

Inference for auto-regulatory genetic networks using diffusion process approximations

Vasileios Giagos

Submitted for the degree of Doctor of Philosophy
at Lancaster University,
September 2010.

Inference for auto-regulatory genetic networks using diffusion process approximations

Vasileios Giagos

Submitted for the degree of Doctor of Philosophy
at Lancaster University, September 2010.

Abstract

The scope of this thesis is to propose new inferential tools, based on diffusion process approximations, for the study of the kinetic parameters in auto-regulatory networks.

In the first part of this thesis, we study the applicability of the EA methodology to Stochastic Differential Equations (SDEs) which approximate biological systems. In principle EA can be applied to any scalar-valued SDE as long as a transformation (known as Lamperti transform) exists that sets the (new) infinitesimal variance to unity. We explore the numerical limitations of this requirement by considering a biological system that can be expressed as a scalar non-linear SDE. Next, we consider the multidimensional extension of this transformation and we show, with a counterexample, that EA can be applied to a class of SDEs which is wider than the class of reducible diffusions.

In the second part of this thesis, we proposed a reparametrization of the kinetic constants that leads to an approximation known as the Linear Noise approximation (LNA). We prove that LNA converges to a linear SDE, as the size of the biological system increases. Since the LNA is a linear SDE, it has a known transition density with parameters given as the solutions of a system of Ordinary Differential Equations (ODEs) which are usually obtained numerically. Furthermore, we compare the LNA's simulation performance to the performance of other (approximate and exact) methods under different modelling scenarios and we relate the performance of the approximate methods to the system size. In addition, we consider LNA as an inferential tool and we use two methods, the Restarting (RE), which we propose, and the Non-Restarting (NR) method, proposed by Komorowski et al. (2009) to derive the LNA's likelihood. The two methods differ on the initial conditions that they pose in order to solve the underlying ODEs. We compare the performance of the two methods by considering data generated under different scenarios. Finally, we discuss the `lnar`, a package for the R

statistical environment, that we developed to implement the LNA methodology.

Acknowledgements

I would like to thank my supervisors, Paul Fearnhead and Chris Sherlock for their support and their patience during my studies. I am grateful to Paul for helping me to identify the research questions of interest and for finding for me additional sources of funding after the end of my studentship. I am indebted to Chris for his endeavour to keep me from sidetracking and also for sharing his research intuition during our meetings.

I am also grateful to Ernst Wit who supervised me during the first year of my studies. Ernst helped me to set the research questions of the first part of this thesis and by the end of the first year we had some preliminary results. I would also like to thank Madhuchhanda Bhattacharjee for giving me the opportunity to start my studies.

I was lucky during my undergraduate studies in Samos because I met Ioannis Ntzoufras and Athanasios Katsis. Ioannis motivated me to continue my studies and Athanasios supported me in my first steps as a researcher.

It was a pleasure studying in such a friendly environment and I would like to thank everyone in the department for that. In particular, I would like to thank Tom Fanshawe, Matt Sperrin, Alex Rodrigues, Ben Taylor, Rebecca Killick, Dennis Prangle for their advice, stimulating conversations and their enthusiasm.

Of course, I would like to thank my family and my friends for their encouragement and in particular Dimitra Petrakaki.

Finally, I would like to thank the Faculty of Sciences and the Department of Mathematics and Statistics for the financial support during my PhD studies.

Declaration

This thesis is my own work and has not been submitted in substantially the same form for the award of a higher degree elsewhere.

Vasileios Giagos

Contents

1	Introduction	4
1.1	Models	4
1.2	Thesis Overview	6
2	Theory Background	8
2.1	Diffusions	8
2.1.1	Brownian Motion and SDEs	9
2.1.2	Itô's Lemma	10
2.1.3	Solution of SDEs	12
2.1.4	Linear SDEs	15
2.1.4.1	Fundamental Matrix Solution	16
2.1.4.2	Normal Transition Density	18
2.1.5	Brownian Bridge	19
2.1.6	Convergence to SDEs	19
2.2	ODE Integration	21
2.3	Numerical Optimisation	23
2.3.1	Nelder–Mead	23
2.3.2	BFGS	24
3	Models	25
3.1	Stochastic Motivation	27
3.2	Interactions as reactions	27
3.2.1	Simple Reactions	28
3.2.1.1	Reactions of order zero	29
3.2.1.2	First order reactions	29

3.2.1.3	Second order reactions	29
3.2.1.4	Higher order reactions	30
3.2.2	Gillespie algorithm	30
3.3	Diffusion approximation	31
3.3.1	Case studies	32
3.3.1.1	Lotka–Volterra Example	33
3.3.1.2	Transcription Example	34
4	Exact Algorithm	37
4.1	Introduction	37
4.2	Rejection Sampling	38
4.3	Simulation using the EA	40
4.3.1	Extensions	42
4.4	Transition densities estimator	43
4.4.1	The Acceptance Method	44
4.4.1.1	Simultaneous Acceptance Method	44
4.4.2	Poisson Estimator	46
4.5	The univariate case	47
4.6	The multivariate case	49
4.6.1	Reducibility	52
4.6.1.1	Reducibility Counterexample	52
4.7	Discussion	54
5	Scaled Rates Approximation	55
5.1	Formulation	55
5.2	Ordinary Differential Equations	57
5.3	SDEs	58
5.4	Convergence	59
5.4.1	Limiting Process	60
5.4.2	Martingale Problem	62
5.5	Transition Density	63
5.6	Case Studies	64
5.6.1	Lotka–Volterra	66

5.6.2	Transcription Example	70
5.6.3	Discussion of the Examples	75
5.7	Discussion	77
6	Inference for Auto-regulatory networks	79
6.1	Methods Overview	79
6.2	Likelihood arising from the Linear Diffusion Approximation	81
6.2.1	Restarting and Non-Restarting	82
6.2.1.1	Stiff ODEs	84
6.3	Case Studies	84
6.3.1	Lotka-Volterra	85
6.3.2	Transcription Network	88
6.3.2.1	Three Datasets	91
6.4	Extending LNA	94
6.4.1	Kalman Filter Recursions	96
6.4.2	Lotka Volterra with partial observations	97
6.5	Discussion and future work	99
7	Implementation	101
7.1	Model Specification	102
7.1.1	The function <code>parsemod</code>	102
7.1.1.1	Details	103
7.1.2	The function <code>compmo</code> d	104
7.1.3	Examples	104
7.1.3.1	Lotka-Volterra	104
7.1.3.2	Transcription Model	105
7.2	Model Usage	106
7.2.1	Functions	106
7.2.1.1	The function <code>lnalik</code>	106
7.2.1.2	The function <code>calcdens</code>	107
7.2.1.3	The function <code>optmo</code> d	109
7.2.2	Examples	110
7.2.2.1	Lotka-Volterra Model (Continued)	110

7.2.2.2	Transcription Model (Continued)	111
7.3	Discussion	112
8	Conclusion	114

List of Figures

2.1	Illustration of the Euler–Maruyama method. The simulation interval ($t = 20s$) is partitioned into $5s$ intervals. At each interval the transition density is approximated with a Normal distribution (<i>illustrated vertically</i>) and a sample is drawn, which in turn becomes the next initial point.	11
3.1	Graph representing Lotka Volterra (Prey-Predator) model.	33
3.2	Graph representing the example prokaryotic auto regulation biochemical network. Edges with solid lines represent stoichiometric coefficients of ones, dotted lines coefficients of twos.	34
4.1	The drift term of the transformed dimerization reaction process plotted for parameter values $(c_1, c_2, k) = (0.1, 0.1, 500), (0.1, 0.6, 500), (0.6, 0.1, 500), (0.6, 0.6, 500)$. 49	
4.2	Plot of $\alpha \{Y_t; \mathbf{c}\}^2 + \alpha \{Y_t; \mathbf{c}\}'$ with $(c_1, c_2) = (0.085, 10^{-6})$ illustrating numerical stability issues.	50
5.1	Dependence graph between the variables (in circles) and parameters $\boldsymbol{\vartheta}$. The deterministic processes, which are given as solutions of ODEs, are positioned inside the “ODEs” frame.	63
5.2	Comparing the Prey empirical transition densities – p.m.f. of the Gillespie Method (<i>solid black</i>) the Chemical Langevin Equation (<i>dashed black</i>) and the Limiting Diffusion approximation (<i>solid grey</i>) at four time points ($t = 0.01, 0.1, 1, \text{ and } 10$) using three different system sizes (Small, Medium, Large). Vertical bars indicate the relative frequencies in samples with a small range of discrete values (< 25).	67

5.3	Q-Q plots of <i>Prey</i> samples; Chemical Langevin Equation vs Gillespie (<i>points</i>) and Linear Noise Approximation vs Gillespie (<i>crosses</i>) at at four time points ($t = 0.01, 0.1, 1, \text{ and } 10$) using three different system sizes (Small, Medium, Large).	70
5.4	Comparisons of <i>Prey</i> means (a) and s.d.s (b) using the sums of the squared log-ratios of the means–s.d. The first four entries at the x -axis correspond to the small system at the time instances $t = 0.01, 0.1, 1, 10$. Similarly the next four correspond to the Medium system and the last four to the Large.	71
5.5	Comparing the empirical transition densities – p.m.f. of the Gillespie Method (<i>solid black</i>) the Chemical Langevin Equation (<i>dashed black</i>) and the Linear Noise Approximation (<i>solid grey</i>) at four time points ($t = 0.01, 0.1, 1, \text{ and } 10$) using three different system sizes (Small, Medium, Large). Vertical bars indicate the relative frequencies in samples with a small range of discrete values (< 25).	73
5.6	Q-Q plots of samples; Chemical Langevin Equation vs Gillespie (<i>circles</i>) and Linear Noise Equation vs Gillespie (<i>crosses</i>) at various time points.	75
5.7	Comparisons of <i>DNA</i> means (a) and s.d.s (b) using the sums of the squared log-ratios of the means–s.d. The first four entries at the x -axis correspond to the small system at the time instances $t = 0.01, 0.1, 1, 10$. Similarly the next four correspond to the medium system and the last four to the large.	76
6.1	Illustrating the solutions obtained by the RE (<i>top left</i>) and the NR (<i>top right</i>) methods which are also plotted in a common plot (<i>bottom left</i>) for a toy dataset of the transcription network (Section 5.6.2). The mean $\mathbf{m}(t)$ of the NR method’s scaled residual process M_t is also plotted (<i>bottom right</i>) for the time interval $[0.05, 0.1)$ which obtains the non-trivial (non-zero) solution.	82
6.2	Q–Q plots of the likelihood ratio test statistic for the Lotka–Volterra model. Likelihood ratio statistics based on the Restarting (<i>points</i>) and the Non–Restarting method (<i>crosses</i>) are plotted against the quantiles of the χ_3^2 distribution (<i>lines</i>). Systems of three sizes (Small, Medium, Large) were considered at three observational intervals (0.1, 0.5, 1) which correspond to samples sizes of (100, 20, 10) observations, indicated with the symbol (#).	86

6.3	Q–Q plots of the Likelihood ratio test statistic for the Transcription model. Likelihood ratio test statistics based on the Restarting (<i>points</i>) and the Non–Restarting method (<i>crosses</i>) are plotted against the quantiles of the χ^2_8 distribution (<i>lines</i>). Systems of three sizes (Small, Medium, Large) were considered at three observational intervals (0.1, 0.5, 1) which correspond to samples sizes of (500, 100, 50) observations, indicated with the symbol (#).	90
6.4	Profile Likelihood ratios (solid lines) and the Quadratic approximation (dotted lines) of the D_1 (top left), D_2 (top right), D_3 (bottom left) datasets. The MLEs are indicated with vertical solid lines and the true vales with vertical dashed. The parameters (x -axis) are on log–scale and a 95% approximate cut–off point is included (horizontal grey line). Contour plot of the profile log–likelihood of (ϑ_5, ϑ_6) parameters in D_2 (top right).	95
6.5	Simulated observations of Preys (<i>diamonds</i>) and Predators (<i>points</i>) of the Lotka Volterra model. The trajectories are indicated with solid faded lines.	98

List of Tables

5.1	System of ODEs for Lotka-Volterra example.	66
5.2	Means (s.d.) of the single step Euler approximation and the corresponding Gillespie Samples for Lotka–Volterra Model	69
5.3	P-values of the multivariate singed rank test based on marginal ranks for the Lotka–Volterra model.	69
5.4	System of ODEs for the Transcription example.	72
5.5	Means (s.d.) of the single step Euler approximation and the corresponding Gillespie Samples for the Transcription model.	74
5.6	P-values of the multivariate rank-sum test keeping ϑ fixed, using 1000 samples from the Transcription example at four time instances.	77
6.1	Counts of statistically significant likelihood ratios for the Lotka–Volterra model.	85
6.2	Missed Wald’s CIs coverage counts for the Lotka-Volterra example.	87
6.3	Counts of significant likelihood ratio tests for the Transcription example.	89
6.4	Missed Wald’s CIs coverage counts for the Transcription example.	91
6.5	MCMC Posterior means and standard deviations for \mathbf{c} estimated on three datasets (D_1, D_2, D_3) of 50 observations. The Table above is a reproduction of Table 1 from Golightly and Wilkinson (2005).	92
6.6	Upper bounds of the 95% C.I. (<i>UCI</i>), Maximum Likelihood Estimates (<i>EST</i>) and the lower bound of ϑ (<i>LCI</i>) and of three datasets (D_1, D_2, D_3) of 50 obser- vations.	93
6.7	Upper bounds of the Wald’s 95% Confidence interval (<i>UCI</i>), Maximum Likeli- hood Estimates (<i>EST</i>) and Lower bounds of the 95% Confidence interval (<i>LCI</i>), of the simulated dataset.	99

Frequently Used Notation

Mathematical Notation

\mathbb{R}	The set of Real numbers
\mathbb{N}	The set of Natural numbers, i.e. positive integers
\mathbb{N}_0	The set of non-negative integers
\mathbb{Z}	The set of integers
\mathcal{F}_t	The generated σ -algebra up to time t
W_t	Brownian Motion
$\alpha \{\cdot\}$	The drift parameter of an SDE
$\sigma \{\cdot\}$	The diffusion coefficient
Σ	The infinitesimal variance covariance matrix
R_i	The i -th reaction
c_i	The rate constant corresponding to R_i
\mathbf{c}	The vector of constant rates
$\mathbf{h}(\mathbf{X}, \mathbf{c})$	The vector of hazard laws
$h_i(\mathbf{X}, c_i)$	The hazard law corresponding to R_i
Z	Standard Normal Distribution
$\mathbb{E}[X_t]$	Expectation of X_t
$\mathbb{E}[X_t \mathcal{F}_t]$	Conditional expectation of X_t given the σ -algebra up to time t
$\text{diag}\{X\}$	A diagonal matrix having X as the main diagonal
N_T	System Size
$\mathbf{y}(t; \boldsymbol{\vartheta})$	The system of ordinary differential equations $\boldsymbol{\vartheta}$ -dependent
$\ell(X_t, \mathbf{c})$	The log-likelihood function
$O(\delta_t^n)$	if $f(\delta_t) \in O(\delta_t^n)$, $ f(\delta_t) \leq M \delta_t^n $ where $M \in \mathbb{R}$ a constant and also $0 < M < \infty$. We usually assume that the terms $O(\delta_t^n)$ go faster to zero than δ_t^{n-1} when $\delta_t \rightarrow 0$. For other variables, e.g. h , we assume that $h \rightarrow \infty$.

Biological – Chemical Notation

- X The biological–chemical system
- N_S Number of Species
- N_R Number of Reactions
- R_i The i -th reaction
- $\{X_1\}$ The number of units (molecules) of the X_1 - species

Chapter 1

Introduction

Modern scientific disciplines have embraced the principle of *reductionism*, where a complex mechanism is explained by simpler ones. The principle of reductionism has affected all aspects of modern science and its origins can be traced back at 17th century in works of Descartes (Boogerd et al. 2007). In the exact opposite direction, the idea of *integration* arises, focusing on a system of simple interacting mechanisms which, as a whole, constitutes the system of study. Systems biology, a rather new branch of biology, is based on the integration principle: it focuses on the interactions that take place in biological systems. The original motivation for a systemic approach in biology can be found in works of Bánáthy (Chong and Ray 2002), whereas a treatment on the philosophical foundations of systems biology was presented by Boogerd et al. (2007). In this thesis we focus on a specialized topic of systems biology: the gene auto-regulatory networks, i.e. biological systems of genes and their functional products (e.g. RNA, proteins) which interact with each other.

1.1 Models

Various approaches have been proposed to represent gene regulatory systems. Biologists seem to favour diagrammatic “block and arrow” representations (Bower and Bolouri 2001) which are very convenient in illustrating the interactions among the individual species of the system. Although they give a very good description of the structure of the systems, the absence of measurable parameters makes the experimental testing particular difficult. A full-scale quantitative model can be considered instead, e.g. each participating species can be represented as a variable in a system of ordinary or partial differential equations. A quantitative model has

the obvious advantage of measurable variables but also introduces a new obstacle: it becomes inconvenient to examine the structural relations of species in a list of mathematical formulae.

A common ground has been proposed by treating the interactions of a biological system as a network of coupled biochemical equations, which provides both quantitative and qualitative representations. Each interaction in the biological system is represented as a chemical reaction and the participating species as reactants or products. Furthermore, each reaction is associated with a particular kinetic constant which is the measurable variable of interest (Wilkinson 2006).

A similar approach make use of the *Stochastic Petri Networks*; although they were first introduced to model concurrency at computer science, they provide, both graphically and quantitatively, a network representation. Goss and Peccoud (1998) introduce them at the context of molecular biology and Wilkinson (2006) presents examples of their use in gene regulatory systems.

As previously mentioned, each biochemical reaction is associated with a kinetic constant and the subject of this thesis is to propose an inferential methodology for these constants based on discrete time observations of the biological system of our interest. Although recent technologies, e.g. confocal microscopy, calcium imaging, fluorescent tagging of proteins (Bower and Bolouri 2001) permit the observation of molecules within living cells, it is generally difficult to directly observe the species of interest. Quite often a strategy of indirect observation is considered instead (Wilkinson 2010), resulting in partial observations subject to measurement error. For instance, a very popular technique binds a fluorescent protein gene with the gene of interest in order to produce proteins which are fluorescent to ultra-violet light (e.g. Wilkinson 2010, Finkenstadt et al. 2008, Henderson et al. 2010). The observation of the fluorescent proteins provides indirect information for the gene activity, but introduces further complexities, i.e. the need of data calibration (Wilkinson 2010). Nevertheless, it is a realistic requirement for an inferential procedure be able to analyse partial data with measurement error as well.

The quantitative description of a biological system encompasses the specification of experimentally measurable parameters which is the only way to formally falsify a model (Bower and Bolouri 2001). Bower and Bolouri (2001) go a step further to suggest a novel research methodology for systems biology models: a researcher can express his current understating of a biological system in a model and, with the help of experimental data, he/she will unveil the model's limitations in order to be further investigated. Therefore, they argue, the distinction between experimentation and modelling is blurring as the two methods become more closely

related.

Many mathematical modelling approaches to biological systems have been suggested; their main distinction lies in whether they take into account the system's intrinsic noise (stochastic models) or not (deterministic models). Typical deterministic models employ *systems of differential equations* or *Boolean networks* (Gibson and Mjolsness 2001). In particular, differential equations seem to be the most popular modelling choice, representing the concentration of the species in the biological system of interest. Also, differential equations are deterministic: a particular set of initial conditions always results to the same outcome. In contrast, a stochastic approach permits different random outcomes, even when the same initial conditions are considered. When the population of the participating species in a system increases, the theory suggests that both approaches, deterministic and stochastic, coincide. Unfortunately, this is not often the case for gene auto-regulatory networks since the number of individual molecules per species is not expected to be large (McAdams and Arkin 1997).

Most traditional analytical tools for the stochastic methods make use of the Master Equation (van Kampen 2007), which is not generally tractable. Gillespie (1976, 1977) proposed an exact model based on Markov Jump processes and also a method of simulating the system's dynamical behaviour which does not rely on the Master Equation formulation. Although, Gillespie's method is very appealing, its computational requirements are unrealistic for systems with a large number of species. Approximations based on diffusions processes have been considered instead, namely the Chemical Langevin Equation and the Linear Noise Approximation (LNA). Since we are interested in employing the models for inferential purposes, we rule out the model of Markov Jump processes since its application to discrete-time observations is particularly difficult (Boys et al. 2008). In this thesis, we examine new inferential methods based on models of diffusion processes and in particular the Linear Noise Approximation.

1.2 Thesis Overview

In Chapter 2, we present some elements from the theory of diffusion processes, numerical optimisation and numerical solution of ordinary differential equations, which will be used in subsequent chapters. Next, in Chapter 3, we focus on the modelling approaches of auto-regulatory gene networks and we apply them to two example networks. Chapter 4 is devoted on the Exact Algorithm (EA), a method proposed by Beskos et al. (2006) for (Monte Carlo)

likelihood-based estimation of discretely observed diffusion processes: we investigate the multidimensional extension of the EA algorithm and we illustrate its application to a very simple auto-regulatory network. In Chapter 5, we return to the general auto-regulatory models, by deriving the Linear Noise Approximation using a reparameterization of the stochastic kinetic constants. Additionally, a thorough comparison of the modelling approaches is also considered in a series of simulated experiments. We continue, in Chapter 6, by considering the Linear Noise Approximation as an inferential tool. In particular, the LNA approximation relies on the solution of a system of ODEs, and starting with different assumptions on their initial conditions, we derive two inferential methods which are compared under different experimental configurations. We also propose an extension of the LNA method for partially observed systems with measurement error and we illustrate it in a simple system. We have developed the `lnar` package which is an implementation of the LNA for the R statistical environment, and in Chapter 7 we describe the package's use with two examples based on the auto-regulatory networks of Chapter 3.

Chapter 2

Theory Background

In this chapter we present briefly some selected topics of probability theory and numerical analysis which are used in the subsequent chapters. In particular, we introduce Stochastic Differential Equations, their (linear) solutions and a convergence theorem of Markov Chains to SDEs. Additionally, we outline some aspects of the numerical solutions of Ordinary Differential Equations and numerical optimisation.

2.1 Diffusions

Diffusion processes provide a mathematical model for various physical phenomena; e.g. the motion of a particle in a fluid container: the particle follows the movement of the fluid but is also exposed to random collisions with the fluid's molecules causing small "fluctuations" to particle's trajectory. Omitting the fluctuations, we can describe the motion of the particle by an ordinary differential equation (ODE):

$$\frac{dX}{dt} = \alpha \{t, X\},$$

i.e. the particle at the position X at time t moves according to the fluid's drift $\alpha \{X, t\}$. Extending the deterministic model, we accommodate for the "random fluctuations" by introducing a "noise" term:

$$\frac{dX_t}{dt} = \alpha \{t, X_t\} + \sigma \{t, X_t\} \times \text{noise}. \quad (2.1)$$

We can restrict the choices for the "noise" term by assuming some characteristic properties:

- *Independent increments* i.e. the present fluctuations do not depend on past or future fluctuations.

- *Stationarity* i.e. its distribution is time-independent.
- *Unbiased* i.e. its increments have a zero mean.
- Its increments follow a *Normal distribution*.

The previous assumptions uniquely define (Øksendal 2005) a stochastic processes known as *Brownian Motion*¹.

2.1.1 Brownian Motion and SDEs

More formally, Brownian Motion, denoted by W_t , is a continuous-time stochastic process with independent Normal increments, i.e. for $0 \leq s < t$ the increment for the 1-d Brownian Motion follows a Normal distribution with zero mean and variance $(t - s)$:

$$W_t - W_s \sim \text{Normal}(0, t - s).$$

The trajectories of W_t process are continuous² and their increments $(W_t - W_s)$ are independent of W_s and of $(W_{t'} - W_t)$, assuming $t' > t$. Since our “noise” candidate is Normal-distributed we can approximate the equation (2.1) by assuming that the increments of the state variables (ΔX_t) are also Normal-distributed for small time intervals (δ_t) :

$$\Delta X_t = X_{t+\delta_t} - X_t \sim \text{Normal}(\alpha\{X_t\}\delta_t, \sigma\{X_t\}^2\delta_t), \quad (2.2)$$

where $\alpha\{\cdot\}$ denotes the infinitesimal mean change, known as the *drift coefficient* and $\sigma\{\cdot\}$ denotes the infinitesimal standard deviation, known as the *diffusion coefficient*³.

Although Brownian Motion is a continuous process, its paths are not smooth; in fact, they are nowhere differentiable (p. 109, Karatzas and Shreve 1991), therefore, we cannot define a “differential” equation using the conventional limiting approach. An alternative route is to consider the stochastic integral first, as the limiting sum of (2.2):

$$X_t \approx X_0 + \sum_{i=0}^k \alpha\{t_i, X_{t_i}\}(t_{i+1} - t_i) + \sum_{i=0}^k \sigma\{t_i, X_{t_i}\}(W_{t_{i+1}} - W_{t_i}) \quad (2.3)$$

with $0 = t_0 < t_1 < \dots < t_k = t$ and X_0 the initial condition, e.g. the starting position of the particle. For convenience, the notation $X(t)$ will also be used together with X_t . Under

¹The name *Wiener process* is also used.

²Locally Hölder continuous, a generalization of continuity (p. 53, Karatzas and Shreve 1991).

³For vector-valued processes, the infinitesimal Covariance matrix is $\sigma\{X_t\}^T \delta_t \sigma\{X_t\}$.

appropriate conditions, (e.g. §3, Øksendal 2005) the sum (2.3) converges to the stochastic integral known as *Itô integral*:

$$X(t, \omega) = X(0, \omega) + \int_0^t \alpha \{s, X(s, \omega)\} ds + \int_0^t \sigma \{s, X(s, \omega)\} dW_s(\omega) \quad (2.4)$$

where $X_t \equiv X(t)$. We have used ω to denote the dependence of each variable X_t, W_t on a particular path of the Brownian motion: at a fixed time t both $X(t, \omega)$ and $W_t(\omega)$ *may* have different values for different trajectories, i.e. $X(t, \omega) \neq X(t, \omega')$. The integral of (2.4) contains the $X(t, \omega)$ variable which, under appropriate conditions (§4, Øksendal 2005), is a stochastic process known as the *Itô process*. Quite often, if $X(t, \omega)$ is an Itô process, a short-hand notation is used to denote the stochastic integral of (2.4):

$$dX(t, \omega) = \alpha \{t, X(t, \omega)\} dt + \sigma \{t, X(t, \omega)\} dW_t, \quad (2.5)$$

which corresponds to a “stochastic” differential equation (SDE). The SDEs that satisfy certain existence and uniqueness conditions on their solutions, outlined in Theorem 2.1.1, are often called *diffusion processes*.

A possible way of simulating, approximately a trajectory $X_t(\omega)$, $t \in [0, T]$ of the SDE (2.5), is to discretize the time interval in small δ_t partitions and employ the (2.2) approximation. The latter is known as *Euler–Maruyama* approximation and approximates (2.5) with the process:

$$\Delta X(t) = \alpha \{t, X(t)\} \delta_t + \sigma \{t, X(t)\} Z \sqrt{\delta_t}, \quad (2.6)$$

where Z is a (multidimensional) standard normal distribution. We can then proceed to the simulation of a $X(t)$ trajectory by applying the Euler–Maruyama approximation sequentially (Figure 2.1). Naturally, as $\delta_t \rightarrow 0$ the approximation converges (§9.6, Kloeden and Platen 1995) to the true process.

2.1.2 Itô’s Lemma

The fundamental theorem of calculus relates the operations of differentiation and integration of real-valued functions; *Itô’s lemma* plays a similar role for functions applied to Itô processes. Specifically, the following lemma holds (Øksendal 2005) for multidimensional processes:

Lemma 2.1.1 Let $X(t, \omega)$ be an n -dimensional Itô process as in (2.5). Let $g(t, x) = (g_1(t, x), g_2(t, x), \dots, g_n(t, x))$ be a C^2 (the space of twice differentiable functions) map from $[0, \infty) \times \mathbb{R}^n \rightarrow \mathbb{R}^p$. Then the

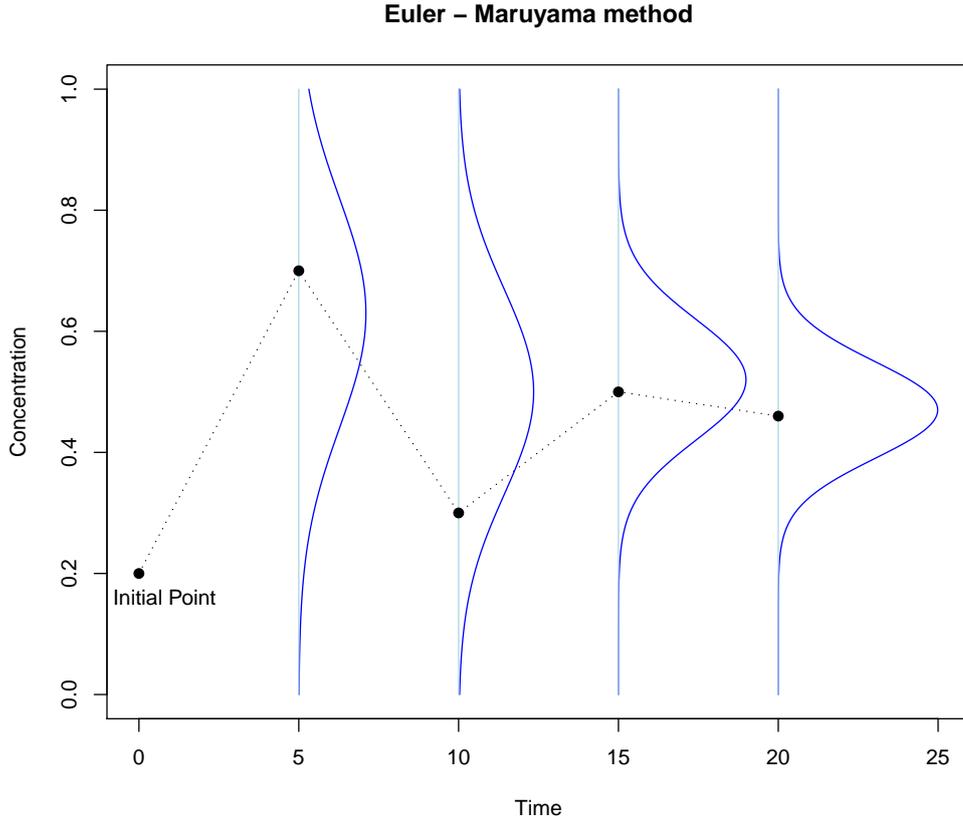


Figure 2.1: Illustration of the Euler–Maruyama method. The simulation interval ($t = 20s$) is partitioned into $5s$ intervals. At each interval the transition density is approximated with a Normal distribution (*illustrated vertically*) and a sample is drawn, which in turn becomes the next initial point.

process:

$$Y(t, \omega) = g(t, X(t, \omega))$$

is again an Itô process whose component number k , Y_k , is given by the formula:

$$dY_k = \frac{\partial g_k}{\partial t}(t, X(t, \omega)) dt + \sum_{i=1}^n \frac{\partial g_k}{\partial x_i}(t, X(t, \omega)) dX_i + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2 g_k}{\partial x_i \partial x_j}(t, X(t, \omega)) dX_i dX_j \quad (2.7)$$

where the following simplification rules apply:

$$dW_i dW_j = \delta_{ij} dt, \quad dW_i dt = dt dW_i = dt dt = 0.$$

Equation (2.7) establishes a closed–form relation between the original process $X(t, \omega)$ and the new process $Y(t, \omega)$ which we will use extensively at the next Sections.

2.1.3 Solution of SDEs

In this Section we will focus on the existence and uniqueness of the solution $X(t)$ of an SDE (2.5). We must note that there are two types of solutions that satisfy (2.5): the *strong* and *weak* solutions and we will present theorems for the existence and uniqueness of the solutions which involve the $\alpha\{\cdot\}$ and $\sigma\{\cdot\}$ coefficients. The distinction lies on whether we are given a *particular* Brownian motion path in advance and we construct a solution $X(t)$ based on it or we just ask for *any* (X_t, W_t) pair that satisfies (2.4). To make the distinction clearer, we consider the uniqueness properties of two solutions X_t, X'_t : if they are strong solutions, they are *pathwise unique*, i.e. $P(X_t = X'_t, \forall 0 \leq t < \infty) = 1$, whereas if they are weak solutions, they are *unique in law*, i.e. they have the same finite-dimensional distributions $P[(X_{t_1}, \dots, X_{t_n}) \in A] = P[(X'_{t_1}, \dots, X'_{t_n}) \in A]$, $n \in \mathbb{N}$.

Example 2.1.1 In order to emphasise the distinction between the weak and strong solutions, we consider the following trivial SDE:

$$dX_t = -dW_t, \quad X_0 = 0.$$

For a given path W_t we can construct, trivially, the strong solution by inverting the sign:

$$X_t = -W_t.$$

In addition, the solution $X'_t = W_t$ is only a weak solution because it follows the same law (Wiener measure), but its trajectory is of opposite sign. \blacklozenge

More formally the existence and pathwise uniqueness of strong solutions of SDEs is given from the following theorem Øksendal (2005):

Theorem 2.1.1. Existence and uniqueness. *Let $T > 0$ and $\alpha\{\cdot, \cdot\} : [0, T] \times \mathbb{R}^n \rightarrow \mathbb{R}^n$, $\sigma\{\cdot, \cdot\} : [0, T] \times \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ be measurable functions satisfying the linear growth condition*

$$|\alpha\{t, x\}| + |\sigma\{t, x\}| \leq C(1 + |x|); \quad x \in \mathbb{R}^n, t \in [0, T] \quad (2.8)$$

for some constant C , (where $|\sigma\{t, x\}|^2 = \sum |\sigma\{t, x\}_{i,j}|^2$) and such that the Lipschitz condition holds:

$$|\alpha\{t, x\} - \alpha\{t, y\}| + |\sigma\{t, x\} - \sigma\{t, y\}| \leq D|x - y|; \quad x, y \in \mathbb{R}^n, t \in [0, T] \quad (2.9)$$

for some constant D . Let Z be a random variable which is independent of the σ -algebra⁴ $\mathcal{F}_\infty^{(m)}$ generated by $W_s(\cdot), s \geq 0$ and such that

$$\mathbb{E} [|Z|^2] < \infty.$$

Then the stochastic differential equation

$$dX_t = \alpha \{t, X_t\} dt + \sigma \{t, X_t\} dW_t, 0 \leq t \leq T, X_0 = Z \quad (2.10)$$

has a unique t -continuous solution $X_t(\omega)$ with the property that $X_t(\omega)$ is adapted to the filtration \mathcal{F}_t^Z generated by Z and $W_s(\cdot); s \leq t$ and

$$\mathbb{E} \left[\int_0^T |X_t|^2 dt \right] < \infty. \quad (2.11)$$

Weak solution As previously mentioned, a weak solution is a pair of processes (X_t, W_t) together with an appropriate probability space (see for more details §V.16 Rogers and Williams 1988, or, §5.3 Karatzas and Shreve 1991) that satisfy (2.4) or more intuitively, has the “right” law.

An equivalent expression of the weak solution can be obtained from the *martingale problem* (§V.20 Rogers and Williams 1988) (formally stated in Definition 2.1.1) which defines a property on the law of the process, i.e. the probability measure on the space of continuous functions. We introduce the martingale problem formulation because it is more convenient and gives more relaxed conditions for the weak solutions of SDEs, instead of working directly with the (X_t, W_t) pair. Informally, we consider a process M_t which is based on the drift and diffusion coefficients of the X_t and on f : any twice continuously differentiable function vanishing outside a bounded interval. The martingale problem asks whether a probability measure exists under which the process M_t is a martingale⁵. If such a measure exists then it is a solution of the martingale problem, and if is unique then the martingale problem is *well-posed*. We provide a formal definition (Øksendal 2005) for completeness:

Definition 2.1.1. Let L be a semi-elliptic differential operator⁶ of the form

$$L = \sum b_i \frac{\partial}{\partial x_i} + \sum a_{ij} \frac{\partial^2}{\partial x_i \partial x_j}$$

⁴Informally, a σ -algebra (\mathcal{F}_t) can be interpreted as the history of the process up to time t .

⁵The conditional expectation of M_s given its previous history up to time $t, s < t$ is M_t

⁶An operator can be considered as a mapping between functions e.g. the partial derivative. The differential operator is a function containing the partial derivative operator.

where the coefficients b_i, a_{ij} are locally bounded Borel measurable functions on \mathbb{R}^N . Then we say that a probability measure \tilde{P}^x on $\left((\mathbb{R}^n)^{[0,\infty)}, \mathcal{B}\right)$ solves the martingale problem for L (starting at x) if the process

$$M_t = f(\omega_t) - \int_0^t Lf(\omega_r)dr, M_0 = f(x) \text{ a.s. } \tilde{P}^x$$

is a \tilde{P}^x martingale w.r.t. the Borel σ -algebras \mathcal{B}_t of $(\mathbb{R}^n)^{[0,t]}$, for all $f \in C_0^2(\mathbb{R}^n)$. The martingale problem is called well-posed if there is a unique measure \tilde{P}^x solving the martingale problem.

It can be shown (§V.20 Rogers and Williams 1988) that when a solution to the martingale problem exists, it is equivalent to the existence of a weak solution. Similarly, when the law of an SDE exists and is unique the martingale problems has at most one solution. Of course if we combine the existence and the uniqueness of the weak solution the martingale problem is well-posed.

Example 2.1.2 We will give an explicit solution to the martingale problem for the simple SDE:

$$dX_t = -dW_t, \quad X_0 = 0,$$

which corresponds to a differential operator L with coefficients $b = 0$ and $a = 1/2$. From Itô's lemma (2.1.1) we know that for any twice differentiable function $f(x)$ the following equality holds:

$$f(X_t) = f(0) + \int_0^t \frac{1}{2} \frac{\partial^2 f(X_s)}{\partial x^2} ds - \int_0^t \frac{\partial f(X_s)}{\partial x} dW_s,$$

which can be used to derive the following expression for M_t :

$$M_t = f(X_t) - \int_0^t \frac{1}{2} \frac{\partial^2 f(X_s)}{\partial x^2} ds = f(0) - \int_0^t \frac{\partial f(X_s)}{\partial x} dW_s.$$

The last term is an Itô integral which is a martingale with respect to the probability measure of the Brownian motion (§3.2 Øksendal 2005). In this trivial example, the law of X_t, \tilde{P}^0 , coincides with the law of W_t which is the Wiener measure. Thus M_t , is a \tilde{P}^0 -martingale w.r.t. \mathcal{B}_t and we can conclude that \tilde{P}^0 solves the martingale problem. In addition, it is a solution to a well-posed martingale problem because \tilde{P}^w is unique. \blacklozenge

A slightly relaxed set of conditions exists which is sufficient in establishing the solution of the well-posed martingale problem (§V.24 Rogers and Williams 1988):

Theorem 2.1.2. Stroock–Varadhan uniqueness theorem. Suppose $\sigma \{ \cdot \} \sigma \{ \cdot \}^T : \mathbb{R}^N \rightarrow S_N^+$, S_N^+ the space of real $N \times N$ nonnegative–definite symmetric matrices, $\alpha \{ \cdot \} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ and:

- (i) $\Sigma = \sigma \{ \cdot \} \sigma \{ \cdot \}^T$ is continuous,
- (ii) Σ is strictly positive definite, i.e. its eigenvalues are positive,
- (iii) for some constant K , for all i, j and x ,

$$|\Sigma_{i,j}| \leq K(1 + |x|^2), \quad |(\alpha \{ x \})_i| \leq K(1 + |x|)$$

Then the martingale problem for $\sigma \{ \cdot \}, \alpha \{ \cdot \}$ is well–posed.

One of the requirements of the Theorem 2.1.2 is that all eigenvalues of Σ to be positive. As we will see in Chapter 3, some kinetic models do not satisfy this requirement because they follow one or more *conservation laws* (§2.3.3 Wilkinson 2006). For instance, a conservation law may state that the total number of a certain species in a system (e.g. copies of a gene) does not change. A usual work-around is to re-express the conserved species as a linear combination of a constant and the other species. The simplified kinetic model has the same dynamics as the original model and the derived SDE approximation has a positive-definite Σ .

2.1.4 Linear SDEs

Generally, very few stochastic differential equations have explicitly known solutions (§4, Kloeden and Platen 1995). If an SDE’s drift and diffusion coefficients are linear functions of the state process, then the SDE belongs to the class of *linear stochastic differential equations* which has known solutions. Our concern is focused on a more restricted class, named as *linear SDEs in narrow–sense*, which takes the following equivalent form:

$$dX(t, \omega) = F(t) X(t, \omega) dt + \sigma \{ t \} dW_t, \quad \text{or, } X(t, \omega) = X(t_0, \omega) + \int_{t_0}^t F(s) X(s, \omega) ds + \int_{t_0}^t \sigma \{ s \} dW_s, \quad (2.12)$$

i.e. the diffusion coefficient depends on time–varying constants, $X(t, \omega), W_t$ are N_S and N_R –dimensional processes respectively, while $F(t), \sigma \{ t \}$ denote time-dependent $N_S \times N_S$ and $N_S \times N_R$ matrices. The SDE (2.12) has known solutions as a member of the linear SDEs class, which will be derived in the next Section using a fundamental matrix solution.

2.1.4.1 Fundamental Matrix Solution

At first look, the deterministic integral of (2.12) involves the state variable, $X(t)$, which is non-trivial to solve. The Fundamental Matrix method overcomes this difficulty by applying a transformation to the $X(t)$ process to express (2.12) independently of $X(t)$. The before-mentioned transformation is the fundamental matrix solution (§5.6, Karatzas and Shreve 1991 or §4.8.6, Kloeden and Platen 1995) which is a matrix that satisfies the following ODE:

$$d\Phi(t; t_0) = F(t)\Phi(t; t_0)dt, \quad \Phi(t_0; t_0) = I, \text{ i.e. the unit matrix.} \quad (2.13)$$

we can identify $\Phi(t; t_0)$ as a first order, matrix-valued, homogeneous ODE with varying coefficients. Solving (2.13) w.r.t. $F(t)$ (assuming $\Phi(t; t_0)$ is non-singular) we have:

$$F(t) = \frac{d\Phi(t; t_0)}{dt} \Phi^{-1}(t; t_0).$$

Using the last expression, we can express the time derivative of the inverse matrix $\Phi^{-1}(t; t_0)$ (note that $\Phi(t; t_0)\Phi^{-1}(t; t_0) = \Phi^{-1}(t; t_0)\Phi(t; t_0) = I$) in terms of $F(t)$:

$$\frac{d}{dt} \Phi^{-1}(t; t_0) = -\Phi^{-1}(t; t_0) \frac{d\Phi(t; t_0)}{dt} \Phi^{-1}(t; t_0) = -\Phi^{-1}(t; t_0) F(t).$$

In addition, the transpose of $\Phi^{-1}(t; t_0)$, $(\Phi^{-1}(t; t_0))^T$ is a fundamental matrix solution of the adjoint system of (2.13) (for more details see §3.2 at Coddington and Levinson 1955). Lets now apply (2.7) to the process $Y(t) := U(t, X_t) = \Phi^{-1}(t; t_0)X(t, \omega)$:

$$\begin{aligned} dY(t, \omega) &= \left[\frac{dU}{dt}(t, X_t)X(t, \omega) + (\nabla U(t, X_t))^T F(t)X(t, \omega) \right] dt + (\nabla U(t, X_t))^T \sigma \{t\} dW_t \\ &= \left[-\Phi^{-1}(t; t_0)F(t)X(t, \omega) + \Phi^{-1}(t; t_0)F(t)X(t, \omega) \right] dt + (\nabla U(t, X_t))^T \sigma \{t\} dW_t \\ &= \Phi^{-1}(t; t_0) \sigma \{t\} dW_t, \end{aligned}$$

where $(\nabla U(t, X_t))^T$ is the transpose of the gradient of $U(t, X_t)$ and coincides with the Jacobian matrix. We can rewrite the last expression in integral form:

$$\begin{aligned} \Phi^{-1}(t; t_0)X(t, \omega) &= \Phi^{-1}(t_0; t_0)X(t_0, \omega) + \int_{t_0}^t \Phi^{-1}(s; t_0) \sigma \{s\} dW_s \\ X(t, \omega) &= \Phi(t; t_0) \left(X(t_0, \omega) + \int_{t_0}^t \Phi^{-1}(s; t_0) \sigma \{s\} dW_s \right), \end{aligned} \quad (2.14)$$

we have used the fact that $\Phi^{-1}(t_0; t_0) = I$. If we consider each term of (2.14) individually we can conclude that the SDE is reduced to an ODE plus a Gaussian noise process: $X(t_0, \omega)$ is a constant, $\Phi^{-1}(t; t_0)$, $\Phi^{-1}(s; t_0)$ and $\sigma \{s\}$ are deterministic functions, while the stochastic

integral $\int_{t_0}^t \Phi^{-1}(s; t_0) \sigma \{s\} dW_s$ can be interpreted as a Gaussian random variable since it is a linear combination of the increments of Brownian motion. Therefore, $X(t, \omega)$ is a Gaussian random variable as well. Taking the expectation of the integral representation of (2.12) we have:

$$\mathbb{E}[X(t, \omega)] = \mathbb{E}\left[X(t_0, \omega) + \int_{t_0}^t F(s)X(s, \omega)ds\right] + \mathbb{E}\left[\int_{t_0}^t \sigma \{s\} dW_s\right],$$

where the expectation of the stochastic integral becomes a vector of zeroes. Additionally, if we differentiate w.r.t. to t , we can express the mean $m(t)$ of (2.14) as the solution of the following ODE:

$$\frac{dm(t)}{dt} = F(t)m(t). \quad (2.15)$$

As a side note, if we set our initial value to $X(t_0) = \mathbf{0}$, a N_S -vector of zeroes, (2.15) becomes:

$$\mathbb{E}[X(t, \omega)|X(t_0, \omega)] = 0, \quad (2.16)$$

because (2.15) attends the trivial solution $m(t) = 0$. For convenience, we define $V(s) = \Phi(t; t_0)\Phi^{-1}(s; t_0)\sigma \{s\}$, and we draw our attention to the variance of the X_t (2.14) given by:

$$S(t) = \int_{t_0}^t V(s)V(s)^T ds = \int_{t_0}^t \Phi(t; t_0)\Phi^{-1}(s; t_0)\sigma \{s\} [\Phi(t; t_0)\Phi^{-1}(s; t_0)\sigma \{s\}]^T ds, \quad (2.17)$$

and a new $N_S \times N_S$ quadratic diffusion $J(t, \omega) = U(X_t) = X(t, \omega)X(t, \omega)^T$ with elements given from the application of Itô's lemma:

$$dJ_{ij}(t, \omega) = \frac{1}{2} \sum_{m=1}^{N_R} \sum_{l,k=1}^{N_S} V_{lm}(t)V_{km}(t) \frac{\partial^2 U_{ij}}{\partial x_l \partial x_k} dt + \sum_{m=1}^{N_R} \sum_{l=1}^{N_S} F_{lm} \frac{\partial U_{ij}}{\partial x_l} dW_t^j,$$

we apply the following simplification to the sum of the first term:

$$\sum_{l,k=1}^{N_S} V_{lm}(t)V_{km}(t) \frac{\partial^2 U_{ij}}{\partial x_l \partial x_k} = \begin{cases} 2V_{im}^2(t), & \text{when } u_{ii} = x_i^2 \\ 2V_{im}(t)V_{jm}(t), & \text{when } u_{ij} = x_i x_j, j \neq i \end{cases}$$

and $\mathbb{E}[dJ_{ij}] = \frac{2}{2} \sum_{m=1}^{N_R} V_{im}(t)V_{jm}(t) = V_{i\cdot}(t)V_{\cdot j}(t)^T$ in vector notation. Therefore, we switch back to the integral form in order to calculate the expectation of the new process:

$$\mathbb{E}[J(t, \omega)] = \int_{t_0}^t V(s)V(s)^T ds = S(t), \quad (2.18)$$

which coincides with the variance of X_t (2.17). We repeat our previous strategy, by taking the derivate of (2.18) to express $S(t)$ as the solution of the following ODE:

$$\begin{aligned}
\frac{dS(t)}{dt} &= \frac{d}{dt} \left(\Phi(t; t_0) \int_{t_0}^t \Phi^{-1}(s; t_0) \sigma \{s\} (\Phi^{-1}(s; t_0) \sigma \{s\})^T ds \Phi(t; t_0)^T \right) \\
&= \frac{d\Phi(t; t_0)}{dt} \int_{t_0}^t \Phi^{-1}(s; t_0) \sigma \{s\} (\Phi^{-1}(s; t_0) \sigma \{s\})^T ds \Phi(t; t_0)^T \\
&\quad + \Phi(t; t_0) \Phi^{-1}(t; t_0) \sigma \{t\} (\Phi^{-1}(t; t_0) \sigma \{t\})^T \Phi(t; t_0)^T \\
&\quad + \Phi(t; t_0) \int_{t_0}^t \Phi^{-1}(s; t_0) \sigma \{s\} (\Phi^{-1}(s; t_0) \sigma \{s\})^T ds \frac{d\Phi(t; t_0)^T}{dt} \\
&= F(t)S(t) + \sigma \{t\} \sigma \{t\}^T + S(t)F(t)^T.
\end{aligned} \tag{2.19}$$

As a side note, the equation (2.19) is also related to the solution of the multidimensional Kalman–Bucy filtering problem from the filtering theory (Øksendal 2005).

2.1.4.2 Normal Transition Density

In Section 2.1.4 we derived the solution of the linear SDEs in the narrow sense and established that $X(t)$ follows a multidimensional normal density. In this Section, we continue with the derivation of the parameters of the transition density. The transition density $P(X(t, \omega)|X(t_0, \omega))$ is a multivariate Normal distribution and we can track its mean and covariance matrix from the estimates (2.15) and (2.19) respectively:

$$X(t, \omega)|X(t_0, \omega) \sim \text{Normal}(m(t), S(t)). \tag{2.20}$$

The evaluation of the transition density (2.20) at a time t , given an initial point $X(t_0)$, requires the solution of the $m(t), S(t)$ ODEs subject to their initial conditions. In addition, due to the Markov property⁷ we can write down the transition density and the related ODEs for arbitrary times $s \leq t$, provided that we observe the process' state at the time instance s :

$$\begin{aligned}
X(t, \omega)|X(s, \omega) &\sim \text{Normal}(m(t), S(t)), \\
\frac{dm(t)}{dt} &= F(t)m(t), \text{ with initial point } m(s) = X(s, \omega), \\
\frac{dS(t)}{dt} &= F(t)S(t) + \sigma \{t\} \sigma \{t\}^T + S(t)F(t)^T, \text{ with i.p. } S(s) = \mathbf{0} \text{ (matrix of zeroes)}.
\end{aligned}$$

Since we are able to track the transition density at any time instance t_1 conditioned on an initial point $X(t_0)$, the simulation of the process $X(t_1)$ becomes trivial. In addition, the

⁷For a discrete-time discrete-state process X_t the Markov Property states: $P(X_t|X_{t-1}, X_{t-2}, \dots, X_0) = P(X_t|X_{t-1})$.

Markov property allows us to condition on $X(t_1)$ in order to simulate a next point $X(t_2)$, given $t_0 < t_1 < t_2$, hence we can simulate a full trajectory of the $X(t)$ process by updating each time the initial state.

2.1.5 Brownian Bridge

We introduce in this subsection a special process which will be used in Chapter 4 extensively. Intuitively, a Brownian Bridge can be considered as a “stochastic interpolation” between the points $a, b \in \mathbb{R}$ of length $t \in (0, \infty)$, or simply, as a Brownian Motion “tied down” at two points. More formally, for $0 \leq s < t$, the following SDE (Øksendal 2005):

$$dY_s = \frac{b - Y_s}{t - s} ds + dW_s, \quad Y_0 = a, \quad (2.21)$$

is solved by a process known as *Brownian Bridge*:

$$Y_s = a(t - s) + bs + (t - s) \int_0^s \frac{1}{t - r} dW_r. \quad (2.22)$$

We denote the distribution of (2.22) with $\mathcal{W}^{(a,b,t)}$. Furthermore, we can obtain a trajectory from a Brownian Bridge following $\mathcal{W}^{(0,0,t)}$, which is a special case of $\mathcal{W}^{(a,b,t)}$, by transforming a path from a Brownian Motion process:

$$W_s - \frac{s}{t} W_t, \quad s \in [0, t].$$

Additionally, the Brownian Bridge holds the relocation property which facilitates the extension of its simulation: if $\omega \sim \mathcal{W}^{(0,0,t)}$ then for arbitrary $a, b \in \mathbb{R}$, $s \in [0, t]$, the transformed path $\omega_s + (1 - s/t)a + (s/t)b$, follows $\mathcal{W}^{(a,b,t)}$.

2.1.6 Convergence to SDEs

In this Section we try to explore the conditions under which a Markov Chain converges to a diffusion process. For instance, let us consider Y_t , a discrete-time Markov Chain. The process Y_t does not resemble a continuous process and a simple idea to overcome this problem is to speed up the time. If Y_t is constant on time intervals $[n, n + 1)$, $n \in \mathbb{N}_0$ we can introduce a new process $Y_{[t/h]}^h$ which is now constant on intervals $[nh, (n + 1)h)$, i.e. h scales one time unit to a time-period of length h . We expect that as h becomes smaller the process will resemble a diffusion process and eventually will converge. In the next chapters we adapt this idea to the

context of chemical reactions: instead of reducing the time step h to speed up the number of events per unit of time, we increase the system's size which, in turn, increases the rate of the occurrence of events and, implicitly, decreases h .

We continue by stating the previous idea formally, using a theorem that sets the conditions for the convergence of a Markov Chain to a diffusion process, originally by Stroock and Varadhan (1979). In the context of the next theorem, X_t^h is a discrete-state discrete-time Markov Chain with a scaled time step h . As $h \rightarrow 0$, X_t^h converges to a diffusion process X_t . The following statement of the theorem is by Durrett (1996):

Theorem 2.1.3 (Stroock-Varadhan). *Suppose in continuous time, for any compact set $K \subset \mathbb{R}^d$, the following holds for the transition rates $Q_h(x, A) \equiv \frac{d}{dt}P(X_t^h \in A | X_0^h = x)$, $x \in K$, $A \in \mathbb{R}^d$:*

$$\sup_{x \in K} Q_h(x, A) < \infty$$

and in either case that the martingale problem is well-posed and for each $i, j, R < \infty$ and $\epsilon > 0$:

- (i) $\lim_{h \rightarrow 0} \sup_{|x| < R} |\sigma_{ij}^h\{X_t\} - \sigma_{ij}\{X_t\}| = 0$, $\sigma_{ij}^h = \int_{|y-z| \leq 1} (y_i - x)(y_j - x_j) Q_h(x, dy)$
- (ii) $\lim_{h \rightarrow 0} \sup_{|x| < R} |\alpha_i^h\{X_t\} - \alpha_i\{X_t\}| = 0$, $\alpha_i^h = \int_{|y-z| \leq 1} (y_i - x) Q_h(x, dy)$
- (iii) $\lim_{h \rightarrow 0} \sup_{|x| < R} Q(x, B(x, \epsilon)^c) = 0$, $B(x, \epsilon) = \{y : |y - x| < \epsilon\}$.

If $X_0^h = x_h \rightarrow x$ then X_t^h converges weakly to X_t , the solution of the martingale problem with $X_0 = x$.

The infinitesimal moments $\alpha^h(\cdot), \sigma^h(\cdot)$ above are truncated, i.e. it is assumed that $|x - y| \leq 1$. Durrett (1996) extends the Theorem above, considering complete infinitesimal moments ($\hat{\alpha}^h(\cdot), \hat{\sigma}^h(\cdot)$), in the following Lemma (p. 306):

Lemma 2.1.1. *If $p \geq 2$ and for all $R < \infty$ we have*

- (a) $\lim_{h \rightarrow 0} \sup_{|x| \leq R} |\hat{\alpha}_i^h\{X_t\} - \alpha_i\{X_t\}| = 0$
- (b) $\lim_{h \rightarrow 0} \sup_{|x| \leq R} |\hat{\beta}_{ij}^h\{X_t\} - \alpha_i\{X_t\}| = 0$
- (c) $\lim_{h \rightarrow 0} \sup_{|x| \leq R} \gamma_p^h(x) = 0$, $\gamma_p^h(x) = \int |y - x|^p Q_h(x, dy)$

then (i), (ii) and (iii) of Theorem 2.1.3 hold.

As we saw in Section 2.1.3, the well-posed martingale problem guarantees the existence and uniqueness of the weak solution of an SDE, or equivalently, the existence and uniqueness of a law that all solutions of the X_t SDE follow. Therefore, as $h \rightarrow 0$, the scaled X_t^h process converges weakly, i.e. has the same unique law as the diffusion process X_t . The conditions of Theorem 2.1.2 are sufficient for the solution of the well-posed martingale problem which are satisfied by the diffusion processes that we will consider in this thesis.

2.2 ODE Integration

Previously, we established that the solutions of linear SDEs depend on the solution of a system of ordinary differential equations. Generally, the resulting ODEs cannot be solved analytically and we resort to numerical solutions. In this Section we introduce the numerical ODEs solvers, also known as numerical integrators, which will be employed in the subsequent chapters. First we use one of the simplest method of numerical ODEs solution, *Euler's method*, to introduce the most basic features of ODEs solvers. Then, we consider two major types of numerical methods of solving systems of ordinary differential equations: *Runge-Kutta* which combine several intermediate approximations (Euler-type steps) with a Taylor series expansion, and *Multistep methods (Predictor-Corrector)* which estimates the solution by extrapolating the solutions obtained from previous steps. In particular, we will use two methods that belong to the latter (multistep) class: *Adams' method* and *Backward Differentiation Formula* which is better suited for *stiff* problems.

Generally, a system of ODEs is usually given in the following form:

$$\frac{dY_t}{dt} = F(t, Y_t),$$

where $Y_t := Y(t)$, and $F(t, Y_t)$ is the derivative of Y_t w.r.t. time t . We are aiming to evaluate Y_t at a certain time point t_1 , given an initial value Y_0 at $t_0 = 0 < t_1$. A simple approximation, known as the *Euler's method*, can be derived using the very definition of derivative:

$$Y_{t+h} \approx Y_t + F(t, Y_t)h, \tag{2.23}$$

where h is the *step-size*, which is assumed to be small; as $h \rightarrow 0$ the approximation converges to the true solution (Butcher 2008). The choice of h is crucial for all numerical ODE solvers: from one hand, a small step-size decreases the approximation error, but on the other hand,

introduces extra computational cost since as the discretization becomes finer, more evaluations of $F(t, Y_t)$ are needed. Unfortunately, Euler's method is numerically unstable and its use is strongly discouraged (Press et al. 2007, Butcher 2008). Nevertheless, it serves as an excellent introduction to numerical ODE solvers.

An important concept in the analysis of numerical integrators is the *order* (n) of the numerical method which indicates the order of (local) error at each step $O(h^{n+1})$, e.g. Euler's method is a first-order, $O(h^2)$, method. It should be noted that although most numerical schemes have higher orders expressions, they do not always yield high accuracy (Press et al. 2007), for instance, it may be more efficient to consider a smaller step-size and a lower order. We considered implementations of the ODEs solvers that support adaptive step-sizes: heuristics are employed to keep the local error estimates under a tolerance level, which is supplied by the user. Multistep methods follow more sophisticated strategies that enable both the step-size and the order of the method to vary (Petzold 1983, Uri M. Ascher 1998) as the solution progresses.

Runge-Kuta is a class of very popular methods (Press et al. 2007) since they are relatively easy to implement and their properties have been extensively analyzed (e.g. Butcher 2008). In our analysis, we have chosen the RK45 method which is a modification of the original method. It is based on the embedded Runge-Kutta formulas that allow for an adaptive step-size control by monitoring the differences between forth and fifth-order solutions estimates. Additionally, we have used the implementation of the GSL library (Galassi et al. 2009).

A system of ordinary differential equations is characterized as *stiff* when, in the course of its numerical solution, its elements have different scales, e.g. some elements are expected to change faster than others in the same unit of time. Although a widely accepted formal definition of *stiffness* does not exist, a usual ad-hock rule (p. 22 Hairer and Wanner 1991) is to look for a dominant eigenvalue in the eigenvalue decomposition of the Jacobian of $F(t, Y_t)$, or alternatively, when the ratio of the largest to the smallest absolute eigenvalues indicates a difference of scale by several orders of magnitude. The previous methods (e.g. RK45) are numerically unstable for stiff ODEs and specialized methods have been suggested which are numerically more stable, e.g. see Hairer and Wanner (1991) for an extensive review. The most popular method for stiff equations is the *Backward Differentiation Formula* (BDF) which belongs to the class of Multistep methods. Its basic idea is to evaluate the derivate at time

$t + h$:

$$Y_{t+h} \approx Y_t + F(Y_{t+h}, t)h,$$

but since Y_{t+h} is not known beforehand, the only way to track it is numerically, by employing a modified Newton method to a system of non-linear equations (Uri M. Ascher 1998).

To overcome the problems associated with stiffness we could employ a specialized method blindly but it would be computationally inefficient for non-stiff problems. Alternatively, we have chosen an adaptive method instead, namely LSODA, proposed by Petzold (1983) which automatically switches between the implicit Adams methods (suitable for non-stiff problems) and the Backward differentiation formulas (BDF). The switching mechanism relies on estimates of the performance and stability of the solutions to make the appropriate decision. As previously mentioned, LSODA implements further optimisations that adapt the order of the numerical integrator using the local error estimates; in particular the order of Adams methods can vary between 1–12 and for BDF method between 1–5.

2.3 Numerical Optimisation

In our case, Maximum likelihood estimates are not analytically known and are obtained with the help of numerical optimisation methods. More specifically, we have used two methods of multidimensional unconstrained non-linear optimisation, namely the *Nelder–Mead* and the *BFGS* method, to optimize the *objective function*, i.e. the likelihood function derived in Section 6.2.

2.3.1 Nelder–Mead

The class of direct search multivariate optimisation algorithms describe the methods of unconstrained optimisation that rely only on evaluations of the objective function without the explicit use of derivatives (Kolda et al. 2003). The *downhill simplex* method or the *Nelder–Mead method* was first proposed by Nelder and Mead (1965) and remains, until today the most popular direct search method. The algorithm assumes a simplex, a geometric object, that consists of $N + 1$ points in N dimensions. At every iteration, the Nelder–Mead method adapts the simplex to the local topology of the objective function, using a series of heuristic moves, and tries to surround a stationary point as close as possible. Kolda et al. (2003) showed

that it is not guaranteed to converge to a stationary point, i.e. a point where the gradient vanishes, unless certain smoothness conditions apply to the objective function (Mckinnon 1999). Although we based our optimisation tasks on alternative methods (BFGS) that guarantee the convergence to stationary points, Nelder–Mead remains a reference method and is used for comparison.

2.3.2 BFGS

The Broyden–Fletcher–Goldfarb–Shanno (BFGS) method belongs to the class of multidimensional unconstrained quasi–Newton methods. Newton’s methods approximate the objective function at each step with a quadratic function based on first (gradient vector) and second–order (Hessian matrix) derivatives. Quasi–Newton methods do not explicitly use the Hessian matrix but an approximation based on the gradient vectors. Surprisingly, the approximation is beneficial (§10.9 Press et al. 2007) in cases where the initial points are located far away from a stationary point. Unfortunately, due to the dependence of the likelihood function to the system of ODEs, it is not possible to derive the gradient vectors analytically, with the rare exception of linear ODEs. We resort to the numerical method of (central) *finite differences* (§5.7 Press et al. 2007) to estimate the gradient vectors.

Chapter 3

Models

In this chapter we begin with a qualitative (biological) description of a genetic regulatory network and we describe all the intermediate steps until we obtain a quantitative (stochastic) model. First, we motivate the expression of the interactions in a genetic regulatory network as biochemical reactions. There is overwhelming theoretical and experimental evidence that the outcomes of regulatory functions in organisms are non-deterministic (e.g. McAdams and Arkin 1997), therefore, it is reasonable to assume a stochastic kinetics framework. Various quantitative approaches have been proposed to model the dynamics of networks of biochemical reactions and we present them according to their modelling assumptions:

- **Markov Jump Processes** Gillespie (1976, 1977) expressed, under appropriate physical assumptions, the kinetics of chemical reactions as Markov Jump processes. His derivation is *exact*, since it “*correctly accounts for the inherent fluctuations and correlations*”¹ (Gillespie 1977). Additionally, the author proposed an algorithm to simulate trajectories from these processes which is presented in Section 3.2.2. In Section 3.3, however, we argue that this algorithm is too demanding, in terms of the computational cost, to be considered in a full analysis of a high dimensional system.
- **Chemical Langevin Equation** Following Wilkinson (2006) and Gillespie (2000), we can compensate for the large computational cost by employing an approximation. The approximation is derived at Section (3.3) and is a diffusion process. The resulting SDEs are non-linear and we employ numerical methods for their solution, e.g. the Euler–

¹More formally it correctly follows the probability distributions (Master Equation) imposed by the chemical laws.

Maruyama discretization.

- **Linear Noise Approximation** This can be viewed as a linear approximation of the Chemical Langevin Equation. It was first introduced by Kurtz (1972) and the resulting SDEs are linear with tractable solutions. We postpone its exposition until the Section 5.3, where we provide an alternative derivation based on a reparametrization of the stochastic kinetic constants.
- **Master Equation** This is a differential equation that expresses the evolution of the transition probability conditional on a initial point. The Master Equation (ME) is an infinite order partial differential equation (Gillespie 1991) and does not generally have analytical solutions (Gillespie 1976), hence cannot be directly employed for the study of chemical or biological systems. Nevertheless, the ME enables the derivation of expressions for the evolution of the moments of the transition density (§4.2.C Gillespie 1991). When non-linear systems are considered, the derived moments are expressed in terms of equations depending on higher order moments, which cannot be solved neither analytically nor numerically. The *moment-closure* approximation (Gillespie 2009) overcomes this obstacle by deriving estimates for the evolution of the moments of the transition density in the form of a system of ODEs which is easier to solve than the ME, e.g. see §4.2.B Gillespie (1991) for more details. The main idea of the moment-closure approximation is to set the moments that exceed a certain order equal to zero, effectively truncating the transition density's high order moments. For instance, the truncation of third and higher order moments leads to a Gaussian approximation with the mean vector and covariance obtained from the solution of a system of ODEs which can be derived numerically. As a side note, both the Linear Noise Approximation and the Chemical Langevin Equation are approximations of the Master Equation (Gillespie 2000, Ferrel et al. 2008).
- **Ordinary Differential Equation** This is a limiting approximation obtained by increasing the size of the system. The ODE is appealing because of its minimal computational requirements. The drawback of the ODE model lies on their deterministic nature: the evolution of the biological system is completely determined by its initial values, which is a strong assumption considering the intrinsic stochasticity associated with the gene expression (McAdams and Arkin 1997).

In this chapter we only focus on the first two methodologies, while the last two are considered at Chapter 5. A detailed comparison of the above methods can be found at Ferm et al. (2008).

3.1 Stochastic Motivation

The activity of genetic regulatory networks is based on the expression rates of genes and is often described as a *cascade*: the product of an expressed gene (e.g. a protein) may regulate the expression of another gene, creating a *genetically coupled link*. After the initiation of the transcription process of the first protein, the process continues until the concentration has reached a critical level which in turn triggers the second process. Therefore, the time interval between the two events depends on the *rate of accumulation* of the protein.

According to the motivational work by McAdams and Arkin (1997) the pattern of protein concentration exhibits short and random *bursts* of newly-produced proteins at random time intervals which resembles a realization of a stochastic process. Additionally, the deterministic principle is violated when a homogeneous population is exposed to the same initial conditions but after a certain time becomes non-homogeneous. In our case, the homogeneous population is an isogenic² population of cells, and the regulatory function of interest is modelled using a small number of species. After a period of time, variations in the population will start to occur creating subpopulations of different phenotypes. These variations can be attributed to the regulatory mechanism, and more specifically, to a random selection of different regulatory pathways. Arkin et al. (1998) exhibited this idea by considering a stochastic kinetic model for the Phage λ -infected *Escherichia Coli* cells. Simulation from this model resulted in different phenotypes and was consistent with their experimental observations.

3.2 Interactions as reactions

Following Wilkinson (2006), we assume that a satisfactory representation of the interactions in a biological system can be expressed as a network of coupled chemical reactions. We consider a system of N_R reactions and N_S reactants, a typical reaction (R_j) possesses the following general form:



²All members have the same genotype.

where $\{X_i\}$ denotes the number of molecules of i -th species of the biochemical system, the vectors \mathbf{m} , \mathbf{n} are the *stoichiometry coefficients* and c_j are the *stochastic rate constants* (Gillespie 1992). The stochastic rate constants are the parameters of interest, denoted by the vector $\mathbf{c} = (c_1, \dots, c_{N_R})$. The notation at (3.1) describes the reaction of m_1 molecules of X_1 with m_2 molecules of X_2 producing n_1 molecules of X_3 , the *product* of R_j , with a rate involving the stochastic rate constants c_j . For convenience, we denote with X_t the system's state (in number of molecules) at the t time instance and we use the same notation to refer to the stochastic process which expresses the dynamical behaviour of \mathbf{X} . Another convention is that $\{\cdot\}$ expresses the number of molecules of the relevant species.

The interpretation of a stochastic rate constant c_j can be better explained by considering the quantity $c_j \delta_t$, which expresses the *average probability* that a particular combination of the reactants associated with the R_j reaction will react accordingly in the next δ_t time interval (Gillespie 1977). Similarly, the *hazard function*³ $h_j(\mathbf{X}, c_j)$, takes into account all possible combinations of the reactants associated with the R_j reaction so that $h_j(\mathbf{X}, c_j)$ expresses *the rate of occurrence* of the R_j reaction. For example if the stoichiometry coefficients are the pair $(m_1, m_2) = (1, 1)$ and there are $\{X_1\}, \{X_2\}$ molecules of X_1, X_2 respectively, then there are $\{X_1\}\{X_2\}$ distinct combinations of reactants. Therefore, the probability of the R_1 reaction occurring in the next δ_t interval is $h_1(\mathbf{X}, c_1) \delta_t = c_1 \{X_1\} \{X_2\} \delta_t$.

Since the number of reactions is usually large, it is more convenient to work with a matrix notation. We define as A a $(N_S \times N_R)$ matrix, the *reaction* or *net effect matrix*, whose j -th column describes the difference at the state X_t , after the occurrence of a single reaction of type R_j :

$$A_{.j} = (-(m_{j1}, \dots, m_{jN_S}) + (n_{j1}, \dots, n_{jN_S}))^T \quad (3.2)$$

where $(m_{j.}, n_{j.})$ are the stoichiometric coefficients associated with R_j and the stoichiometry matrix is the transpose of A : $S = A^T$.

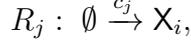
3.2.1 Simple Reactions

In this thesis we will consider only three types of reactions which can be classified according to their order, i.e. how many reactants participate. We introduce each reaction type and we also derive the corresponding hazard functions.

³Also known as *rate law*.

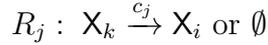
3.2.1.1 Reactions of order zero

Reactions of the following form:



describe the introduction of new molecules in the system. Since there is an absence of reactants (\emptyset), the hazard function coincides with the stochastic rate $h_j(\mathbf{X}, c_j) = c_j$.

3.2.1.2 First order reactions



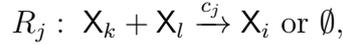
Express the alteration of the X_k biochemical species to the X_i species, as in radioactive decay, or to extinction (\emptyset). The hazard function is given by:

$$h_j(\mathbf{X}, c_j) = c_j \{X_k\},$$

since that each molecule of X_k could change to X_j with stochastic rate c_j .

3.2.1.3 Second order reactions

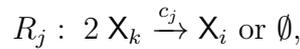
Second order reactions consist of two types. The first describes the reaction of two molecules of *different* species:



where (X_k, X_l) molecule pairs could react with a stochastic rate c_j , resulting the corresponding hazard function:

$$h_j(\mathbf{X}, c_j) = c_j \{X_k\} \{X_l\}.$$

When two molecules of the same species react:



we have $\{X_k\}(\{X_k\} - 1)/2$ possible pairs which leads to the following hazard function:

$$h_j(\mathbf{X}, c_j) = \frac{1}{2} c_j \{X_k\} (\{X_k\} - 1), \quad (3.3)$$

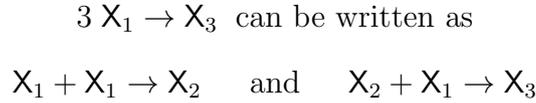
which can be approximated it with the following hazard function:

$$h_j(\mathbf{X}, c_j) = \frac{1}{2} c_j \{X_k\}^2. \quad (3.4)$$

In the next chapters, we always employ (3.3) in our exact (Gillespie) simulations and (3.4) to the ODE and SDE approximations.

3.2.1.4 Higher order reactions

It has been suggested (Wilkinson 2006) that higher order reactions can be written as a combination of intermediate products and lower order reactions. For example:



The drawback is that by decomposing a higher order reaction to simpler ones the dynamics are altered compared to the original reaction. Nevertheless, in this thesis we will only consider reactions up to second order.

3.2.2 Gillespie algorithm

So far we have discussed how we can obtain a network of coupled chemical reactions from a biological system. In this Section we are concerned with the analysis of a biochemical system's time evolution. Traditional methodologies involve the Master Equation. As mentioned at the introduction of this Chapter, the Master Equation expresses the time evolution of the transition density of the system conditional on an initial state. In most of the cases, the Master Equation is neither analytically nor numerically tractable (Gillespie 1976). Gillespie (1976, 1977) overcame this obstacle by suggesting a stochastic simulation algorithm which follows the system's exact dynamics without considering approximations nor time-discretization. The algorithm simulates realizations of the underlying stochastic (Markov Jump) process by iterating between two steps: it first simulates the time of the next reaction occurrence (any reaction allowed) and then chooses a particular reaction.

When reaction R_j occurs at t , the corresponding state $X(t)$ changes *deterministically* according to the stoichiometry of R_j or the j -th row of A . Assuming that our current state is $X(t)$, the next reaction occurs at the time instance $(t+t')$, where t' is exponentially distributed:

$$t' \sim \text{Exponential}(h_0(\mathbf{X}, \mathbf{c})),$$

where $h_0(\mathbf{X}, \mathbf{c}) = \sum_{i=1}^{N_R} h_i(\mathbf{X}, c_i)$ is the total rate. Afterwards, a reaction h_j is selected with probability $h_j(\mathbf{X}, c_j)/h_0(\mathbf{X}, \mathbf{c})$ and the state is updated accordingly. It should be noted that the process is constant during the time interval before the occurrence of the next reaction, i.e. for all $t^* \in [t, t+t')$, $X(t^*) = X(t)$.

Since the rate laws depend on the current state $X(t)$, the time distribution of the next reaction occurrence is *state dependent* also. It should be stressed that if we allow the state variable to change, the distribution of the time of the next reaction occurrence is no longer Exponential, with the exception of reactions of order zero.

More formally, assuming a fixed volume, a uniform mixture of N_S chemical elements which react through N_R reactions and thermal equilibrium the *Gillespie algorithm* at the j -th iteration involves the following steps (Algorithm 1) which iterate until the desired duration has been reached:

Algorithm 1 Gillespie Algorithm

1. Calculate $h_0(\mathbf{X}, \mathbf{c})^{(j)}$
 2. Sample the time to next occurrence ($t+t'$): $t' \sim \text{Exponential}(h_0(\mathbf{X}, \mathbf{c}))$
 3. Choose the j -th reaction with probability $P(j) = h_j(\mathbf{X}, c_j)/h_0(\mathbf{X}, \mathbf{c})$.
 4. Update the state X_t according the stoichiometry of R_j .
-

3.3 Diffusion approximation

The Gillespie algorithm simulates exact trajectories from X_t , the continuous time discrete-state stochastic process corresponding to the number of molecules of species ($X_i, i = 1, \dots, N_S$) of the system. However, it is computationally very demanding, and even impractical for simulation (Wilkinson 2006) and inference (Boys et al. 2008) of large biological systems. An alternative approach is to approximate the exact dynamics via a diffusion process known as the *Chemical Langevin equation*; Gillespie (2000) provides a formal derivation, whereas we give an informal introduction, motivated again from Wilkinson (2006).

To derive a diffusion approximation for the discrete-state system X_t at time point t we consider its behavior in a small fixed time interval δ_t . The number of occurrences of the j th reaction (\mathcal{N}_j), will be approximately:

$$\mathcal{N}_j \sim \text{Poisson}(h_j(\mathbf{X}, c_j) \delta_t). \quad (3.5)$$

In (3.5) we have assumed that the rate of the i th reaction is constant over δ_t which is in principle true for the limiting case $\delta_t \rightarrow 0$. Furthermore, if we expect \mathcal{N}_j occurrences of the

j -th reaction (R_j), then the current state X_t is expected to change according to the vector $A_j^T \mathcal{N}_j$, where A_j is the j -th column of A , i.e. the change that results from the occurrence of a single R_j reaction. Consequently, the state change ΔX_t in a infinitesimal time interval can be expressed as:

$$\Delta X_t = A^T \mathcal{N} \quad (3.6)$$

where $\mathcal{N} = (\mathcal{N}_1, \dots, \mathcal{N}_{N_R})$ is a N_R -vector of Poisson (3.5) random variables; their mean is provided by the vector of hazard functions $\mathbf{h}(\mathbf{X}, \mathbf{c}) = (h_1(\mathbf{X}, c_1), \dots, h_{N_R}(\mathbf{X}, c_{N_R}))^T$. Due to the Poisson assumption, each reaction R_j is independent of the others and the mean change of the overall system becomes:

$$\mathbb{E}[\Delta X_t] = \mathbb{E}[A^T \mathcal{N}] = A^T \mathbf{h}(\mathbf{X}, \mathbf{c}) \delta_t, \quad (3.7)$$

and the corresponding variance:

$$\mathbb{E}[\Delta X_t (\Delta X_t)^T] = \mathbb{E}[A^T \mathcal{N} \mathcal{N}^T A] = A^T \text{diag}\{\mathbf{h}(\mathbf{X}, \mathbf{c})\} A \delta_t + O(\delta_t^2) \quad (3.8)$$

where $\text{diag}\{\mathbf{h}(\mathbf{X}, \mathbf{c})\}$ is a diagonal $N_R \times N_R$ matrix with main diagonal $(h_1(\mathbf{X}, c_1), \dots, h_{N_R}(\mathbf{X}, c_{N_R}))$. Using the same arguments we can also calculate the third raw moment, which will be used in the next chapters. The third moment (μ'_3) is a multidimensional array of three indices⁴ ($\mathbf{i}, \mathbf{j}, \mathbf{k}$), with elements:

$$\mathbb{E}[(\Delta X_t)_i (\Delta X_t)_j (\Delta X_t)_k] = \sum_{z=1}^{N_R} A_{zi} A_{zj} A_{zk} h_z(\mathbf{X}, c_z) \delta_t + O(\delta_t^2). \quad (3.9)$$

In Chapter 2 we introduced diffusion processes; here we consider one, which we denote by V_t , and we adapt it to the Poisson process (3.5). For infinitesimally small time intervals dt , the infinitesimal state change dV_t can be approximated, according to the Euler–Maruyama (2.6) approximation, by a Normal distribution by matching the corresponding mean (3.7) and variance (3.8) of (3.5):

$$\mu(V_t, \mathbf{c}) = A^T \mathcal{N} \text{ and } \Sigma(V_t, \mathbf{c}) = \sigma(V_t, \mathbf{c}) \sigma(V_t, \mathbf{c})^T = A^T \mathcal{N} A$$

For simulation purposes, we resort to the Euler–Maruyama approximation (2.6) by discretizing the desired interval $[0, T]$ into smaller time increments δ_t .

⁴Or more generally a *tensor*.

3.3.1 Case studies

We consider two biological models: the *Transcription* model represents the transcription and repression of protein in a prokaryotic organism; the *Lotka–Volterra* model, although not a biological model *per se*, is widely used to illustrate a non-linear dynamical auto-regulatory behavior. We apply the methodology of this chapter to derive the relevant quantitative models, i.e. the corresponding hazard functions and net effect matrices.

3.3.1.1 Lotka–Volterra Example

The *Lotka–Volterra* model describes a population of two competitive species; it was proposed independently by Lotka (1925) and Volterra (1926). We consider two species (Prey, Predators) which interact in the following way: each prey reproduces (R_1) with rate c_1 ; a predator reproduces by consuming a prey (R_2) with rate c_2 and has a natural death (R_3) with rate c_3 . The dependences between the species and the reactions are presented as a graph in Figure 3.1, where the species are represented as *ellipses* and the reactions as *rectangles*. In addition, the reactions of the model are summarized at (3.10).

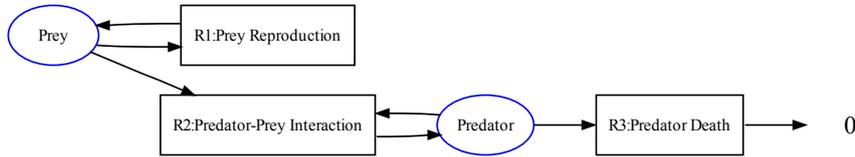
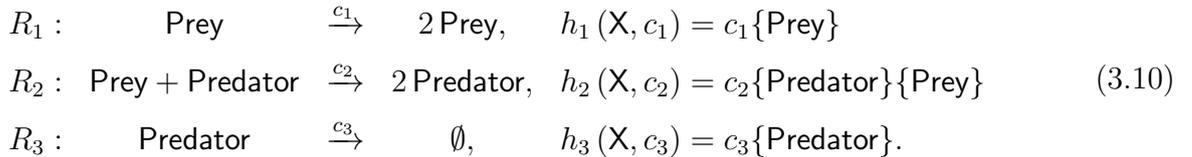


Figure 3.1: Graph representing Lotka Volterra (Prey-Predator) model.



The stoichiometry matrix is:

$$A^T = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}. \tag{3.11}$$

Therefore, the mean vector and variance matrix are respectively

$$\alpha \{X, \mathbf{c}\} = A^T \mathbf{h}(X, \mathbf{c}) = \begin{pmatrix} c_1 \{\text{Prey}\} - c_2 \{\text{Prey}\} \{\text{Predator}\} \\ c_2 \{\text{Prey}\} \{\text{Predator}\} - c_3 \{\text{Predator}\} \end{pmatrix} \quad (3.12)$$

and

$$\sigma \{X, \mathbf{c}\}^T \sigma \{X, \mathbf{c}\} = \begin{pmatrix} c_1 \{\text{Prey}\} + c_2 \{\text{Prey}\} \{\text{Predator}\} & -c_2 \{\text{Prey}\} \{\text{Predator}\} \\ -c_2 \{\text{Prey}\} \{\text{Predator}\} & c_2 \{\text{Prey}\} \{\text{Predator}\} + c_3 \{\text{Predator}\} \end{pmatrix} \quad (3.13)$$

◆

3.3.1.2 Transcription Example

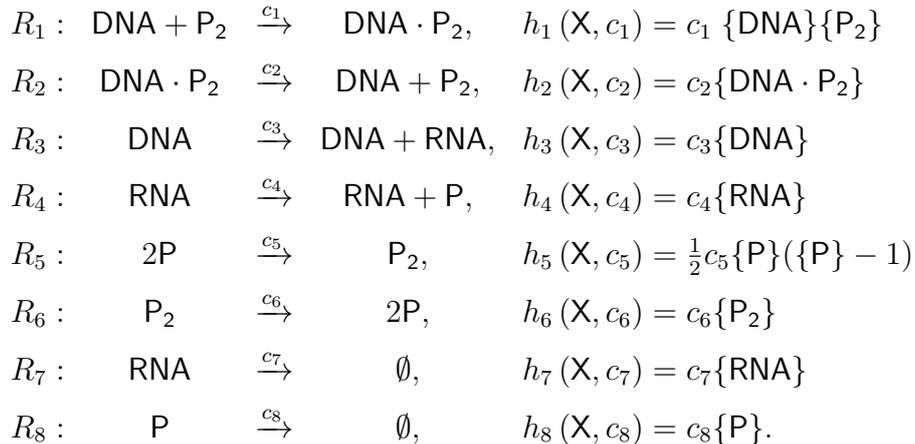
For our modelling purposes we use the example in Golightly and Wilkinson (2005) which is represented graphically in Figure 3.2. This biochemical network describes a prokaryotic auto-regulatory network illustrating three cellular processes: transcription, degradation and repression. Schematically, the biochemical network is expressed by five reactants:

$$X = (\text{RNA}, P, P_2, \text{DNA} \cdot P_2, \text{DNA}),$$

illustrated as *ellipses*, and eight reactions, illustrated as *rectangles*, which occur with stochastic rate constants $\mathbf{c} = (c_1, \dots, c_8)$ according to the dependence graph of Figure (3.2).

A gene (DNA) is transcribed (R_3) to RNA which in turn is translated (R_4) to protein P. Both RNA and P degrade as time passes, which is expressed by the reactions R_7 and R_8 respectively. Finally, two molecules of P bind together (R_5) to form protein dimers P_2 which can either split again (R_6) or repress the DNA transcription by binding (R_1) to the operator region. Additionally, the attached repressor $\text{DNA} \cdot P_2$ can also be detached (R_2).

The corresponding rate laws are:



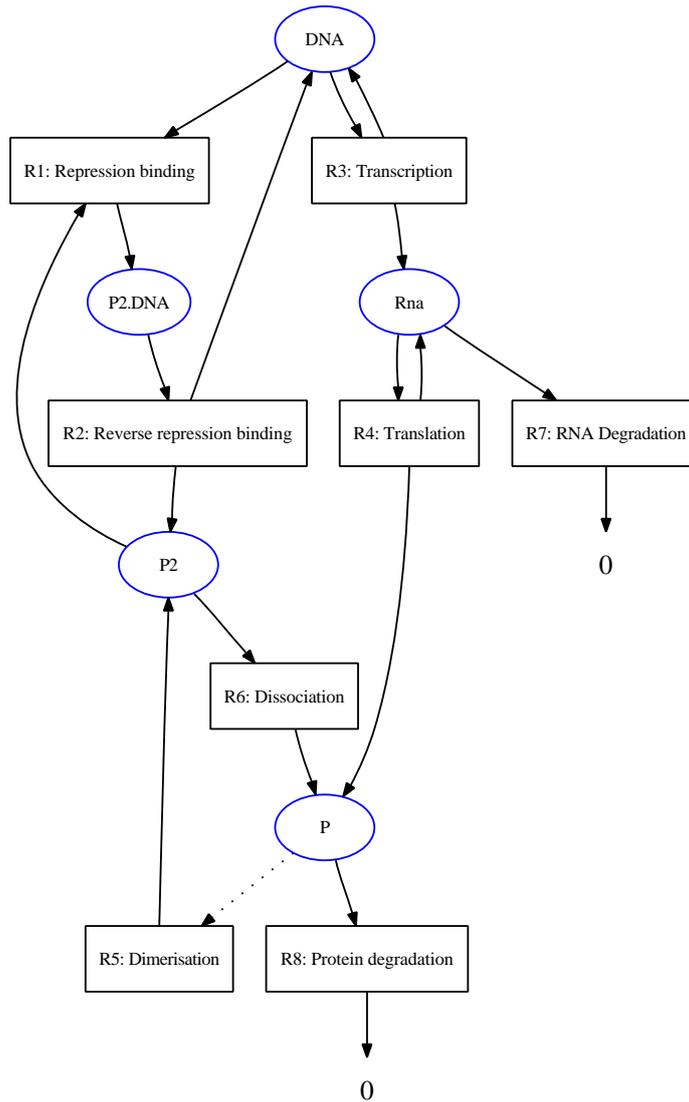


Figure 3.2: Graph representing the example prokaryotic auto regulation biochemical network. Edges with solid lines represent stoichiometric coefficients of ones, dotted lines coefficients of twos.

Here $\{\cdot\}$ denotes the number of molecules of the relevant species and the net effect matrix (A) is:

$$A^T = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 2 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & -2 & 2 & 0 & -1 \\ -1 & 1 & 0 & 0 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (3.14)$$

Moreover, (3.14) is linear dependent, i.e. $A_4^T = -A_5^T$, which results to one or more zero eigenvalues, violating the assumptions of Theorem 2.1.2 (p. 15). This can be attributed to the following conservation law, as discussed by Golightly and Wilkinson (2005):

$$\{\text{DNA} \cdot \text{P}_2\} + \{\text{DNA}\} = k,$$

where k is the total number of repressed and non-repressed genes and is assumed to be constant and predefined. Substituting $\{\text{DNA} \cdot \text{P}_2\} = k - \{\text{DNA}\}$ and omitting A_4^T we have:

$$A^T = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & -2 & 2 & 0 & -1 \\ -1 & 1 & 0 & 0 & 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (3.15)$$

and $h_2(\mathbf{X}, c_2) = c_2\{\text{DNA} \cdot \text{P}_2\}$, the remaining hazard rates are unaffected by the reparametrization. The mean for the corresponding diffusion approximation is:

$$\alpha\{\mathbf{X}, \mathbf{c}\} = A^T \mathbf{h}(\mathbf{X}, \mathbf{c}) = \begin{pmatrix} c_3\{\text{DNA}\} - c_7\{\text{RNA}\} \\ c_4\{\text{RNA}\} - c_5(\{\text{P}\} - 1)\{\text{P}\} - c_8\{\text{P}\} + 2\{\text{P}_2\}c_6 \\ \frac{c_5(\{\text{P}\}-1)\{\text{P}\}}{2} - c_1\{\text{P}_2\}[\text{DNA}] + c_2(k - \{\text{DNA}\}) - \{\text{P}_2\}c_6 \\ c_2(k - \{\text{DNA}\}) - c_1\{\text{P}_2\}\{\text{DNA}\} \end{pmatrix}, \quad (3.16)$$

and the infinitesimal variance-covariance matrix is $\Sigma = \sigma\{\mathbf{X}, \mathbf{c}\}^T \sigma\{\mathbf{X}, \mathbf{c}\} = A^T \text{diag}\{\mathbf{h}(\mathbf{X}, \mathbf{c})\} A =$

$$\begin{pmatrix} c_7\{\text{RNA}\} + c_3\{\text{DNA}\} & 0 & 0 & 0 \\ 0 & c_4\{\text{RNA}\} + 2c_5(\{\text{P}\} - 1)\{\text{P}\} + c_8\{\text{P}\} + 4\{\text{P}_2\}c_6 & -c_5(\{\text{P}\} - 1)\{\text{P}\} - 2\{\text{P}_2\}c_6 & 0 \\ 0 & -c_5(\{\text{P}\} - 1)\{\text{P}\} - 2\{\text{P}_2\}c_6 & \frac{c_5(\{\text{P}\}-1)\{\text{P}\}}{2} + C + \{\text{P}_2\}c_6 & C \\ 0 & 0 & C & C \end{pmatrix}, \quad (3.17)$$

where, $C = c_1\{\text{P}_2\}\{\text{DNA}\} + c_2(k - \{\text{DNA}\})$. \blacklozenge

Chapter 4

Exact Algorithm

4.1 Introduction

At Section 2.1.1 we introduced the Euler–Maruyama method to numerically approximate the solution of an SDE. The Euler–Maruyama method converges to the “true” solution when the time increments of the approximation tend to zero. An alternative approach to discretization schemes, the Exact Algorithm (EA), has been proposed by Beskos et al. (2006) which can be used for the simulation of trajectories of diffusion process as well as for inferential purposes. As the authors comment “the algorithm is *exact* in the sense that no discretization exists” therefore the sampled paths follow the same finite-dimensional distributions with the original process. When the EA is employed for inferential purposes, the estimation error can be attributed exclusively to the Monte Carlo estimation.

Currently, the Exact algorithm is defined for the class of univariate diffusion processes having *weak solutions* (see Section 2.1.2) and *unit variance*, i.e. diffusions of the form:

$$dX_s = \alpha \{X_s; \mathbf{c}\} ds + dW_s. \quad (4.1)$$

It can be extended to a wider class, e.g. the 1-d SDEs with arbitrary parameters:

$$dV_t = \mu \{V_t; \mathbf{c}\} dt + \sigma \{V_t; \mathbf{c}\} dW'_t,$$

which can be transformed to (4.1) by finding a suitable transformation $\eta(\cdot; \mathbf{c})$, them *Lamperti transform*, that sets the diffusion coefficient to unity:

$$\eta(\cdot; \mathbf{c}) = \int \frac{1}{\sigma \{x; \mathbf{c}\}} dx. \quad (4.2)$$

Now, if we apply $\eta(\cdot; \mathbf{c})$ to V_t , the transformed process $X_t = \eta(V_t; \mathbf{c})$, has a unit diffusion coefficient, while the drift coefficient is given by the following expression (Beskos et al. 2006):

$$\alpha \{x; \mathbf{c}\} = \frac{\mu \{ \eta^{-1}(x; \mathbf{c}); \mathbf{c} \}}{\sigma \{ \eta^{-1}(x; \mathbf{c}); \mathbf{c} \}} - \frac{\sigma \{ \eta^{-1}(x; \mathbf{c}); \mathbf{c} \}'}{2},$$

where $\eta^{-1}(x; \mathbf{c})$ is the inverse transformation of $\eta(x; \mathbf{c})$ and $\sigma \{ \cdot \}'$ the derivate of the diffusion coefficient w.r.t. the state variable. If the transformed stochastic differential equation (SDE) satisfies certain growth and limiting conditions the exact algorithm can be applied. The output is a sample from a weak solution, i.e. a process having the same finite-dimensional probability distributions as (4.1) which can be transformed back to the original diffusion. More specifically, the EA outputs an exact skeleton, realisations of the process at finite time-points, in a given time interval, which can be “filled-in” up to an arbitrary number of time-points and then is back-transformed to the original process using the inverse of (4.2).

Furthermore, the authors suggest that the algorithm can be extended to multivariate and inhomogeneous diffusions, as long as a suitable n -dimensional reversible transformation can be found and Girsanov’s formula is tractable. In Section 4.2, we start with a rejection sampler and we showcase its limitations in the SDE setting. Then, in Section 4.3 we show how the Exact Algorithm addresses these limitations; in Section 4.4 we give an overview of the EA-based inferential methods and we attempt to apply it on a scalar-valued SDE (Section 4.5), which arises from a simple auto-regulatory model. Towards the end of the chapter (Section 4.6) we will discuss the limitations of a multidimensional extension.

4.2 Rejection Sampling

Essentially the EA is a rejection sampling algorithm (for more details on the rejection sampling see e.g. §2.3, Robert and Casella 2004), which involves a proposal and an acceptance–rejection step. A tradition approach to rejection sampling would consider the proposal of candidate trajectories from a known process, the Brownian Bridge (Section 2.1.5) in our case, which in turn are accepted (or rejected), through a stochastic mechanism. In this hypothetical case, the acceptance criterion is given by the Radon–Nikodym derivative which, intuitively, compares the transition probability distributions of the two processes. Girsanov’s theorem provides us with a very convenient form of the Radon-Nikodym derivative, assuming that the relevant

conditions (see for instance Øksendal 2005) hold:

$$\frac{d\mathcal{Q}_\theta^{(t,x_i,x_{i+1})}}{d\mathcal{W}^{(t,x_i,x_{i+1})}} = \frac{\mathcal{N}_t(x_{i+1} - x_i)}{p_t(x_i, x_{i+1}; \mathbf{c})} \exp \left\{ - \int_0^t \alpha \{s, \omega; \mathbf{c}\} dB_s - \frac{1}{2} \int_0^t \alpha \{s, \omega; \mathbf{c}\}^2 ds \right\} \quad (4.3)$$

where \mathcal{Q}_θ is the measure of the stochastic process of interest for the transition $(x_i, 0) \rightarrow (x_{i+1}, t)$, \mathcal{W} is the measure underlying the corresponding Brownian Bridge, $\mathcal{N}_t(x_{i+1} - x_i)$ is the Normal distribution with mean 0 and variance t and $p_t(x_i, x_{i+1}; \mathbf{c})$ is the transition density for the transition $(x_i, 0) \rightarrow (x_{i+1}, t)$. Essentially (4.3) is the ratio of the transition density of interest over the known Normal transition density of a Brownian Bridge times a correction term, similar to the idea of standardizing a Gaussian random variable.

In Chapter 3, we considered diffusion processes as alternative models for the biological systems. Assuming that the kinetic constants \mathbf{c} are the parameters of interest, we can derive the likelihood function, from the Markov Property, as a product of transition densities:

$$L(\mathbf{c}|X_{obs}) = p(x_0; \mathbf{c}) \prod_{i=1}^n p_t(x_{i-1}, x_i; \mathbf{c}) \quad (4.4)$$

where $p(x_0; \mathbf{c})$ is the probability density for the initial point and $p_t(x_{i-1}, x_i; \mathbf{c})$ is the transition density for the transition $(x_i, 0) \rightarrow (y, t)$ which is generally unknown and analytically intractable. The EA, as an acceptance algorithm, can be used to estimate the individual transition densities, hence the likelihood function as well. We proceed by highlighting the difficulties associated with the estimation of (4.4). Our first step is to match the likelihood (4.4) with the acceptance probability as specified by (4.3), which can be shown that is proportional to:

$$\frac{d\mathcal{Q}_\theta^{(t,x_i,y)}}{d\mathcal{W}^{(t,x_i,y)}} \propto \exp \left(-r(\omega, \mathbf{c}) \int_0^t \phi(\omega_s, \mathbf{c}) ds \right), \quad (4.5)$$

where $l(\mathbf{c})$, $r(\omega, \mathbf{c})$ and $\phi(\omega_s, \mathbf{c})$ are bounds of functions which contain the following term:

$$\frac{\alpha \{u; \mathbf{c}\}^2 + \alpha \{u; \mathbf{c}\}' }{2}, \quad (4.6)$$

which can be obtained from Ito's Lemma when we assume that the transformation $A(u) = \int_0^u \alpha \{z\} dz$ is applied to the biased Brownian Bridge (for more details see p. 2425 in Beskos and Roberts 2005). We denote with $\alpha \{u; \mathbf{c}\}^2$ the squared term and $\alpha \{u; \mathbf{c}\}'$ the derivative w.r.t. u . The expression $l(\mathbf{c})$ is a lower bound of (4.6) and depends only on the parameters \mathbf{c} :

$$l(\mathbf{c}) \leq \inf_{u \in \mathbb{R}} \{ (\alpha \{u; \mathbf{c}\}^2 + \alpha \{u; \mathbf{c}\}') / 2 \},$$

similarly, $r(\omega, \mathbf{c})$ is an upper bound to $\alpha \{u; \mathbf{c}\}^2 + \alpha \{u; \mathbf{c}\}' - l(\mathbf{c})$ which depend on the path ω :

$$r(\omega_s, \mathbf{c}) \geq \sup_{s \in [0, t]} \{ (\alpha \{\omega_s; \mathbf{c}\}^2 + \alpha \{\omega_s; \mathbf{c}\}') / 2 - l(\mathbf{c}) \},$$

and we can use the last two bounds to produce a “standardized” version of (4.6):

$$\phi(\omega_s, \mathbf{c}) = \frac{1}{r(\omega, \mathbf{c})} \left(\frac{\alpha \{\omega_s; \mathbf{c}\}^2 + \alpha \{\omega_s; \mathbf{c}\}'}{2} - l(\mathbf{c}) \right),$$

which satisfies $0 \leq \phi(\omega, \mathbf{c}) \leq 1$. A number of versions of the EA algorithm have been proposed which pose different assumptions on $r(\omega, \mathbf{c})$. For instance, EA1 assumes that $r(\omega, \mathbf{c})$ does not depend on the path ω i.e. $r(\omega, \mathbf{c}) = r(\mathbf{c})$. Through this introduction we choose to illustrate the main ideas of the EA using the EA1 variation. In Section 4.3.1 we will consider two further extensions of EA, namely, EA2 and EA3.

For our purposes it would be sufficient to have a scheme that allows to sample from \mathcal{Q}_θ . A traditional approach using a rejection sampler would propose ω from the law of dW , which is the law of Brownian Bridge, and accept ω with probability relative to Equation (4.5). Unfortunately, we can neither evaluate the path integral in the r.h.s. of Equation (4.5) nor can we sample a continuous path ω , therefore, more specialized methods are needed.

4.3 Simulation using the EA

Previously, we mentioned that there are two problems associated with the estimation of (4.5): the evaluation of the path integral and sampling a continuous path ω . EA1 does not attempt to evaluate the path integral but it re-expresses the acceptance probability of a path as the probability of a “special event” of a Poisson marked process. In addition, the simulation of the continuous path ω_s is avoided completely by employing the idea of *retrospective sampling*: the Poisson Marked Process is simulated first and then a sample of (4.1) at discrete times can be obtained from a Brownian Bridge. Once a finite sample, a *skeleton*, has been obtained, it can be “filled-in” using Brownian Bridges in order to obtain a sample with an arbitrary number of observations which is also exact. We continue by presenting these two ideas in more detail and we assume that the following conditions on the drift term $\alpha \{X_s; \mathbf{c}\}$ hold:

1. $\alpha \{ \cdot \}$ is everywhere differentiable. An essential property in order to proceed with Girsanov formula.

2. The *Novikov condition* (Øksendal 2005):

$$\mathbb{E} \left[\exp \left(\frac{1}{2} \int_0^t \alpha \{X_s; \mathbf{c}\}^2 ds \right) \right] < \infty,$$

holds which is sufficient to guarantee that the Girsanov formula (4.3) is a martingale w.r.t. Wiener measure. It follows from the Girsanov Theorem that (4.3) exists and can be used as an acceptance criterion.

3. $\alpha \{\cdot\}^2 + \alpha \{\cdot\}'$ is bounded below, otherwise identifiability issues arise and (4.3) would not be finite.

The use of a marked Poisson process to re-express (4.5) seems counter-intuitive but it is completely legitimate. Let N be the number of points below the graph $t \rightarrow \phi(\omega_t; \mathbf{c})$, $k \sim \text{Poisson}(r(\omega; \mathbf{c})t)$, $\Psi : \Psi_1, \dots, \Psi_k \sim \text{Uniform}(0, t)$, $Y_1, \dots, Y_k \sim \text{Uniform}(0, 1)$, hence (Ψ, \mathbf{Y}) is a marked Poisson process. The probability that all points of the marked Poisson process (Ψ, \mathbf{Y}) lie above the graph $\phi(\omega_t; \mathbf{c})$ is derived in the following steps:

$$\begin{aligned} P(N = 0) &= \sum_{k=0}^{\infty} P(N = 0 | K = k) P(K = k) \\ &= \sum_{k=0}^{\infty} \int \dots \int P(Y_1 > \phi(\omega_{(\Psi_1)}; \mathbf{c}), \dots, Y_k > \phi(\omega_{(\Psi_k)}; \mathbf{c}) | \Psi) P(\Psi) d\Psi P(K = k) \\ &= \sum_{k=0}^{\infty} \int \dots \int P(Y_1 > \phi(\omega_{(\Psi_1)}; \mathbf{c} | \Psi_1)) \dots P(Y_k > \phi(\omega_{(\Psi_k)}; \mathbf{c} | \Psi_k)) P(\Psi) d\Psi P(K = k) \\ &= \sum_{k=0}^{\infty} \prod_{i=1}^k \int_0^t \left(1 - \int_0^{\phi(\omega_{(\Psi_i)}; \theta)} dY_i \right) \frac{1}{t} d\Psi_i P(K = k) \\ &= \sum_{k=0}^{\infty} \left[\int_0^t (1 - \phi(\omega_{(\Psi_1)}; \theta)) \frac{1}{t} d\Psi_1 \right] \dots \left[\int_0^t (1 - \phi(\omega_{(\Psi_k)}; \theta)) \frac{1}{t} d\Psi_k \right] P(K = k) \\ &= \sum_{k=0}^{\infty} \left[1 - \frac{1}{t} \int_0^t \phi(\omega_s; \mathbf{c}) ds \right]^k e^{-r(\omega; \theta)t} \frac{(r(\omega; \mathbf{c})t)^k}{k!} \\ &= \exp[-r(\omega; \mathbf{c})t] \exp \left[r(\omega; \mathbf{c})t \left(1 - \frac{1}{t} \int_0^t \phi(\omega_s; \mathbf{c}) ds \right) \right] \\ &= \exp \left[-r(\omega; \mathbf{c}) \int_0^t \phi(\omega_s; \mathbf{c}) ds \right] \end{aligned} \tag{4.7}$$

where $d\Psi \equiv d\Psi_1 \dots d\Psi_k$, $P(Y_i > \phi(\omega_{(\Psi_i)}; \mathbf{c}) | \Psi_i) \times P(\Psi_i)$ for any $1 \leq i \leq k$ is equal to the complement of the normalized area under $\phi(\omega_s; \mathbf{c})$ graph. It should be stressed that the total range of the uniform (Ψ_i, Y_i) variables spawns the area of the rectangle $t \times 1$. Therefore, we

only need to ensure that all points are accepted with probability $\phi(\omega_{\Psi_j}; \mathbf{c})$, or equivalently, all marks \mathbf{Y} are above the graph of $t \rightarrow \phi(\omega_t; \mathbf{c})$ by introducing the indicator function:

$$I(\mathbf{c}, \Phi, \omega) := \prod_{j=1}^k \mathcal{I}[\phi(\omega_{\Psi_j}; \mathbf{c}) < v_j] \quad (4.8)$$

which also denotes the acceptance or rejection of the skeleton.

A simple rejection sampler based on (4.8) would require first the full path $\omega_s, s \in [0, t]$ of a Brownian Bridge and then proceed with the realization of the marked Poisson process with rate $r(\omega; \mathbf{c})$. Beskos et al. (2006) employed the idea of retrospective sampling (Papaspiliopoulos and Roberts 2008): if the rate $r(\omega; \mathbf{c}) = r(\mathbf{c})$, i.e. does not depend from ω , we can simulate the Poisson process first and then proceed with the simulation of the finite sample of $(\omega_{\Psi_1}, \dots, \omega_{\Psi_k})$ which will be used at the indicator function I . Putting everything together the Exact Algorithm can be described by the following pseudo code:

Algorithm 2 Exact Algorithm 1 (EA1).

1. Initiate a path $k(\omega)$ of a biased Brownian motion with initial point $\omega_0 = x$ and endpoint $\omega_t = y$.
 2. Generate a realization of the marked Poisson process $\Phi = \{\Psi, \Upsilon\}$ of rate $r\{k(\omega), \mathbf{c}\}$, $\Psi : \Psi_1, \dots, \Psi_k \sim \text{Uniform}(0, t)$, $\Upsilon : \Upsilon_1, \dots, \Upsilon_k \sim \text{Uniform}(0, 1)$, $k \sim \text{Poisson}(r(\omega; \mathbf{c})t)$.
 3. Simulate the skeleton $\{\omega_{(\Psi_1)}, \dots, \omega_{(\Psi_k)}\}$ conditionally on $k(\omega)$.
 4. compute the acceptance indicator I (4.8).
 5. If $I = 1$ then return $k(\omega)$ and $\text{Skel}(\omega) = \{(x, 0), (\omega_{(\Psi_1)}, \Psi_1), \dots, (\omega_{(\Psi_k)}, \Psi_k), (y, t)\}$ else return to 1.
-

4.3.1 Extensions

All extensions of the EA assume that the conditions 1–3 given at the beginning of the chapter hold. When the diffusion process of interest is conditioned only on the starting point and the ending point is left unspecified, the Exact Algorithm cannot be applied directly. In order to proceed, a variation of EA has been suggested, the *unconditional* EA, which has an extra step that involves the simulation of an ending point. After an ending point has been obtained the EA can be applied directly. If we assume that $h(u) = \exp\{A(u) - (u - x_i)^2/(2t)\}$, $A(u) =$

$\int_0^u \alpha \{z\} dz$, is integrable, then $h(\cdot)$ is the unnormalized conditional density of the endpoint (X_{T+t}) of a diffusion conditioned at (X_T) . Knowing $h(\cdot)$ we can obtain an ending point X_{T+t} by rejection sampling.

Beskos and Roberts (2005) proposed the first variation of the exact algorithm where the decision of acceptance or rejection of the current skeleton is not reached immediately. The decision is reached only when a realization of a sequence of special events have occurred, otherwise the problem is undecidable and continues to iterate.

We saw that EA1 poses some restrictions on the rate of the Poisson process since it assumes that does not depend on the path of the Brownian Bridge, i.e. $r(\omega; \mathbf{c}) = r(\mathbf{c})$. EA2 (Beskos et al. 2006) relaxes this condition by assuming that either:

$$\limsup_{u \rightarrow \infty} \{\alpha \{u; \mathbf{c}\}^2 + \alpha \{u; \mathbf{c}'\}\} < \infty, \text{ or, } \limsup_{u \rightarrow -\infty} \{\alpha \{u; \mathbf{c}\}^2 + \alpha \{u; \mathbf{c}'\}\} < \infty$$

depends on the maximum (or minimum) value of the path ω . Given the starting, the ending and an extreme (maximum or minimum) point the rest of the path can be proposed using Bessel Bridges (Beskos and Roberts 2005).

Beskos et al. (2008) proposed a variation of exact algorithm (EA3) which does not put any assumptions on the $r(\omega; \mathbf{c})$ rate. The main idea is to randomly select the range of the path ω and conditional on the range, an extreme point of the path (minimum or maximum with probability 1/2) is simulated.

Nevertheless, the output of EA is a skeleton, a finite set of points, which can be used to obtain finer trajectory samples by interpolating between the accepted points either with Brownian Bridges (EA1), Bessel Bridges (EA2) or Layered Brownian Bridges (EA3). The finer sample is still an exact sample of the unit-variance process (4.1) which can be back-transformed using $\eta^{-1}(\cdot)$, the inverse of (4.2), to the original process.

4.4 Transition densities estimator

In this Section we will review the inferential capabilities of the EA algorithm. As previously mentioned, the Likelihood function can be written as the product of the transition densities (4.4). Specifically, we will focus on the work of Beskos et al. (2006) which proposed unbiased estimators for the transition densities of SDEs, but for convenience, we will only consider SDEs that satisfy the EA1 assumptions.

We begin with a very important result by Dacunha-Castelle and Florens-Zmirou (1986) which relates the transition density of the unit-variance SDE (4.1) with the transition density of the Brownian Bridge:

$$p_t(x, y; \mathbf{c}) = \mathcal{N}_t(y - x) \mathbb{E}_{W^{(t,x,y)}} \left[\exp \left\{ A(y; \mathbf{c}) - A(x; \mathbf{c}) - \int_0^t \frac{1}{2} \alpha \{ \omega_s; \mathbf{c} \}^2 + \alpha \{ \omega_s; \mathbf{c} \}' ds \right\} \right]. \quad (4.9)$$

Beskos et al. (2006) propose two estimators based on (4.9), namely the *Acceptance method* and the *Poisson estimator*. In the next section we will go through the details of the Acceptance Method, assuming that the SDE satisfies the conditions of the EA1. Its extension to more general EA settings can be found in Beskos et al. (2009). Additionally, Beskos et al. (2006) proposed an additional estimator, the *Bridge Estimator* applicable only to diffusions that satisfy the conditions of EA1.

4.4.1 The Acceptance Method

We denote by $\gamma(x, y, \mathbf{c})$ the acceptance probability of a proposed path in the EA algorithm. We can identify $\gamma(x, y, \mathbf{c})$ as the proportional quantity of the Radon-Nikodym derivative (4.5). Therefore by integrating w.r.t. $\mathcal{W}^{(t,x,y)}$, we get the following expression:

$$\gamma(x, y, \mathbf{c}) = \mathbb{E}_{\mathcal{W}^{(t,x,y)}} \left[\exp \left\{ -r(\omega, \mathbf{c}) \int_0^t \phi(\omega_s; \mathbf{c}) ds \right\} \right], \quad (4.10)$$

and (4.9) can be rewritten in terms of $\gamma(x, y, \mathbf{c})$:

$$p_t(x, y; \mathbf{c}) = \mathcal{N}_t(y - x) \exp\{A(y; \mathbf{c}) - A(x; \mathbf{c}) - l(\mathbf{c})t\} \gamma(x, y, \mathbf{c}). \quad (4.11)$$

Additionally we can express the acceptance probability as an expectation of the indicator function (4.8):

$$\gamma(x, y, \mathbf{c}) = \mathbb{E}[I(\mathbf{c}, \Phi, \omega)]. \quad (4.12)$$

Therefore, the proportion of accepted skeletons in a finite number of EA simulation attempts, denoted with $\hat{\gamma}(x, y, \mathbf{c})$, is a Monte-Carlo estimator of (4.12). Consequently, $\hat{\gamma}(x, y, \mathbf{c})$ can be used in (4.11) to obtain an estimator for $p_t(x, y; \mathbf{c})$, referred to as the *Acceptance Method* by Beskos et al. (2006).

4.4.1.1 Simultaneous Acceptance Method

The authors identified that the estimator $\hat{\gamma}(x, y, \mathbf{c})$ depends on the choice of \mathbf{c} and for different values of \mathbf{c} , say \mathbf{c}' , new simulations of $\hat{\gamma}(x, y, \mathbf{c}')$ are needed. They proposed a new estimator, the *Simultaneous Acceptance Method* (SAM), that explores the whole parameter space of \mathbf{c} from a common set of simulations. We briefly sketch the ideas of Beskos et al. (2006) that lead to the derivation of SAM for diffusions that satisfy the EA1 assumptions.

In the EA1 setting the indicator function (4.8) is evaluated on the finite sample ω_{Ψ_i} which is simulated from a Brownian Bridge following a $\mathcal{W}^{(x,y,t)}$ distribution. It can be rewritten, using the relocation property (Section 2.1.5), in terms of a standard Brownian Bridge, $\omega \sim \mathcal{W}^{t,0,0}$:

$$I(x, y, \mathbf{c}, \Phi, \omega) = \prod_{j=1}^k \mathcal{I} \left[\phi \left(\omega_{\Psi_j} + \left(1 - \frac{\Psi_j}{t}\right)x + \frac{\Psi_j}{t}y; \mathbf{c} \right) < v_j \right]. \quad (4.13)$$

In the equation above, the parameters vector \mathbf{c} is only required for the simulation of Φ , the marked Poisson process. The authors, suggested that the thinning property can be used to express the indicator function in terms of a marked Poisson process $\Phi_{\max} = \{\Psi_{\max}, \Upsilon_{\max}\}$ which has constant intensity $r_{\max} \geq r(\mathbf{c})$, for all \mathbf{c} . The thinning property suggests that if each point of Φ_{\max} was omitted with probability $1 - r(\mathbf{c})/r_{\max}$ then the remaining points would be realizations of the Poisson process of interest Φ . Concisely, the indicator function (4.13) can be expressed in terms of realizations of Φ_{\max} :

$$I(x, y, \mathbf{c}, \Phi_{\max}, \omega, U) = \prod_{j=1}^k \mathcal{I} \left[\mathcal{I} \left(U_j < \frac{r(\mathbf{c})}{r_{\max}} \right) \phi \left(\omega_{\Psi_j} + \left(1 - \frac{\Psi_j}{t}\right)x + \frac{\Psi_j}{t}y; \mathbf{c} \right) < v_j \right] \quad (4.14)$$

where $U = (U_1, \dots, U_k)$ are *iid* Uniform(0,1) random variables. The indicator (4.14) can either:

- Accept the sample when all points are accepted. A point is *accepted* when:
 - It belongs to Φ Poisson process ($U_j < r(\mathbf{c})/r_{\max}$) and is above the graph ($v_j > \phi(\cdot)$).
 - It does not belong to the Φ process ($U_j > r(\mathbf{c})/r_{\max}$).
- Reject the sample when at least one point belongs to Φ Poisson process ($U_j < r(\mathbf{c})/r_{\max}$) and is under the graph, ($v_j < \phi(\cdot)$).

Since a single realization of Φ_{\max} is adequate to evaluate (4.14) at the full \mathbf{c} range simultaneously, simulation-based estimation of (4.8) becomes more efficient than producing a random

sample for each new candidate vector \mathbf{c}' . We will work towards a Monte Carlo estimator of $\gamma(x, y, \mathbf{c})$ by taking, first, the expectation of (4.14) conditional on (Φ_{\max}, ω) :

$$\begin{aligned}
\mathbb{E}_U [\mathcal{I}(x, y, \mathbf{c}, \Phi_{\max}, \omega, U | \Phi_{\max}, \omega)] &= \prod_{j=1}^k \left\{ P \left[U_j > \frac{r(\mathbf{c})}{r_{\max}} \right] \right. \\
&\quad \left. + P \left[U_j < \frac{r(\mathbf{c})}{r_{\max}}, \phi \left(\omega_{\Psi_j} + \left(1 - \frac{\Psi_j}{t}\right)x + \frac{\Psi_j}{t}y; \mathbf{c} \right) < v_j \right] \right\} \\
&= \prod_{j=1}^k \left\{ 1 - \frac{r(\mathbf{c})}{r_{\max}} + \frac{r(\mathbf{c})}{r_{\max}} \left[1 - \phi \left(\omega_{\Psi_j} + \left(1 - \frac{\Psi_j}{t}\right)x + \frac{\Psi_j}{t}y; \mathbf{c} \right) \right] \right\} \\
&= \prod_{j=1}^k \left\{ 1 - \frac{r(\mathbf{c})}{r_{\max}} \phi \left(\omega_{\Psi_j} + \left(1 - \frac{\Psi_j}{t}\right)x + \frac{\Psi_j}{t}y; \mathbf{c} \right) \right\} \\
&= r_{\max}^{-k} \prod_{j=1}^k \left\{ r_{\max} + \left(l(\mathbf{c}) - \frac{1}{2}(\alpha \{ \omega_{\Psi_j}; \mathbf{c} \} + \alpha \{ \omega_{\Psi_j}; \mathbf{c}' \}) \right) \times \right. \\
&\quad \left. \phi \left(\omega_{\Psi_j} + \left(1 - \frac{\Psi_j}{t}\right)x + \frac{\Psi_j}{t}y; \mathbf{c} \right) \right\}. \tag{4.15}
\end{aligned}$$

We can identify that $\gamma(x, y, \mathbf{c})$ is the expectation of (4.15) w.r.t (Φ_{\max}, ω) . Hence, we can replicate the argument we used in the Monte Carlo estimation of the Acceptance Method: a Monte Carlo estimator of $\gamma(x, y, \mathbf{c})$ is obtained by averaging independent realizations of (4.15), which in turn can be used in (4.11) to estimate the transition density.

4.4.2 Poisson Estimator

The Poisson estimator follows the idea of expressing the expectation of (4.9) w.r.t. a Brownian Bridge as an expectation w.r.t. a Poisson process. Let $f(\cdot)$ any arbitrary function and the finite expectation w.r.t. a diffusion bridge measure $P(t, x)$ needs to be estimated over a path ω :

$$\mathbb{E}_{P(t,x)} \left[\exp \left\{ - \int_0^t f(\omega_s) ds \right\} \right] < \infty$$

let $c \in \mathbb{R}, \lambda > 0$, we derive the following formulation for the above path integral (which proves the result in Beskos et al. 2006):

$$\begin{aligned}
\exp \left\{ - \int_0^t f(\omega_s) ds \right\} &= \exp \left\{ -ct + ct - \int_0^t f(\omega_s) ds \right\} \\
&= \exp \left\{ -ct - \int_0^t \lambda t \frac{f(\omega_s) - c}{\lambda t} ds \right\} \\
&= \exp(-ct) \sum_{k=0}^{\infty} \frac{1}{k!} \left\{ - \int_0^t \lambda t \frac{f(\omega_s) - c}{\lambda t} ds \right\}^k \\
&= \exp(-ct) \sum_{k=0}^{\infty} \frac{e^{-\lambda t} e^{\lambda t}}{k!} (\lambda t)^k \left\{ \int_0^t \frac{c - f(\omega_s)}{\lambda t} ds \right\}^k \\
&= e^{(\lambda-c)t} \sum_{k=0}^{\infty} \frac{e^{-\lambda t} (\lambda t)^k}{k!} \left\{ \int_0^t \frac{c - f(\omega_s)}{\lambda t} ds \right\}^k \\
&= e^{(\lambda-c)t} \mathbb{E}_k \left[\left\{ \int_0^t \frac{c - f(\omega_s)}{\lambda t} ds \right\}^k \right]
\end{aligned}$$

where we can identify $k \sim \text{Poisson}(\lambda t)$. As a side note, if $\Psi \sim \text{Uniform}[0, t]$ then $[f(\omega_\Psi) - c]/\lambda$ is an unbiased estimator of $\int_0^t [f(\omega_s) - c]/\lambda ds$. Consequently, let $\Psi = [\Psi_1, \dots, \Psi_k]$ we have:

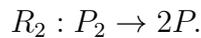
$$\prod_{j=1}^k \left(\frac{c - f(\omega_{\Psi_j})}{\lambda} \right) \text{ is estimating } \left\{ \int_0^t \frac{c - f(\omega_s)}{\lambda t} ds \right\}^k \quad (4.16)$$

which gives the following explicit estimator given an accepted EA skeleton:

$$\exp\{(\lambda - c)t\} \lambda^{-k} \prod_{j=1}^k \left[c - f(\omega_{\Psi_j}) + \left(1 - \frac{\Psi_j}{t}\right) x + \left(\frac{\Psi_j}{t}\right) y \right], \omega \sim \mathcal{W}^{(t,0,0)}. \quad (4.17)$$

4.5 The univariate case

We are considering the application of EA algorithm to diffusion processes motivated by biochemical reactions and more specifically the example in Golightly and Wilkinson (2005), which describes the *dimerization* of two reactants (P, P_2):



The above model is one of the simplest cases involving second order reactions. Let k be a constant representing the maximum possible number of P -molecules. Setting $P_2 = \frac{1}{2}(k - P)$

we obtain the scalar valued SDE:

$$dP_t = [c_2(k - P_t) - c_1P_t(P_t - 1)] dt + \sqrt{2c_1P_t(P_t - 1) + 2c_2(k - P_t)}dW_t \quad (4.18)$$

where $\mathbf{c} = (c_1, c_2)$. We investigate whether the methodology suggested by Beskos et al. (2006) can be applied. Assuming $P_t, c_1, c_2 > 0$ the proposed transformation is the anti-derivative of:

$$\begin{aligned} \eta(P_t, \mathbf{c}) &= Y_t = \int (2c_1P_t(P_t - 1) + 2c_2(k - P_t))^{-1/2} dW_t \\ &= \frac{\sqrt{2}}{2\sqrt{c_1}} \ln \left[\frac{\sqrt{2}(2c_1P - c_1 - c_2)}{2\sqrt{c_1}} + (2c_1P^2 - 2(c_1 + c_2)P + 2c_2k)^{1/2} \right] \end{aligned} \quad (4.19)$$

and the inverse transformation is given by:

$$\eta^{-1}(Y_t, \mathbf{c}) = e^{\sqrt{2}c_1Y} a_1 + a_2 + e^{-\sqrt{2}c_1Y} a_3 \quad (4.20)$$

where (a_1, a_2, a_3) are:

$$\begin{aligned} a_1 &= \frac{\sqrt{2}}{4\sqrt{c_1}} \\ a_2 &= \frac{1 + c_2}{2} \\ a_3 &= \frac{\sqrt{2c_1} + \sqrt{8c_2c_1}^{-1/2} - \sqrt{8c_1}^{-1/2}c_2k + \sqrt{2c_1}^{-5/2}c_2^2}{8}. \end{aligned}$$

A part of the inverse transformation (4.20) can be interpreted as an *hyperbolic* function.

Applying Ito's lemma the transformed process becomes:

$$dY_t = \alpha \{Y_t; \mathbf{c}\} dt + dW_t \quad (4.21)$$

where

$$\alpha \{Y_t; \mathbf{c}\} = \frac{-c_1\eta^{-1}(Y_t, \mathbf{c})^2 + \eta^{-1}(Y_t, \mathbf{c})(-c_2 + c_1) + c_2 k - 2 c_1\eta^{-1}(Y_t, \mathbf{c}) \eta^{-1}(Y_t, \mathbf{c})' - \eta^{-1}(Y_t, \mathbf{c})'(c_1 + c_2)}{\sqrt{2}\sqrt{\eta^{-1}(Y_t, \mathbf{c})^2 c_1 - \eta^{-1}(Y_t, \mathbf{c})(c_1 + c_2) + c_2 k}} \quad (4.22)$$

The drift coefficient of the transformed process (4.22) involves a combination of exponential functions. One of the EA assumptions, regardless of its variation, is the existence of a lower bound for the function:

$$\alpha \{Y_t; \mathbf{c}\}^2 + \alpha \{Y_t; \mathbf{c}\}', \quad (4.23)$$

which, in our setting, is not trivial to check. Equation (4.23) involves the minimization of exponential functions of Y_t and \mathbf{c} due to the inverse transformation (4.20). In addition, the

behavior of the drift term changes drastically from the choice of parameters. In Figure 4.1 we plotted $\alpha\{Y_t; \mathbf{c}\}$ using different values of \mathbf{c} . For $(c_1, c_2) = (0.1, 0.1)$ or $(c_1, c_2) = (0.1, 0.6)$ the drift term seems to have an upper bound, but setting $(c_1, c_2) = (0.6, 0.1)$ or $(c_1, c_2) = (0.6, 0.6)$ the drift term does not have any bounds at all. Nevertheless, we applied a numerical minimization method to (4.23) assuming the constraints $Y_t \in [-4000, 4000]$ and $c_1, c_2 \in (0, 3]$ failed to converge and also produced warnings of numerical issues (undefined values, see also Figure 4.5 for an illustration). Due to the involved numerical instabilities we did not proceed with the implementation of an EA-based simulation.

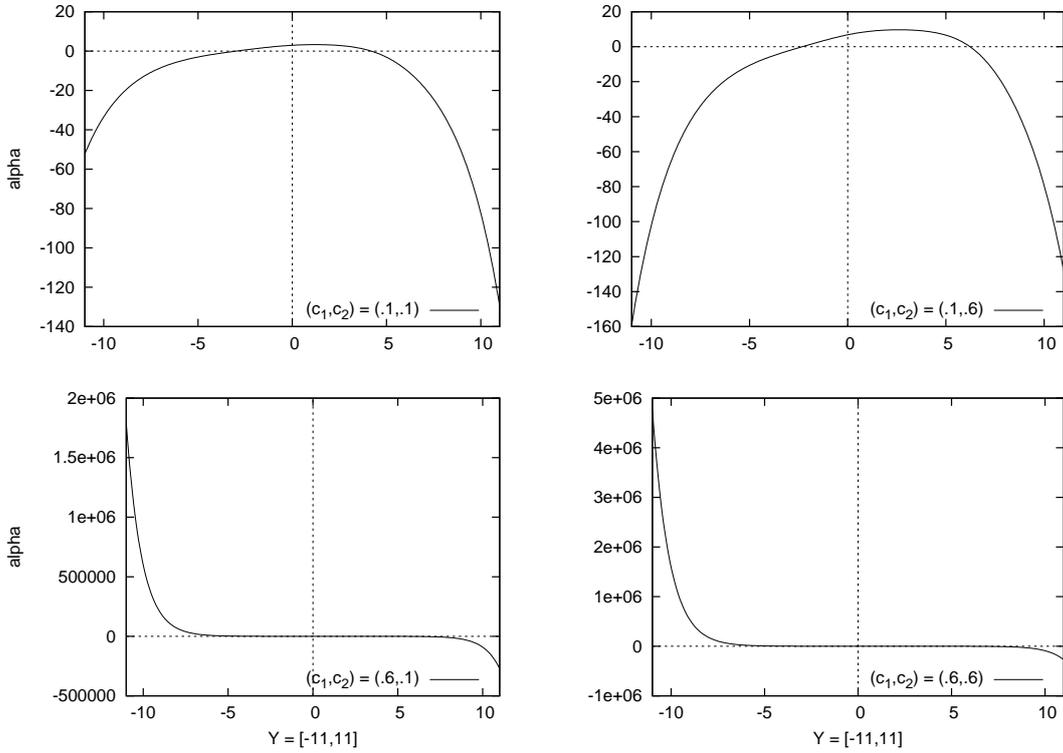


Figure 4.1: The drift term of the transformed dimerization reaction process plotted for parameter values $(c_1, c_2, k) = (0.1, 0.1, 500), (0.1, 0.6, 500), (0.6, 0.1, 500), (0.6, 0.6, 500)$.

4.6 The multivariate case

The multivariate case introduces more difficulties, we specify our original untransformed diffusion process or more generally the stochastic differential equation:

$$dX_t = \mu\{X_t, \mathbf{c}\}dt + \sigma\{X_t, \mathbf{c}\}dW_t, \quad (4.24)$$

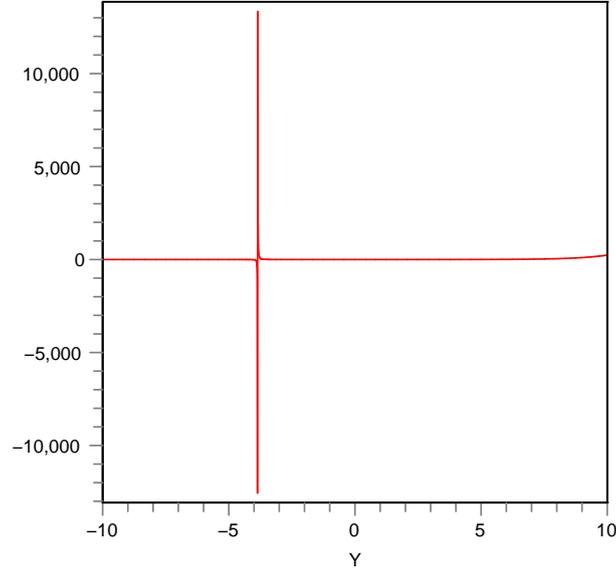


Figure 4.2: Plot of $\alpha \{Y_t; \mathbf{c}\}^2 + \alpha \{Y_t; \mathbf{c}\}'$ with $(c_1, c_2) = (0.085, 10^{-6})$ illustrating numerical stability issues.

where $\mu\{X_t, \mathbf{c}\}$ a $m \times 1$ vector corresponding to the *drift* coefficient and a $m \times n$ matrix $\sigma\{X_t, \mathbf{c}\}$ corresponding to the *diffusion* coefficient. Let $\mathbb{S}_x \subseteq \mathbb{R}^m$ the domain of the diffusion X_t , and as a usual choice we consider the whole \mathbb{R}^m .

We want to find, although is not always possible, a suitable *unit-variance* transformation $g : \mathbb{R}^m \rightarrow \mathbb{R}^m$ which would result an SDE with unit variance-covariance matrix of infinitesimal change:

$$dY_t = \alpha \{Y_t, \mathbf{c}\} dt + \beta \{Y_t, \mathbf{c}\} dW_t, \quad \text{var}(dY_t) = \beta \beta^T = I. \quad (4.25)$$

By applying Ito's lemma at X_t , the variance-covariance matrix associated with a infinitesimal change dY_t of the transformed process Y_t becomes:

$$\text{var}(dY_t) = (J_g \sigma)(J_g \sigma)^T = I, \quad (4.26)$$

where J_g is the *Jacobian matrix* of the transformation w.r.t. the n -th dimensional state variables $X_{t,i}, i = 1, \dots, n$. For convenience, we can write the (4.26) as:

$$A = J_g \sigma, \text{ where the elements given by: } a_{i,j} = \nabla g_i \cdot \sigma_{\cdot j} = \sum_{l=1}^m \frac{\partial g_i}{\partial x_l} \sigma_{lj}, \quad (4.27)$$

$\sigma_{\cdot k}$ corresponds to the k -th row of the original diffusion coefficient (σ), $g_i, i \in (1, \dots, n)$ the coordinate functions of the g mapping and ∇g_i the vector corresponding to the gradient of the g_i function.

Finally by expressing (4.26) in terms of (4.27) we can derive a system of first order non-linear (second degree), partial differential equations for the *unit-variance* transformation $g(\mathbf{x}; \mathbf{c})$:

$$\begin{aligned} \sum_{l=1}^n (a_{il})^2 &= \sum_{l=1}^n (\nabla g_i \cdot \sigma_{\cdot l})^2 = 1, \quad i \in \mathbb{N} \\ \sum_{l=1}^n a_{il} a_{jl} &= \sum_{l=1}^n (\nabla g_i \cdot \sigma_{\cdot l}) (\nabla g_j \cdot \sigma_{\cdot l}) = 0, \end{aligned} \quad (4.28)$$

where (\cdot) the dot product, $l, i, j \in \mathbb{N}$ and $i \neq j$. For illustration consider the SDE:

$$dX_t = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} dt + \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{pmatrix} dW_t,$$

the resulting system of PDEs is:

$$\begin{aligned} 1 &= \left(\sigma_{11} \frac{\partial g_1}{\partial X_{t,1}} + \sigma_{21} \frac{\partial g_1}{\partial X_{t,2}} \right)^2 + \left(\sigma_{11} \frac{\partial g_2}{\partial X_{t,1}} + \sigma_{21} \frac{\partial g_2}{\partial X_{t,2}} \right)^2 \\ 0 &= \left(\sigma_{12} \frac{\partial g_1}{\partial X_{t,1}} + \sigma_{22} \frac{\partial g_1}{\partial X_{t,2}} \right) \left(\sigma_{11} \frac{\partial g_1}{\partial X_{t,1}} + \sigma_{21} \frac{\partial g_1}{\partial X_{t,2}} \right) + \\ &\quad \left(\sigma_{12} \frac{\partial g_2}{\partial X_{t,1}} + \sigma_{22} \frac{\partial g_2}{\partial X_{t,2}} \right) \left(\sigma_{11} \frac{\partial g_2}{\partial X_{t,1}} + \sigma_{21} \frac{\partial g_2}{\partial X_{t,2}} \right) \\ 1 &= \left(\sigma_{12} \frac{\partial g_1}{\partial X_{t,1}} + \sigma_{22} \frac{\partial g_1}{\partial X_{t,2}} \right)^2 + \left(\sigma_{12} \frac{\partial g_2}{\partial X_{t,1}} + \sigma_{22} \frac{\partial g_2}{\partial X_{t,2}} \right)^2 \end{aligned} \quad (4.29)$$

using a short-hand notation:

$$\begin{cases} a_{11}^2 + a_{12}^2 = 1 \\ a_{11}a_{21} + a_{12}a_{22} = 0 \\ a_{21}^2 + a_{22}^2 = 1 \end{cases} \rightarrow \begin{cases} a_{11} = \pm \sqrt{1 - a_{12}^2} \\ a_{11}a_{21} = -a_{12}a_{22} \\ a_{22} = \pm \sqrt{1 - a_{21}^2} \end{cases} \rightarrow \begin{cases} a_{21} \sqrt{1 - a_{12}^2} = -a_{12} \sqrt{1 - a_{12}^2} \\ a_{11}a_{21} = -a_{12}a_{22} \\ a_{21} \sqrt{1 - a_{12}^2} = a_{12} \sqrt{1 - a_{12}^2} \end{cases}$$

using $(\pm\sqrt{1-a_{12}^2}) a_{21} = -a_{12} (\pm\sqrt{1-a_{21}^2})$ reduces to two linear systems:

$$\begin{cases} a_{21} = a_{12} \\ a_{11} = -a_{22} \end{cases} \quad \text{or} \quad \begin{cases} a_{21} = -a_{12} \\ a_{11} = a_{22} \end{cases} \quad \text{and} \quad a_{11}^2 + a_{12}^2 = 1. \quad (4.30)$$

It is very interesting to note that two linear systems in the equations (4.30) are the *Cauchy-Riemann* equations (p. 17 Abramowitz and Stegun 1964) and the non-linear is the *Eikonal* equation (Evans 1998).

4.6.1 Reducibility

Aït-Sahalia (2008) introduces the definition of a reducible diffusion. The concept of reducibility derives from the same idea as the one dimensional analogue: the unit-variance transformation can be found as an integral of the inverse variance term. In the multidimensional setting, further complications are introduced. One can work with the diffusion coefficient, setting the inverse of the diffusion coefficient equal to the Jacobian of the unit-variance transformation:

$$J_g = \sigma^{-1},$$

assuming σ is invertible, which spawns a system of *partial differential equations*. Investigating the *compatibility* (or consistency) of the system of PDEs, the following necessary and sufficient condition, is given:

Aït-Sahalia's proposition: The diffusion Y_t is *reducible* if and only if

$$\sum_{l=1}^m \frac{\partial \sigma_{ik}(x)}{\partial x_l} \sigma_{lj}(x) = \sum_{l=1}^m \frac{\partial \sigma_{ij}(x)}{\partial x_l} \sigma_{lk}(x), \quad (4.31)$$

for each $x \in S_x$, $(i, j, k) = (1 \dots m), k > j$. If σ is *non-singular* the condition can be expressed as:

$$\frac{\partial \sigma_{ij}^{-1}(x)}{\partial x_k} = \frac{\partial \sigma_{ik}^{-1}(x)}{\partial x_j}.$$

If we set $\beta = J_g \sigma = I$, we can derive the relevant system of PDES (4.28):

$$a_{ij} = \sum_{l=1}^m \frac{\partial g_i}{\partial x_l} \sigma_{lj} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}.$$

This proposition enables the closed-form Hermite Series expansion for the transition density of multivariate diffusions processes. Aït-Sahalia (2008) also provides with an alternative expansion for diffusion processes that fail to satisfy the conditions (4.31). The transformation g is known as the *Lamperti transform* and it has been used in other instances e.g. Doss (1977).

4.6.1.1 Reducibility Counterexample

The reducibility condition sets the diffusion coefficient of the transformed process to a unit matrix, i.e. $\beta\{Y_t; \mathbf{c}\} = I$, which sets, trivially, the infinitesimal variance–covariance matrix of dY_t to a unit matrix as well. We claim that EA can be applied to a class of diffusion processes wider than the class of reducible diffusions. To support this claim, we use a counterexample based on (4.28), a less restrictive transformation which allows the diffusion coefficient of the transformed process, $\beta\{Y_t; \mathbf{c}\}$, to be non-constant while the corresponding infinitesimal variance $\text{var}(dY_t)$ is constant (unit matrix).

Our original SDE $X_t = (X_{t,1}, X_{t,2})$ is:

$$dX_t = \alpha \{X_t\} dt + \sigma \{X_t\} W_t,$$

with $\alpha \{X_t\}$ being an arbitrary vector-valued function and the diffusion coefficient given by the matrix:

$$\sigma \{X_t\} = \begin{pmatrix} -A & 0 \\ -A \sin(X_{t,2} - X_{t,1}) & 1 \end{pmatrix},$$

where $A = 1/\cos(X_{t,2} - X_{t,1})$ and we assume that $\cos(X_{t,2} - X_{t,1}) \neq 0$. The inverse of σ^{-1} is:

$$\sigma^{-1} = \begin{pmatrix} -1/A & 0 \\ -\sin(X_{t,2} - X_{t,1}) & 1 \end{pmatrix}.$$

We assume that $\sigma \{X_t\}$ is non-singular, i.e. $A \neq 0$, and we investigate the reducibility conditions 4.6.1. For $i = 2, j = 1, k = 2$, we have:

$$\begin{aligned} \frac{\partial \sigma_{21}^{-1}}{\partial X_{t,2}} &= -\cos(X_{t,2} - X_{t,1}), \\ \frac{\partial \sigma_{22}^{-1}}{\partial X_{t,1}} &= 0, \end{aligned}$$

which are not equal and hence Ait-Sahalia's conditions are not satisfied. On the other hand, if we assume the following transformation:

$$g(x_1, x_2) = (\cos(x_1) - \sin(x_2), \sin(x_1) + \cos(x_2)),$$

then we end up with a transformed SDE $Y = g(X_{t,1}, X_{t,2})$ with the following diffusion coefficient:

$$\beta = \begin{pmatrix} A [\sin(X_{t,2} - X_{t,1}) \cos(X_{t,2}) + \sin(X_{t,1})] & -\cos(X_{t,2}) \\ A [\sin(X_{t,2} - X_{t,1}) \sin(X_{t,2}) - \cos(X_{t,1})] & -\sin(X_{t,2}) \end{pmatrix},$$

which is clearly non-constant. We can show that $\beta\beta^T = I$ but in our case it is easier to show that $B^T = \beta^T\beta = I$. We quickly can go through the trigonometric manipulations, by replacing the $(X_{t,1}, X_{t,2})$ notation with (a, b) :

$$\begin{aligned}
B_{2,2}^T &= \sin^2(b) + \cos^2(b) \\
&= 1, \\
B_{2,1}^T &= A[-\sin(b)(\sin(b)\sin(b-a) - \cos(a)) - \cos(b)(\cos(b)\sin(b-a) + \sin(a))] \\
&= A[-\sin(b-a)(\sin^2(b) + \cos^2(b)) + \sin(b)\cos(a) - \cos(b)\sin(a)] \\
&= 0, \\
B_{1,1}^T &= A^2[(\sin(b)\sin(b-a) - \cos(a))^2 + (\cos(b)\sin(b-a) + \sin(a))^2] \\
&= A^2\sin(b-a)[\sin^2(b)\sin(b-a) - 2\sin(b)\cos(a) + \sin(b-a)\cos^2(b) + 2\cos(b)\sin(a)] + A^2 \\
&= A^2\sin(b-a)[\sin(b-a) - 2(\sin(b)\cos(a) - \cos(b)\sin(a))] + A^2 \\
&= A^2[1 - \sin^2(b-a)] \\
&= 1,
\end{aligned}$$

and since $B_{2,1}^T = B_{1,2}^T$, we end up with the identity matrix, i.e. a transformed process with unit infinitesimal variance. \blacklozenge

4.7 Discussion

The unit variance transformation widens the class of diffusions coefficients. Furthermore a criterion describing that class seems possible, either by looking the consistency of the system of PDEs or with a possible extension of the (exact) differential forms argument of Aït-Sahalia (2008). On the other hand the idea of reducibility derives a linear system of PDEs which is much easier to handle. Overall, we have found that there is a wider class of diffusions than those termed reducible by Aït-Sahalia (2008) which can be transformed to diffusions with unit infinitesimal variance. Nevertheless, a multidimensional extension of the EA for general SDEs seems difficult considering the effort that should be invested in solving the involved non-linear PDEs. Additionally, we saw in §4.5 that an SDEs arising from a very simple biological network introduced numerical obstacles when we attempted to transform it to a unit-variance SDE.

Chapter 5

Scaled Rates Approximation

In this chapter we introduce a reparametrization $\boldsymbol{\vartheta}$ of the stochastic rate constants \mathbf{c} which leads, in Sections 5.2 and 5.3, to two approximations of the discrete–state process $X(t; \boldsymbol{\vartheta})$: as a system of Ordinary Differential Equations (ODEs) and as a system of linear Stochastic Differential Equations (SDEs), called the Linear Noise Approximation (LNA). The LNA was first proposed by Kurtz (1972), who also considered an ODE approximation, using an approximation with respect to the system size. Additionally, in Section 5.4 we give an alternative proof of convergence of $X(t; \boldsymbol{\vartheta})$, the exact process, to the LNA as the system size increases. In the final Section (5.6) we apply the methods of Chapter 3 to two specific biological models to compare their transition distributions.

5.1 Formulation

We denote the discrete–state continuous–time process as $X(t; \mathbf{c})$, to highlight the dependence on the stochastic rates \mathbf{c} . The change of the state ($\Delta X(t; \mathbf{c})$) in an infinitesimal time interval (δ_t) can be expressed as:

$$\Delta X(t; \mathbf{c}) = A^T \boldsymbol{\kappa}(X(t; \mathbf{c}), \mathbf{c}, t) \quad (5.1)$$

where A the net effect matrix and $\boldsymbol{\kappa}(X(t; \mathbf{c}), \mathbf{c}, t) : (\kappa_1(X(t; \mathbf{c}), c_1, t), \dots, \kappa_{N_R}(X(t; \mathbf{c}), c_{N_R}, t))$ is a vector of the random variables. Each r.v. $\kappa_j(X(t; \mathbf{c}), c_j, t)$ expresses the occurrence (1) or not (0) of a particular (j) reaction:

$$\kappa_j(X(t; \mathbf{c}), c_j, t) = \begin{cases} 1, & \text{w. prob. } h_j(X_t, c_j) \delta_t \\ 0, & \text{otherwise.} \end{cases} \quad (5.2)$$

Analogously to the derivation in Section 3.3, the moments of (5.1) become:

$$\mathbb{E}[\Delta X(t; \mathbf{c}) | \mathcal{F}_t] = A^T \mathbb{E}[\boldsymbol{\kappa}(X(t; \mathbf{c}), \mathbf{c}, t) | \mathcal{F}_t] = A^T \mathbf{h}(X(t; \mathbf{c}), \mathbf{c}) \delta_t \quad (5.3)$$

where \mathcal{F}_t is a σ -algebra (informally the history of the process up to time t), $\mathbf{h}(X(t; \mathbf{c}), \mathbf{c}) = \{h_1(\mathbf{X}, c_1), \dots, h_{N_R}(\mathbf{X}, c_{N_R})\}$ the vector of hazard functions. Similarly, we track the variance which coincides with the second moment:

$$\begin{aligned} \mathbb{E}[\Delta X(t; \mathbf{c}) \Delta X(t; \mathbf{c})^T | \mathcal{F}_t] &= A^T (\mathbb{E}[\Delta X(t; \mathbf{c}) \Delta X(t; \mathbf{c})^T | \mathcal{F}_t] - \mathbb{E}[\Delta X(t; \mathbf{c}) | \mathcal{F}_t] \mathbb{E}[\Delta X(t; \mathbf{c})^T | \mathcal{F}_t]) A \\ (\text{Product of means is } O(\delta_t^2)) &= A^T \mathbb{E}[\boldsymbol{\kappa}(X(t; \mathbf{c}), \mathbf{c}, t) \boldsymbol{\kappa}(X(t; \mathbf{c}), \mathbf{c}, t)^T | \mathcal{F}_t] A + O(\delta_t^2) \\ &= A^T \text{diag}\{\mathbf{h}(X(t; \mathbf{c}), \mathbf{c})\} A \delta_t + O(\delta_t^2). \end{aligned} \quad (5.4)$$

where $\text{diag}\{\mathbf{h}(X(t; \mathbf{c}), \mathbf{c})\}$ denotes a diagonal matrix having as main diagonal the vector of hazard rates and for first equality we have used the definition of variance: $\text{Var}(x) = \mathbb{E}[x^2] - (\mathbb{E}[x])^2$.

Without loss of generality, we assume that the hazard rates in $\mathbf{h}(X(t; \mathbf{c}), \mathbf{c})$ are ordered according to their reaction order:

- $N_R^{(0)}$ reactions of order zero $\left(h_1^{(0)}(X(t, \mathbf{c}), c_1), \dots, h_{N_R^{(0)}}^{(0)}(X(t, \mathbf{c}), c_{N_R^{(0)}}) \right)^T$, with hazard function: $h_j^{(0)}(X(t, \mathbf{c}), c_j) = c_j^{(0)}$.
- $N_R^{(1)}$ reactions of first order $\left(h_1^{(1)}(X(t, \mathbf{c}), c_1), \dots, h_{N_R^{(1)}}^{(1)}(X(t, \mathbf{c}), c_{N_R^{(1)}}) \right)^T$, with hazard function: $h_j^{(1)}(X(t, \mathbf{c}), c_j) = c_j^{(1)} X_j^{(1)}$. Where $X_j^{(1)}$ is a reactant participating in the j -th first order reaction.
- $N_R^{(2)}$ reactions of second order $\left(h_1^{(2)}(X(t, \mathbf{c}), c_1), \dots, h_{N_R^{(2)}}^{(2)}(X(t, \mathbf{c}), c_{N_R^{(2)}}) \right)^T$, with hazard rate: $h_j^{(2)}(X(t, \mathbf{c}), c_j) = c_j^{(2)} X_j^{(2,1)} X_j^{(2,2)}$. Where $X_j^{(2,1)} X_j^{(2,2)}$ are the two reactants participating in the j -th second order reaction.

We introduce the following family of parameterizations for the rate constants \mathbf{c} which depend on the reaction order:

$$\begin{aligned} \boldsymbol{\vartheta} &= \left(\vartheta_1^{(0)}, \dots, \vartheta_1^{(1)}, \dots, \vartheta_1^{(2)}, \dots \right)^T \\ &= \left(\frac{c_1^{(0)}}{N_T}, \dots, c_1^{(1)}, \dots, c_1^{(2)} N_T, \dots \right)^T, \end{aligned} \quad (5.5)$$

or alternatively $\vartheta_j = c_j N_T^{r-1}$, where r is the order of the reaction and N_T is some measure of the size of the system, e.g. the total number of molecules in the system at $t = 0$ or the volume

of the system. In the next sections we are interested in the behaviour of the system when both the number of molecules and N_T tend to infinity but their ratio, e.g. their concentration remains fixed which resembles the idea of the thermodynamic limit in statistical mechanics. Assuming that we have elementary reactions with polynomial rates, the parametrization (5.5) satisfies the following relation:

$$h_j(X_t, \mathbf{c}) = h_j(X_t/N_T, \boldsymbol{\vartheta}) N_T. \quad (5.6)$$

The reparametrization can be extended to incorporate more general reaction rates as long as (5.6) is satisfied. Kurtz (1972) proposes a similar relation to (5.6) which keeps the same stochastic rate constants on both sides.

5.2 Ordinary Differential Equations

The moments (5.3, 5.4) can be rewritten in terms of $\boldsymbol{\vartheta}$:

$$\begin{aligned} \mathbb{E} \left[\frac{\Delta X(t; \mathbf{c})}{N_T} \middle| \mathcal{F}_t \right] &= A^T \left(\frac{c_1^{(0)}}{N_T}, \dots, c_1^{(1)} \frac{X_1^{(1)}}{N_T}, \dots, \frac{c_1^{(2)}}{N_T} X_1^{(2,1)} X_1^{(2,2)}, \dots \right)^T \delta_t \\ &= A^T \left(\vartheta_1^{(0)}, \dots, \vartheta_1^{(1)} \frac{X_1^{(1)}}{N_T}, \dots, \vartheta_1^{(2)} \frac{X_1^{(2,1)}}{N_T} \frac{X_1^{(2,2)}}{N_T}, \dots \right)^T \delta_t \\ &= A^T \mathbf{h}(X(t; \boldsymbol{\vartheta})/N_T, \boldsymbol{\vartheta}) \delta_t. \end{aligned} \quad (5.7)$$

Similarly:

$$\mathbb{E} \left[\left(\frac{\Delta X(t; \mathbf{c})}{N_T} \right) \left(\frac{\Delta X(t; \mathbf{c})}{N_T} \right)^T \middle| \mathcal{F}_t \right] = \frac{1}{N_T} A^T \text{diag} \{ \mathbf{h}(X(t; \boldsymbol{\vartheta})/N_T, \boldsymbol{\vartheta}) \} A \delta_t. \quad (5.8)$$

We note that in the Equation (5.8) we have omitted the terms of order $O(\delta_t^2)$. We also derive the third moment, similar to (3.9), which is a multidimensional array of three indexes consisting of the following elements:

$$\begin{aligned} \mu'_3(\Delta X(t; \mathbf{c})/N_T)_{i,j,k} &= \mathbb{E} \left[\frac{(\Delta X(t; \mathbf{c}))_i (\Delta X(t; \mathbf{c}))_j (\Delta X(t; \mathbf{c}))_k}{N^3} \middle| \mathcal{F}_t \right] \\ &= \sum_{z=1}^{N_R} A_{zi} A_{zj} A_{zk} h(X(t; \boldsymbol{\vartheta}_z)/N_T, \boldsymbol{\vartheta}_z) \frac{\delta_t}{N_T^2} + O(\delta_t^2). \end{aligned} \quad (5.9)$$

As $N_T \rightarrow \infty$ the moments (5.8) and (5.9) approach zero, and

$$\mathbf{y}(t; \boldsymbol{\vartheta}) = \lim_{N_T \rightarrow \infty} X(t; \boldsymbol{\vartheta})/N_T < \infty,$$

converges to the following system of ODEs:

$$d\mathbf{y}(t; \boldsymbol{\vartheta}) = A^T \mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \boldsymbol{\vartheta}) dt. \quad (5.10)$$

Kurtz (1972) proves that (5.10) is the ODE limiting solution of $X(t; \boldsymbol{\vartheta})/N_T$ as the volume of the system increases and the concentrations remain fixed. We henceforth reserve the notation $\mathbf{y}(t; \boldsymbol{\vartheta})$ to denote the solution of this system of ODEs and to distinguish between the ODE $\mathbf{y}(t; \boldsymbol{\vartheta})$ and the scaled process $X(t; \boldsymbol{\vartheta})/N_T$. In addition, we note that (5.7) expressed in terms of $\mathbf{y}(t; \boldsymbol{\vartheta})$ may contain an error $O(\delta_t^2)$ term, since we are approximating a differential function with its differential.

5.3 SDEs

We saw in Section 5.2 that the system dynamics converge to a deterministic model (ODE) as the size of the system increases and the concentrations remain constant. In this section we investigate the fluctuations of the system along the solution proposed by the deterministic model and we compute them to order $N_T^{-1/2}$ relative to the system size. We introduce a new process \hat{M}_t to assess the discrepancy between the stochastic and deterministic models:

$$\hat{M}_t = \sqrt{N_T} \left(\frac{X(t; \boldsymbol{\vartheta})}{N_T} - \mathbf{y}(t; \boldsymbol{\vartheta}) \right), \quad (5.11)$$

and proceed by finding the moments:

$$\begin{aligned} \mathbb{E} \left[\Delta \hat{M}(t; \boldsymbol{\vartheta}) \middle| \mathcal{F}_t \right] &= \mathbb{E} \left[\sqrt{N_T} \left(\frac{\Delta X(t; \boldsymbol{\vartheta})}{N_T} - \Delta \mathbf{y}(t; \boldsymbol{\vartheta}) \right) \middle| \mathcal{F}_t \right] \\ &= \sqrt{N_T} \mathbb{E} \left[\frac{\Delta X(t; \boldsymbol{\vartheta})}{N_T} \middle| \mathcal{F}_t \right] - \sqrt{N_T} A^T \mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \boldsymbol{\vartheta}) \delta_t + O(\delta_t^2) \\ &= \sqrt{N_T} A^T \left(\mathbf{h}(X(t; \boldsymbol{\vartheta})/N_T, \boldsymbol{\vartheta}) - \mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \boldsymbol{\vartheta}) \right) \delta_t + O(\delta_t^2) \\ &= \sqrt{N_T} A^T \left(\vartheta_1^{(0)} - \vartheta_1^{(0)}, \dots, \vartheta_1^{(1)} \left(\frac{X_1^{(1)}}{N_T} - y_1^{(1)} \right), \dots + O(\delta_t^2) \right. \\ &\quad \left. \dots, \vartheta_1^{(2)} \left(\frac{X_1^{(2,1)} X_1^{(2,2)}}{N_T^2} - y_1^{(2,1)} y_1^{(2,2)} \right), \dots \right)^T \delta_t + O(\delta_t^2) \\ &= A^T \left(0, \dots, \vartheta_1^{(1)} \hat{M}_1^{(1)}, \dots + O(\delta_t^2) \right. \\ &\quad \left. \dots, \vartheta_1^{(2)} \left(\frac{\hat{M}_1^{(2,1)} \hat{M}_1^{(2,2)}}{\sqrt{N_T}} + \hat{M}_1^{(2,1)} y_1^{(2,2)} + y_1^{(2,1)} \hat{M}_1^{(2,2)} \right), \dots \right)^T \delta_t + O(\delta_t^2), \end{aligned}$$

the second moment:

$$\begin{aligned}
\mathbb{E} \left[\Delta \hat{M}(t; \boldsymbol{\vartheta})(\Delta \hat{M}(t; \boldsymbol{\vartheta}))^T \middle| \mathcal{F}_t \right] &= \mathbb{E} \left[(\Delta X(t; \boldsymbol{\vartheta})/N_T - \Delta \mathbf{y}(t; \boldsymbol{\vartheta})) (\Delta X(t; \boldsymbol{\vartheta})/N_T - \Delta \mathbf{y}(t; \boldsymbol{\vartheta}))^T \middle| \mathcal{F}_t \right] N_T \\
&= \mathbb{E} \left[\Delta X(t; \boldsymbol{\vartheta})/N_T (\Delta X(t; \boldsymbol{\vartheta})/N_T)^T \middle| \mathcal{F}_t \right] N_T + O(\delta_t^2) \\
&= \frac{1}{N_T} A^T \mathbf{h} (X(t; \boldsymbol{\vartheta})/N_T, \boldsymbol{\vartheta}) A N_T \delta_t + O(\delta_t^2) \\
&= A^T \text{diag} \left\{ \vartheta_1^{(0)}, \dots, \vartheta_1^{(1)} \left(\frac{\hat{M}_1^{(1)}}{\sqrt{N_T}} + y_1 \right), \dots \right. \\
&\quad \left. \dots, \vartheta_1^{(2)} \left(\frac{\hat{M}_1^{(2,1)}}{\sqrt{N_T}} + y_1^{(2,1)} \right) \left(\frac{\hat{M}_1^{(2,2)}}{\sqrt{N_T}} + y_1^{(2,2)} \right), \dots \right\} A \delta_t + O(\delta_t^2).
\end{aligned}$$

As $N_T \rightarrow \infty$ the terms:

$$\left(\frac{\hat{M}_1^{(2,2)}}{\sqrt{N_T}}, \frac{\hat{M}_1^{(2,1)} \hat{M}_1^{(2,2)}}{N_T}, \frac{\hat{M}_1^{(1)}}{\sqrt{N_T}} \right) \rightarrow (0, 0, 0),$$

and the first two moments become:

$$\begin{aligned}
\mathbb{E} \left[\Delta \hat{M}(t; \boldsymbol{\vartheta}) \middle| \mathcal{F}_t \right] &= \alpha \left\{ \hat{M}(t; \boldsymbol{\vartheta}), \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \right\} \delta_t \\
&= A^T \left(0, \dots, \vartheta_1^{(1)} \hat{M}_1^{(1)}, \dots \right. \\
&\quad \left. \dots, \vartheta_1^{(2)} \left(\hat{M}_1^{(2,1)} y_1^{(2,2)} + y_1^{(2,1)} \hat{M}_1^{(2,2)} \right), \dots \right)^T \delta_t, \quad (5.12)
\end{aligned}$$

$$\begin{aligned}
\mathbb{E} \left[\Delta \hat{M}(t; \boldsymbol{\vartheta})(\Delta \hat{M}(t; \boldsymbol{\vartheta}))^T \middle| \mathcal{F}_t \right] &= \beta \left\{ \hat{M}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \right\} \delta_t = \sigma \left\{ \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \right\}^T \sigma \left\{ \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \right\} \delta_t \\
&= A^T \text{diag} \left\{ \vartheta_0, \dots, \vartheta_1^{(1)} y_1^{(1)}, \dots \right. \\
&\quad \left. \dots, \vartheta_1^{(2)} y_1^{(2,1)} y_1^{(2,2)}, \dots \right\} A \delta_t. \quad (5.13)
\end{aligned}$$

We note that we have omitted the $O(\delta_t^2)$ terms from the last two moments. In the next Section we will show that these correspond to the drift, $\alpha \{M(t; \boldsymbol{\vartheta}), \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta}\}$, and diffusion, $\sigma \{ \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \}$, coefficients respectively of a diffusion process:

$$dM_t = \alpha \{M(t; \boldsymbol{\vartheta}), \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta}\} dt + \sigma \{ \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \} dW_t. \quad (5.14)$$

Specifically, we identify (5.14) as a system of *linear SDEs in the narrow-sense* (see Equation 2.12), i.e. $\alpha \{M(t; \boldsymbol{\vartheta}), \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta}\}$ depends linearly on $M(t; \boldsymbol{\vartheta})$ while $\sigma \{ \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \}$ does not depend at on $M(t; \boldsymbol{\vartheta})$ all. Linear SDEs have known solutions which facilitate the evaluation of the transition densities as we shall see in Section 5.5.

5.4 Convergence

The convergence of a Markov process to SDEs is usually motivated either by expressing it as a limiting process and employ a Central limit theorem (Kurtz and Protter 1991) or by

exploiting the characterisation of the Markov process as a solution of the Martingale Problem. In the next subsections we give an overview of the convergence of the $\sqrt{N_T}(X(t)/N - \mathbf{y}_t)$ to an SDE $M(t)$ using the limiting process approach (Section 5.4.1) and the Martingale Problem characterisation (Section 5.4.2). We also assume that $\beta\{\cdot; \boldsymbol{\vartheta}\}$ is of full rank, i.e. there are no zero-eigenvalues. This often is not an issue because we can modify the kinetic model in order to re-express the conservation laws¹ to obtain a full rank matrix.

5.4.1 Limiting Process

We will give an overview of the convergence proof from Chapter 11 of Ethier and Kurtz (1986) adapted to our case. The kinetic parameters are re-parametrised as $\boldsymbol{\alpha} = \mathbf{c}/N_T$ and the transition intensities becomes:

$$q_{k,k+l} = \sum_{m_j - n_j = l} N_T^{-|m_j|+1} \alpha_j \prod_{i=1}^{N_S} \binom{\{\mathbf{X}_i\}}{m_{ji}} = N_T \left[\sum_{m_j - n_j = l} \frac{\alpha_j}{\prod_{i=1}^{N_S} m_{ij}!} \binom{\{\mathbf{X}_j\}}{m_{ij}} + O(N_T^{-2}) \right]$$

where, in this proof, m_{ij} denotes the stoichiometry coefficient of the j -th reaction i -th species, as in (3.1), and $m_j = (m_{j1}, \dots, m_{jN_S}), n_j = (n_{j1}, \dots, n_{jN_S})$. We note that the intensities can be written as:

$$q_{k,k+l} = N_T \beta_l(\{\mathbf{X}\}/N_T) + O(N_T^{-1})$$

which holds without the error term for reactions with stoichiometry coefficients m_{ij}, n_{ik} either 0 or 1. The authors assume that $\beta_l(\cdot)$ are non-negative functions, $l \in \mathbb{Z}^{N_S}$, defined on a subject $E \subset \mathbb{R}^{N_S}$. Let $E_{N_T} = E \cap \{N_T^{-1}k : k \in \mathbb{Z}^{N_S}\}$ it is required that $x \in E_{N_T}, \beta_l(x) > 0$ which implies $x + N_T^{-1}l \in E_{N_T}$.

For t less than the first infinity of jumps, we can write (Theorem 4.1 in Chapter 6, Ethier and Kurtz 1986) the markov jump process as:

$$X(t) = X(0) + \sum_l l Y_l \left(N_T \int_0^t \beta_l \left(\frac{X(s)}{N_T} \right) ds \right),$$

Y_l are independent standard Poisson processes. Let $F(x) = \sum_l l \beta_l(x)$ and $\hat{X}(t) = N_T^{-1}X(t)$, $\hat{X}(t)$ becomes:

$$\hat{X}(t) = \hat{X}(0) + \sum_l l N_T^{-1} \tilde{Y}_l \left(N_T \int_0^t \beta_l \left(\hat{X}(s) \right) ds \right) + \int_0^t F \left(\hat{X}(s) \right) ds, \quad (5.15)$$

¹See also the discussion at the end of Section 2.1.3

where $\tilde{Y}_l(u) = Y_l(u) - u$ is a compensated Poisson processes, centered at its expectation. As a side note, Ethier and Kurtz (1986) note that $\lim_{N_T \rightarrow \infty} \tilde{Y}_l(N_T u) = 0$, a.s., $u \geq 0$ and $\hat{W}_l = N_T^{-1/2} \tilde{Y}_l(N_T u)$ converges weakly to W_l , the standard Brownian Motion. They prove that \hat{X} converges to the deterministic (ODE) model \mathbf{y}_t using the following theorem (for the proof see §2.1, p. 456 in Ethier and Kurtz 1986):

Theorem 5.4.1. *Suppose that for each compact $K \subset E$*

$$\sum_l |l| \sup_{x \in K} \beta_l(x) < \infty$$

and there exists $M_K > 0$ such that

$$|F(x) - F(y)| \leq M_K |x - y|, x, y \in K.$$

Suppose \hat{X} satisfies (5.15), $\lim_{N_T \rightarrow \infty} \hat{X}(0) = x_0$, and \mathbf{y}_t satisfies

$$\mathbf{y}_t = x_0 + \int_0^t F(\mathbf{y}_s) ds, \quad t \geq 0,$$

assuming also global existence for $\frac{d\mathbf{y}_t}{dt} = F(\mathbf{y}_t)$. Then for every $t \geq 0$,

$$\lim_{N_T \rightarrow \infty} \sup_{s \leq t} |\hat{X}(s) - \mathbf{y}_s| = 0, \quad a.s.$$

They follow the same idea for the SDE approximation: $\hat{M}(t) = \sqrt{N_T} (\hat{X}(t) - \mathbf{y}_t)$, satisfying:

$$\hat{M}(t) = \hat{M}(0) + \sum_l l \hat{W}_l \left(\int_0^t \beta_l(\hat{X}(s)) ds \right) + \int_0^t \sqrt{N_T} [F(\hat{X}(s)) - F(\mathbf{y}_s)] ds. \quad (5.16)$$

The latter is shown to converge to a Gaussian process $M(t)$, which coincides with (5.14):

$$M(t) = M(0) + \sum_l l W_l \left(\int_0^t \beta_l(\mathbf{y}_s) ds \right) + \int_0^t DF(\mathbf{y}_s) M(t) ds, \quad (5.17)$$

where $(DF(x))_{ji} = \frac{\partial F_j(x)}{\partial x_i}$, a $N_R \times N_S$ matrix. The result is derived using the (martingale) Central Limit theorem and also the following theorem (the proof is in §2.3, p.458, Ethier and Kurtz 1986)):

Theorem 5.4.2. *Suppose for each compact $K \subset E$,*

$$\sum_l |l|^2 \sup_{x \in K} \beta_l(x) < \infty,$$

and that the β_l and DF are continuous. Suppose \hat{X} , \mathbf{y}_t , $\hat{M}(t)$, $M(t)$ as above and $\lim_{N_T \rightarrow \infty} \hat{M}(0) = M(0)$, $M(0)$ constant. Then \hat{M} converges weakly to M , the solution of (5.17).

5.4.2 Martingale Problem

In this section we sketch a proof for the same result using the Stroock–Varadhan Theorem 2.1.3 and its extension, Lemma 2.1.1, for $h = 1/N_T \rightarrow 0$. We note that $\beta_l(\cdot) = \sum_j \mathcal{I}(l = A_j) h_j(\cdot, \boldsymbol{\vartheta})$, where $\mathcal{I}(\cdot)$ is the indicator function. The assumptions of the theorems on $\beta_l(\cdot)$, $F(\cdot)$ above hold since we only consider reactions up to order two and $\mathbf{y}_0 = \hat{X}(0)$. For each compact $K \subset E$, we assume $x \in K$ and $|x| \leq R < \infty$ and $N_T > 0$. We use the same notation as above for $\hat{X}(t) = X(t)/N_T$ and $\hat{M}(t) = \sqrt{N_T}(\hat{X}(t) - \mathbf{y}_t)$.

The conditions (a) (b) of Lemma 2.1.1 (Section 2.1.6) for the drift $\alpha \left\{ \hat{M}(t; \boldsymbol{\vartheta}), \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \right\}$ and infinitesimal covariance $\beta \left\{ \hat{M}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta} \right\}$ terms can be derived from (5.12) and (5.13) respectively. We note that we know the transition rates for $\hat{X}(t)$ and $\mathbf{y}(t; \boldsymbol{\vartheta})$ beforehand and the terms δ_t and $O(\delta_t^2)$ are finite due to the assumptions on t . As $h(= N_T^{-1}) \rightarrow 0$ they converge to $\alpha \{M(t; \boldsymbol{\vartheta}), \mathbf{y}(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta}\}$ and infinitesimal covariance $\beta \{M(t; \boldsymbol{\vartheta}); \boldsymbol{\vartheta}\}$ respectively. We also stress that the condition (b) holds for a full rank $\beta \{ \cdot; \boldsymbol{\vartheta} \}$ which often requires some modifications to the kinetic model in order to re-express the conservation laws.

Now for (c), we rearrange the terms in (5.16) and we use the assumptions on $F(\cdot)$:

$$\frac{\hat{M}(t) - \hat{M}(0)}{\sqrt{N_T}} = \sum_l l \frac{1}{\sqrt{N_T}} \hat{W}_l \left(\int_0^t \beta_l(\hat{X}(s)) ds \right) + \int_0^t M_K |\hat{X}(s) - \mathbf{y}_s| ds$$

by applying Gronwall's inequality (Appendix 5, Ethier and Kurtz 1986):

$$\frac{\hat{M}(t) - \hat{M}(0)}{\sqrt{N_T}} \leq \sum_l l \frac{1}{\sqrt{N_T}} \hat{W}_l \left(\int_0^t \beta_l(\hat{X}(s)) ds \right) e^{M_K t}.$$

We can now derive a bound using the rate of the compensated Poisson processes:

$$N_T \int_0^t \beta_l(\hat{X}(s)) ds \leq N_T \bar{\beta}_l t$$

where $\bar{\beta}_l = \sup_{s \leq t} \beta_l(\hat{X}(s)) < \infty$. We can exploit the fact that \hat{W} converges to a standard Brownian motion, we denote with $\mu'_3(\xi_l)$ the third raw moment of Gaussian variable with mean 0 and (bounded) variance $\bar{\beta}_l t$:

$$\lim_{h \rightarrow 0} \gamma_3^h(x) \leq \mu'_3 \left(\sum_l \xi_l e^{t M_K} \right)$$

which satisfies the condition (c).

5.5 Transition Density

In Equation (5.11) we established a relation between the discrete process $X(t; \boldsymbol{\vartheta})$ and the ODE $\mathbf{y}(t; \boldsymbol{\vartheta})$ as well as the SDE $M(t; \boldsymbol{\vartheta})$. We assume that the initial values of $M(t; \boldsymbol{\vartheta})$ are the scaled differences at points of observation. Furthermore, we will be assuming that the observations have no error and in the special case of $\mathbf{y}(t_{\text{obs}}; \boldsymbol{\vartheta}) = X(t_{\text{obs}}; \boldsymbol{\vartheta})/N_{\text{T}}$, $M(t_{\text{obs}}; \boldsymbol{\vartheta})$ is reset to $\mathbf{0}$ at each observation time t_{obs} . Hence, by (5.12) $\mathbb{E}[M(t; \boldsymbol{\vartheta})] = \mathbf{0}$, $\forall t$ and we will refer to this special case as the *restarting method*.

Additionally, $M(t; \boldsymbol{\vartheta})$ is a linear in the narrow-sense SDE and from (2.20) we know that its transition density coincides with a multivariate Normal distribution with mean (\mathbf{m}_t or a vector of zeros) and covariance matrix (S_t), which are tractable up to the numerical solution of a system of ODEs (2.19). The discrete stochastic process $X(t; \boldsymbol{\vartheta})$ has a scaled mean which approximately evolves according to $\mathbf{y}(t; \boldsymbol{\vartheta})$ and the mean of $M(t; \boldsymbol{\vartheta})$, \mathbf{m}_t . Similarly, the scaled covariance of $X(t; \boldsymbol{\vartheta})$ evolves according to the covariance of $M(t; \boldsymbol{\vartheta})$, S_t , as illustrated in Figure 5.1. Therefore, we exploit (5.11) to approximate the mean of the discrete

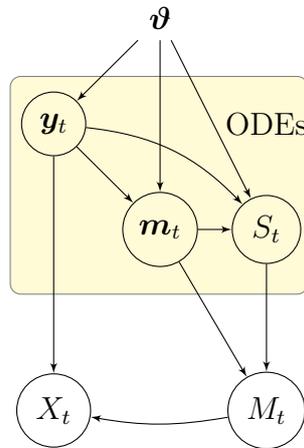


Figure 5.1: Dependence graph between the variables (in circles) and parameters $\boldsymbol{\vartheta}$. The deterministic processes, which are given as solutions of ODEs, are positioned inside the “ODEs” frame.

process $X(t; \boldsymbol{\vartheta})$ with $\mathbf{y}(t; \boldsymbol{\vartheta})$, \mathbf{m}_t and similarly, the covariance matrix of $X(t; \boldsymbol{\vartheta})$ in terms of the covariance of $M(t; \boldsymbol{\vartheta})$. A single simulation of a point $X(t; \boldsymbol{\vartheta})$ given a starting point at $X(0; \boldsymbol{\vartheta})$ would involve the solution of $\mathbf{y}(t; \boldsymbol{\vartheta})$, S_t up to time t , subject to initial conditions $(\mathbf{y}(0; \boldsymbol{\vartheta}), S(0; \boldsymbol{\vartheta})) = (X(0; \boldsymbol{\vartheta})/N_{\text{T}}, 0)$ and the generation of $M(t; \boldsymbol{\vartheta})|M(0; \boldsymbol{\vartheta}) = 0$ from the

transition density $\text{Normal}(\mathbf{0}, S_t)$. Putting everything together we have:

$$X_t|X_0 = x_0 \sim \text{Normal}(\mathbf{y}_t N_T, S_t N_T)$$

and if we drop the condition of the restarting method ($M(0; \boldsymbol{\vartheta}) \neq 0$):

$$X_t|X_0 = x_0 \sim \text{Normal}\left(\mathbf{y}_t N_T + \mathbf{m}_t \sqrt{N_T}, S_t N_T\right),$$

where $\mathbf{y}_t, \mathbf{m}_t, S_t$ are the solutions of the following system of Ordinary Differential Equations:

$$\frac{d\mathbf{y}_t}{dt} = A^T \mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \boldsymbol{\vartheta}), \quad (5.18)$$

$$\frac{d\mathbf{m}_t}{dt} = F \mathbf{m}_t, \quad (5.19)$$

$$\frac{dS_t}{dt} = F S_t + S_t F^T + A^T \text{diag}\{\mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \boldsymbol{\vartheta})\} A, \quad (5.20)$$

and F is the matrix of coefficients which satisfies the equation:

$$F \mathbf{m}_t = A^T \mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \mathbf{m}_t, \boldsymbol{\vartheta}),$$

$\mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \mathbf{m}_t, \boldsymbol{\vartheta})$ is the main diagonal vector of (5.13) or alternatively, we can define it by (Kurtz 1972) taking the partial derivatives w.r.t. \mathbf{m}_t :

$$F = D_{\mathbf{m}_t} A^T \mathbf{h}(\mathbf{y}(t; \boldsymbol{\vartheta}), \mathbf{m}_t, \boldsymbol{\vartheta}).$$

5.6 Case Studies

So far we have introduced both exact and approximate methods in a general context: a class of abstract biological networks. In this Section we use two biological models, the Transcription example and the Lotka–Volterra model which were introduced in Section 3.3.1.1, to illustrate the methods of the Chemical Langevin Equation (CLE), the linear noise approximation (LNA) and the exact Gillespie (SSA²) method. The Chemical Langevin Equation leads to a system of non-linear SDEs which are subsequently solved using a Euler-Maruyama discretization scheme with discretization step $\Delta t = 10^{-3}$, as well as with a single ($\Delta t = t$) step. It should be stressed that although the transition densities that emerge from the LNA and the single-step Euler approximations are both multivariate Normals, the computational cost of calculating their parameters is disproportional. The LNA requires the solution of a system of non-linear ODEs,

²Stochastic Simulation Algorithm.

whereas the parameters of the single-step Euler are readily available, in closed-form, from (2.6). Therefore, the comparison between the LNA and the single-step Euler approximation questions whether the introduction of a computationally more demanding method can be justified.

The LNA method relies on a numerical ODEs solver and we have used the method proposed by Petzold (1983) (refer to Section 2.2 for more details) while specifying the tolerance levels of relative and absolute errors at 10^{-6} . We produced 1000 samples at each of 4 different time points (0.01, 0.1, 1, 10) using 3 different system sizes: (Small, Medium, Large) defined for each system separately at Sections 5.6.1 and 5.6.2. We choose the system configurations as such to keep $\boldsymbol{\vartheta}$ fixed (rather than \mathbf{c}) while varying the size of the system. The last consideration renders the concentration-based solutions of the ODEs (Equation 5.10 and Tables 5.4, 5.1) *independent* of the system size. As a result, the solution of the means for the LNA's transition densities remain unchanged (again in concentrations) irrespectively of the system size. Tables 5.4, 5.1 detail the system of ODEs (5.10, 2.19) which estimate the mean vectors ($\mathbf{y}(t), \mathbf{m}(t)$) and the covariance matrices ($S_{1,2} := \text{Cov}(X_1(t), X_2(t)N_T^{-1})$) of the LNA's transition density. Results are presented in terms of number of molecules $X(t; \mathbf{c})$; this is obtained by multiplying the concentrations by the system size N_T .

We now compare the samples obtained by the approximate methods to the corresponding samples generated by the Gillespie algorithm: marginal comparisons include Q-Q plots and Gaussian kernel density estimates (using Silverman's rule of thumb). In particular, the LNA's actual marginal density was used and vertical bars were preferred to indicate the relative frequencies in samples with a small range of values (< 25), that render smoothing methods misleading. Joint comparisons also were included by plotting the sum of the squares of the log-ratios of the means and standard deviations for each possible size-time combination:

$$\sum_{i=1}^{N_S} \left(\log \left(\frac{\mu_{i,\text{SSA}}}{\mu_{i,\text{CLE}}} \right) \right)^2, \sum_{i=1}^{N_S} \left(\log \left(\frac{\mu_{i,\text{SSA}}}{\mu_{i,\text{LNA}}} \right) \right)^2 \text{ and } \sum_{i=1}^{N_S} \left(\log \left(\frac{\sigma_{i,i,\text{SSA}}}{\sigma_{i,i,\text{CLE}}} \right) \right)^2, \sum_{i=1}^{N_S} \left(\log \left(\frac{\sigma_{i,i,\text{SSA}}}{\sigma_{i,i,\text{LNA}}} \right) \right)^2.$$

Also, a non-parametric multivariate signed rank test (§5.4 Puri and Sen 1971) based on marginal ranks was used to test if the combined observations (LNA-SSA and repeated for CLE-SSA) follow the same distribution for all configurations. It should be noted that if the variables are dependent, the efficiency of the non-parametric test may become poor (for more details see Sections 5.8 and 4.6 of Puri and Sen 1971).

The implementation of the Gillespie algorithm (`gillespie2`) was developed by Darren

$$\begin{array}{ll}
\frac{dy_1}{dt} = \vartheta_1 y_1 - y_1 \vartheta_2 y_2 & \frac{dS_{1,2}}{dt} = S_{1,2} (-\vartheta_3 - \vartheta_2 y_2 + y_1 \vartheta_2 + \vartheta_1) \\
\frac{dy_2}{dt} = y_1 \vartheta_2 y_2 - y_2 \vartheta_3 & -y_1 \vartheta_2 S_{2,2} + S_{1,1} \vartheta_2 y_2 - y_1 \vartheta_2 y_2 \\
\frac{dS_{1,1}}{dt} = 2S_{1,1} (\vartheta_1 - \vartheta_2 y_2) + y_1 \vartheta_2 y_2 - 2y_1 S_{1,2} \vartheta_2 & \frac{dS_{2,2}}{dt} = y_2 \vartheta_3 + 2S_{2,2} (y_1 \vartheta_2 - \vartheta_3) + 2S_{1,2} \vartheta_2 y_2 \\
+ \vartheta_1 y_1 & + y_1 \vartheta_2 y_2
\end{array}$$

Table 5.1: System of ODEs for Lotka-Volterra example.

Wilkinson and Carole Proctor using model specifications in `sbml`. The Euler scheme and LNA method was implemented in the programming language `C` using the numerical library `GSL` (Galassi et al. 2009) and `LSODA` (Petzold 1983) for the numerical solution of ODEs.

5.6.1 Lotka–Volterra

We consider three systems with initial (Predator, Prey) numbers of (30, 50), (300, 500), and (3000, 5000), leading to system sizes (N_T) of (80, 800, 8000) respectively. Among all configurations, we consider fixed scaled rates $\boldsymbol{\vartheta} = (.25, 0.2, 0.125)$ while the corresponding (unscaled) rates \boldsymbol{c} are adjusted according to $(0.25, 0.2/N_T, 0.125)$; as discussed in Section 5.6 this leaves the deterministic process \mathbf{y}_t unaffected by changes in system size.

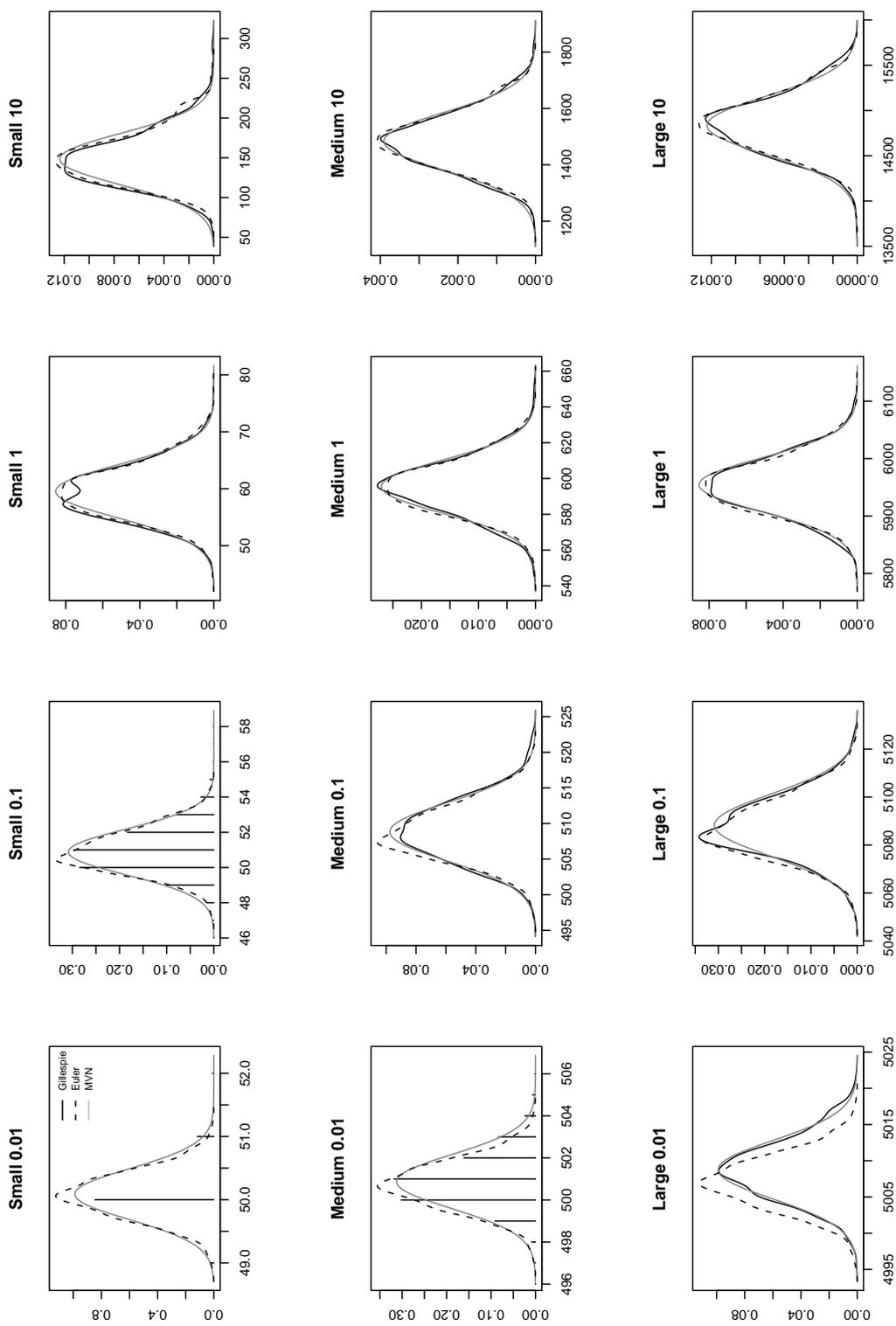


Figure 5.2: Comparing the Prey empirical transition densities – p.m.f. of the Gillespie Method (*solid black*) the Chemical Langevin Equation (*dashed black*) and the Limiting Diffusion approximation (*solid grey*) at four time points ($t = 0.01, 0.1, 1, \text{ and } 10$) using three different system sizes (Small, Medium, Large). Vertical bars indicate the relative frequencies in samples with a small range of discrete values (< 25).

Kernel estimates of the marginal (Prey) transition densities for the three methods (SSA, LNA, CLE) are illustrated in Figure 5.2. The Normal Density of the LNA method provides an adequate approximation to the empirical SSA density or mass function for all configurations. The empirical transition density of the CLE approximation generally provides similar support as the LNA method but with one striking difference: at $t = 0.01$ and for systems of a Large size the CLE’s empirical transition density seems off-centre compared to the LNA’s and SSA’s densities.

The single step Euler–Maruyama approximation was omitted from the plot but the means and the variances of the corresponding samples can be found in Table 5.5. For all $t = 0.01, 0.1$ and 1 , the estimates of the mean are close to the SSA’s but they become worse for $t = 10$. For the Large system at $t = 10$ the transition density of the single-step Euler–Maruyama approximation is concentrated in a completely different range compared to the corresponding range of SSA. The single-step standard deviations provide reasonable approximations to the SSA’s empirical standard deviations up to $t = 0.1$.

Figure 5.3 illustrates the Q–Q plots of the empirical Prey quantiles of the CLE vs SSA and LNA vs SSA samples. The lines at each figure illustrate the empirical quantiles of the SSA samples. The CLE’s empirical transition densities are satisfactory for medium–large configurations with the only exception of the Large–0.01 seconds configuration: the CLE’s quantiles do not follow SSA’s which confirms the “off-centre” observation of the kernel estimates. The exact samples of the small configuration have empirical transition densities concentrated on a few point masses (Figure 5.3) discouraging the application of continuous approximations in small system sizes for simulation purposes. The quantiles of the LNA’s transition density are generally similar to the SSA’s, except the cases of Small and Medium sized systems, for $t = 0.01$, where the discreteness of SSA is most prominent.

The sum of squared log–ratios of means favor the LNA’s estimates over the CLE’s method in the small system configurations but they are equivalent for the rest. Furthermore, the LNA’s estimates consistently (Figure 5.4–b) outperform CLE’s, in terms of the sum of squared log–ratios of standard deviations. It must be noted that the LNA is a (linear) approximation to the CLE and any observed differences can be attributed to the numerical method for the solution of SDEs. In particular, the performance of the Euler–Maruyama method is linked to the choice of the time step, therefore, a choice of a smaller time step is expected to improve the numerical approximations.

Size	Method	Time Points			
		0.01	0.1	1	10
Small	1-Step	50.0875 (0.40)	50.875 (1.27)	58.75 (4.03)	137.5 (12.75)
	SSA	50.1120 (0.42)	50.933 (1.34)	59.38 (4.63)	148.6 (33.03)
Medium	1-Step	500.875 (1.27)	508.75 (4.03)	587.5 (12.75)	1375 (40.31)
	SSA	500.916 (1.28)	509.02 (4.21)	595.4 (15.3)	1486 (101.14)
Large	1-Step	5008.75 (4.03)	5087.5 (12.75)	5875 (40.31)	13750 (127.48)
	SSA	5008.81 (4.13)	5087.8 (12.46)	5954 (47.77)	14851 (321.78)

Table 5.2: Means (s.d.) of the single step Euler approximation and the corresponding Gillespie Samples for Lotka–Volterra Model

Size	Test	Time Points			
		0.01s	0.1s	1s	10s
Small	LNA vs SSA	0.0024	0.5542	0.3013	0.4162
	CLE vs SSA	0.2497	0.9375	0.1800	0.3239
Medium	LNA vs SSA	0.2586	0.6296	0.7054	0.2680
	CLE vs SSA	0.0028	0.3973	0.3024	0.6484
Large	LNA vs SSA	0.1487	0.3245	0.2489	0.1779
	CLE vs SSA	0.0000	0.1542	0.8586	0.0583

Table 5.3: P-values of the multivariate singed rank test based on marginal ranks for the Lotka–Volterra model.

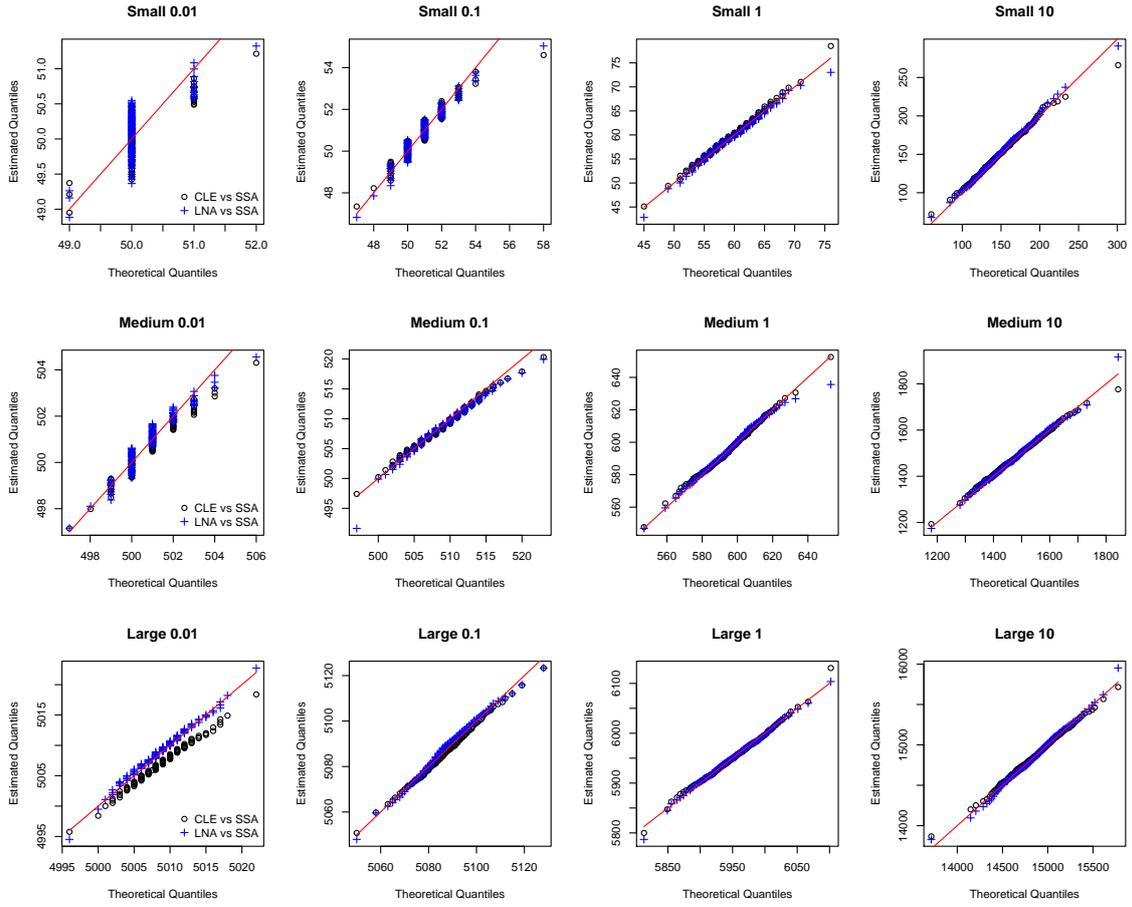


Figure 5.3: Q-Q plots of Prey samples; Chemical Langevin Equation vs Gillespie (*points*) and Linear Noise Approximation vs Gillespie (*crosses*) at at four time points ($t = 0.01, 0.1, 1,$ and 10) using three different system sizes (Small, Medium, Large).

The multivariate signed rank tests (Table 5.3) resulted fewer significant differences (at 95% s.l.) of the LNA samples over the CLE at short (0.01s) transition intervals; although, overall, the performance of the two methods in the non-parametric tests is close for both.

5.6.2 Transcription Example

We now describe the details of the simulations of the Transcription example (3.3.1.2). For the small system we set the initial state to $\{X\} = (8, 8, 8, k - 5, 5)$ and $k = 10$ while the unscaled (c) parameters are set equal to $(3.4, 0.7, 0.35, 0.2, 3.4, 0.9, 0.3, 0.1)$ which matches the parameter choice for c considered by Golightly and Wilkinson (2005). For larger system sizes $i = 10$ and 100 , corresponding respectively to medium and large sizes, the initial number of

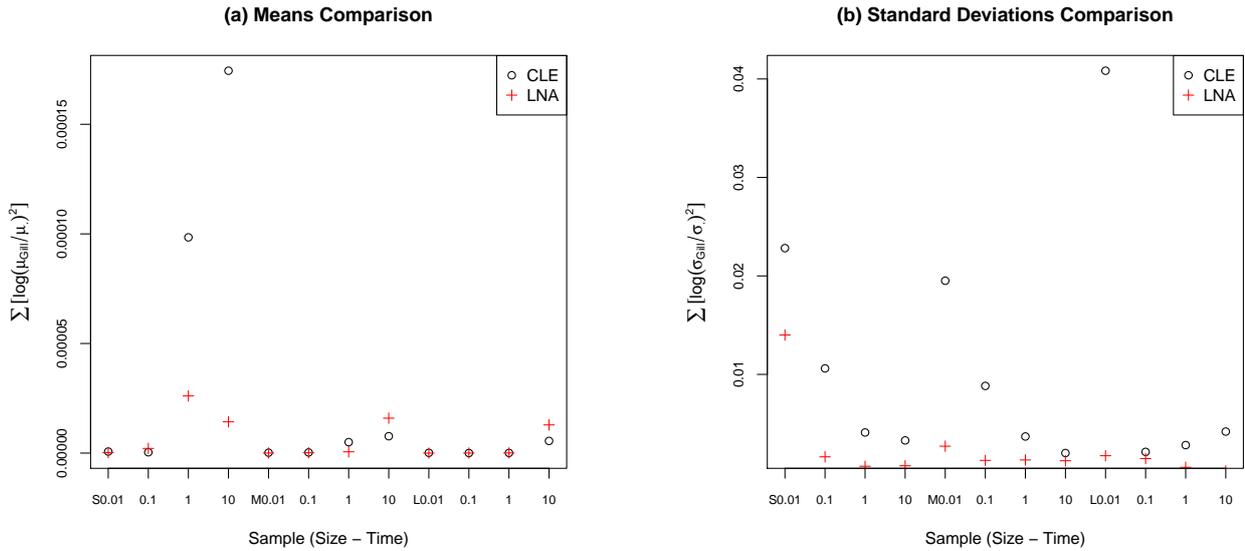


Figure 5.4: Comparisons of Prey means (a) and s.d.s (b) using the sums of the squared log-ratios of the means–s.d. The first four entries at the x -axis correspond to the small system at the time instances $t = 0.01, 0.1, 1, 10$. Similarly the next four correspond to the Medium system and the last four to the Large.

molecules are increased by a factor of i . The molecule-level rate constants \mathbf{c} are adjusted according to the formula $((10i)^{-1}, 0.7, 0.35, 0.2, (10i)^{-1}, 0.9, 0.3, 0.1)$, and the $\boldsymbol{\vartheta}$ remain fixed.

We choose the DNA species to start our marginal comparisons. The empirical transition densities of DNA molecules (Figure 5.5) show that the transition densities of all methods provide similar support. The single step Euler–Maruyama approximation, leads to a Normal distribution which for $t = 1$ and $t = 10$ is outside the range of the previous methods; the mean and variance estimates of the transition densities are provided in Table 5.5.

$$\begin{array}{ll}
\frac{dy_1}{dt} = \vartheta_3 y_4 - y_1 \vartheta_7 & \frac{dS_{2,3}}{dt} = S_{2,3} (-\vartheta_8 - \vartheta_6 - 2y_2 \vartheta_5 - \vartheta_1 y_4) \\
\frac{dy_2}{dt} = -y_2 \vartheta_8 + 2y_3 \vartheta_6 - y_2^2 \vartheta_5 + y_1 \vartheta_4 & \quad + 2S_{3,3} \vartheta_6 - 2y_3 \vartheta_6 + y_2 S_{2,2} \vartheta_5 - y_2^2 \vartheta_5 \\
\frac{dy_3}{dt} = \vartheta_2 \left(\frac{k}{N} - y_4 \right) - y_3 \vartheta_6 + \frac{y_2^2 \vartheta_5}{2} - \vartheta_1 y_3 y_4 & \quad + S_{1,3} \vartheta_4 + S_{2,4} (-\vartheta_1 y_3 - \vartheta_2) \\
\frac{dy_4}{dt} = \vartheta_2 \left(\frac{k}{N} - y_4 \right) - \vartheta_1 y_3 y_4 & \frac{dS_{2,4}}{dt} = S_{2,4} (-\vartheta_8 - 2y_2 \vartheta_5 - \vartheta_1 y_3 - \vartheta_2) \\
\frac{dS_{1,1}}{dt} = -2S_{1,1} \vartheta_7 + y_1 \vartheta_7 + \vartheta_3 y_4 + 2S_{1,4} \vartheta_3 & \quad + 2S_{3,4} \vartheta_6 - \vartheta_1 S_{2,3} y_4 + S_{1,4} \vartheta_4 \\
\frac{dS_{1,2}}{dt} = S_{1,2} (-\vartheta_8 - \vartheta_7 - 2y_2 \vartheta_5) + 2S_{1,3} \vartheta_6 + S_{1,1} \vartheta_4 & \frac{dS_{3,3}}{dt} = \vartheta_2 \left(\frac{k}{N} - y_4 \right) + y_3 \vartheta_6 + \vartheta_1 y_3 y_4 + \frac{y_2^2 \vartheta_5}{2} \\
\quad + S_{2,4} \vartheta_3 & \quad + 2\vartheta_5 y_2 S_{2,3} + 2(-\vartheta_1 y_3 - \vartheta_2) S_{3,4} \\
\frac{dS_{1,3}}{dt} = S_{1,3} (-\vartheta_7 - \vartheta_6 - \vartheta_1 y_4) + S_{1,2} y_2 \vartheta_5 + \vartheta_3 S_{3,4} & \quad + 2S_{3,3} (-\vartheta_6 - \vartheta_1 y_4) \\
\quad + S_{1,4} (-\vartheta_1 y_3 - \vartheta_2) & \frac{dS_{3,4}}{dt} = -S_{3,4} (\vartheta_6 + \vartheta_1 y_4 + \vartheta_1 y_3 + \vartheta_2) \\
\frac{dS_{1,4}}{dt} = S_{1,4} (-\vartheta_7 - \vartheta_1 y_3 - \vartheta_2) + \vartheta_3 S_{4,4} - \vartheta_1 S_{1,3} y_4 & \quad - (\vartheta_1 y_3 + \vartheta_2) S_{4,4} - \vartheta_1 S_{3,3} y_4 + \vartheta_1 y_3 Y_4 \\
\quad + \vartheta_2 \left(\frac{k}{N} - y_4 \right) + y_2 S_{2,4} \vartheta_5 & \\
\frac{dS_{2,2}}{dt} = y_2 \vartheta_8 + 2S_{2,2} (-\vartheta_8 - 2y_2 \vartheta_5) + 4y_3 \vartheta_6 & \frac{dS_{4,4}}{dt} = \vartheta_2 \left(\frac{k}{N} - y_4 \right) + 2(-\vartheta_1 y_3 - \vartheta_2) S_{4,4} \\
\quad + 4S_{2,3} \vartheta_6 + 2y_2^2 \vartheta_5 + 2S_{1,2} \vartheta_4 + y_1 \vartheta_4 & \quad - 2\vartheta_1 S_{3,4} y_4 + \vartheta_1 y_3 y_4
\end{array}$$

Table 5.4: System of ODEs for the Transcription example.

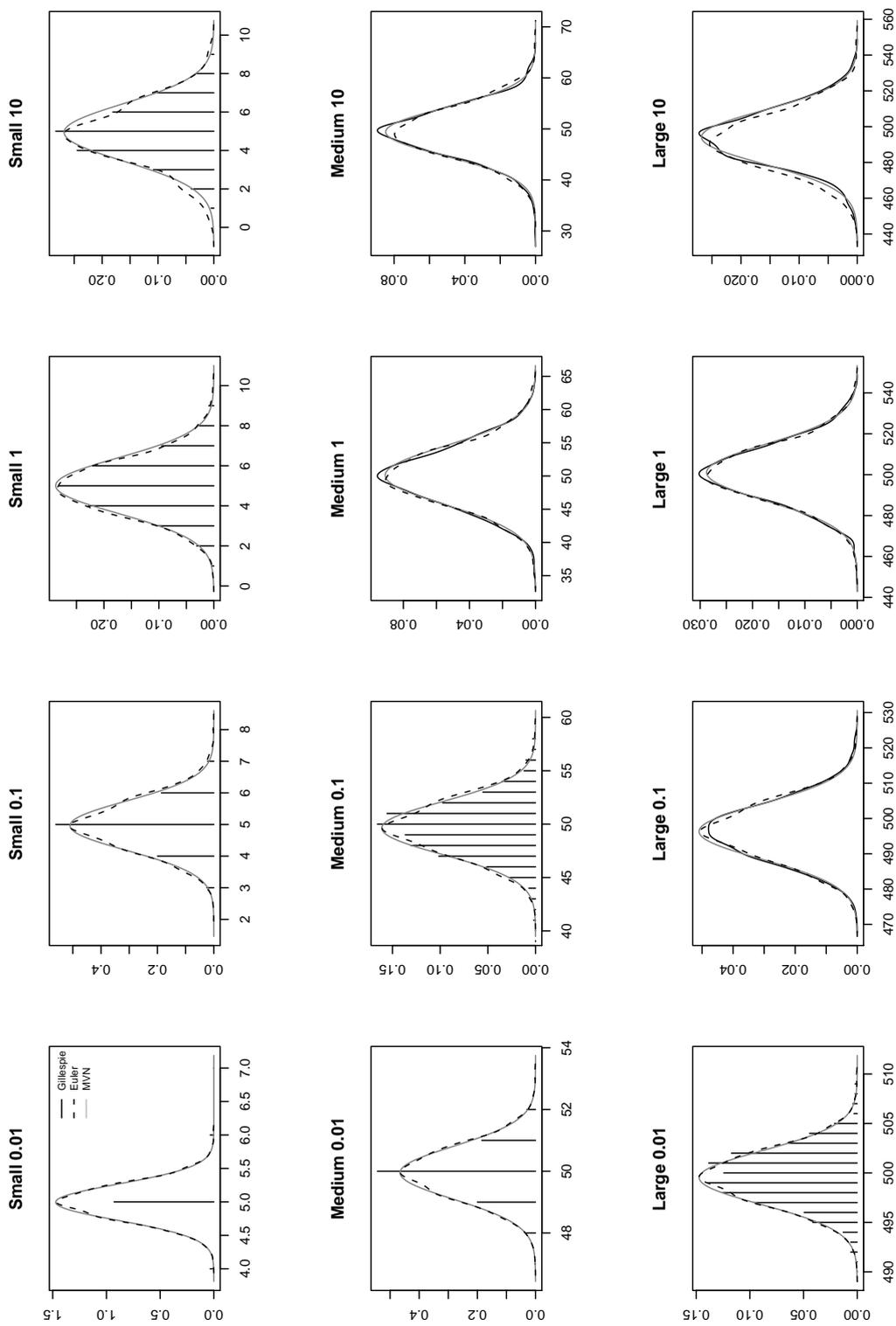


Figure 5.5: Comparing the empirical transition densities – p.m.f. of the Gillespie Method (*solid black*) the Chemical Langevin Equation (*dashed black*) and the Linear Noise Approximation (*solid grey*) at four time points ($t = 0.01, 0.1, 1,$ and 10) using three different system sizes (Small, Medium, Large). Vertical bars indicate the relative frequencies in samples with a small range of discrete values (< 25).

Size	Method	Time Points			
		0.01	0.1	1	10
Small	1-Step	4.995 (0.27)	4.95 (0.87)	4.5 (2.74)	$-8.8 \cdot 10^{-16}$ (8.66)
	SSA	7.994 (0.18)	7.968 (0.59)	7.32 (1.72)	5.891 (2.47)
Medium	1-Step	49.95 (0.87)	49.5 (2.74)	45 (8.66)	$-7.1 \cdot 10^{-15}$ (27.39)
	SSA	79.959 (0.63)	79.326 (2.02)	74.131 (5.54)	59.577 (8.04)
Large	1-Step	499.5 (2.74)	495 (8.66)	450 (27.39)	$-1.13 \cdot 10^{-13}$ (86.60)
	SSA	799.346 (1.99)	793.29 (6.29)	743.242 (17.84)	589.198 (24.99)

Table 5.5: Means (s.d.) of the single step Euler approximation and the corresponding Gillespie Samples for the Transcription model.

Q–Q plots (Figure 5.6) highlight the discreteness of SSA samples in all instances of the small configuration and also at short transition intervals ($t = 0.01, 0.1$) and ($t = 0.01$) of the medium and large configurations respectively. Both approximate methods generate samples that have similar quantiles to the exact samples with the exception of the largest time transitions: the upper CLE quantiles seem to indicate a slight positive skew. Omitting the case of the small system at $t = 0.01$, where both approximations are not satisfactory, the LNA quantiles seem overall closer to the exact quantiles.

The overall comparison of means and standard deviations (Figure 5.7) does not indicate large differences between the summed squares of the log-ratios of the means; the largest is observed at the transition time of $t = 10$ of the small configuration. Contrary to the means, the comparison of the standard deviations (Figure 5.7 b) reveals large differences in scale between the CLE and SSA samples.

Finally, the non-parametric multivariate signed rank test showcases fewer significant differences (at 95% s.l.) between the exact and the LNA method (4/12), compared to the CLE v.s. SSA tests for all size–time combinations. The samples from the CLE method exhibit consistently significant differences from the exact sample in large transition times for all system sizes, which is in accord with the pattern in Figure 5.7–b.

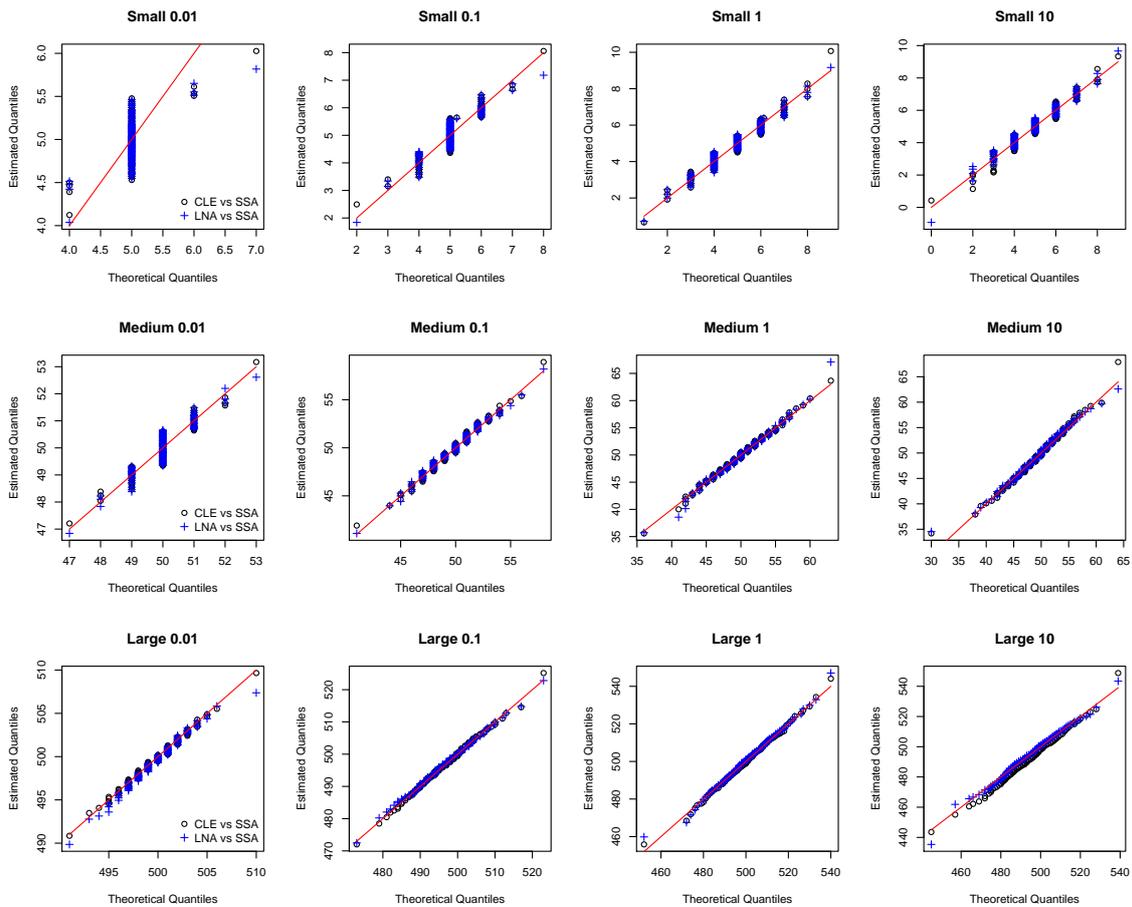


Figure 5.6: Q-Q plots of samples; Chemical Langevin Equation vs Gillespie (*circles*) and Linear Noise Equation vs Gillespie (*crosses*) at various time points.

5.6.3 Discussion of the Examples

The simulated samples from the two models that we have considered share some common features, in terms of their empirical transition densities. First we consider the behavior of the exact discrete process in short transition times. In small time intervals, e.g. $t = 0.01$, we expect that few reactions will occur, therefore, the changes on state of the process will involve a small number of molecules. Hence, the concentration of the transition probabilities at few discrete points, as seen in Figures 5.3 and 5.6, is explained by the occurrence of a small number of reactions. Furthermore, both LNA and the Euler–Maruyama-based approximations rely on continuous densities which do not approximate well enough densities concentrated on a small number of discrete points.

Following the previous argument, we expect that as the LNA and CLE methods will provide

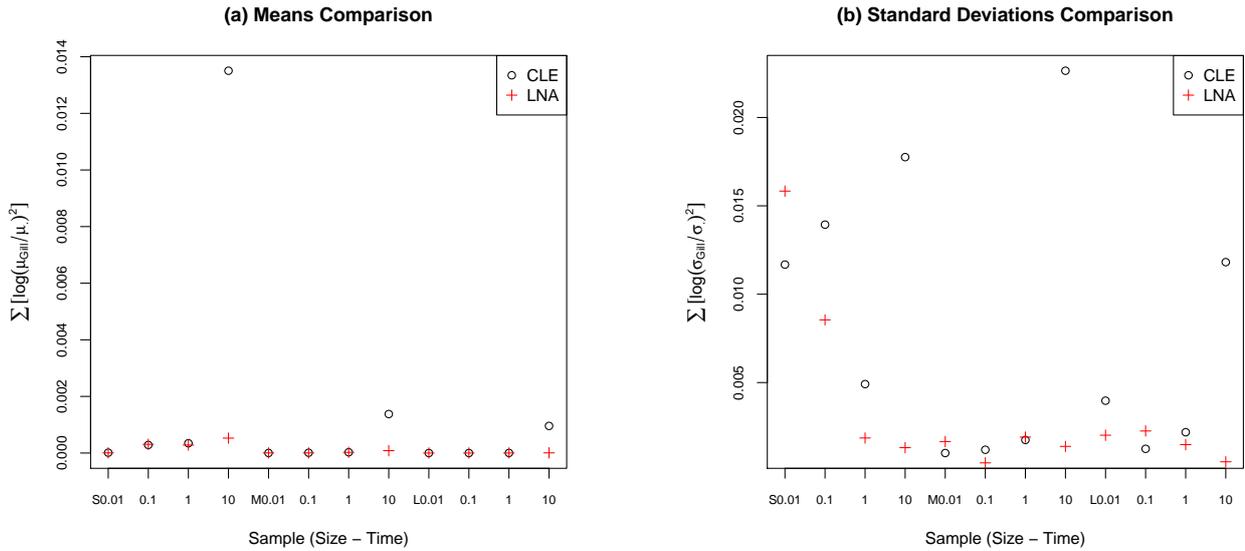


Figure 5.7: Comparisons of DNA means (a) and s.d.s (b) using the sums of the squared log-ratios of the means–s.d. The first four entries at the x -axis correspond to the small system at the time instances $t = 0.01, 0.1, 1, 10$. Similarly the next four correspond to the medium system and the last four to the large.

good approximations when the transition densities involve a large range of values. When the number of reactions increases, the corresponding range of the transition distributions increases as well. In systems with a small number of molecules, or with slow reactions, the number of reaction occurrences will become sufficiently large after a large time interval; whereas in system with larger number of molecules, the number of reaction occurrences will become sufficiently large sooner. Hence, we can confirm in Figures 5.3 and 5.6 that the LNA and CLE methods provide good approximations for the transition densities of systems of small size for both models at $t = 1$ and $t = 10$ but not at $t = 0.1$. Contrary, both methods provide good approximations for systems with medium and large number of molecules sooner, i.e. $t = 0.1, 1$ and 10 .

We established previously that for both models, the single–step Euler–Maruyama approximation becomes very poor as the transition time increases (Tables 5.2 and 5.5). Under the single–step approximation the mean and the variance of the Normal transition density is estimated from the initial point. Therefore, it is acceptable locally, i.e. for small transition intervals $t = 0.01$, but as t increases, the approximation cannot follow the behaviour of

Size	Test	Time Points			
		0.01s	0.1s	1s	10s
Small	LNA vs SSA	0.0001	0.0162	0.2382	0.4431
	CLE vs SSA	0.0001	0.0730	0.2117	0.0000
Medium	LNA vs SSA	0.8920	0.4902	0.3589	0.2064
	CLE vs SSA	0.2708	0.7367	0.3335	0.0000
Large	LNA vs SSA	0.2026	0.9331	0.7700	0.0412
	CLE vs SSA	0.8849	0.8881	0.8330	0.0000

Table 5.6: P-values of the multivariate rank-sum test keeping ϑ fixed, using 1000 samples from the Transcription example at four time instances.

the system dynamics. In contrast, the CLE method which is based on a discretized Euler–Maruyama follows the system dynamics more closely. The rather poor performance of the CLE at estimating the transition of the Transcription model at $t = 10$, in the Large system, could be attributed to the discretization step: a smaller step may improve the accuracy of the approximation.

In both examples the LNA seems to provide satisfactory approximations for the medium and large system sizes while in some cases the CLE samples diverged from the exact samples. For small system sizes both approximate methods seems to struggle since the system’s discreteness is approximated poorly. The LNA’s s.d. estimates seems to be closer to the exact s.d. estimates than the CLE’s. Overall, most configurations show that the LNA’s approximation is as good as or even has a small improvement over CLE’s. In addition, LNA is tractable, and as we will see in Chapter 6, more amenable to use for inference than the CLE method. Nevertheless, the LNA is a linear approximation of the CLE method and we can attribute its superior performance to the numerical schemes: the numerical solution of ODEs are much easier to handle compared to the numerical solution of non-linear SDEs.

5.7 Discussion

In this chapter we introduced a reparametrization which leads to the LNA approximation. In order to investigate the performance of the LNA we considered the Lotka–Volterra model and a

model for prokaryotic Transcription. In particular, we compared the LNA's transition density to the empirical transition densities of the Gillespie algorithm and the Chemical Langevin Equation approximation under different modelling scenarios. The LNA method provided very satisfactory approximations for the transition densities of large time intervals. For small transition intervals, we saw that LNA, being a continuous approximation, is difficult to account for a (discrete) probability mass function. Nevertheless, the estimates of the mean and the standard deviation were close to the exact method's empirical estimates. Overall, LNA's performance was as good as the CLE's approximation and for some cases, it showed a small improvement.

Chapter 6

Inference for Auto-regulatory networks

In Chapter 5 we concluded that, with the exception of small systems at small time intervals, the Linear Noise Approximation provides a reasonable approximation to the system's dynamics and we proceed, in this chapter, by exploring the inferential possibilities. First, in Section 6.1, we outline the methodologies that have been proposed for the inference of auto-regulatory networks. In Section 6.2 we derive the two inferential methods based on the LNA which differ in the solutions of the underlying ODEs. Finally, in Section 6.3 we compare the two LNA methods by considering a series of simulated experiments based on two case studies.

6.1 Methods Overview

The inference for auto-regulatory networks can be perceived as a twofold problem: inferring the model structure (Bower and Bolouri 2001) and estimation of the stochastic kinetic constants. We only address the latter problem and assume that the methodology described in Chapter 3 is applicable, i.e. the underlying network structure is known and can be expressed as a set of biochemical reactions. The problem of stochastic rate constants estimation from discrete time observations has recently attracted the research interest; similar to Chapter 3, we outline the proposed methods classified according to their modelling assumptions:

- **Deterministic Approximations:** A system of ODEs is considered as an approximation to the stochastic kinetics. Usually, the procedure of fitting ODE models is considered

as a non-linear regression, e.g. the COPASI software developed by Hoops et al. (2006) employs a minimisation criterion based on weighted residual sum of squares.

- **Exact methods:** The exact Markov Jump processes are employed to estimate the corresponding stochastic rate constants. Tian et al. (2007) follow a simulated likelihood approach by applying non-parametric transition density estimators on SSA and SDE generated samples. Boys et al. (2008) propose a method that tracks the exact likelihood of the discrete process using MCMC algorithms which accommodates for missing data and data with completely unobservable species. Although the exact methods are theoretically very appealing, in practice they are very computationally demanding and the application to realistic biochemical networks is an open problem (Boys et al. 2008).
- **Chemical Langevin Equation:** as discussed in Section 3.3, CLE is a non-linear SDE which is used as an approximation to the exact process. Golightly and Wilkinson (2005) combined a data imputation scheme based on Euler-Maruyama transition densities with a MCMC algorithm which had deteriorating mixing properties as the data augmentation increased. Their proposed methodology was applied in a auto-regulatory network assuming both complete and partial observations. The same authors (Golightly and Wilkinson 2007) addressed effectively the mixing issue of their previous MCMC methodology and also extended its scope to partial observations subject to measurement error. Furthermore, in parallel with their work on static inference, Golightly and Wilkinson (2006) proposed a sequential MCMC method that handles partial observations with measurement error as well.
- **Linear noise approximation:** as we saw in Section 5.3, the linear noise approximation (LNA) leads to a *linear in the narrow sense* SDE which has a known (Normal) transition density with parameters expressed as solutions of systems of ODEs. Komorowski et al. (2009) express the likelihood in terms of the LNA's transition densities and employ a Bayesian analysis in conjunction with a MCMC algorithm and apply it to a time dependent model of gene expression. It should be noted that the method considered by Komorowski et al. (2009) corresponds to the Non-Restarting method, derived in the next Sections of this chapter. Rutter et al. (2010) consider two methods, the LNA and a mean field approximation, which are derived from a *variational formulation* of exact inference.

- **Moment closure:** Approximations of the transition densities can be achieved by considering the moments of the Master Equation up to a certain order. In the presence of non-linear reaction rates a recursive relation occurs where lower order moments are expressed in terms of higher order moments. Based on the recursion, analytical approximations can be derived by assuming a *closure*, i.e. all moments whose order exceeds a preset threshold are set to zero. Although a full-scale inferential methodology has not yet been proposed, it is an area of active research, moment closure has been applied to the analysis of regulatory genetic systems: Singh and Hespanha (2007) used the moment closure technique to estimate the coefficient of variation in two regulatory models and Gillespie (2009) to estimate the transition density of certain species in a chaperone model of realistic size.

6.2 Likelihood arising from the Linear Diffusion Approximation

We assume complete observations of the state of the system, $\{\mathbf{X}\}$, at a set of times t_0, t_1, \dots, t_n . We denote with $X(t; \boldsymbol{\vartheta})$ the LNA approximation of $\{\mathbf{X}\}$ at time t and for $i = 0, \dots, n - 1$, $X(t_i; \boldsymbol{\vartheta})$ is set equal to the corresponding observations of $\{\mathbf{X}\}$. A given observation at t_i , provides initial conditions for the ODEs described in the Section 2.1.4; the ODEs are then integrated forward until time t_{i+1} to provide the transition kernel for the system at time t_{i+1} given its state at time t_i . Explicitly,

$$X(t_{i+1}; \boldsymbol{\vartheta}) | X(t_i; \boldsymbol{\vartheta}) \sim \text{Normal} \left(N_{\text{T}} \mathbf{y}(t_{i+1}; \boldsymbol{\vartheta}) + \sqrt{N_{\text{T}}} \mathbf{m}(t_{i+1}; \boldsymbol{\vartheta}), N_{\text{T}} S(t_{i+1}; \boldsymbol{\vartheta}) \right) \quad (6.1)$$

$\mathbf{y}(t; \boldsymbol{\vartheta})$ is the ODE solution of the “deterministic approximation” detailed in Section 5.2, and $\mathbf{m}(t; \boldsymbol{\vartheta})$, $S(t; \boldsymbol{\vartheta})$ are the solutions to the ODEs expressing the mean (2.15) and variance (2.17) of the deviation process $M(t)$.

The product of the transition densities provides a conditional likelihood¹ for the data (conditional on $X(t_0; \boldsymbol{\vartheta})$). A numerical optimisation algorithm (BFGS, introduced in Section 2.3.2) is then used to obtain the Maximum Likelihood Estimators (MLEs) $\hat{\boldsymbol{\vartheta}}$ of $\boldsymbol{\vartheta}$, the vector of scaled stochastic rate constants. The parameters of interest express the rate of occurrence of a reaction in a single unit of time per unit of “concentration”, and are constrained

¹The log of which we denote by $\ell(X_t, \mathbf{c})$.

to the non-negative plane: $(\boldsymbol{\vartheta} \in [0, \infty)^{N_R})$. Since the BFGS algorithm is unconstrained, we use a log-transformation $(\log(\boldsymbol{\vartheta}) \in \mathbb{R}^{N_R})$ for consistency. In addition, we preserve the log-parametrisation in the derivation of approximate confidence intervals (CIs), based on the Wald's approximation and the observed Fisher information.

6.2.1 Restarting and Non-Restorting

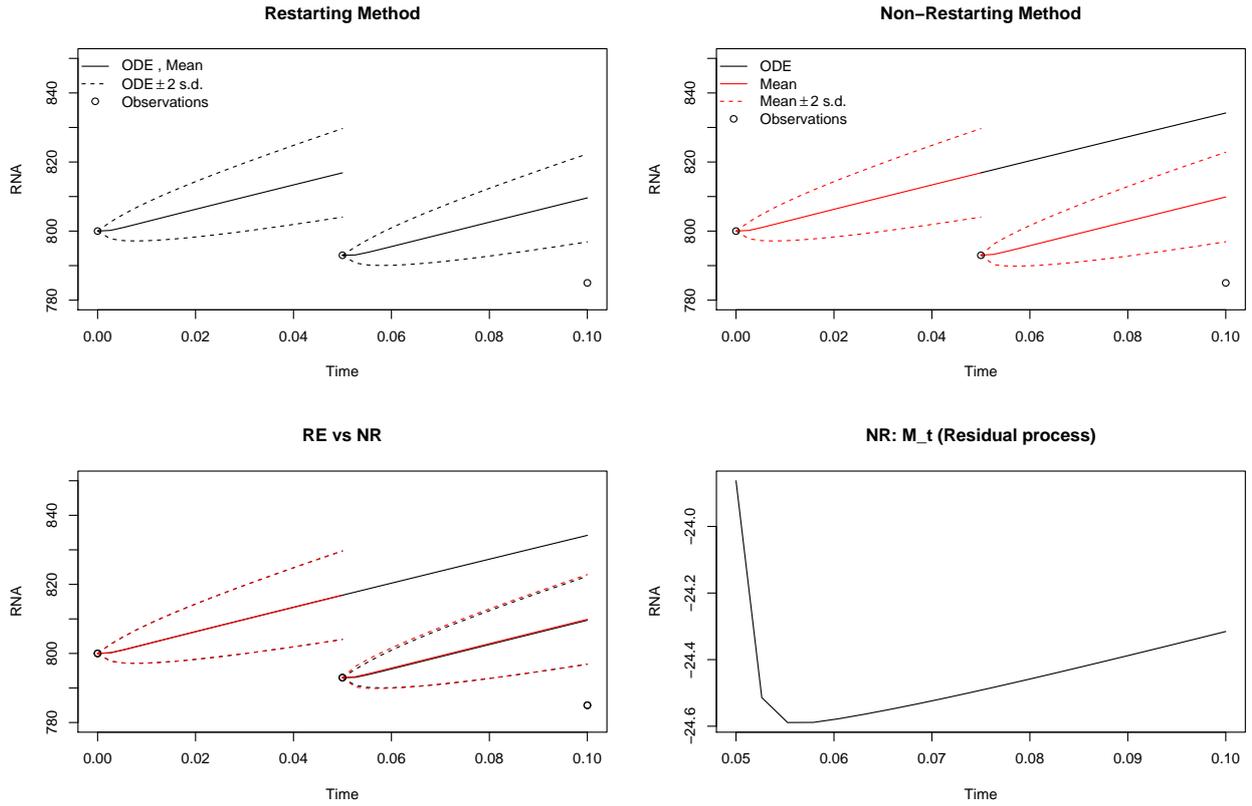


Figure 6.1: Illustrating the solutions obtained by the RE (*top left*) and the NR (*top right*) methods which are also plotted in a common plot (*bottom left*) for a toy dataset of the transcription network (Section 5.6.2). The mean $\mathbf{m}(t)$ of the NR method's scaled residual process M_t is also plotted (*bottom right*) for the time interval $[0.05, 0.1)$ which obtains the non-trivial (non-zero) solution.

As mentioned before, we obtain the solutions of the system of ODEs $(\mathbf{y}(t; \boldsymbol{\vartheta}), \mathbf{m}(t; \boldsymbol{\vartheta}), S(t; \boldsymbol{\vartheta}))$ by numerically resetting the initial conditions at every observed time point. In this section we present two different methods to update the initial conditions: the *Restarting method* (RE) and the *Non-Restorting* (NR) method.

In the NR method we solve the ODEs (\mathbf{y}_t) of the deterministic model using the (scaled) initial observation as the initial condition: $\mathbf{y}(t_0; \boldsymbol{\vartheta}) = x(t_0)/N_T$, which corresponds to the solid black line in the two top right plot of Figure 6.1, and we let it evolve forward in time until the requested duration T . At each time instance t_i , associated with an observation $x(t_i)$, we re-initialise the residual process M_t by updating its mean $\mathbf{m}(t_i; \boldsymbol{\vartheta}) = \sqrt{N_T}(x(t_i)/N_T - \mathbf{y}(t_i; \boldsymbol{\vartheta}))$ and its covariance matrix $S(t_i; \boldsymbol{\vartheta}) = \mathbf{0}$, i.e. a vector of zeroes². The combined mean ($\sqrt{N_T} \mathbf{m}(t; \boldsymbol{\vartheta}) + N_T \mathbf{y}_t$) is illustrated as a solid red line and the (scaled) covariance matrix which is associated with $(\sqrt{N_T} \mathbf{m}(t; \boldsymbol{\vartheta}) + N_T \mathbf{y}_t) \pm 2$ s.d., illustrated as dashed red lines, in the top right plot of Figure 6.1. As a side note, the Non-Restarting method has been used for inferential purposes previously by Komorowski et al. (2009).

In the RE method, we solve the ODEs (\mathbf{y}_t) associated with the deterministic model by reconsidering our initial conditions at each time point t_i associated with an observation $x(t_i)$: $\mathbf{y}(t_i; \boldsymbol{\vartheta}) = x(t_i)/N_T$. By restarting the deterministic model, we set implicitly the mean $\mathbf{m}(t)$ of the residual process M_t equal to zero $\mathbf{m}(t_i; \boldsymbol{\vartheta}) = \mathbf{0}$ at each t_i which makes $\mathbf{m}(t)$ to obtain the trivial solution $\mathbf{m}(s; \boldsymbol{\vartheta}) = \mathbf{0}, \forall s \in [0, T]$. Therefore, in the RE method the combined mean $\sqrt{N_T} \mathbf{m}(t; \boldsymbol{\vartheta}) + N_T \mathbf{y}_t$ coincides with the (scaled) solution of the deterministic model, illustrated as a solid black line in the top left plot in Figure 6.1. In addition, at each t_i we restart the initial conditions for the covariance matrix $S(t_i; \boldsymbol{\vartheta}) = \mathbf{0}$ which is used to estimate the standard deviation in the formula $\sqrt{N_T} \mathbf{m}(t; \boldsymbol{\vartheta}) + N_T \mathbf{y}_t \pm 2$ s.d, illustrated as the dashed black lines in the top left plot.

From the plots in Figure 6.1 we can observe that one difference between the two methods is the way that the combined mean ($\sqrt{N_T} \mathbf{m}(t; \boldsymbol{\vartheta}) + N_T \mathbf{y}_t$) is handled by the two methods: in the RE method, coincides with the deterministic model because $\mathbf{m}(t)$ vanishes, but in the NR method the combined mean involves $\mathbf{m}(t)$ in addition to \mathbf{y}_t . Both solutions seems to be very close when they are combined in a single plot (bottom left in Figure 6.1). As we saw in Section 2.1.4.2, the ODEs associated with the covariance matrix $S(t)$ involve only terms of \mathbf{y}_t , the deterministic model, and not $\mathbf{m}(t; \boldsymbol{\vartheta})$. Therefore, as the solutions for the \mathbf{y}_t diverge between the two methods we would expect the same to happen for $S(t)$ the estimate of the covariance matrix. In the subsequent Sections we will investigate the performance of both methods in a series of numerical examples.

²We choose to work with a vectorised version of the upper triangular of $S(t)$.

6.2.1.1 Stiff ODEs

In Section 2.2 we characterized a system of ODEs as stiff when, in the same time interval, certain of its elements exhibit rapid changes compared to the rest subject to their initial conditions. Taking into account the fact that we employ the ODE solver within the optimisation procedure, it is not possible to foresee, in the course of the optimisation, if a certain parameter choice would lead to stiff behavior. Furthermore, general ODE numerical methods do not usually address the stiffness problem, which leads to numerical instabilities (Hairer and Wanner 1991). We chose the numerical ODE solver LSODA (Petzold 1983) which has the advantage of automatically switching between stiff and non-stiff methods. For testing purposes, we considered a non-stiff method, RK45 introduced at Section 2.2, but this resulted in a very long completion time.

6.3 Case Studies

As in the Section 5.6, we employ the Transcription and Lotka–Volterra models to put the estimating capabilities of both methods (RE, NR) to the test. We generated 100 datasets, using the SSA algorithm, for each combination of the following two factors: the *observation intervals* chosen to be every 0.1, 0.5, 1 time units and the *system size* considering an initial population (small) a tenfold increase (medium) and a successive tenfold increase as well (large). In order to assess the maximum likelihood estimators (MLEs) we apply the asymptotic result:

$$T := 2 \left(\ell \left(X_t, \hat{\boldsymbol{\vartheta}} \right) - \ell \left(X_t, \boldsymbol{\vartheta}_{\text{True}} \right) \right) \sim \chi_{N_R}^2$$

where N_R is the number of parameter which in our setting corresponds to the number of reactions. We examine Q–Q plots of T compared to its anticipated distribution and we also count the number of datasets for which T is significant at the 5% level, which should account for 5% of the datasets. We also investigate the coverage probabilities of the estimated 95% confidence intervals (CIs) through counts, per parameter, of CIs that do not include the true parameter.

Finally, we analyze three specific datasets of the Transcription model, which have been previously analyzed by Golightly and Wilkinson (2005) in a Bayesian setting, to assess the performance of LNA relative to other methods.

6.3.1 Lotka–Volterra

For the generation of Lotka–Volterra data we considered three observational intervals (0.1, 0.5, 1) corresponding to (100, 20, 10) observations of a system evolving for 10 seconds. The choice of the initial conditions and the parameters was identical to the example presented in Section 5.6.1, i.e. the initial number of species for systems of small size ($N_T = 80$) is ($\{\text{Prey}\}, \{\text{Predator}\}$) = (50, 30) with fixed scaled reaction rate constants $\boldsymbol{\vartheta} = (.25, 0.2, 0.125)$ while the corresponding unscaled rates \mathbf{c} are adjusted according to the formula: $\mathbf{c} = (0.25, 0.2/N_T, 0.125)$.

Obs. Interval	Method	System Size		
		Small	Medium	Large
0.1	NR	17	8	7
	RE	13	7	6
0.5	NR	14	6	5
	RE	9	6	5
1	NR	13	7	3
	RE	7	8	3

Table 6.1: Counts of statistically significant likelihood ratios for the Lotka–Volterra model.

Table 6.1 shows the counts of statistically significant likelihood ratio test statistics for both the Restarting and Non–Restarting method. As the system size increases the number of significant log–likelihood differences decreases since the LNA approximation, an asymptotic approximation³ with respect to N_T , becomes more accurate. The latter problem becomes more prominent in the case of a system of small size and dense (100) observations. In addition, RE method provides a better coverage than the NR for each possible system size and observational interval combination.

Figure 6.2 illustrates Q–Q plots of the likelihood ratio test statistics, for each configuration of system size–observations number, v.s. the theoretical χ_3^2 distribution. When we focus on systems of small size, the likelihood ratio test statistics depart from the χ_3^2 distribution

³Can also be viewed as a central limit theorem of the transition densities as the system size increases (Kurtz 1972).

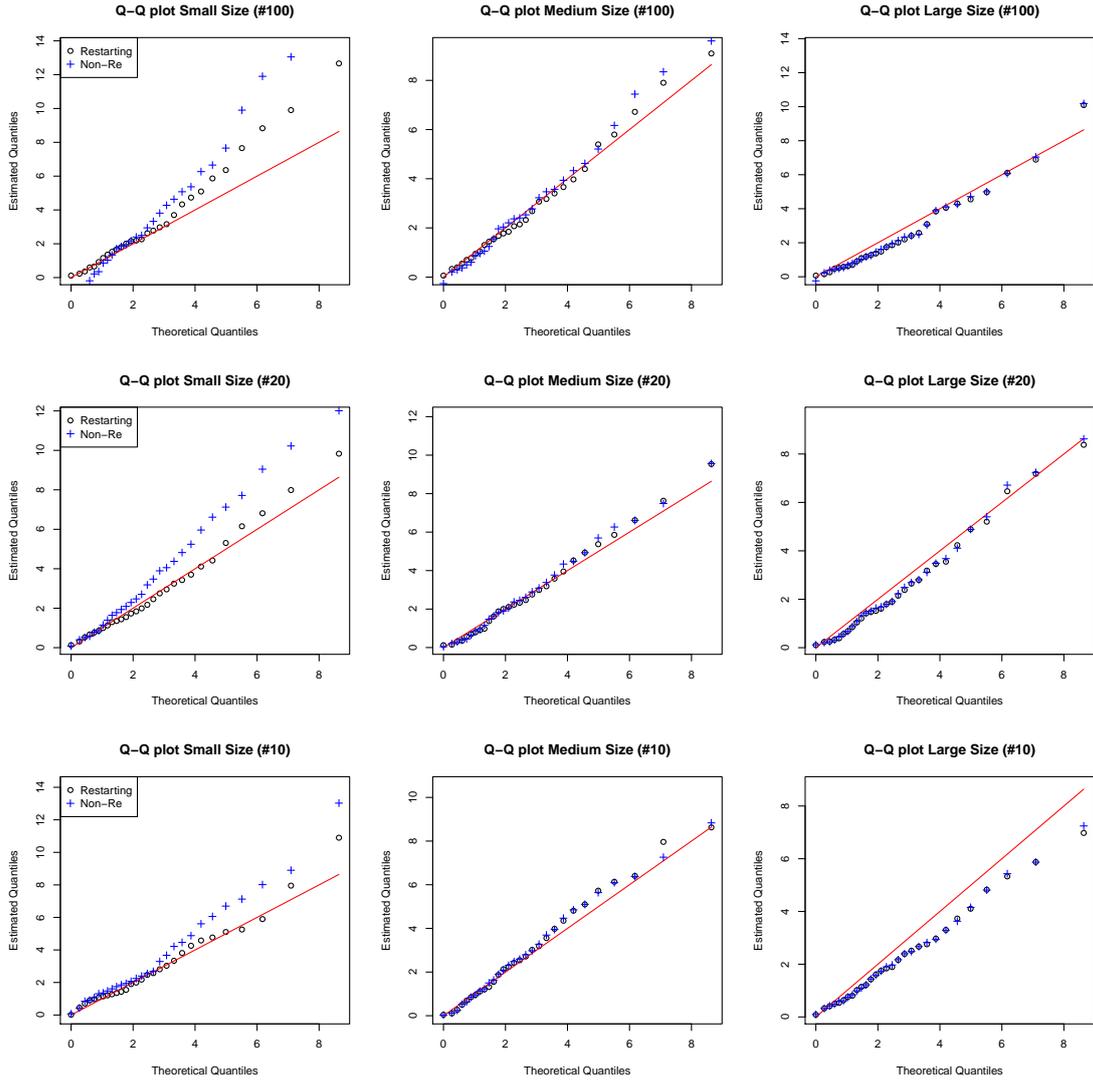


Figure 6.2: Q–Q plots of the likelihood ratio test statistic for the Lotka–Volterra model. Likelihood ratio statistics based on the Restarting (*points*) and the Non–Restarting method (*crosses*) are plotted against the quantiles of the χ^2_3 distribution (*lines*). Systems of three sizes (Small, Medium, Large) were considered at three observational intervals (0.1, 0.5, 1) which correspond to sample sizes of (100, 20, 10) observations, indicated with the symbol (#).

for both methods. In particular, as the number of observations increase the deviation from the χ^2 assumption increases which is in accordance with the high number of significant log–likelihood differences. Furthermore, the marginal CIs of the RE method result a coverage of true parameters which is at least as good as or even better than the coverage provided by the NR method.

We summarize the number of the approximate CIs that fail to cover the true value of the

Increment	System Size	Method	ϑ_1	ϑ_2	ϑ_3
0.1	Small	NR	31	10	9
		RE	9	11	14
	Medium	NR	5	4	5
		RE	4	4	5
	Large	NR	6	8	8
		RE	4	7	8
0.5	Small	NR	18	9	8
		RE	9	8	5
	Medium	NR	5	8	4
		RE	3	7	5
	Large	NR	4	9	5
		RE	5	9	6
1	Small	NR	18	8	4
		RE	13	9	4
	Medium	NR	6	10	4
		RE	5	9	3
	Large	NR	3	3	3
		RE	2	2	2

Table 6.2: Missed Wald's CIs coverage counts for the Lotka-Volterra example.

parameters of interest in Table 6.2. The parameter ϑ_1 corresponds to the first order R_1 reaction that expresses the reproduction of Prey. Coverage is especially poor for this parameter in the small system, for both methods, although the restarting method does perform better than the Non-Restarting method. In general, coverage is worse for the small system, and for the most part the Non-Restarting method performs better than the Restarting method, which is accordance with the findings from Table 6.1

6.3.2 Transcription Network

We considered three levels of observation time intervals of $(0.1, 0.5, 1)$ time units leading to $(500, 100, 50)$ observations over a simulation interval of 50 time units. The initial states and parameter choices are similar to the simulation choices at Section 5.6.2. To be more specific, we assumed that a system of small size has initial state:

$$(\{\text{RNA}\}, \{\text{P}\}, \{\text{P}_2\}, \{\text{DNA}\}) = (8, 8, 8, k - 5, 5)$$

with $k = 10$ and parameters:

$$\boldsymbol{\vartheta} = (3.4, 0.7, 0.35, 0.2, 3.4, 0.9, 0.3, 0.1)$$

which correspond to the unscaled parameters:

$$\boldsymbol{c} = (0.1, 0.7, 0.35, 0.2, 0.1, 0.9, 0.3, 0.1).$$

For systems of medium and large sizes we considered subsequent tenfold increases of the initial state. The $\boldsymbol{\vartheta}$ remain fixed, while \boldsymbol{c} are adjusted to the $[(10i)^{-1}, 0.7, 0.35, 0.2, (10i)^{-1}, 0.9, 0.3, 0.1]$ formula with $i = \{10, 100\}$ corresponding to the medium and large system respectively.

The LNA approximation is based on the asymptotic consideration that the system size N_T increases, which is most strongly violated by the small systems. Table 6.3 shows counts of significant likelihood ratio tests for the Transcription example; as might be expected, of the three system sizes, the χ^2 fit is poorest for the small system. As with the Lokta–Volterra example, the introduction of more observations exacerbates this problem as evidenced by the increase in the number of significant log-likelihood differences. Conversely, sparser observations (50) exhibit a smaller number of significant ratios: the smaller number of observations contributes to the estimation uncertainty leading to wider confidence intervals (Table 6.4). In

Obs. Interval	Method	System Size		
		Small	Medium	Large
0.1	NR	69	11	5
	RE	40	7	4
0.5	NR	25	3	4
	RE	12	5	3
1	NR	13	4	5
	RE	6	5	5

Table 6.3: Counts of significant likelihood ratio tests for the Transcription example.

terms of significant log-likelihood differences, the RE method consistently outperforms the NR in systems of small size and both methods provide a good fit for systems of larger sizes.

The Q–Q plots (Figure 6.3) of the Likelihood ratio test statistic v.s. the theoretical χ^2_8 distribution provide further evidence that the combination of large number of observations (500) with small system size leads to large likelihood ratios that cannot be attributed to chance. Nevertheless, the Restarting method in the sparser dataset (50) is consistent with the asymptotic χ^2 assumption as in the Lotka–Volterra case, i.e. the smaller number of significant test statistics of Table 6.3 can be attributed to the uncertainty introduced by the smaller number of observations. In contrast, with the exception of the medium system with observations at 0.1 time intervals, the test statistics of larger system sizes seem to be consistent with the χ^2 approximation for both methods.

In Table 6.4 we investigate each parameter separately by considering the counts of CIs that do not cover the true parameters. Both methods provide adequate coverage when they are applied to datasets generated from observations of large systems, and observations of 0.1 and 0.5 time intervals from medium systems. As expected from the likelihood ratio test, the coverage for small systems is poor; the only exception, again, is traced at the combination of the small system with an observational interval of 1 time units: the small number of observations increases the range of CIs leading to an improved coverage. Generally, the performance of RE method seems to be as good as the performance of NR or better.

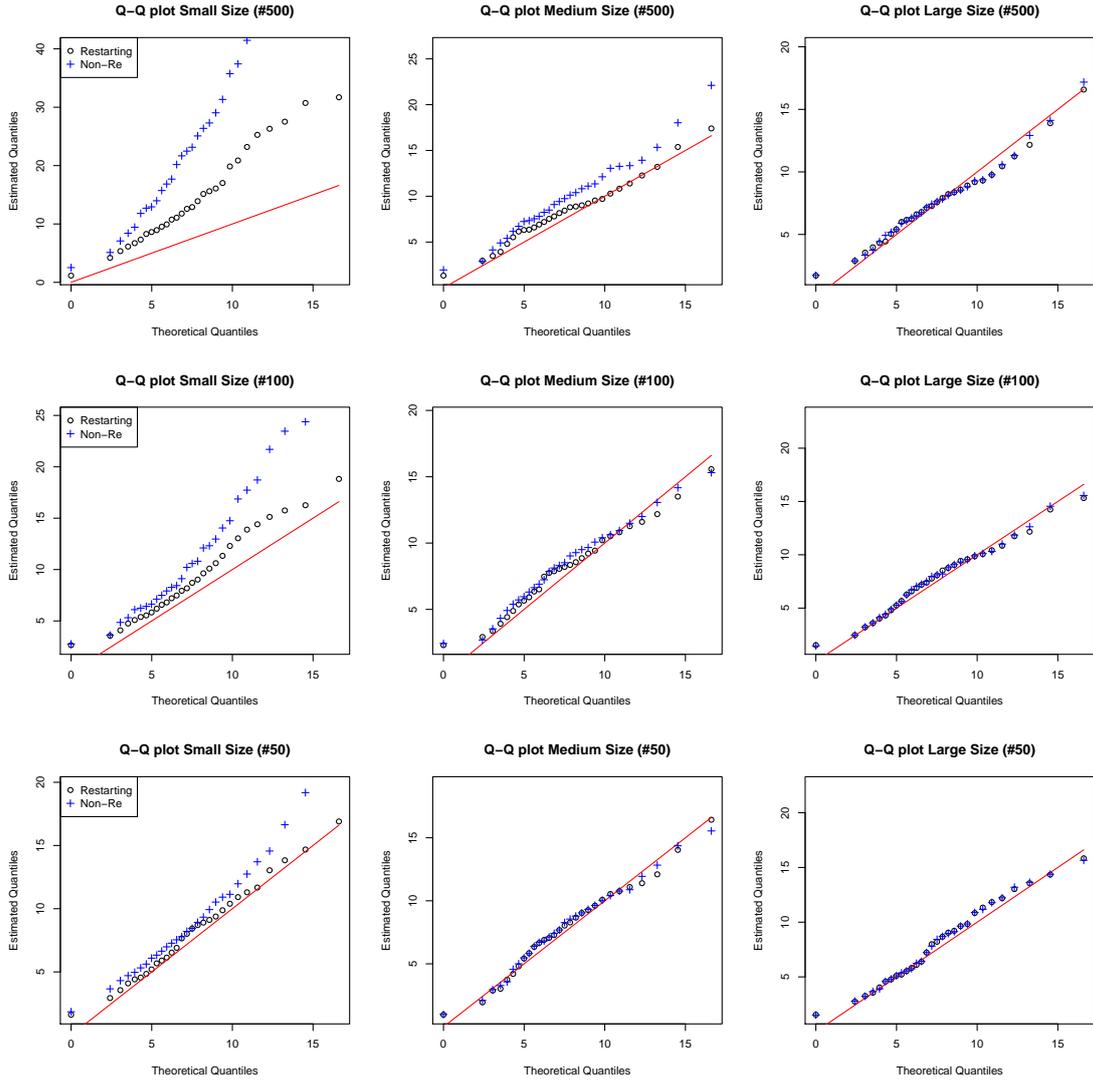


Figure 6.3: Q–Q plots of the Likelihood ratio test statistic for the Transcription model. Likelihood ratio test statistics based on the Restarting (*points*) and the Non–Restarting method (*crosses*) are plotted against the quantiles of the χ^2_3 distribution (*lines*). Systems of three sizes (Small, Medium, Large) were considered at three observational intervals (0.1, 0.5, 1) which correspond to samples sizes of (500, 100, 50) observations, indicated with the symbol (#).

It should be noted that incidents of missed coverage are not independent between parameters. If we take into account the graph of Figure 3.2 which depicts the species–reactions dependencies, we can identify two reversible reactions, Repression Binding and Dimerisation, associated with the parameter pairs $(\vartheta_1, \vartheta_2)$ and $(\vartheta_5, \vartheta_6)$, respectively. The presence of reversible reactions, imposes a correlation structure on each pair which can be traced in almost all configurations of Table 6.4. The smallest system with short (0.1) observational interval

Obs. Interval	Method	Size	ϑ_1	ϑ_2	ϑ_3	ϑ_4	ϑ_5	ϑ_6	ϑ_7	ϑ_8
0.1	Small	NR	10	12	23	8	3	9	10	24
		RE	6	11	16	14	15	9	14	28
	Medium	NR	6	3	11	8	3	2	3	6
		RE	7	6	9	8	4	3	5	6
	Large	NR	1	3	3	8	8	7	5	8
		RE	4	4	3	9	8	8	4	9
0.5	Small	NR	7	9	10	12	5	4	4	11
		RE	8	8	11	11	6	4	6	10
	Medium	NR	6	6	5	2	2	3	3	1
		RE	6	6	6	2	3	5	3	2
	Large	NR	4	5	11	8	4	5	9	8
		RE	4	5	9	8	4	5	9	7
1	Small	NR	2	4	5	7	10	10	2	10
		RE	3	5	3	8	10	10	2	7
	Medium	NR	10	11	4	6	12	13	3	3
		RE	9	9	3	6	12	13	3	3
	Large	NR	3	3	3	9	11	11	5	10
		RE	4	3	2	9	8	9	4	10

Table 6.4: Missed Wald’s CIs coverage counts for the Transcription example.

seems to be an exception which can be attributed to the overall large number of missed coverage incidents.

6.3.2.1 Three Datasets

In this Section, we consider three datasets (D_1, D_2, D_3) of sparse (50) observations generated by Golightly and Wilkinson (2005). As mentioned before, we have selected the configuration of the small system to match the configuration considered by the authors for the three datasets. Additionally, the duration of the simulation remains unchanged (50) and the states are observed per unit of time.

Golightly and Wilkinson (2005) proposed a Bayesian inferential methodology for the parameters \mathbf{c} which they applied to (D_1, D_2, D_3) . For convenience, we reproduce their results in Table 6.5. According to Golightly and Wilkinson (2005), for each dataset a single chain of 1,000,000 iterations was run and the first 100,000 iterations was discarded as a burn-in period. Therefore, Table 6.5 contains the estimated posterior means and s.d. from the final 900,000 iterations. The posterior means and variances of the parameters which are not associated with a reversible reaction, provide a good support for the corresponding true values. In addition, the authors commented that each pair of parameters that is linked with a reversible reaction, raises identifiability issues. To overcome this problem, they introduced a new parametrisation by considering the ratio of the kinetic constants of each pair, i.e. c_1/c_2 and c_5/c_6 in this case, which improved the accuracy of the Posterior estimates.

	Values	c_1	c_2	c_1/c_2	c_3	c_4	c_5	c_6	c_5/c_6	c_7	c_8
	True	0.1	0.7	0.1429	0.35	0.2	0.1	0.9	0.1111	0.3	0.1
D_1	Mean	0.064	0.474	0.141	0.360	0.252	0.043	0.475	0.094	0.288	0.143
	SD	0.022	0.148	0.035	0.125	0.079	0.013	0.154	0.025	0.099	0.044
D_2	Mean	0.058	0.363	0.157	0.372	0.240	0.048	0.477	0.105	0.285	0.121
	SD	0.020	0.120	0.090	0.131	0.071	0.014	0.154	0.047	0.095	0.039
D_3	Mean	0.052	0.346	0.153	0.416	0.213	0.044	0.488	0.092	0.321	0.115
	SD	0.020	0.120	0.046	0.151	0.061	0.011	0.145	0.021	0.108	0.036

Table 6.5: MCMC Posterior means and standard deviations for \mathbf{c} estimated on three datasets (D_1, D_2, D_3) of 50 observations. The Table above is a reproduction of Table 1 from Golightly and Wilkinson (2005).

We employed the RE method to the three datasets in order to get MLEs as well as 95% confidence intervals for the parameters of interest. To facilitate the comparisons, we report, in Table 6.6, the results in terms of \mathbf{c} instead of $\boldsymbol{\vartheta}$. In order to provide (Wald) Confidence Intervals for the ratios c_1/c_2 , and c_5/c_6 we have used an approximation based on the Delta method:

$$\text{Var} \begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \end{bmatrix} \approx \frac{\text{Var}[\hat{c}_1]}{\hat{c}_2^2} + \frac{\hat{c}_1^2}{\hat{c}_2^4} \text{Var}[\hat{c}_2] - 2 \frac{\hat{c}_1}{\hat{c}_2^3} \text{Cov}[\hat{c}_1, \hat{c}_2], \quad (6.2)$$

Values	c_1	c_2	c_1/c_2	c_3	c_4	c_5	c_6	c_5/c_6	c_7	c_8	
True	0.1	0.7	0.1429	0.35	0.2	0.1	0.9	0.1111	0.3	0.1	
D_1	UCI	0.297	2.213	0.164	0.591	0.337	0.273	2.636	0.122	0.499	0.200
	EST	0.113	0.851	0.133	0.366	0.222	0.080	0.793	0.101	0.298	0.133
	LCI	0.043	0.327	0.102	0.226	0.147	0.023	0.238	0.080	0.177	0.088
D_2	UCI	0.196	1.285	0.193	0.599	0.320	72.69	664.581	0.134	0.476	0.178
	EST	0.088	0.572	0.154	0.381	0.216	0.243	2.171	0.112	0.289	0.114
	LCI	0.039	0.255	0.115	0.242	0.146	0.001	0.007	0.090	0.175	0.073
D_3	UCI	0.098	0.665	0.186	0.766	0.285	0.420	3.891	0.129	0.596	0.171
	EST	0.053	0.372	0.142	0.477	0.191	0.098	0.915	0.107	0.357	0.109
	LCI	0.029	0.208	0.099	0.297	0.128	0.023	0.215	0.085	0.213	0.070

Table 6.6: Upper bounds of the 95% C.I. (*UCI*), Maximum Likelihood Estimates (*EST*) and the lower bound of ϑ (*LCI*) and of three datasets (D_1, D_2, D_3) of 50 observations.

and the observed Fisher information. Both ratios are reported in terms of the original \mathbf{c} parameters.

Table 6.6 shows the results of the RE method. Specifically, the 95% confidence intervals of D_1 provide good coverage of the true values. The CIs of the second dataset (D_2) are extremely wide for the c_5, c_6 parameters which may indicate identifiability issues as they are linked with a reversible reaction. In D_3 we observe a similar behavior for the c_1, c_2 kinetic constants associated, again, with a single reversible reaction. Nevertheless, the \hat{c}_2/\hat{c}_1 and \hat{c}_5/\hat{c}_6 estimate the ratios of the true parameters accurately, while the corresponding CIs provide excellent coverage of the true values. It is interesting to note, that although the \hat{c}_5, \hat{c}_6 of D_2 dataset seem off-location their estimated ratio is surprisingly close to the true value. Since the remaining parameters are unaffected from the miss-estimation of \hat{c}_5, \hat{c}_6 , we are led to the conclusion that the relative ratio of the reversible rates plays a more important role than the individual parameters.

The results from both methodologies (Tables 6.5 and 6.6) suffer from the same identifiability issues introduced by the reversible equations. Fortunately, these issues vanish when the ratios of the involved parameter pairs are considered, which both methods estimate with a high

degree of accuracy. The performance of the two methods is not directly comparable because in some instances, e.g. for the c_1, c_2, c_8 parameters, RE produces more accurate estimates, while in some others, e.g. for the D_2 dataset the Bayesian estimates give much more reasonable estimates for c_5, c_6 parameters. The last example can be attributed to the fact that Bayesian estimates benefit from prior information that constrains the range of parameters. As a side note, it is possible to constrain the range of parameters in the RE method as well, by using a constrained optimization method. If we consider the ratios of the kinetic constants of reversible equations then, with the exception of c_3 , LNA provides estimates with a small improvement to their accuracy.

In Figure 6.4 we compare the quadratic approximations with the profile likelihood ratios for the logarithm of the scaled stochastic kinetic constants ($\log(\vartheta)$) using the Restarting method. From one hand, parameters associated with non-reversible first-order reactions ($\vartheta_3, \vartheta_4, \vartheta_7, \vartheta_8$) obtain almost exact quadratic approximations of their profile likelihoods. On the other hand, in most of cases the quadratic approximations for parameters associated with reversible equations and non-linear reactions ($\vartheta_1, \vartheta_2, \vartheta_5, \vartheta_6$) are very poor and so the Wald CIs are likely to be inaccurate; a possible solution is to employ likelihood-based confidence intervals. In addition, Figure 6.4 highlights the identifiability issues of D_2 for the ϑ_5, ϑ_6 pair associated with reversible reactions: *marginally* the likelihood ratios indicate a very wide set of acceptable parameter values while *jointly* the log-likelihood contour plot of the joint profile log-likelihood strongly supports pairs of parameters along a line with slope equal to the true value of identifiable reparametrisation $\vartheta_5/\vartheta_6 = 3.81$ ($c_5/c_6 = 0.11206$).

6.4 Extending LNA

So far we assumed that, in our system of interest, we can observe all species without any measurement error. In this section we extend the previous methodology to handle datasets with unobserved species and measurement error. For convenience, we denote vectors with bold lower case letters \mathbf{x} and the matrices are denoted with bold capital letters \mathbf{X} . Our approach is based on the idea of *Extended Kalman Filter* (See for instance §8.2 in Anderson and Moore 2005), we use LNA to “linearise” the stochastic process of the system’s state, the *unobserved system* $\boldsymbol{\eta}_i$, and then we assume that our observations \mathbf{d}_i are obtained from the *observed system* \mathbf{o}_i which is a combination of a linear function of $\boldsymbol{\eta}_i$ and a Gaussian measurement error $\boldsymbol{\varepsilon}_i$.

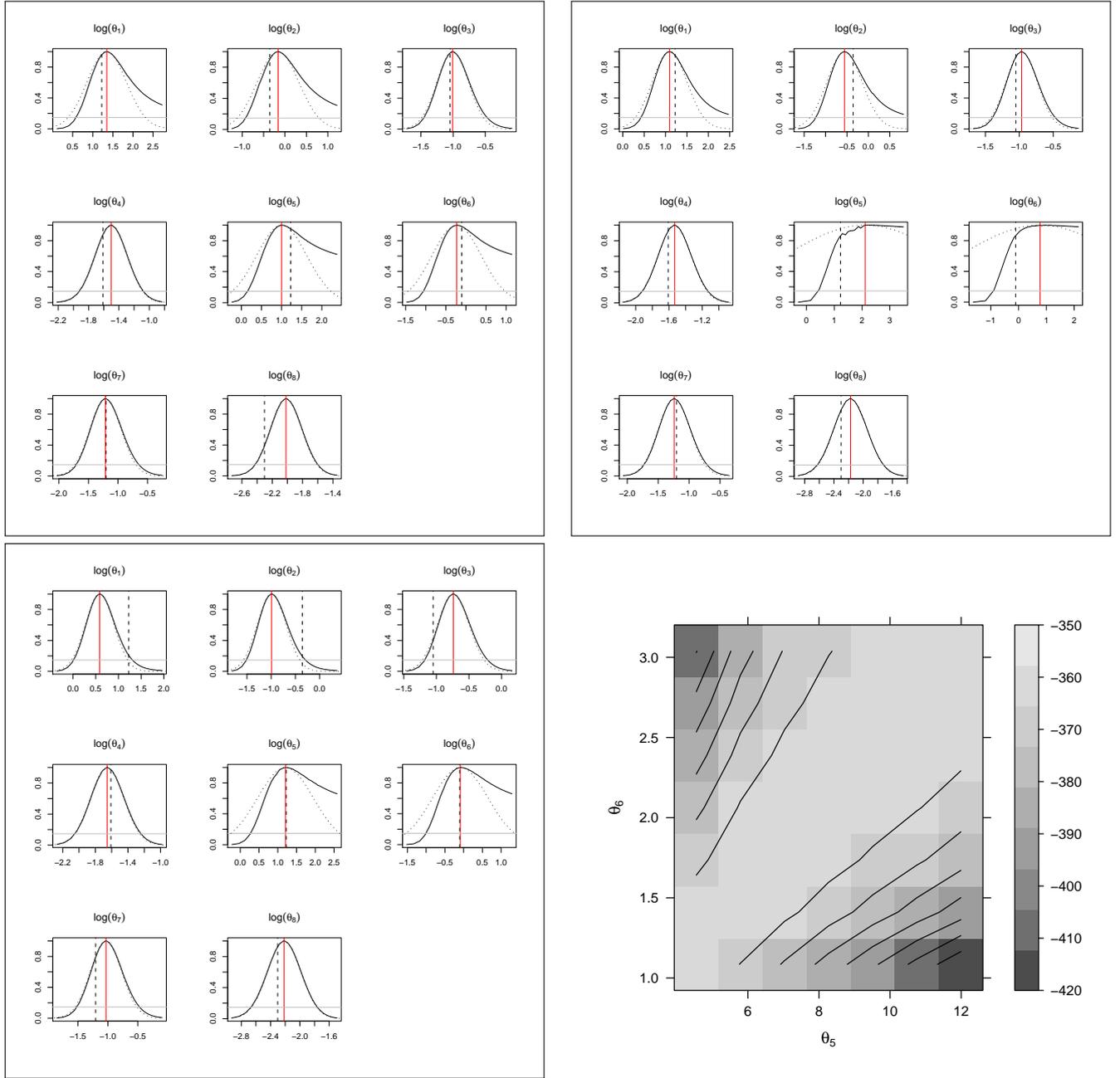


Figure 6.4: Profile Likelihood ratios (solid lines) and the Quadratic approximation (dotted lines) of the D_1 (top left), D_2 (top right), D_3 (bottom left) datasets. The MLEs are indicated with vertical solid lines and the true values with vertical dashed. The parameters (x -axis) are on log-scale and a 95% approximate cut-off point is included (horizontal grey line). Contour plot of the profile log-likelihood of $(\vartheta_5, \vartheta_6)$ parameters in D_2 (top right).

We assume that only the initial state of the d_1 -dimensional unobserved system $\boldsymbol{\eta}_0 \equiv \boldsymbol{\eta}_{t_0}$ is known and at the time instances $t_i, i \in (1, \dots, N_D)$ we have N_D vector-valued observations

$\mathbf{d}_i \equiv \mathbf{d}_{t_i}$, of d_2 elements each, from the observed system $\mathbf{o}_i \equiv \mathbf{o}_{t_i}$:

$$\begin{aligned} (\text{unobserved system}) \quad \boldsymbol{\eta}_i &= f(\boldsymbol{\eta}_{i-1}, \mathbf{w}_i, \boldsymbol{\vartheta}), \\ (\text{observed system}) \quad \mathbf{o}_i &= \mathbf{G}\boldsymbol{\eta}_i + \boldsymbol{\varepsilon}_i, \end{aligned} \tag{6.3}$$

where \mathbf{w}_i a vector-valued r.v. following a standard multivariate Normal distribution, $f(\boldsymbol{\eta}_{i-1}, \mathbf{w}_i, \boldsymbol{\vartheta})$ a non-linear vector-valued function with d_1 elements, \mathbf{G} is a constant $d_2 \times d_1$ matrix, $\boldsymbol{\varepsilon}_i, i \in (1, \dots, N_D)$ is a random vector of d_2 variables distributed according to $\text{Normal}(\mathbf{0}, \rho \mathbf{I}_{d_2})$, where \mathbf{I}_{d_2} is the unity $d_2 \times d_2$ matrix and ρ a scalar constant. We can assume a more general covariance matrix of $\boldsymbol{\varepsilon}_i$, but for simplicity we stick to the i.i.d. case. The matrix \mathbf{G} is the *observability* matrix and, in the case of partial observations, we assume that it is not of full rank, i.e. some elements of the state system $\boldsymbol{\eta}$ are not observed in the system \mathbf{o} . The function $f(\boldsymbol{\eta}_{i-1}, \mathbf{w}_i, \boldsymbol{\vartheta})$ represents the solution of the LNA process which combines the non-linear deterministic functions $\mathbf{y}_i(\cdot), \mathbf{m}_i(\cdot), \mathbf{S}_i(\cdot)$ which are solutions of the system of Ordinary Differential Equations (5.18–5.20) associated with the Linear Noise Approximation.

6.4.1 Kalman Filter Recursions

In the special case where the function $f(\cdot)$ of the system (6.3) is linear we can estimate the unobserved states conditional on the observations by employing the *Kalman Filter* method (§3, Anderson and Moore 2005). The solution is given as a recursive estimator which, at each iteration, repeats two steps: first *predicts* the mean ($\boldsymbol{\alpha}_{i|i-1}$) and variance ($\mathbf{P}_{i|i-1}$) of $\boldsymbol{\eta}_i$ conditional on \mathbf{d}_{i-1} and then *updates* the estimates ($\boldsymbol{\alpha}_i, \mathbf{P}_i$) conditional on \mathbf{d}_i . Additionally, we can use the Kalman Filter to write the Likelihood function (L) of the observed system.

In our case, we assume that the unobserved system in (6.3) follows the solution of the linear SDE of the LNA approximation and as our recursive estimates we use the equations for the Extended Kalman filter method (see for instance Anderson and Moore 2005, §8.2) combined with the solutions of the system of ODEs (5.18–5.20):

$$\begin{aligned} \boldsymbol{\alpha}_{i|i-1} &= N_T \mathbf{y}_i + \sqrt{N_T} \mathbf{m}_i \\ \mathbf{P}_{i|i-1} &= N_T \mathbf{S}_i \\ \mathbf{K}_i &= \mathbf{G}\mathbf{P}_{i|i-1}\mathbf{G}^T + \rho \mathbf{I}_{d_2} \\ \boldsymbol{\alpha}_i &= \boldsymbol{\alpha}_{i|i-1} + \mathbf{P}_{i|i-1}\mathbf{G}^T\mathbf{K}_i^{-1} [\mathbf{d}_i - \mathbf{G}\boldsymbol{\alpha}_{i|i-1}] \\ \mathbf{P}_i &= \mathbf{P}_{i|i-1} - \mathbf{P}_{i|i-1}\mathbf{G}^T\mathbf{K}_i^{-1}\mathbf{G}\mathbf{P}_{i|i-1}, \end{aligned} \tag{6.4}$$

where $\boldsymbol{\alpha}_{i|i-1}, \boldsymbol{\alpha}_i$ are the prior and posterior estimates of the mean respectively. Likewise, $\mathbf{P}_{i|i-1}, \mathbf{P}_i$ are the prior and posterior estimates of the covariance matrix and \mathbf{K}_i is the Kalman gain matrix. In (6.3) the subscript $i|i-1$ denotes that we condition on the latest observation, at time t_{i-1} , to estimate either the mean or the covariance at the time instance t_i . Furthermore, the subscript i denotes the posterior estimates, i.e. the estimates after observing \mathbf{d}_i . We notice that we can obtain the current distribution of the unobserved state:

$$\boldsymbol{\eta}_i | \mathbf{d}_{0:i}, \boldsymbol{\eta}_0 \sim \text{Normal}(\boldsymbol{\alpha}_i, \mathbf{P}_i), \quad \mathbf{d}_{0:i} \equiv (\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_i).$$

Similarly, the predictive distribution of the observed state is:

$$\mathbf{o}_i | \mathbf{d}_{0:(i-1)}, \boldsymbol{\eta}_0 \sim \text{Normal}(\mathbf{G}\boldsymbol{\alpha}_{i|i-1}, \mathbf{K}_i),$$

which coincides with the transition density:

$$P(\mathbf{o}_i | \mathbf{d}_{0:(i-1)}, \boldsymbol{\vartheta}) = \int_{-\infty}^{\infty} P(\mathbf{o}_i | \boldsymbol{\eta}_i, \rho) P(\boldsymbol{\eta}_i | \mathbf{d}_{i-1}, \boldsymbol{\vartheta}) dH_i = \text{Normal}(\mathbf{G}\boldsymbol{\alpha}_{i|i-1}, \mathbf{K}_i).$$

The latter expression allows us to write the Likelihood function as:

$$\begin{aligned} L(\boldsymbol{\vartheta} | \mathbf{d}_{0:i}) &= P(\mathbf{d}_0 | \boldsymbol{\vartheta}) \prod_{i=1}^n P(\mathbf{d}_i | \mathbf{d}_{i-1}, \boldsymbol{\vartheta}) \\ &= \delta(\mathbf{d}_0 - \mathbf{G}\boldsymbol{\eta}_0) \prod_{i=1}^n \text{Normal}(\mathbf{G}\boldsymbol{\alpha}_{i|i-1}, \mathbf{K}_i), \end{aligned} \quad (6.5)$$

where $\delta(\cdot)$, is the Dirac's delta function, the distribution of the initial (known) conditions, which is omitted in the next examples.

6.4.2 Lotka Volterra with partial observations

We illustrate the extension of LNA in an example based on the Lotka Volterra model of Section 6.3.1. We followed the modelling choices of Boys et al. (2008). In particular, we used the following kinetic parameters and initial state:

$$\mathbf{c} = (0.5, 0.0025, 0.3), \quad (\{\text{Prey}\}, \{\text{Predator}\}) = (79, 71),$$

and we simulated the evolution of the system until the time instance 40 which is plotted as a time-series in Figure 6.5. We obtained equidistant observations (with a time step 1.025 and $N_D = 40$), represented in Figure 6.5 as diamonds and points for the Prey and Predator species respectively. We proceeded to the estimation of the kinetic parameters by assuming

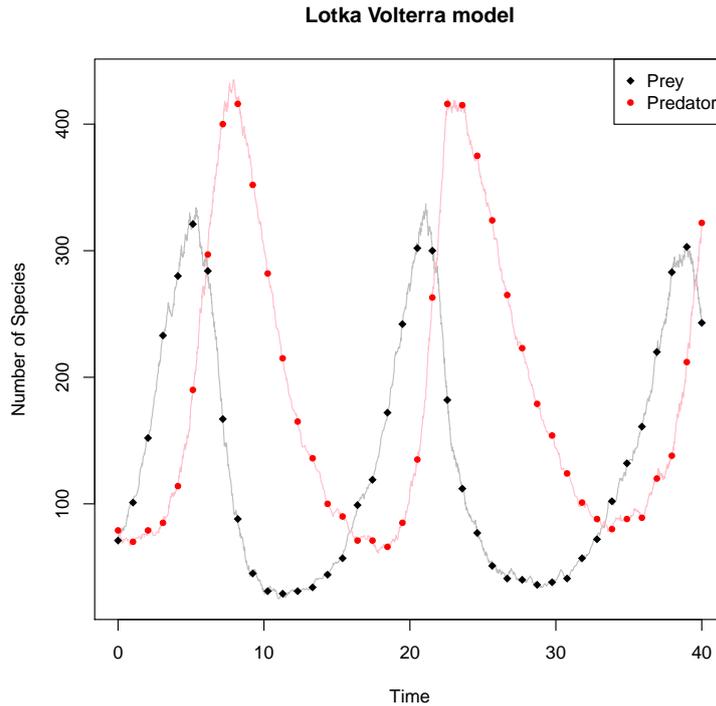


Figure 6.5: Simulated observations of Preys (*diamonds*) and Predators (*points*) of the Lotka Volterra model. The trajectories are indicated with solid faded lines.

two different observational scenarios. First, in the *full* dataset, we assumed that all species are observed and we applied the error-free methodology of the previous section. Second, in the *partial* dataset, the full initial state (\mathbf{d}_0) is observed for both species and in the subsequent observations ($\mathbf{d}_i, i > 0$) only the levels of the **Prey** species are observed. Therefore, for the partial dataset we chose the (1×2) observational matrix $\mathbf{G} = (0, 1)$. Finally, in both datasets, we did not consider adding measurement error and we assumed $\rho = 0$.

In Table 6.7 we summarise the MLEs together with the bounds of the 95% approximate (Wald's) Confidence Intervals (CIs). The confidence intervals have been derived for the $\log(\mathbf{c})$ parameters and then transformed to the original (\mathbf{c}) scale. In both cases, the MLEs are close to the true values and the corresponding CIs provide good coverage as well. In addition, the CIs become narrower when we considered the full dataset as a result of smaller standard errors. It should be noted that Boys et al. (2008) compared various Bayesian estimation methods under similar observational scenarios. Although the interpretation of the MLEs and their corresponding CIs in Table 6.7 is completely different from the posterior summaries of Boys et al. (2008), we can get an informal indication for their performance by looking at the

Observations	Estimates	c_1	c_2	c_3
Full	UCI	0.54591	0.00281	0.33826
	EST	0.51564	0.00264	0.31678
	LCI	0.48705	0.00249	0.29666
Partial	UCI	0.60965	0.00292	0.35955
	EST	0.54026	0.00254	0.30454
	LCI	0.47876	0.00222	0.25795

Table 6.7: Upper bounds of the Wald’s 95% Confidence interval (*UCI*), Maximum Likelihood Estimates (*EST*) and Lower bounds of the 95% Confidence interval (*LCI*), of the simulated dataset.

numerical values of the parameter estimates and the corresponding posterior means – which are close. To be more precise, the absolute differences between the MLEs and the true values are within the range of the corresponding absolute differences of the posterior means.

6.5 Discussion and future work

We have presented two likelihood-based methods to estimate the stochastic kinetic constants and their extension to handle observations from partially observed systems. These methods both depend on the LNA but differ on the initial conditions presented to the ODEs associated with the transition densities of the linear SDEs. In most cases, the Restarting method provided 95% CIs with coverage properties at least as good as or even better than those of the Non-Restarting method. An intuitive explanation, is that the RE method follows the stochastic process more closely by updating the initial conditions at every observation. The linear noise approximation assumes a large system size, and, unsurprisingly, this assumption is violated for datasets from systems of small size, for which both methods did not provide satisfactory estimates. Perhaps more surprisingly, the coverage properties worsened as the temporal density of the observations increased; a likely explanation is that more observations result in narrower confidence intervals, which in turn, highlight in the form of biased estimation

the poor approximation approximation.

The accuracy of RE estimates for kinetic constants of non-reversible equations presented a small improvement compared to Bayesian estimates of Golightly and Wilkinson (2005) which also applies for the ratios of the parameters of reversible equations. In contrast, the RE method did not incorporate any prior information which resulted, for the D_2 dataset, very wide confidence intervals for parameters associated with reversible equations. Fortunately, model identifiability issues can be diagnosed using profile likelihoods and identifiable parameterizations can be preferred instead.

The NR method was faster, in computational time, than the RE method when the BFGS numerical optimization routine was employed: in the Transcription model the mean time of a single dataset was 108 (± 11) and 65 (± 8) secs for the RE and NR methods respectively, considering datasets from a large system with 500 observations⁴. The replication of the experiment using the Nelder–Mead optimisation routine, yielded similar parameter estimates with a smaller mean computational time per dataset of: 15.3 (± 2.62) and 14.5 (± 2.59) secs for the RE and NR methods respectively.

Motivated by the Extended Kalman Filter method, we extended the RE/NR methods to support partially observed data subject to measurement error. The RE method handled quite well the case of a partially observed LV system, by providing MLEs close to the true values of the kinetic constants and CIs with satisfactory coverage which are in agreement to the Bayesian estimates of Boys et al. (2008) for a similar modelling scenario.

Finally, this Chapter presents many opportunities for applications of the LNA method as well as future extensions:

- Its employment in a Bayesian setting. Obviously, LNA can be used as an approximation to the discrete process. Also, since LNA's transition density is known and can be computed fast, it may be a reasonable proposal density for simulating trajectories of the state process.
- Its application to models motivated from other disciplines, provided that a suitable scaling of the parameters is available.

⁴Tested on a GNU/Linux system with eight Intel[®] Xeon[™] CPU 3 GHz processors. The optimizations were enabled but no parallelization was employed. The datasets of 100 and 50 observations were less computationally demanding.

- As a methodology to fit models based on ODEs. Both the Non-Restarting and the Restarting method provide two deterministic (ODEs) approximations for the state process: RE introduces a *local* model which tries to approximate the dynamics between each pair of observations and NR introduces a *global* model and which tries to approximate the overall dynamics.

Chapter 7

Implementation

In this chapter we discuss the `lnar` package, for the R statistical environment, which we developed to implement the LNA methodology of Chapter 6. The core functions of the package are coded in C programming language for efficiency but no knowledge of C is required for its use. In fact, after the specification of a biological model, `lnar` tries to automatically generate optimized C code in order to be passed at the LSODA ODEs solver which is the most demanding computationally aspect of the model. Alternatively, the code generation can be skipped by letting the user to specify the relative C code. After the generation of the relevant C code, the model can be fitted to data via a maximum likelihood estimation procedure. The `lnar` package is hosted in R-Forge (Theußl and Zeileis 2009) in the following url: <http://r-forge.r-project.org/projects/lnar/> and can be installed¹ by issuing the following command:

```
1 install.packages(c("inline", "Ryacas"))
2 install.packages("lnar", repos="http://R-Forge.R-project.org")
```

We split the exposition of `lnar` into two Sections: in Section 7.1 we discuss how to specify a model, and then, in Section 7.2 we show how to use it for inferential purposes. Also, in parallel to the general discussion, we consider two example applications of `lnar` based on the Lotka–Volterra and the prokaryotic transcription models. Both examples are available as demos for the `lnar` which can be run by issuing the commands `demo(lv)` and `demo(autoreg)` corresponding to the Lotka–Volterra and the prokaryotic transcription models respectively.

¹Currently is available only for the Linux platform.

7.1 Model Specification

In Section 5.5 we established that we can approximate the (discrete) state process with a new stochastic processes, induced by LNA, which has the following transition density:

$$X_t|X_0 = x_0 \sim \text{Normal} \left(Y_t N_T + \mathbf{m}_t \sqrt{N_T}, S_t N_T \right),$$

where $\mathbf{y}_t, \mathbf{m}_t, S_t$ are given as a single system of ODEs. The first step is to generate the code for these ODEs using the function `parsemod`. The functions `parsemod` expects as arguments a stoichiometry matrix (A^T), a vector of hazard functions ($\mathbf{h}(\mathbf{X}, \mathbf{c})$), the names of \mathbf{c} kinetic constants and the names of species. The final step of the model specification involves the compilation of the code either using the `compmod` function or the standard mechanisms of R (R Development Core Team 2010).

7.1.1 The function `parsemod`

Description

Given as input the reaction rates, stochastic constants and model constants, the function outputs the C code (via `yacas`) of the underlying ODEs. The system of odes express the macroscopic approximation (ODE) as well as the estimates for the instantaneous mean and covariance of the linear SDE from the LNA approximation.

Usage

```
parsemod(y, rfun, thetas, species, constants=NA)
```

Arguments

<code>y</code>	The stoichiometry matrix, note that the dimensions are assumed to be: $N_S \times N_R$.
<code>rfun</code>	A character vector with each elements expressing the reaction rates. Make sure that the <code>ryacas</code> package is able to parse the formula of each reaction.
<code>thetas</code>	A character vector which denotes the stochastic constant names associated with each reaction.
<code>species</code>	A character vector which denotes the species names.

constants A character vector which denotes the model constants. The model constants are substituted in the C code by their numeric value.

Return values

Returns a list with the following elements:

ccode The actual C code as text.

cspecies A character vector of the species names in the C code and their corresponding model names are given in the **names** attribute.

cthetas A character vector of the names of the stochastic constants in the C code and their corresponding model names are given in the **names** attribute.

Cov A character vector of the functions names that corresponds the upper triangular matrix (given as the names attribute) of the instantaneous variance-covariance matrix.

Means A character vector of the functions names that corresponds to the instantaneous means (given as the names attribute).

Orders A numerical vector indicating the order of each reaction.

7.1.1.1 Details

The derivation of the ODEs $(\mathbf{y}_t, \mathbf{m}_t, S_t)$ is accomplished with the help of **Ryacac** package (Goedman et al. 2010), an interface between R and the symbolic computer algebra system **Yacas**. The derived ODEs are joined together, i.e. treated as a single system, and are expressed as C code. In particular, they are expressed as a C function of the following form:

```
1 double * name (double * t, double * y, double * fout, double * vthetas)
```

Listing 7.1: The of the function expressing system of ODEs.

Where t is a pointer to the variable denoting the state's time, y is a pointer to the state of the $(\mathbf{y}_t, \mathbf{m}_t, S_t)$ ODEs, $fout$ is a pointer that returns the numerical values of the derivatives w.r.t time and $vthetas$ is a pointer to the values of the kinetic constants.

The hazard functions are assumed to express reactions of zeroth to second orders. It is possible to specify more general hazard functions as long as they can be parsed from **lnar**, **Yacas** and the C compiler. Alternatively, they can be specified directly in the C code, i.e. the element **ccode** of the returned list.

7.1.2 The function `compmod`

Description

Compiles the generated C source code of a parsed model using the **inline** package.

Usage

```
compmod(cout, name = "derivs")
```

Arguments

cout A parsed model.
name A string indicating the name of the compiled function, defaults to "derivs".

Details

Uses the **inline** package to compile the generated C code.

Return values

Returns a compiled function named as the **name** argument and is included in R's environment automatically.

7.1.3 Examples

As mentioned previously we consider the Lotka–Volterra and prokaryotic transcription models to illustrate the usage of `lnar`. First, we use the biological descriptions of the Section 3.3.1.1 to specify, for each model, the stoichiometric matrix, the hazard the names of the parameters of interest and the names of the species. Then, we call the `parsemod` function using the previous defined variables as arguments to generate the C code, which in turns is compiled with the help of the `compmod` function.

7.1.3.1 Lotka-Volterra

```
1 require(lnar)
2 tt <- matrix(c(1, -1, 0, 0, 1, -1), nrow=2, ncol=3, byrow=TRUE)
3 rfun <- c("con1 * Prey", "con2 * Prey * Predator", "con3 * Predator")
4 thetas <- paste("con", 1:3, sep="")
5 species <- c("Prey", "Predator")
```

```

6 cout <- parsemod(tt, rfun, thetas, species)
7 compmod(cout, "derivs") # Compile the model

```

Listing 7.2: Specification of the Lotka-Volterra Model.

7.2 we specify the Lotka–Volterra model: the variable `tt` is the stoichiometric matrix, `rfun` is a character vector with the hazard rates, `thetas` contains the namers of the parameters and the variable `species` contains the corresponding names of the species. The parsed model is saved in the variable `cout` and contains the C implementation of the system of ODEs defined in Table 5.1. Note that if the names of the species or the parameters do not match the ones of `rfun` the generated model will be inaccurate. The last line compiles the c code and loads it to the R environment as the `derivs` variable, *without* the need of an assignment (`<-` or `=`).

7.1.3.2 Transcription Model

```

1 library(lnar)
2 #Number of Species: 4
3 species=c('RNA', 'P', 'P2', 'DNA')
4 #Number of Parameters: 8
5 params=c('k1', 'r1', 'k2', 'k3', 'k4', 'r4', 'k5', 'k6')
6 stoich=matrix(c( 0,0,1,0,0,0, -1,0,
7                 0,0,0,1, -2,2,0, -1,
8                 -1,1,0,0,1, -1,0,0,
9                 -1,1,0,0,0,0,0,0),4,8,byrow=TRUE)
10 #Number of Reactions: 8
11 reac=c('k1*DNA*P2',
12        'r1*(10-DNA)',
13        'k2*DNA',
14        'k3*RNA',
15        'k4*0.5*P*P',
16        'r4*P2',
17        'k5*RNA',
18        'k6*P')
19 #generate the model and c code
20 modell<-parsemod(stoich, reac, params, species)
21 compmod(modell, "tder") #Compile model

```

Listing 7.3: Specification of the prokaryotic transcription model.

We specify the Transcription Model in Listing 7.3. In this example, the `stoich` variable is the stoichiometric matrix, `species` contain the names of the species, `params` variable contains the names of the kinetic constants and `reac` variable contains the vector of hazard function. Finally the `model1` variable contains the generated model and `tder` the compiled derivatives function.

7.2 Model Usage

After the generation and the compilation of the C code, we are ready to use it. The function `lnalik` calculates the log-likelihood given a set of data-points. The function `calcdens`, given an initial state, estimates the parameters (mean and variance) and the value of the transition density at a number of time-points. The function `optmod` fits the LNA approximation to a dataset using a maximum likelihood estimation procedure based on numerical optimization.

7.2.1 Functions

7.2.1.1 The function `lnalik`

Description

Estimates the log-likelihood of the LNA approximation.

Usage

```
lnalik(cout, nthetas, mydata, syssize = sum(mydata[1, -1]),
      relerr = 1e-09, abserr = 1e-09, method = 0, dfunction)
```

Arguments

<code>cout</code>	The parsed model.
<code>nthetas</code>	The vector of the parameters.
<code>mydata</code>	Either a matrix or a data frame of the data to be evaluated. The first column is assumed to correspond to the time of each observation.
<code>syssize</code>	Optional, a scalar indicating the system size.
<code>relerr</code>	Optional, a scalar indicating the relative error for the ODE solver.
<code>abserr</code>	Optional, a scalar indicating the absolute error for the ODE solver.
<code>method</code>	Optional, a scalar with possible options:

- 0: Restarting method using concentrations. The parameters are assumed to be scaled, i.e. ϑ .
- 1: Restarting method using number of molecules. The parameters are assumed to be un-scaled, i.e. \mathbf{c} .
- 3: Non-Restarting method using concentrations. The parameters are assumed to be scaled as well.

`dfunction` The compiled model.

Return values

Returns the estimated log-likelihood.

7.2.1.2 The function `calcdens`

Description

The system of ODEs is solved subject to initial conditions and the estimates of the mean, the variance, the macroscopic equations and the transition density are returned.

Usage

```
calcdens(initdata, edata=NA, tstart=0, tend,
         initode=NA, initmean=rep(0,length(initdata)),
         initvar=rep(0,length(initdata)*(length(initdata)+1)/2),
         thetas, relerr=1e-9, abserr=1e-9, syssize, dfunction)
```

Arguments

`initdata` A numerical vector indicating the initial point. It is unscaled, e.g. expressed as number of molecules.

`edata` Optional, a numerical vector indicating the ending point. It also is unscaled, e.g. expressed as number of molecules.

`tstart` The starting time, defaults to 0.

`tend` Either a vector or a scalar with the time-points to be estimated.

`initode` Optional, the initial values of the macroscopic ODEs, defaults to the scaled `initdata`, e.g. the concentration of the species.

<code>initmean</code>	A numerical vector indicating the initial values for the means. Defaults to a vector of zeroes, otherwise it is expected to be scaled by the inverse of the square root of the system size.
<code>initvar</code>	Either a matrix indicating the initial Variance-Covariance matrix or a vector representing the upper diagonal (including the main diagonal) following a row orientation. Defaults to a matrix of zeroes and is expected to be on the scale of macroscopic ODEs.
<code>thetas</code>	A numerical vector with the parameter values.
<code>relerr</code>	Numerical, the relative error for the numerical ordinary differential equations (ODEs) solver.
<code>abserr</code>	Numerical, the absolute error for the numerical ordinary differential equations (ODEs) solver.
<code>sysize</code>	Numerical, indicating the system size.
<code>dfunction</code>	The compiled function, given as a loaded dynamic library in R.

Return values

A list of the following components, estimated at each `tend` time-point:

<code>Time</code>	The time instance of the estimates.
<code>ODE</code>	The value of the ODE equation (the macroscopic model)
<code>MEAN</code>	The mean of the SDE process.
<code>VAR</code>	The covariance of the SDE process.
<code>prob</code>	Optional, expresses the estimated transition probability density, available only if <code>ifedata</code> is defined.

Note

All densities are conditioned on the initial time-point `tstart`. The `MEAN` and `VAR` elements are not at the same scale but they depend on the scale of the initial values. We assume that the initial values are given as number of molecules.

7.2.1.3 The function `optmod`

Description

Fits the compiled model to a given dataset using a numerical Maximum Likelihood Estimation procedure.

Usage

```
optmod(cout,nthetas, mydata, maxiter=300,  
       syssize=sum(mydata[1,-1]), tcrit=.0001,  
       relerr=1e-9, abserr=1e-9, hessianh=1e-4,  
       method=1, usebfgs=0, dfunction)
```

Arguments

<code>cout</code>	The parsed model containing the C code and the name relations.
<code>nthetas</code>	A numerical vector with the initial values for the scaled parameters to be optimized.
<code>mydata</code>	A <code>data.frame</code> or a matrix with the data. The first column must indicate the time of the observations.
<code>maxiter</code>	Numerical, indicated the maximum number of iterations for the optimization algorithm.
<code>syssize</code>	Numerical, the system size defaults to the initial population.
<code>tcrit</code>	Numerical, the convergence criterion for the optimization algorithm.
<code>relerr</code>	Numerical, the relative error for the numerical ordinary differential equations (ODEs) solver.
<code>abserr</code>	Numerical, the absolute error for the numerical ordinary differential equations (ODEs) solver.
<code>hessianh</code>	Numerical, indicates the approximation step for the central differences calculations of the Hessian matrix.
<code>method</code>	Numerical which takes the following integer values: <ul style="list-style-type: none">• 0: Restarting Method using number of molecules.

- 1: Restarting Method using concentrations.
 - 3: Non-Restarting Method using concentrations.
- usebfgs** Specify whether to use the BFGS algorithm (1), or the default Nelder-Mead simplex algorithm (0)
- dfunction** The compiled function, given as a loaded dynamic library in R.

Details

By default the L-BFGS-B optimization procedure is employed (see `optim` for more details).

Return values

A list with the following elements:

- UP** The upper confidence bound.
- ES** The MLEs
- LO** The lower confidence bound.

Note

Note that the `tcrit` has a different interpretation for the Nelder-Mead algorithm and different for BFGS.

7.2.2 Examples

In this Section we try to fit each model to a corresponding dataset. We consider two datasets to fit, one for each model, which correspond to the large and small systems configurations we considered in Section 6.3 for the Lotka–Volterra and the Transcription model respectively. The dataset of Lotka–Volterra Model is inputted in the example and the dataset of the Transcription model is contained in the package’s distributed datasets. The initial parameter values are expected to be in terms of ϑ , for any method choice.

7.2.2.1 Lotka-Volterra Model (Continued)

```

8 mydata<-c(0.0, 5000.0, 3000, 1, 5989, 2992, 2, 7165, 3107, 3, 8534, 3306,
9           4, 10041, 3709, 5, 11624, 4265, 6, 13306, 5181, 7, 14741, 6492,
10          8, 15867, 8337, 9, 16025, 10981)

```

```

11 mydata2 <- matrix(mydata,10,3,byrow=TRUE) # Example dataset
12 ntheta<-c(.4,.1,0.4) # The initial parameter values
13
14 #Find the Maximum Likelihood Estimates and Wald CIs
15 (run1<-optmod(cout,ntheta=ntheta, mydata=mydata2, method=1,
16             maxiter=300, tcrit=1e-5, relerr=1e-9,
17             abserr=1e-9, hessianh=1e-4,
18             dfunction=derivs))
19
20 ##Calculate the transition density 's parameters at t=1
21 calcdens(mydata2[1,],tend=1,theta=run1$ES,ssize=8000,dfunction=derivs)
22
23 ##Evaluate the log-likelihood at the mles
24 (ll<-lnalik(cout,ntheta=run1$ES, mydata=mydata2, method=1,
25           relerr=1e-9, abserr=1e-9,
26           dfunction=derivs) )

```

Listing 7.4: Maximum Likelihood Estimation for the Lotka-Volterra model.

Listing 7.4 fits the Lotka Volterra model to the mydata2 dataset. We choose the (0.4, 0.1, 0.4) as our initial parameter values, contained in variable ntheta. We set the option method=1, at the arguments of optmod function, in order to work with the concentrations. Next, we calculate (calcdens) the transition density and its parameters at $t = 1$ conditional on the first set of observation of the dataset (mydata2[1,]). Finally, we evaluate the log-likelihood (lnalik) for the values of the mles that we estimated previously.

7.2.2.2 Transcription Model (Continued)

```

22 ##load the data
23 data(ardata)
24 ##We set all c's to .2 for our initial values
25 ntheta <- rep(.2,8)
26 ntheta[1]=ntheta[1]*34 # corresponds to a 2nd order reaction
27 ntheta[5]=ntheta[5]*34 # corresponds to a 2nd order reaction
28 ##Optimize with Nelder-Mead
29 (modellopt<-optmod(model1,ntheta=ntheta, mydata=ardata, method=0,
30                 maxiter=1800, tcrit=1e-5, relerr=1e-12,
31                 abserr=1e-12, hessianh=1e-4,ssize=34,
32                 dfunction=tder))

```

```

33 ##Continue with BFGS
34 (model2opt<-optmod(model1,nthetas=modellopt$ES, mydata=ardata, method=0,
35     maxiter=25, tcrit=1, relerr=1e-12,
36     abserr=1e-12, hessianh=1e-4, syssize=34, usebfgs=1,
37     dfunction=tder))
38 ##Calculate the transition density 's parameters at t=1
39 calcdens(as.numeric(ardata[1,]), tend=1,
40     thetas=model2opt$ES, syssize=34, dfunction=tder)
41
42 ##Evaluate the log-likelihood at the mles
43 (ll<-lnalik(model1,nthetas=model2opt$ES, mydata=ardata, method=0,
44     relerr=1e-9, abserr=1e-9,
45     dfunction=tder) )

```

Listing 7.5: Maximum Likelihood Estimation for the Transcription Model.

Listing 7.5 fits the prokaryotic transcription model to the dataset `ardata`. The initial values (`nthetas`) of the parameters \mathbf{c} are set equal to `.2` and converted to $\boldsymbol{\vartheta}$. Then the `optmod` is called once to find the MLEs using the Nelder–Mead algorithm. The termination criterion (`tcrit`) is 10^{-5} , i.e. when the maximum distance between the points of the simplex and its center becomes smaller than 10^{-5} the algorithm ends. Then, we repeat the `optmod` call but this time we select the BFGS method for the numerical optimization. BFGS uses a different termination criterion: when the (Euclidean) norm of the gradient becomes less than 1 the algorithm stops.

In this example we choose to work with the number of molecules (`method=0`) because in Section 3.3.1.2 we have specified the constant $k = \{\text{DNA} \cdot \text{P}_2\} + \{\text{DNA}\}$ in term of molecules ($k = 10$). Choosing `method=1` would make the `optmod` function to use concentrations ($\{\mathbf{X}\}/N_T$) instead of number of molecules ($\{\mathbf{X}\}$), but in this case, the rate at line 12 in Listing 7.3 must be rewritten in terms of concentrations: `r1*((10/34) - DNA)`.

Finally, as with the Lokta–Volterra model previously, we estimate (`calcdens`) the transition density conditional on the first observation of the `ardata` dataset and we evaluate (`lnalik`) the log-likelihood at the `mles` values, found by the BFGS algorithm.

7.3 Discussion

In this Section we show how to use `lnar` package to specify and analyze two biological models. The advantage of the `lnar` package is that generates, implicitly, the C code for the solution of

the ODEs related to LNA. Therefore, the computations performance benefits from the use of compiled code, which offers a great improvement compared to the performance of interpreted code. In addition, the `lnar` package can also be used to fit a deterministic model based on ODEs. The restarting method fits a *local* model, i.e. restarts at every observation, while the non-restarting method fits a *global*.

Although, the `lnar` package has been tested only for simple reactions, presented in Section 3.2.1, it can accept hazard functions of arbitrary type, as long as, the underlying code can be parsed from `lnar` package. Nevertheless, one can specify the ODEs directly in C to overcome this limitation. The `lnar` package is far from complete and many features can be added: e.g. support for incomplete and noisy datasets, profile likelihoods, parallelization.

Chapter 8

Conclusion

In the first part of this thesis, We studied the possibility of extending the Exact Algorithm to multidimensional SDEs which approximate gene auto-regulatory systems. We showed that it is non-trivial to find a suitable unit-variance transformation. Even a scalar SDE, expressing a simple protein dimerization network, has a unit-variance transformation which is computationally difficult to handle. For the multidimensional case, we expressed the unit-variance transformation as a system of Partial Differential Equations and we showed that the EA can be applied to a class of diffusions wider than the one implied by the definition of reducibility (Ait-Sahalia 2008). To support our claim, we provided a counterexample of a diffusion that can be transformed to an SDE with non-constant diffusion coefficient but with a constant (unit matrix) instantaneous variance.

In the second part of the thesis we presented the Linear Noise Approximation (LNA) with applications to gene auto-regulatory networks. First, we introduced a reparameterization of the kinetic constants that leads to the LNA and we proved that converges to a linear diffusion process. Secondly, we investigated to what degree the approximation to the system dynamics is satisfactory, under different modelling scenarios. We concluded that the LNA's simulation performance was as good as or even better than the performance of the Chemical Langevin Equation's approximation. But as we have seen, both approximations are weak for the cases where the probability density is concentrated in few points. Next, we employed the LNA for inferential purposes and, more specifically, we used two methods to derive the LNA's likelihood: the Restarting (RE), which we proposed, and Non-Restarting (NR) method proposed by Komorowski et al. (2009). These two methods are similar and they only differ on the initial conditions that they use to solve the ODEs involved in the estimation of the

transition density. We compared these two methods in a series of simulated datasets using likelihood ratios statistics and the coverage of the approximated confidence intervals. We concluded that the performance of RE was better than the performance of NR. In addition, we extended RE method to handle the case of partially observed systems with good inferential power in the example of a simple dynamic model (Lokta–Volterra). Finally, we presented `lnar`, an implementation of the LNA method for the R statistical environment, which facilitates the LNA analysis without compromising its performance.

Bibliography

- Abramowitz, M. and Stegun, I. A.: 1964, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, ninth dover printing, tenth gpo printing edn, Dover, New York.
- Aït-Sahalia, Y.: 2008, Closed-form likelihood expansions for multivariate diffusions, *The Annals of Statistics* **36**(2), 906–937.
- Anderson, B. D. O. and Moore, J. B.: 2005, *Optimal Filtering*, Dover Publications.
- Arkin, A., Ross, J. and McAdams, H. H.: 1998, Stochastic kinetic analysis of developmental pathway bifurcation in phage λ -infected escherichia coli cells, *Genetics* **149**, 1633–1648.
- Beskos, A., Papaspiliopoulos, O. and Roberts, G.: 2009, Monte carlo maximum likelihood estimation for discretely observed diffusion processes, *The Annals of Statistics* **37**(1), 223–245.
- Beskos, A., Papaspiliopoulos, O. and Roberts, G. O.: 2008, A factorisation of diffusion measure and finite sample path constructions, *Methodology and Computing in Applied Probability* **10**(1), 85–101.
- Beskos, A., Papaspiliopoulos, O., Roberts, G. O. and Fearnhead, P.: 2006, Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processes (with discussion), *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 333–382.
- Beskos, A. and Roberts, G. O.: 2005, Exact simulation of diffusions, *Annals of Applied Probability* **15**(4), 2422–2444.
- Booger, F., Bruggeman, F. J., Hofmeyr, J.-H. S. and Westerhoff, H. (eds): 2007, *Systems Biology: Philosophical Foundations*, first edn, Elsevier.

- Bower, J. M. and Bolouri, H. (eds): 2001, *Computational Modeling of Genetic and Biochemical Networks*, first edn, MIT Press.
- Boys, R. J., Wilkinson, D. J. and Kirkwood, T. B. L.: 2008, Bayesian inference for a discretely observed stochastic kinetic model, *Statistics and Computing* .
- Butcher, J. C.: 2008, *Numerical Methods for Ordinary Differential Equations*, 2 edn, Wiley.
- Chong, L. and Ray, L. B.: 2002, Whole-ist Biology, *Science* **295**(5560), 1661.
- Coddington, E. A. and Levinson, N.: 1955, *Theory of Ordinary Differential Equations*, McGraw–Hill.
- Dacunha-Castelle, D. and Florens-Zmirou, D.: 1986, Estimation of the coefficients of a diffusion from discrete observations, *Stochastics* **19**(4), 263.
- Doss, H.: 1977, Liens entre équations différentielles stochastiques et ordinaires, *Ann. Inst. Henri Poincaré* **13**(2), 99–125.
- Durrett, R.: 1996, *Stochastic Calculus: A Practical Introduction*, CRC Press.
- Ethier, S. N. and Kurtz, T. G.: 1986, *Markov processes: characterization and convergence*, Wiley.
- Evans, L. C.: 1998, *Partial Differential Equations*, Graduate Studies in Mathematics, American Mathematical Society.
- Ferm, L., Lötstedt, P. and Hellander, A.: 2008, A hierarchy of approximations of the master equation scaled by a size parameter, *Journal of Scientific Computing* **34**(2), 127–151.
- Finkenstadt, B., Heron, E. A., Komorowski, M., Edwards, K., Tang, S., Harper, C. V., Davis, J. R. E., White, M. R. H., Millar, A. J. and Rand, D. A.: 2008, Reconstruction of transcriptional dynamics from gene reporter data using differential equations, *Bioinformatics* **24**(24), 2901–2907.
- Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Alken, P., Booth, M. and Ross, F.: 2009, *GNU Scientific Library Reference Manual*, third edn, Network Theory Limited.

- Gibson, M. A. and Mjolsness, E.: 2001, Modeling the activity of single genes, *in* J. M. Bower and H. Bolouri (eds), *Computational Modeling of Genetic and Biochemical Networks*, first edn, MIT Press, chapter 1, pp. 1–48.
- Gillespie, C.: 2009, Moment-closure approximations for mass-action models, *IET Systems Biology* **3**(1), 52–58.
- Gillespie, D. T.: 1976, A general method for numerically simulating the stochastic time evolution of coupled chemical reactions, *Journal of Computational Physics* **22**, 403–434.
- Gillespie, D. T.: 1977, Exact stochastic simulation of coupled chemical reactions, *The Journal of Physical Chemistry* **81**(25), 2340–2361.
- Gillespie, D. T.: 1991, *Markov Processes: An Introduction for Physical Scientists*, Academic Press.
- Gillespie, D. T.: 1992, A rigorous derivation of the chemical master equation, *Physica A* **188**(1-3), 404–425.
- Gillespie, D. T.: 2000, The chemical langevin equation, *The Journal of Chemical Physics* **113**(1), 297–306.
- Goedman, R., Grothendieck, G., Højsgaard, S. and Pinkus, A.: 2010, *Ryacas: R interface to the Yacas computer algebra system*. R package version 0.2-10.
URL: <http://CRAN.R-project.org/package=Ryacas>
- Golightly, A. and Wilkinson, D.: 2006, Bayesian sequential inference for stochastic kinetic biochemical network models, *Journal of Computational Biology* **13**(3), 838–851.
- Golightly, A. and Wilkinson, D. J.: 2005, Bayesian inference for stochastic kinetic models using a diffusion approximation, *Biometrics* **61**(3), 781–788.
- Golightly, A. and Wilkinson, D. J.: 2007, Bayesian inference for nonlinear multivariate diffusion models observed with error, *Computational Statistics & Data Analysis* .
- Goss, P. J. E. and Peccoud, J.: 1998, Quantitative modeling of stochastic systems in molecular biology by using stochastic petri nets, *Proceedings of the National Academy of Sciences of the United States of America* **95**(12), 6750–6755.

- Hairer, E. and Wanner, G.: 1991, *Solving Ordinary Differential Equations II: Stiff and Differential–Algebraic problems*, Springer–Verlag.
- Henderson, D. A., Boys, R. J., Proctor, C. J. and Wilkinson, D. J.: 2010, Linking systems biology models to data: a stochastic kinetic model of p53 oscillations, *in* A. O’Hagan and M. West (eds), *The Oxford Handbook of Applied Bayesian Analysis*, Oxford University Press.
- Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P. and Kummer, U.: 2006, COPASI—a COmplex PAthway SIMulator, *Bioinformatics* **22**(24), 3067–3074.
- Karatzas, I. and Shreve, S. E.: 1991, *Brownian Motion and Stochastic Calculus*, 2 edn, Springer.
- Kloeden, P. and Platen, E.: 1995, *Numerical solution of stochastic differential equations*, Springer-Verlag.
- Kolda, T. G., Lewis, R. M. and Torczon, V.: 2003, Optimization by direct search: New perspectives on some classical and modern methods, *SIAM Review* **45**(3), 385–482.
- Komorowski, M., Finkenstädt, B., Harper, C. and Rand, D.: 2009, Bayesian inference of biochemical kinetic parameters using the linear noise approximation, *BMC Bioinformatics* **10**(1).
- Kurtz, T. G.: 1972, The relationship between stochastic and deterministic models for chemical reactions, *The Journal Chemical Physics* **57**(7), 2976–2978.
- Kurtz, T. G. and Protter, P.: 1991, Weak limit theorems for stochastic integrals and stochastic differential equations, *The Annals of Probability* **19**(3), 1035–1070.
- Lotka, A. J.: 1925, *Elements of Physical Biology*, Williams and Wilkins, Baltimore.
- McAdams, H. and Arkin, A.: 1997, Stochastic mechanisms in gene expression, *Proceedings of the national academy of sciences* **94**, 814–819.
- Mckinnon, K. I. M.: 1999, Convergence of the Nelder-Mead simplex method to a non-stationary point, *SIAM J. OPTIM* **9**, 148—158.
- Nelder, J. A. and Mead, R.: 1965, A simplex method for function minimization, *Computer Journal* **7**, 308–313.

- Øksendal, B.: 2005, *Stochastic Differential Equations: An Introduction with Applications (Universitext)*, sixth edn, Springer.
- Papaspiliopoulos, O. and Roberts, G. O.: 2008, Retrospective markov chain monte carlo methods for dirichlet process hierarchical models, *Biometrika* **95**(1), 169–186.
- Petzold, L.: 1983, Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations, *SIAM Journal on Scientific and Statistical Computing* **4**(1), 136–148.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P.: 2007, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3 edn, Cambridge University Press.
- Puri, M. L. and Sen, P. K.: 1971, *Nonparametric methods in multivariate analysis*, Wiley: New York.
- R Development Core Team: 2010, *Writing R Extensions*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-11-9.
URL: <http://www.R-project.org>
- Robert, C. P. and Casella, G.: 2004, *Monte Carlo Statistical Methods*, second edn, Springer Verlag, New York.
- Rogers, L. C. G. and Williams, D.: 1988, *Diffusions, Markov processes, and martingales. Vol. 2*, Cambridge Mathematical Library, Cambridge University Press, Cambridge.
- Ruttur, A., Sanguinetti, G. and Opper, M.: 2010, Approximate inference for stochastic reaction processes, in N. D. Lawrence, M. Girolami, M. Rattray and G. Sanguinetti (eds), *Learning and Inference in Computational Systems Biology*, MIT Press, chapter 10, pp. 241–259.
- Singh, A. and Hespanha, J. P.: 2007, Stochastic analysis of gene regulatory networks using moment closure, *American Control Conference, 2007. ACC'07*, pp. 1299–1304.
- Stroock, D. W. and Varadhan, S. R. S.: 1979, *Multidimensional Diffusion Processes*, Springer, New York.

- Theußl, S. and Zeileis, A.: 2009, Collaborative Software Development Using R-Forge, *The R Journal* **1**(1), 9–14.
- Tian, T., Xu, S., Gao, J. and Burrage, K.: 2007, Simulated maximum likelihood method for estimating kinetic rates in gene expression, *Bioinformatics* **23**(1), 84.
- Uri M. Ascher, L. R. P.: 1998, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM.
- van Kampen, N. G.: 2007, *Stochastic Processes in Physics and Chemistry*, 3rd edn, Elsevier.
- Volterra, V.: 1926, Variazioni e fluttuazioni del numero d'individui in specie animali conviventi, *Mem. R. Accad. Naz. dei Lincei. Ser. VI* **2**, 31–113.
- Wilkinson, D. J.: 2006, *Stochastic Modelling for Systems Biology*, Chapman & Hall/CRC.
- Wilkinson, D. J.: 2010, Parameter inference for stochastic kinetic models of bacterial gene regulation: a bayesian approach to systems biology, in S. Bayarri, J. O. Berger, J. M. Bernardo, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West (eds), *Bayesian Statistics*, Vol. 9, Oxford University Press.