \quad m_{t_0} = m_0 \Rightarrow \\ &m_t = m_0 e^{-a(t-t_0)} \end{aligned} \tag{18}$$

For the variance we have the following:

$$\begin{aligned} &\frac{dV_t}{dt} = -2\alpha V_t + \sigma^2, \quad V_{t_0} = V_0 \Rightarrow \\ &V_t = e^{-2\alpha(t-t_0)}V_0 + \int_{t_0}^{t} e^{-2\alpha(t-\tau)}\sigma^2 d\tau\,, \\ &V_t = e^{-2\alpha(t-t_0)}V_0 + \frac{\sigma^2}{2\alpha}(1 - e^{-2\alpha(t-t_0)}) \end{aligned} \tag{19}$$

For the solution of the integrated OU process $dY_t/dt = X_t$ we need to calculate its mean, covariance and variance. The initial conditions for these ODEs will be set to 0, since the integated process starts from 0. The solutions are making use of the results A, B, C from part (A.6.1).

First we find the mean:

$$\begin{aligned} &\frac{dmy_t}{dt} = m_t = m_0 e^{-\alpha(t-t_0)}, \quad my(t_0) = 0 \Rightarrow \\ &my_t = \int_{t_0}^{t} m_0 e^{-\alpha(\tau-t_0)} d\tau \overset{A}{\Rightarrow} \\ &my_t = \frac{m_0}{\alpha}(1 - e^{-\alpha(t-t_0)}) \end{aligned} \tag{20}$$

For the covariance we have:

$$\begin{aligned} &\frac{d\mathbb{E}[X_t Y_t]}{dt} = -\alpha\mathbb{E}[X_t Y_t] + \mathbb{E}[{X_t}^2], \quad \mathbb{E}[X_0 Y_0] = 0 \Rightarrow \\ &\mathbb{E}[X_t Y_t] = \int_{t_0}^{t} \mathbb{E}[{X_t}^2] e^{-\alpha(t-\tau)} d\tau \overset{\mathbb{E}[{X_t}^2] = V_t + {m_t}^2}{\Rightarrow} \\ &\mathbb{E}[X_t Y_t] = \int_{t_0}^{t} ((m_0^2 - \frac{\sigma^2}{2\alpha} + V_0)e^{-2\alpha(\tau-t_0)} + \frac{\sigma^2}{2\alpha})e^{-\alpha(t-\tau)} d\tau \overset{A,C}{\Rightarrow} \\ &\mathbb{E}[X_t Y_t] = \\ &= \frac{\sigma^2}{2\alpha^2}(1 - e^{-\alpha(t-t_0)}) + \frac{1}{\alpha}(m_0^2 - \frac{\sigma^2}{2\alpha} + V_0)(e^{-\alpha(t-t_0)} - e^{-2\alpha(t-t_0)}) \end{aligned} \tag{21}$$

Now the covariance can be calculated from:

$$\begin{aligned} &Cov(X_t, Y_t) = \mathbb{E}[X_t Y_t] - m_t m y_t \Rightarrow \\ &Cov(X_t, Y_t) = \\ &= \frac{\sigma^2}{2\alpha^2} + (-\frac{\sigma^2}{\alpha^2} + \frac{V_0}{\alpha})e^{-\alpha(t-t_0)} + (\frac{\sigma^2}{2\alpha^2} - \frac{V_0}{\alpha})e^{-2\alpha(t-t_0)} \end{aligned} \tag{22}$$

For the variance we need to calculate:

$$\begin{aligned} &\frac{d\mathbb{E}[{Y_t}^2]}{dt} = 2\mathbb{E}[X_t Y_t], \quad \mathbb{E}[{Y_0}^2] = 0 \Rightarrow \\ &\mathbb{E}[{Y_t}^2] = 2\int_{t_0}^{t} \mathbb{E}[X_\tau Y_\tau] d\tau \overset{B}{\Rightarrow} \\ &\mathbb{E}[{Y_t}^2] = \frac{{m_0}^2}{\alpha^2}(1 - 2e^{-\alpha(t-t_0)} + e^{-2\alpha(t-t_0)}) \end{aligned} \tag{23}$$

Now we can derive the variance:

$$\begin{aligned} Vy_t =& \mathbb{E}[{Y_t}^2] - m{y_t}^2 \Rightarrow \\ Vy_t =& \frac{\sigma^2}{\alpha^2}(t - t_0) + (\frac{\sigma^2}{2\alpha^3} - \frac{V_0}{\alpha^2})(1 - e^{-2\alpha(t-t_0)}) + \\ &+ 2(-\frac{\sigma^2}{\alpha^3} + \frac{V_0}{\alpha^2})(1 - e^{-\alpha(t-t_0)}) \end{aligned} \tag{24}$$

A.6.1 Frequently used integrals for part (A.6)

$$A = \int_{t_0}^{t} e^{-\alpha(\tau - t_0)} d\tau \tag{25a}$$

$$= \tfrac{1}{\alpha}(1 - e^{-\alpha(t-t_0)}) \tag{25b}$$

$$B = \int_{t_0}^{t} e^{-2\alpha(\tau - t_0)} d\tau \tag{26a}$$

$$= \tfrac{1}{2\alpha}(1 - e^{-2\alpha(t-t_0)}) \tag{26b}$$

$$C = \int_{t_0}^{t} e^{-\alpha(t-\tau)} e^{-2\alpha(\tau-t_0)} d\tau \tag{27a}$$

$$= \frac{1}{\alpha}(e^{-\alpha(t-t_0)} - e^{-2\alpha(t-t_0)}) \tag{27b}$$

A.7 Exact Updating formula of OU process

The OU process $dX_t = -\alpha X_t dt + \sigma dW_t$ admits an exact update formula given by [26]:

$$X_{t+dt} = X_t e^{-\alpha dt} + \sqrt{\sigma^2 \frac{1}{2\sigma} e^{-2\alpha dt}} N(0,1), \tag{28}$$

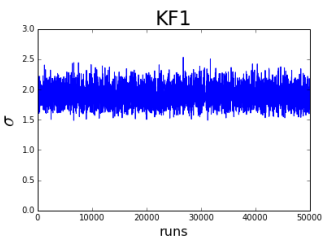
A.8 Priors for SGE model

θ	Prior
γ_R	Gamma(19.36,1/44)
k_P	Exp(10)
γ_R	Gamma(27.04,1/52)
k	Exp(1)
b_0	Exp(10)
b_1	Exp(1)
b_2	Exp(10)
b_3	Exp(10)

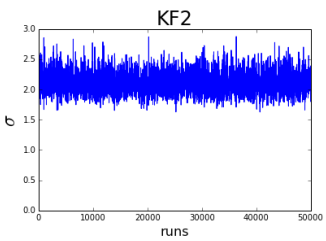
Table A.1: Table of priors used with the SGE model.

Appendix B

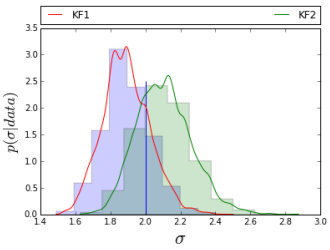
Additional plots



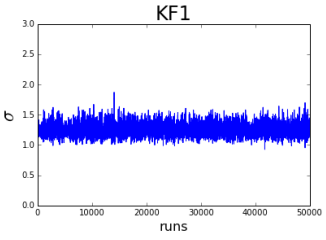
(a) MH trace of σ from KF1 for $\Delta = 0.1$.



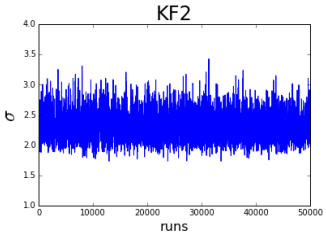
(b) MH trace of σ from KF2 for $\Delta = 0.1$.



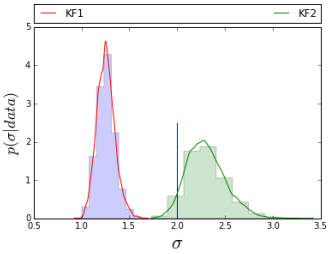
(c) Posterior of σ using KF1(blue) and KF2(green)



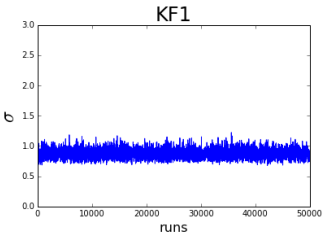
(d) MH trace of σ from KF1 for $\Delta = 0.5$.



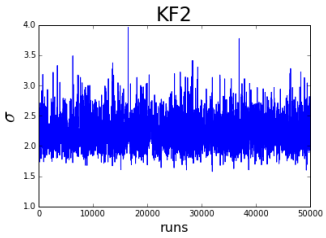
(e) MH trace of σ from KF2 for $\Delta = 0.5$.



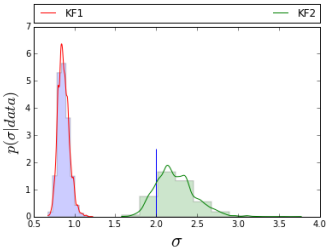
(f) Posterior of σ using KF1(blue) and KF2(green)



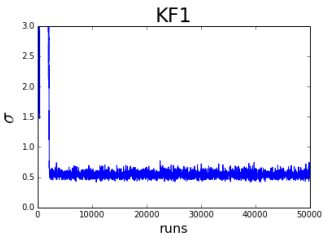
(g) MH trace of σ from KF1 for $\Delta = 1.0$.



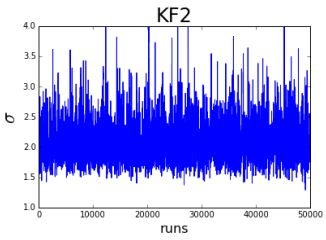
(h) MH trace of σ from KF2 for $\Delta = 1.0$.



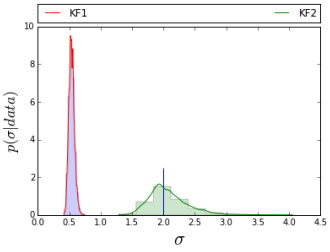
(i) Posterior of σ using KF1(blue) and KF2(green)



(j) MH trace of σ from KF1 for $\Delta = 2.0$.



(k) MH trace of σ from KF2 for $\Delta = 2.0$.



(l) Posterior of σ using KF1(blue) and KF2(green)

Figure B.1: MCMC traces and histograms of the posterior of σ using a MH for both KF1 and KF2. Ground truth for $\alpha = 2$ and is indicated by the vertical blue line on the histogram plots.

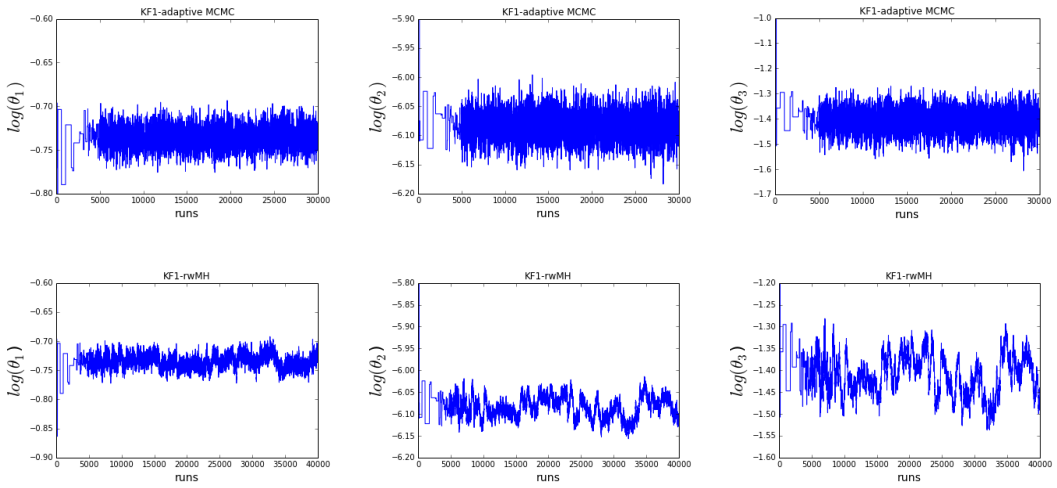
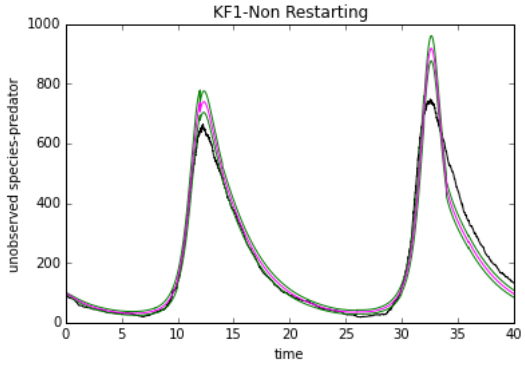
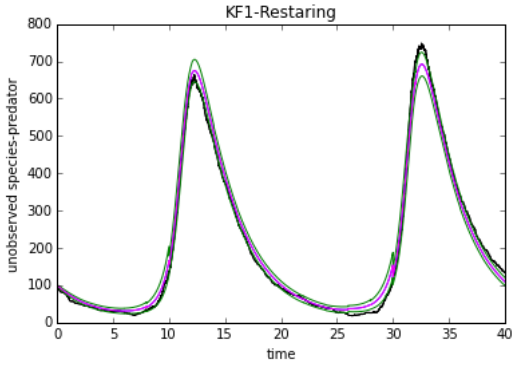


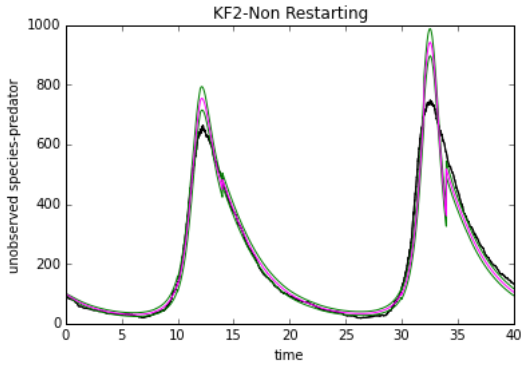
Figure B.2: Traceplots of the Lotka-Volterra parameters using KF1 with an adaptive MCMC (first row) and a random walk MH (second row).



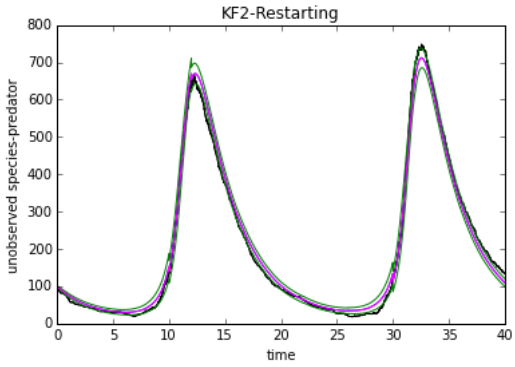
(a) Filtering results of the Non-Restarting KF1.



(b) Filtering results of the Restarting KF1.

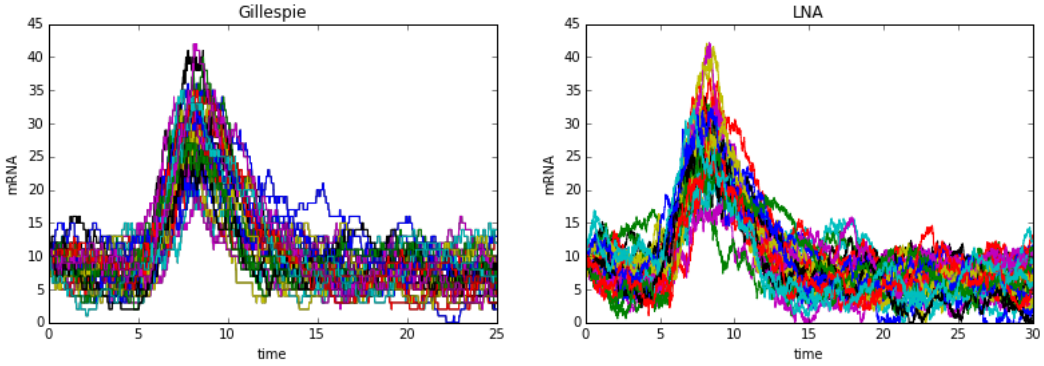


(c) Filtering results of the Non-Restarting KF2.



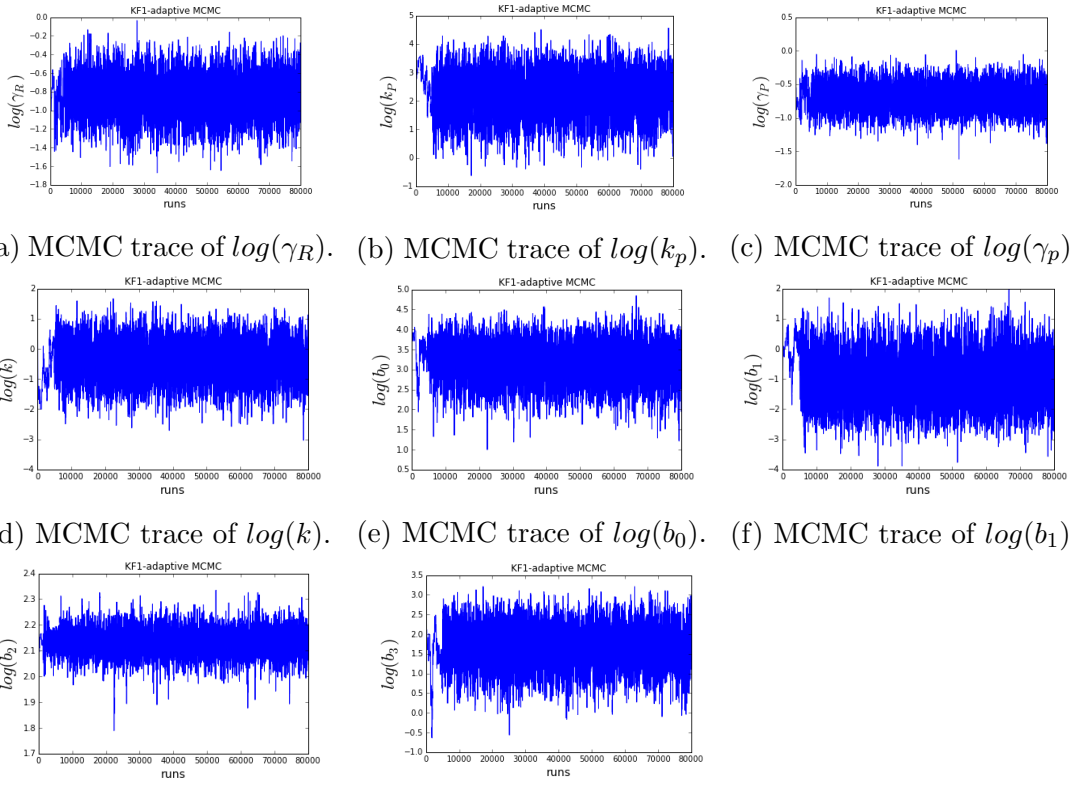
(d) Filtering results of the Restarting KF2.

Figure B.3: Filtering plots for the predator population with (KF2) and without (KF1) aggregate data. The Non-Restarting method is shown on the first column and the Restarting on the second column. The predator population is unobserved. Black lines represent the actual process, while purple lines represent the mean estimate and green 1 s.d. .



(a) Gillespie (b) LNA

Figure B.4: Simulated trajectories of mRNA using the Gillespie algorithm (a) and the LNA (b).

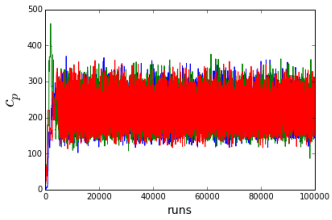


(a) MCMC trace of $log(\gamma_R)$. (b) MCMC trace of $log(k_p)$. (c) MCMC trace of $log(\gamma_p)$.

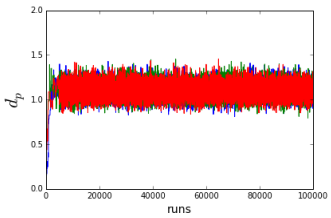
(d) MCMC trace of $log(k)$. (e) MCMC trace of $log(b_0)$. (f) MCMC trace of $log(b_1)$.

(g) MCMC trace of $log(b_2)$. (h) MCMC trace of $log(b_3)$.

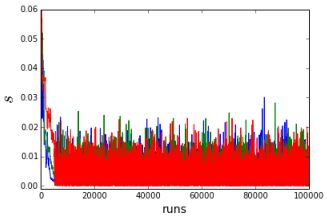
Figure B.5: Adaptive MCMC traces for the log parameters of the SGE model using KF1 with aggregated data.



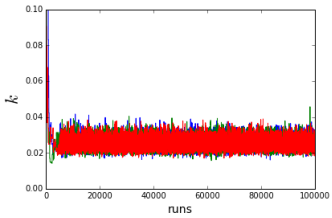
(a) MCMC trace of c_P.



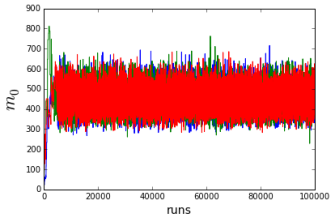
(b) MCMC trace of d_P.



(c) MCMC trace of s.



(d) MCMC trace of k.



(e) MCMC trace of m_0.

Figure B.6: Adaptive MCMC traces of the Translation inhibition model parameters using KF1 with real data.