Variance reduction techniques for chemical reaction network simulation



Casper H. L. Beentjes
New College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy
Hilary Term 2020

Acknowledgements

What started as a plan to discover the secrets of the Oxford bubble through an MSc has become quite the long haul journey with this DPhil thesis as its pièce de résistance. Looking back, I am indebted to countless people and institutions for their support and advice along the way. Too many, I realise, to recognise all individually; my apologies, but your aid and friendship is acknowledged nonetheless.

First of all, I would like to express my deepest gratitude to my supervisor Ruth Baker for her guidance through good and bad times, her enthusiasm and care in supervising, and above all for supporting me to become an independent researcher. A wise man once wrote *non omnia possumus omnes*, but thanks to her help I know a thing or two about academic research at last.

I would like to gratefully acknowledge the funding received from the Clarendon Fund and New College. Additional support for travel and collaboration was provided by the Mathematical Institute, Oxford; the Society for Mathematical Biology; the Isaac Newton Institute for Mathematical Sciences, Cambridge; the International Centre for Mathematical Sciences, Edinburgh; UK MultiScale Biology network and the MATRIX institute, Australia. Trinity College also deserves a special mention for giving me the opportunity to teach during my final years and, as a bonus, for welcoming me to their fantastic Senior Common Room.

In addition, I must thank New College for providing me with a (forever) home (not just in the literal sense) and the most stimulating postgraduate environment I could have ever wished for. I would also like to thank St. Catz MCR football team, the NC croquet swaz team and Kees' infectious love of cycling for keeping me moving rather than slowly glueing to my desk. Furthermore, many thank yous go to those friends whose paths I was lucky enough to cross at some point during our time in Oxford. In particular, I would like to thank Matteo for his friendship, time, helpful discussions and insights (maybe, one day, we will write a paper together!).

Pa, Ma, bedankt voor alle steun in het najagen van mijn dromen, zowel academisch als daarbuiten. Zonder jullie goede zorgen had ik nooit zo ver kunnen komen.

Finally, Lauren, you have been there all along the way for me in more ways than you can possibly imagine. Keeping me sane in times of despair and sharing with me life in all of its foolishness in times of joy, thank you.

And now for something completely different.

The bottom line for mathematicians is that the architecture has to be right. In all the mathematics that I did, the essential point was to find the right architecture. It's like building a bridge. Once the main lines of the structure are right, then the details miraculously fit. The problem is the overall design.

Freeman Dyson (1923-2020)

Abstract

In recent decades stochastic models have become an indispensable tool when analysing quantitative biological data, which are often subject to noise, both from intrinsic and extrinsic sources. Though such models generate richer, and perhaps more realistic, behaviour than their counterpart deterministic models, they are also inherently more difficult to study. Since exact solutions describing the dynamics of stochastic models are scarce, one is often forced to resort to simulation of these models using Monte Carlo methods, which are generally easy to implement. However, the output of such computational methods is itself subject to statistical error, and consequently it is often computationally demanding to get numerical predictions from stochastic models to a satisfactory degree of accuracy.

In this thesis we therefore consider methods that improve the efficiency of stochastic simulation algorithms, specifically in the context of well-mixed chemical reaction network models. We primarily achieve this by utilising variance reduction techniques to reduce statistical error in the outputs of simulation algorithms without incurring extra computational costs. In particular, as our key contribution, we study three different ideas leading to more efficient simulation approaches.

Firstly, we derive an efficient implementation of the uniformisation technique for continuous-time Markov chain models. Using stratified sampling we can leverage the extra structure in a uniformised chemical reaction network to subsequently produce a more efficient simulation method. In addition we show how to use the uniformisation technique to reduce statistical error via what is effectively a low-pass filter.

Secondly, we study the synthesis of quasi-Monte Carlo methodology with simulation methods for chemical reaction network models. Whilst such methods can significantly improve efficiency, we show that, due to the problem structure of chemical reaction network models, care must be taken in the specific implementation choices, both regarding the quasi-Monte Carlo techniques and simulation methods.

Lastly, we provide a significantly improved method for generating unit-rate Poisson processes via Poisson bridge constructions in combination with antithetic sampling. This method is subsequently used to improve the efficiency of standard simulation algorithms for chemical reaction networks.

Contents

1 Introduction **1**
1.1 Challenges . . . 2
1.2 Aims and outline . . . 3

2 Stochastic models and simulation **5**
2.1 Mathematical models of chemical reactions . . . 5
2.2 Simulation of well-mixed systems . . . 12
2.2.1 Exact simulation algorithms . . . 13
2.2.2 Approximate simulation algorithms . . . 14
2.3 Monte Carlo methods . . . 18
2.3.1 Distribution estimation . . . 20
2.3.2 Computational complexity and efficiency . . . 24
2.4 Variance reduction methodology . . . 31
2.4.1 Correlation based techniques . . . 32
2.4.2 Coupling based techniques . . . 44
2.4.3 Importance sampling . . . 53
2.5 Outlook . . . 56
Chapter appendix
2.A Analytic moments for the τ-leap method in systems with affine propensity functions . . . 57
2.B Histograms and self-distance . . . 64
2.C Probability generating functions for test systems . . . 70

3 Uniformisation methods **75**
3.1 Uniformisation . . . 75
3.2 Efficient simulation . . . 80
3.2.1 Firing virtual reactions . . . 80
3.2.2 Adapting the uniformisation rate . . . 83
3.3 Variance reduction via stratification . . . 87
3.3.1 Stratification of the number of reactions . . . 88
3.3.2 Examples . . . 91
3.4 Transient analysis . . . 98
3.4.1 A weighted uniformisation method . . . 98
3.4.2 Examples . . . 103
3.5 Discussion . . . 110
Chapter appendix

3.A Unbiased adapting of the uniformisation rate . . . 114
3.B Time-inhomogeneous models . . . 116
3.C Unbiasing the wUDM via randomisation . . . 120

4 Quasi-Monte Carlo methods **122**
4.1 Quasi-Monte Carlo methodology . . . 123
4.1.1 Randomised quasi-Monte Carlo . . . 127
4.1.2 Array-RQMC . . . 130
4.2 RQMC applied to the τ-leap method . . . 132
4.2.1 Monomolecular reaction networks . . . 135
4.2.2 Discrete toy model . . . 141
4.2.3 Non-linear example . . . 151
4.3 Array-RQMC for chemical reactions . . . 155
4.3.1 Array-RQMC and the τ-leap method . . . 155
4.3.2 Array-RQMC and CLE discretisation . . . 162
4.3.3 Array-RQMC and uniformisation . . . 164
4.4 Discussion . . . 181
Chapter appendix
4.A Computational effort to generate quasi-random numbers . . . 185
4.B Model parameters . . . 185

5 Poisson bridge methods **188**
5.1 Standard sampling of Poisson processes . . . 189
5.2 Midpoint-based Poisson bridge . . . 191
5.2.1 Implementation details . . . 193
5.3 Median-based Poisson bridge . . . 193
5.3.1 Implementation details . . . 196
5.4 Antithetic construction of Poisson bridges . . . 197
5.4.1 Midpoint bridge parameters . . . 200
5.4.2 Comparison of different Poisson bridges . . . 204
5.4.3 Chemical reaction network examples . . . 207
5.5 Discussion . . . 221
Chapter appendix
5.A Midpoint-based Poisson bridge parameter optimisation details . . . 225
5.B Algorithm details . . . 244

6 Discussion **249**
6.1 Review . . . 249
6.2 Future work and challenges . . . 252
6.3 Conclusions . . . 264

Bibliography **266**

List of notation

Notation	Description
$\mathbb{N}, \mathbb{R}, \mathbb{R}^+$	Natural, real and positive real numbers, respectively
$I_x(a, b)$	Regularised incomplete beta function
$\text{Beta}(\alpha, \beta)$	Beta distribution with shape parameters α, β
$\mathcal{B}(N, p)$	Binomial distribution with N trials and success probability p
$\text{Exp}(\lambda)$	Exponential distribution with rate parameter λ
$\text{Gamma}(\alpha, \beta)$	Gamma distribution with shape parameter α and scale parameter β
$\text{Geom}(p)$	Geometric distribution with parameter p
$\mathcal{M}(N, \boldsymbol{\pi})$	Multinomial distribution with N trials and event probabilities $\boldsymbol{\pi}$
$\text{NB}(r, p)$	Negative binomial distribution for r successes with success probability p
$\mathcal{N}(\mu, \sigma^2)$	Normal distribution with mean μ and variance σ^2
$\mathcal{P}(\lambda)$	Poisson distribution with rate parameter λ
$\mathcal{U}(a, b)^d$	Uniform distribution on $[a, b)^d$

Acronyms

ABC	approximate Bayesian computation
ANOVA	analysis of variance
BSL	Bayesian synthetic likelihood
CDF	cumulative distribution function
CLE	chemical Langevin equation
CME	chemical master equation
CRN	common random numbers
CTMC	continuous-time Markov chain
DM	(Gillespie's) direct method
DTMC	discrete-time Markov chain
EDF	empirical distribution function
IMSE	integrated mean squared error
LLN	law of large numbers
MC	Monte Carlo
MCMC	Markov chain Monte Carlo
MFMC	multifidelity Monte Carlo
MISE	mean integrated squared error
MLMC	multilevel Monte Carlo
MLQMC	multilevel quasi-Monte Carlo
MNRM	modified next reaction method
MSE	mean squared error
NRM	next reaction method
NUDM	naïve uniformised direct method

ODE	ordinary differential equation
PCA	principal component analysis
PDE	partial differential equation
PDF	probability distribution function
PGF	probability generating function
QMC	quasi-Monte Carlo
RMSE	root mean squared error
RQMC	randomised quasi-Monte Carlo
RRE	reaction rate equation
RTCR	random time change representation
SDE	stochastic differential equation
SMC	sequential Monte Carlo
SSA	stochastic simulation algorithm
UDM	uniformised direct method
VRF	variance reduction factor
wSSA	weighted stochastic simulation algorithm
wUDM	weighted uniformised direct method

Chapter 1

Introduction

In the past decades research in molecular biology has generated vast amounts of quantitative data. This growing amount of data has inspired the development of a variety of mathematical modelling and simulation techniques aiming to support the experimental study of the intricate processes taking place in cells and other molecular systems. As a result we now often have the option to carry out pen and paper analysis or perform *in silico* experiments alongside the more traditional *in vivo* and *in vitro* approaches to study complex cellular pathways, giving us a detailed view of the different components in these often intricate networks [219].

When modelling chemical reaction kinetics, which form a key building block for intracellular pathways and intercellular interactions, we can take a wide variety of approaches, varying in detail and complexity [199]. One specific feature which nowadays appears prominently in many such models of chemical reaction networks is randomness, for example in the context of gene expression [146, 177] and cellular decision making [16]. The aim of including randomness is to mimic the effects of intrinsic and extrinsic noise sources present in molecular systems, as found in experiments [25, 57, 147, 168]. Though the analysis of a stochastic model is often more challenging compared to a similar deterministic model, it is easy to make a case for the relevance of stochasticity as it can be responsible for a wide variety of observed phenomena that

are not present in deterministic counterpart models, such as stochastic focussing [170], multistable species distributions [24, 54] or resonance-inducing oscillations [102].

1.1 Challenges

The addition of noise, however, also comes at a price in terms of our ability to analyse models relevant to quantititative biological data. Despite considerable effort from the scientific community, analytic solutions to stochastic models remain few and far between. For example, standard models for gene expression without extrinsic noise, which could be considered to be of relatively simple nature, have only been solved under limiting assumptions, e.g. [196]. To study more detailed or complex models we therefore often have to resort to *in silico* experiments. Due to the presence of randomness, single experiments will have to be run many times using Monte Carlo (MC) methods to yield results in the form of summary statistics to a satisfactory degree of certainty, which can result in large computation times or even make a problem intractable with existing computational methods and resources.

A key challenge in the development of computational techniques for these models therefore is finding ways to improve the efficiency of *in silico* simulations. Efforts towards this goal mainly follow either of two strands of research, those of algorithmic and statistical improvements, respectively. Approaches in the former category improve efficiency by increasing the number of simulations we can carry out within a given computational budget by "speeding up" current computational methods, e.g. [1, 68, 138, 145, 148, 198]. In this thesis, however, we focus on the latter approach, and aim to improve efficiency through the use of variance reduction methods. This approach generally uses specific model structures to refine currently available computational methods for these models such that the variance of the summary statistics, which form the output of the simulation algorithms used, is reduced. As a result we can use a lower number of simulations to achieve results of a satisfactory degree of

accuracy compared to standard computational approaches.

The history of variance reduction techniques can be traced back to the early 1950s, e.g. [92, 106, 187], almost immediately after the start of the use of MC methods using computers in the late 1940s, and an array of methods has since been devised to improve the efficiency of MC simulations in, amongst others, computational finance and particle physics. Though these variance reduction techniques have great potential, their success in many applications is intricately related to the particular model problems of interest. In the specific context of chemical reaction networks relevant to this thesis, the main challenge is therefore determining if and when variance reduction techniques, developed with different applications in mind, are appropriate in this specific context, which is characterised by testing features such as discrete, non-negative, high-dimensional state spaces and multiple scales.

1.2 Aims and outline

The central theme of this thesis is the construction of more efficient simulation methods in the context of chemical reaction networks that are modelled by continuous-time Markov chains (CTMCs), primarily via the exploration of variance reduction techniques. Crucially, we solely focus on techniques that have previously proven to be successful in different contexts and we study not only the applicability of these methods, but also aim to understand the limitations arising from the specific challenges posed by chemical reaction network models.

To this end we start in Chapter 2 with a comprehensive review of the relevant stochastic modelling and simulation approaches for chemical reaction networks. In addition we discuss, in detail, widely used variance reduction techniques and, where possible, show how these techniques are already used, or can be used, in the context of chemical reaction networks.

In the three research chapters that follow we discuss three separate themes lead-

ing to more efficient simulation methods. The content of these chapters is almost completely independent and the chapters can thus largely be read in isolation.

Chapter 3 revisits a previously coined idea in the literature to use the uniformisation technique to study chemical reaction networks. First we discuss an algorithmic improvement to speed up standard simulation methods for such uniformised systems, removing all computational overhead from applying uniformisation. Coincidentally this allows us to efficiently leverage the extra structure in uniformised systems in two different ways that both yield reduced variance estimators.

Chapter 4 studies how quasi-Monte Carlo (QMC) methods can be used to improve the efficiency of simulations of chemical reaction networks. Though such methods are very popular and effective in, for example, computational finance, we show that direct application of standard QMC methodology in chemical reaction network simulation does not yield the big improvements observed in other fields. We explain how the models specific to chemical reaction networks severely limit the effectiveness of QMC methods and discuss how an alternative method, called array-RQMC, can overcome some of these limitations to yield impressive results in practice.

Chapter 5 provides an in-depth analysis of Poisson bridge methods, which can be used to apply variance reduction techniques to the simulation of unit-rate Poisson processes. We show how to optimise the parameters in such Poisson bridge methods to achieve the largest possible variance reduction, and show how we can then use this methodology to improve the efficiency of standard simulation methods for chemical reaction networks.

Finally, in Chapter 6 we review the outcomes of this thesis and discuss new and remaining questions and challenges to create efficient simulation approaches for chemical reaction networks. By considering future suggestions of research related to this thesis we also show how our results tie in with various directions the field is taking.

Chapter 2

Stochastic models and simulation

In this chapter we provide a literature review of the field of stochastic simulation and modelling with an eye towards application to the field of chemical reaction networks. First we give an overview of commonly used models of chemical reactions and discuss how to perform and interpret simulations of these models. The accuracy of these stochastic simulation methods is inherently hindered and limited by statistical error and in the second part we review variance reduction methods, which are devised to reduce this statistical error and form the main focus of this thesis.

2.1 Mathematical models of chemical reactions

In this work we consider stochastic simulation of a model describing the temporal evolution of molecule copy numbers in a well-mixed system of volume V, i.e. we ignore any spatial information and assume that the molecules are homogeneously distributed within V. There exists a natural extension of this model that includes spatial information and we refer to [59, 80] for more detail on how to include spatial movement of molecules via diffusion in the framework described here.

Suppose we have a collection of J types of chemical species $S_1, \ldots, S_J$ that can interact via K different types of reactions $R_1, \ldots, R_K$, often denoted as reaction

channels. In generic form we can describe such an interaction R_k as

$$\alpha_{1,k}S_1 + \cdots + \alpha_{J,k}S_J \xrightarrow{c_k} \beta_{1,k}S_1 + \cdots + \beta_{J,k}S_J, \tag{2.1}$$

where $\alpha_{j,k}, \beta_{j,k} \in \mathbb{N}$ and c_k is the reaction rate constant for this reaction. We define $\mathbf{X}(t)$ to be the state (column) vector[1] describing the evolution of all the species as time evolves, i.e. $X_j(t)$ is the number of molecules S_j at time t. Upon the firing of reaction R_k the number of molecules will often change and we will use $\boldsymbol{\zeta}_k$, the stoichiometric vector[2] for reaction R_k, to denote this change of the copy number, i.e. due to reaction R_k we see $\mathbf{X} \to \mathbf{X} + \boldsymbol{\zeta}_k$. We can therefore describe the evolution of the chemical species using

$$\mathbf{X}(t) = \mathbf{X}_0 + \sum_{k=1}^{K} N_k(t)\boldsymbol{\zeta}_k, \tag{2.2}$$

where now $N_k(t)$ denotes the number of times that reaction channel R_k fires in the time interval $[0, t]$ and $\mathbf{X}_0$ represents the initial state vector, i.e. $\mathbf{X}(0)$. This evolution equation (2.2) is, of course, not very useful yet as we have no way to calculate these $N_k(t)$. In order to model these counting functions $N_k(t)$ we will assign to every reaction channel, R_k, a propensity function, $a_k(\mathbf{X}(t))$, which describes the probability that the reaction channel fires in the infinitesimal time interval $[t, t + \mathrm{d}t)$ as follows

$$\mathbb{P}\left(R_k \text{ fires in } [t, t + \mathrm{d}t)\right) = a_k(\mathbf{X}(t))\,\mathrm{d}t. \tag{2.3}$$

We also define the total reaction propensity $a_0(\mathbf{X}(t)) = \sum_{k=1}^{K} a_k(\mathbf{X}(t))$. For this work we primarily use *mass action kinetics*, under which the propensity for reaction R_k to fire is proportional to the number of possible combinations of reactants, see

[1] Throughout this thesis we use column vectors, unless stated otherwise. Vectors and matrices are written in boldface. We identify the i-th vector element of a generic vector $\mathbf{a}$ as a_i and similarly $\mathbf{B}_i$ represents the i-th column of a matrix $\mathbf{B}$.

[2] Using the stoichiometric vectors $\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_K$ we define the stoichiometric matrix $\boldsymbol{\zeta}$ such that its k-th column is equal to $\boldsymbol{\zeta}_k$.

for instance [98] or Example 2.1 for practical implications and [75] for a rigorous derivation from microphysical principles. However, note that the selection of this function is a modelling choice and in no way essential for anything that follows.

Example 2.1 (Mass action kinetics)**.**
We assume reactions take place in a well-mixed system of volume V. Now suppose we have species S_1, S_2 and S_3 undergoing a reaction of the form

$$S_1 + S_2 \xrightarrow{c} S_3. \tag{2.4}$$

If we use the previously described notation then see that there are $X_1 \cdot X_2$ possible pairs of S_1 and S_2 molecules for which this reaction is possible, but such pairs need to collide and react in a volume V. It is therefore assumed that $a_k(\mathbf{X}) \propto X_1 \cdot X_2/V$ with the proportionality constant defined by the reaction rate constant, c. Similarly, if we have a higher-order reaction such as

$$S_1 + 2S_2 \xrightarrow{c} 4S_3, \tag{2.5}$$

we find $X_1 \cdot X_2(X_2 - 1)/2$ possible combinations of S_1 and S_2 molecules that can undergo the reaction. Combinatorial factors such as $1/2$ are often implicitly incorporated in reaction rate constants and the Law of Mass Action therefore assumes $a_k(\mathbf{X}) \propto X_1 \cdot X_2(X_2 - 1)/V^2$. Finally we note that for zeroth order reactions such as

$$\emptyset \xrightarrow{c} S_1 \tag{2.6}$$

the reaction propensity is independent of molecule numbers, but instead depends on the reaction volume, V, i.e. $a_k(\mathbf{X}) \propto V$. Unless states otherwise, we let $V = 1$ in this thesis and thus absorb the dependency on V into the reaction rate constants.

We will now present different ways to define $N_k(t)$ based on these propensity functions. The order of presentation is incorrect from a historical angle (it is in fact

in reverse historical order), but it allows for a good exposition of the relations between the different methods.

An accurate way to model these counting functions would be to view the above description as a CTMC where, given the current state, $\mathbf{X}(t)$, we can experience K different state transitions based on the various reaction channels. Note that the transition rates are generally state-dependent, which means the CTMC is density-dependent. An inhomogeneous Poisson process will then describe the number of times reaction R_k fires. We can construct this inhomogenous Poisson process from a unit rate Poisson process, Y_k, using a time scale change. The intensity function or (counting) rate for this Poisson process Y_k in this framework is related to the propensity functions for the reaction channels and this leads to the Kurtz random time change representation (RTCR) [10]

$$\mathbf{X}(t) = \mathbf{X}_0 + \sum_{k=1}^{K} Y_k \left(\int_0^t a_k(\mathbf{X}(s)) \, \mathrm{d}s \right) \boldsymbol{\zeta}_k. \tag{2.7}$$

Note that we use K independent unit-rate Poisson counting processes Y_k, counting the number of times reaction channel R_k fires for all k. This representation forms the base for many (stochastic) simulation approaches.

To derive alternative and approximate models based on the RTCR we start by noting that we can use equation (2.7) to describe the evolution of the system over a time interval $[t, t+\tau)$ by

$$\mathbf{X}(t+\tau) = \mathbf{X}(t) + \sum_{k=1}^{K} Y_k \left(\int_t^{t+\tau} a_k(\mathbf{X}(s)) \, \mathrm{d}s \right) \boldsymbol{\zeta}_k. \tag{2.8}$$

Chemical Langevin equation. To proceed we assume that, loosely speaking, it is possible to choose a small time scale τ so that the reaction propensities remain (approximately) constant, but still such that we expect many reactions to fire on this time scale. Based on precise versions of these two approximations we can derive the

chemical Langevin equation (CLE) following [79]. The CLE is also known as the diffusion approximation to the original CTMC and it first appeared in [117], albeit with a different motivation and derivation.

The first condition requires a small time scale τ such that for each reaction channel we have $a_k(\mathbf{X}(s)) \approx a_k(\mathbf{X}(t))$ for $s \in [t, t+\tau]$, i.e. the reaction propensity remains approximately constant over the interval $[t, t+\tau]$. Using this assumption we can write

$$\mathbf{X}(t+\tau) \approx \mathbf{X}(t) + \sum_{k=1}^{K} Y_k \left(a_k(\mathbf{X}(t))\tau\right) \boldsymbol{\zeta}_k, \tag{2.9}$$

where we now have K homogeneous Poisson processes (over the interval $[t, t+\tau]$). As pointed out in [79] this condition can always be satisfied if the reactant species in the system are abundant (or if we take the trivial time scale $\tau \to 0$). Equation (2.9) forms the basis for a popular computational method, τ-leap [76], which we discuss in more detail in Section 2.2.2.

The second condition that we need balances the first condition by requiring each reaction channel to still fire many times over the interval $[t, t+\tau]$; more precisely for all k we require $a_k(\mathbf{X}_t)\tau \gg 1$. Note that this rules out taking $\tau \to 0$, i.e. the trivial choice that would guarantee the first condition to hold. Furthermore, it could be possible that there exists no time scale τ such that both conditions hold, in which case one should doubt the validity of the CLE approximation. Assuming, however, both conditions to be satisfied we next use the normal approximation of a Poisson process with large rate, from the condition $a_k(\mathbf{X}(t))\tau \gg 1$, to give

$$\begin{aligned}
\mathbf{X}(t+\tau) &\approx \mathbf{X}(t) + \sum_{k=1}^{K} \mathcal{N}_k\left(a_k(\mathbf{X}(t))\tau, a_k(\mathbf{X}(t))\tau\right) \boldsymbol{\zeta}_k \\
&= \mathbf{X}(t) + \sum_{k=1}^{K} \left[a_k(\mathbf{X}(t))\tau + \sqrt{a_k(\mathbf{X}(t))\tau}\mathcal{N}_k\left(0,1\right)\right] \boldsymbol{\zeta}_k,
\end{aligned} \tag{2.10}$$

where we abuse notation to let $\mathcal{N}_k(\mu, \sigma^2)$ denote a normal random variable with mean μ and variance σ^2. To derive the CLE in canonical form note that equation (2.10)

in the formal limit of $\tau \to 0$[3] gives an evolution equation for $\mathbf{X}(t)$ in the form of a stochastic differential equation (SDE);

$$\mathrm{d}\mathbf{X}_t = \left[\sum_{k=1}^{K} a_k(\mathbf{X}_t)\boldsymbol{\zeta}_k\right] \mathrm{d}t + \sum_{k=1}^{K} \sqrt{a_k(\mathbf{X}_t)}\boldsymbol{\zeta}_k \, \mathrm{d}W_k, \tag{2.11}$$

where now the W_k denote K independent Wiener processes and, by abuse of notation, $\mathbf{X}_t$ represents $\mathbf{X}(t)$ as is customary notation for SDEs. An alternative derivation of the CLE follows from taking a Kramers-Moyal expansion of the master equation describing the dynamics of the RTCR, resulting in a Fokker-Planck equation of which equation (2.11) is the pathwise representation. For more details on the latter approach, and a comparison of the validity of the CLE, see [90].

We note that the standard representation of the CLE in equation (2.11) is equivalent to an alternative SDE which is now driven by J Wiener processes W_j

$$\mathrm{d}\mathbf{X}_t = \left[\sum_{k=1}^{K} a_k(\mathbf{X}_t)\boldsymbol{\zeta}_k\right] \mathrm{d}t + \sum_{j=1}^{J} \mathbf{C}_j\left(\mathbf{X}_t\right) \mathrm{d}W_j, \tag{2.12}$$

where we recall J is the number of species. The functions $\mathbf{C}_j$ are such that the covariance structure of the noise is equal to equation (2.11), in particular, if we let $\mathbf{C}$ be the matrix with columns $\mathbf{C}_j(\mathbf{X}_t)$, then we need $\mathbf{C}\mathbf{C}^\intercal = \boldsymbol{\zeta}\mathrm{diag}(a_1(\mathbf{X}_t), \ldots, a_K(\mathbf{X}_t))\boldsymbol{\zeta}^\intercal$. A drawback of equation (2.12) on the level of interpretation is that this formulation seems to implicate that the noise term, which should reflect the stochastic nature of the reactions, is independent between species. Taking this interpretation in a literal sense would imply that the noise is therefore external to the reaction dynamics, which goes against what the model is trying to capture. Note that equation (2.11) does not suffer from this criticism, as pointed out in [76]. From a simulation perspective, however, both equations (2.11) and (2.12) are equivalent and if $J \ll K$ it could be more advantageous to use equation (2.12). The alternative representation

[3]Note that the second condition, $a_k(\mathbf{X}_t)\tau \gg 1$ for all k, does imply a lower limit on the time scale τ for which the CLE (and its SDE representation) can be valid. We refer the reader to [79] for a more in-depth discussion of this issue.

of the CLE via equation (2.12) has also been used to study the breakdown of the CLE approximation [195] and we refer the reader to [151] for more information on alternative representations of the CLE.

Reaction rate equation. Using the SDE formulation of the CLE approximation we can derive the deterministic reaction rate equations (RREs) which have been in use for over a century. We take the thermodynamic limit, letting the number of molecules and the volume go to infinity whilst their ratio, often referred to as the concentration and denoted by $\mathbf{x} = \mathbf{X}/V$, remains constant. In this limit[4] the random fluctuations become negligibly small compared to the deterministic contributions and we convert equation (2.10) into

$$\mathbf{x}(t+\tau) = \mathbf{x}(t) + \sum_{k=1}^{K} \tilde{a}_k(\mathbf{x}(t))\tau\boldsymbol{\zeta}_k, \tag{2.13}$$

where we defined $\tilde{a}_k(\mathbf{x}) = a_k(\mathbf{X})/V$. We can rewrite this into a system of ordinary differential equations (ODEs) by taking the limit $\tau \to 0$:

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \sum_{k=1}^{K} \tilde{a}_k(\mathbf{x}(t))\boldsymbol{\zeta}_k. \tag{2.14}$$

For a more in-depth discussion of the thermodynamic limit and its validity in the context of well-mixed chemical reaction networks see [77].

We now have three different mathematical models describing the temporal evolution of the molecular species $S_1, \ldots, S_J$ which can be seen as a chain of approximations going from a CTMC with discrete state space to a CTMC with a continuous state space to a deterministic ODE system. Each of these models can be analysed and simulated in different ways and it is often an open question, and an area of active research, which of these models is most applicable in a given circumstance, see for example [41, 54, 89, 90, 116, 216].

[4]Note that we also implicitly assume that $a_k(\mathbf{X})/V$ is a function solely of $\mathbf{X}/V$ for every reaction channel in the thermodynamic limit, which is true for the mass action kinetics and other reaction kinetics encountered in practice [77].

2.2 Simulation of well-mixed systems

Having presented the three different ways to model the evolution of interacting chemical species in the preceding section we can now ask how to perform a mathematical analysis on them. For the deterministic rate equations (2.14) we can use a wide array of well-known analytical and numerical techniques and we will not go in detail here but rather refer the reader to [98] and references therein.

Similarly for the CLE we can often use standard simulation procedures that are used for SDEs and we refer the reader to [111] for an extensive exposition of that subject. Note that, unlike general SDEs, the CLE should technically only be defined on the positive orthant and for mass action kinectis no guarantee can be made as to whether its diffusion and drift field obey this constraint, see for example [218, Section 1]. Different local and global changes to the CLE have been put forward that attempt to remedy this issue, see for example [6, 47, 128, 195, 218].

The RTCR, equation (2.7), yields a pathwise representation of the dynamics of the species in the system. An alternative description of the same dynamics is by use of the chemical master equation (CME), which comprises a (high-dimensional) system of ODEs [65]. The CME fully describes the evolution in time of the distribution of the occupation probability of the state space of $\mathbf{X}(t)$. Though an attractive feature of this approach is that the solution to the CME exactly describes the systems dynamics, it also has a severe disadvantage, the dimension of this system of ODEs. As the dimension is equal to the size of the state space this will generally be so high that the problem is analytically intractable in practice. Only for a small class of problems does the full CME have a known analytical solution [104, 184].

One therefore often has to rely on approximate methods or stochastic simulation to explore the behaviour of the system. In this thesis we will focus on the stochastic simulation approach and we refer the reader to [193] for an overview of alternative methods that build on (computational) approximations, often only feasible for prob-

lems of moderate dimensions, to the CME, such as the finite state projection method [91, 140, 157], moment-closure methods [28, 194] and system-size expansions [107].

2.2.1 Exact simulation algorithms

Despite the existence of good (approximate) methods, approaches working directly with the CME are bound to become too computationally expensive if the problem dimension is high and this means that one often relies on a different approach to get a handle on the model dynamics. Instead of looking at the evolution of the probability over the whole state space at once we generate single sample paths which evolve according to the rules of equation (2.7). The exact algorithm to compute such sample paths in the context of chemical reaction networks is often called "the" stochastic simulation algorithm (SSA) or (Gillespie's) direct method (DM) after the author who introduced it in this setting [78]. Given a current state $\mathbf{X}(t)$ the algorithm, as depicted in Algorithm 2.1, provides a way to compute the time until the next reaction fires and determines which reaction this is. In that way we can progress the Markov chain one reaction at a time.

Algorithm 2.1 (Gillespie's) direct method (DM).
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Final time T

$\mathbf{X} \leftarrow \mathbf{X}_0$
$t \leftarrow 0$
while $t < T$ **do**
    Generate $u_1, u_2 \sim \mathcal{U}(0,1)$
    $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate reaction propensities.
    $a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total reaction propensity.
    $\tau \leftarrow -\log u_1 / a_0$ ▷ Calculate the next reaction time.
    $t \leftarrow t + \tau$ ▷ Update time.
    Choose p such that $\sum_{k=1}^{p-1} a_k < a_0 u_2 \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
    $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$ ▷ Update state vector.
end while

Algorithm 2.2 Modified next reaction method (MNRM).
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Final time T

$\mathbf{X} \leftarrow \mathbf{X}_0$
$t \leftarrow 0$
$T_k \leftarrow 0$ ▷ Current internal time for each channel.
$P_k \leftarrow \text{Exp}(1)$ ▷ Generate internal time till next reaction for each channel.
while $t < T$ **do**
  $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate reaction propensities.
  $\Delta_k \leftarrow (P_k - T_k)/a_k$ ▷ Calculate absolute time till next reaction for each channel.
  $p \leftarrow \arg\min_k \Delta_k$ ▷ Find the next reaction to fire.
  $\tau \leftarrow \Delta_p$ ▷ Find the time till the next reaction.
  $t \leftarrow t + \tau$ ▷ Update time.
  $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$ ▷ Update state vector.
  $T_k \leftarrow T_k + a_k\tau$ ▷ Update current internal time for each channel.
  $P_p \leftarrow P_p + \text{Exp}(1)$ ▷ Generate internal time till next reaction for channel p.
end while

This approach can be made more computationally efficient by, for example, using the next reaction method (NRM) by Gibson and Bruck [68] or the modified next reaction method (MNRM) by Anderson [1], depicted in Algorithm 2.2. These latter two methods use just a single random variable per reaction fired by keeping track of the individual Poisson processes in the RTCR in equation (2.7), which encode the reaction times for each reaction channel.

2.2.2 Approximate simulation algorithms

Still, all variants of exact SSAs suffer from a drawback, namely their computational costs. As these methods simulate each reaction individually their run time can be significant if there are many molecules and reactions involved. This is the rationale behind the development of approximate methods to simulate sample paths from equation (2.7) of which we will highlight two methods in particular.

Algorithm 2.3 (Naïve) τ-leap method.
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Time step τ
Input: Final time T
$\mathbf{X} \leftarrow \mathbf{X}_0$
$t \leftarrow 0$
while $t < T$ **do**
    $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate reaction propensities.
    $t \leftarrow t + \tau$ ▷ Update time.
    Generate $Y_1, \ldots, Y_K$ Poisson random variables, s.t. $Y_k \sim \mathcal{P}(a_k \tau)$
    $\mathbf{X} \leftarrow \mathbf{X} + \sum_{k=1}^{K} Y_k \boldsymbol{\zeta}_k$ ▷ Update state vector.
end while

τ-leap One of the most widely used methods is the τ-leap scheme [76]. To derive this method we go back to equation (2.9) and this time we do not approximate the Poisson process by Gaussian random variates to yield the CLE. In essence the τ-leap method follows from the rationale that in a given small enough time interval $[t, t+\tau)$ the propensities of the reactions do not change much and therefore can be assumed constant. This approach yields a discrete-time Markov chain (DTMC) with a discrete state space, where the time between each state update is given by the time step τ and the transitions are computed by

$$\mathbf{X}(t+\tau) = \mathbf{X}(t) + \sum_{k=1}^{K} Y_k \left(a_k(\mathbf{X}(t))\tau\right) \boldsymbol{\zeta}_k. \tag{2.15}$$

The computational gain with this method comes from the fact that in order to calculate $Y_k \left(a_k(\mathbf{X}(t))\tau\right)$ we can simply generate a single Poisson random variable with parameter $a_k(\mathbf{X}(t))\tau$. This means that we can fire multiple reactions at once and therefore progress quicker than was the case for the DM (and variants of the DM). An algorithmic representation of the τ-leap method is depicted in Algorithm 2.3.

Being an approximate method the τ-leap method does not come without caveats, a first major one being the possibility of achieving negative molecule numbers. Many

possible workarounds to avoid negative copy numbers have been proposed [35, 36, 42, 211]. Furthermore, employing the τ-leap method has the potential of yielding a significant systematic error in the outcome, which is generally of order $\mathcal{O}(\tau)$, see e.g. [4]. This is due to the fact that the sample paths from the τ-leap algorithm are not drawn from the probability distribution described by the CME. Such a systematic error is also known as bias and is discussed further in Section 2.3. The quality of the approximation of the τ-leap sample paths to those of the exact SSAs depends on the time step chosen in relation to the typical reaction time scale of the system, see for example [36, 81] for a discussion on (adaptive) step size selection and its effect on accuracy.

A different view on the τ-leap method is that it is a variant of the explicit Euler method for ODEs applied to equation (2.7), where we approximate the time integral by a left Riemann sum. This method therefore parallels the widely used Euler-Maruyama scheme for SDEs and one could therefore ask the question whether it is possible to adapt some other ODE time-stepping approaches to the CTMC simulation case. Indeed this turns out to be possible for a large class of ODE methods such as linear multistep methods [17] or Runge-Kutta methods [31], of which we highlight here only the implicit Euler method from which implicit τ-leap approaches [183] can be derived that perform better for systems exhibiting stiff behaviour.

R-leap Of the other approximations to equation (2.7) available in the literature we only highlight R-leap [13], which parallels τ-leap in many respects. Where the τ-leap method assumes constant propensities over a fixed time interval, the R-leap method assumes constant propensities over a fixed number of reactions, say L. Using the approximation of constant propensities we can, compared to the SSA, fire more reactions at once and therefore progress in time more quickly. The R-leap approach, like τ-leap, yields a DTMC. Under the assumption of constant propensities over the next L reactions we know that the number of reactions per reaction channel is

Algorithm 2.4 (Naïve) R-leap method.
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Reaction step L
Input: Final time T

$\mathbf{X} \leftarrow \mathbf{X}_0$
$t \leftarrow 0$
while $t < T$ **do**
    $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate reaction propensities.
    $a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total reaction propensity.
    Generate $\tau \sim \text{Gamma}\,(L, 1/a_0)$ ▷ Time span for L reactions.
    if $t + \tau > T$ **then**
        Generate $L \sim \mathcal{B}\,(L-1, (T-t)/\tau)$ ▷ Find number of reactions in $[t, T]$.
        $t \leftarrow T$ ▷ Update time.
    else
        $t \leftarrow t + \tau$ ▷ Update time.
    end if
    $\pi_k \leftarrow a_k/a_0$ ▷ Relative channel propensities.
    Generate $\mathbf{Y} \sim \mathcal{M}\,(L, \boldsymbol{\pi})$ ▷ Reactions firing per channel.
    $\mathbf{X} \leftarrow \mathbf{X} + \sum_{k=1}^{K} Y_k \boldsymbol{\zeta}_k$ ▷ Update state vector.
end while

multinomially distributed, with event probabilities given by the channel propensities relative to the total propensity. This makes it easy to sample how many reactions happen per channel per step of L overall reactions and therefore progress quicker than was the case for the DM (and variants on the DM).

However, in contrast to τ-leap, we do not know *a priori* at which times t the different state updates of the algorithm take place. If one desires to know the evolution of t alongside $\mathbf{X}$, i.e. to give a time stamp to every state in the algorithm, then this can be calculated by adding an extra step to the algorithm. This extra step, however, involves using an extra (gamma-distributed) random variable. Note that, in general, this step is necessary to be able to ascertain whether a final simulation time has been reached. An algorithmic representation of the R-leap method is depicted in Algorithm 2.4.

It is therefore clear that the τ-leap and R-leap algorithms, although similar in

spirit, control a different parameter in order to approximate the inhomogeneous Poisson processes, time for τ-leap versus number of reactions for R-leap. A recent study shows that marginal gains can be achieved when one combines the benefits from τ-leap and R-leap, resulting in a new method the authors called S-leaping [139].

2.3 Monte Carlo methods

The previously described simulation methods for well-mixed systems all provide a means to generate (approximate) sample paths of chemical reaction networks, but a remaining question is how one should infer information from these paths. Often one is interested in expressions like $f(\mathbf{X}(t))$, where f is a function of the state variable at time t. For example, we could use $f(x) = x$ if the population level is of interest, or $f(x) = \mathbb{1}_{\{x>0\}}(x)$ if we are to determine whether a species becomes extinct or not. However, as $\mathbf{X}(t)$ is a random variable it is often more practical to consider the expectation of this function f applied to $\mathbf{X}(t)$, i.e.

$$Q = \mathbb{E}\left[f(\mathbf{X}(t))\right], \tag{2.16}$$

a quantity often called the summary statistic[5]. However, such summary statistics are often intractable for reaction networks of interest due to the complex distribution of $\mathbf{X}(t)$. To overcome this hurdle we can employ MC methods to estimate the quantity in equation (2.16). In this methodology we generate N independent sample paths $\mathbf{X}^{(1)}(t), \ldots, \mathbf{X}^{(N)}(t)$ from the distribution of possible outcomes in the state space of $\mathbf{X}(t)$ and construct from this the MC estimator of equation (2.16) by averaging over the sample paths

$$\hat{Q} = \frac{1}{N}\sum_{n=1}^{N} f\left(\mathbf{X}^{(n)}(t)\right) \approx \mathbb{E}\left[f\left(\mathbf{X}(t)\right)\right]. \tag{2.17}$$

Individual realisations of $\hat{Q}$ will in general not be equal to Q and are random vari-

[5]Note that for notational ease we often drop the explicit dependency of the summary statistic Q on t in this thesis.

ates which therefore have an inherent uncertainty related to them. This uncertainty can be quantified by the mean squared error (MSE)

$$\mathrm{MSE}\left[\hat{Q}\right] = \mathbb{E}\left[\left(\hat{Q} - Q\right)^2\right]. \tag{2.18}$$

This MSE can be decomposed into two separate sources of error,

$$\mathrm{MSE}\left[\hat{Q}\right] = \mathrm{Var}\left[\hat{Q}\right] + \mathrm{Bias}^2\left[\hat{Q}\right], \tag{2.19}$$

where

$$\mathrm{Var}\left[\hat{Q}\right] = \mathbb{E}\left[\left(\hat{Q} - \mathbb{E}\left[\hat{Q}\right]\right)^2\right], \tag{2.20a}$$

$$\mathrm{Bias}\left[\hat{Q}\right] = \mathbb{E}\left[\hat{Q}\right] - Q = \mathbb{E}\left[\hat{Q} - Q\right], \tag{2.20b}$$

denote the variance, which quantifies the statistical error, and bias, representing the systematic error, respectively. An unbiased estimator for the statistical error based on N samples is $\hat{s}^2/N$, where $\hat{s}^2$ is the usual sample variance

$$\hat{s}^2 = \frac{1}{N-1}\sum_{n=1}^{N}\left(f\left(\mathbf{X}^{(n)}(t)\right) - \hat{Q}\right)^2. \tag{2.21}$$

The bias, however, is often hard to estimate if no information is available about the true summary statistic Q (which is typically the case when one employs a MC method).

Example 2.2 (MSE for Gillespie's DM)**.**

The sample paths generated with the DM are free of systematic error and therefore unbiased. The MSE is thus purely determined by statistical error due to finite sample size, i.e. the variance of the estimator $\hat{Q}$. Given a sample variance σ^2 of the desired summary statistic we know that $\mathrm{Var}\left[\hat{Q}\right] = \sigma^2/N$ with N the number of sample paths [129, Chapter 1]. As a result we see that the more samples, N, we use, the smaller

the statistical error, and therefore the MSE, becomes. The squared error between $\hat{Q}$ and the (unknown) true summary statistic Q thus behaves like $\mathcal{O}\left(N^{-1}\right)$ for this unbiased method.

This section thus far has described the MC framework used to derive point estimates for certain scalar summary statistics Q like (2.16). It is, however, possible to extend this to vector summary statistics, $\mathbf{f}(\mathbf{X}(t))$, for example by using element-wise application of previous material. A particular extension of relevance to the context of this thesis is the estimation of the probability distribution giving rise to the observed data. Such an estimate of the distribution can often give more information to the observer than single point estimates such as the mean or variance, especially when the underlying distribution is multimodal. We will therefore discuss the problem of distribution estimation next before we move on to the question of the computational complexity of MC methods.

2.3.1 Distribution estimation

Suppose we have a quantity $\mathbf{Y}$ taking values in the state space Ω which can be described by some process $\mathbf{Y} \sim p$. If we have no explicit description of p, but are in the position to simulate from it, we can attempt to derive a MC estimate for p. For a fixed $\omega \subseteq \Omega$ we first note that, similar to (2.16), we have the identity

$$p(\omega) = \mathbb{E}\left[\mathbb{1}_{\mathbf{Y}\in\omega}\left(\mathbf{Y}\right)\right]. \tag{2.22}$$

Because ω is taken fixed this is nothing else than a point summary statistic so that we can use N samples $\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(N)}$ and apply (2.17) to find a MC point estimate for (2.22) of the form

$$\widehat{p(\omega)} = \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}_{\mathbf{Y}\in\omega}\left(\mathbf{Y}^{(n)}\right) \approx p(\omega). \tag{2.23}$$

A distribution is, however, not a point estimate. To proceed we note that the fixed ω was completely arbitrary which means we can define an (approximate) distribution function $\hat{p} : \Omega \mapsto \mathbb{R}^+$ by

$$\hat{p}(\cdot) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\mathbf{Y} \in \cdot} \left(\mathbf{Y}^{(n)} \right). \tag{2.24}$$

This specific approximate distribution function is known as the empirical distribution function (EDF). Note, however, that there is a common class of approximate distribution functions derived via kernel density estimation methods which extend the EDF. Such methods define estimators of the form

$$\hat{p}(\cdot) = \frac{1}{N} \sum_{n=1}^{N} K_{h,\cdot} \left(\mathbf{Y}^{(n)} \right), \tag{2.25}$$

where $K_{h,\cdot}$ is a non-negative kernel function with a bandwidth parameter h used in order to smooth the resulting distribution. One of the most popular alternatives to distribution estimators, the histogram, is a blend between the aforementioned methods and is discussed in more detail in Appendix 2.B. Note that the EDF is a special case of a kernel density estimator where the kernel function is given by the indicator function. For discrete distributions the EDF is also equivalent to a histogram where the bin size is equal to unity.

Due to the finite sample size, N, these approximate distribution functions $\hat{p}$ are random functions and carry intrinsic uncertainty, just like we saw for the estimate $\hat{Q}$ in (2.17). To quantify this uncertainty we note that the equivalent to the MSE in the context of distribution estimation is the mean integrated squared error (MISE), with respect to p, given by

$$\mathrm{MISE}\left[\hat{p} \parallel p\right] = \mathbb{E}\left[\|\hat{p} - p\|_2^2\right] = \mathbb{E}\left[\int_{\Omega} \left(\hat{p}(\omega) - p(\omega)\right)^2 \, \mathrm{d}\omega\right], \tag{2.26}$$

which can again be decomposed into statistical error and systematic error components

via

$$\mathrm{MISE}\left[\hat{p} \parallel p\right] = \underbrace{\mathbb{E}\left[\int_{\Omega} \left(\hat{p}(\omega) - \mathbb{E}\left[\hat{p}(\omega)\right]\right)^2 \, \mathrm{d}\omega\right]}_{\text{statistical error}} + \underbrace{\mathbb{E}\left[\int_{\Omega} \left(\mathbb{E}\left[\hat{p}(\omega) - p(\omega)\right]\right)^2 \, \mathrm{d}\omega\right]}_{\text{systematic error}}. \quad (2.27)$$

Note that the MISE is an example of a measure between two distributions on Ω, $\hat{p}$ and p in this case. Such measures are generally known as statistical distances between two distributions q and p on Ω. We will only mention two other notable examples of statistical distance here; the total variation distance which is given in this context by

$$\delta_{\mathrm{TV}}\left[q \parallel p\right] = \sup_{\omega \subseteq \Omega} |q(\omega) - p(\omega)|, \quad (2.28)$$

and Kolmogorov-Smirnov distance which considers cumulative discrepancies between distributions p and q, given by

$$D_{KS}\left[F_q \parallel F_p\right] = \sup_{y \in \Omega} |F_q(y) - F_p(y)|, \quad (2.29)$$

where F_q and F_p are the cumulative distribution functions (CDFs) for q and p, respectively. Note that the choice of statistical distance to compare distributions can have a significant influence in statistical testing procedures, see [40] for a comparison in the context of discrete random variables. In this thesis, however, we are mainly concerned with convergence of empirical distributions based on an increasing number of samples, N, and therefore limit ourselves to the MISE and total variation distance.

The aforementioned general kernel density methods are particularly useful in problems where $\mathbf{Y}$ is a continuous random variable, in which case the discrete nature of the EDF forms a severe hindrance. However, in this work we are mainly interested in models where $\mathbf{Y}$ has a discrete state space. Discrete kernel densities have been proposed for these problems such as (negative) binomial or Poisson distribution kernels, see for example [112]. However, these kernel functions introduce a bias into $\hat{p}$

dependent on the bandwidth h. The resulting distribution approximations therefore do not have a MISE that automatically converges to zero. It was also shown in [112] that the benefit of such discrete kernels lies predominantly in the regime of very few samples, N, where the bandwidth bias is frequently the less important contribution to the MISE compared to the statistical error. Unless one's computational budget is very restrictive it is, in the context of chemical reaction networks, therefore often sufficient to simply use the EDF given by (2.24), or a histogram, see Appendix 2.B, which is what we will do from now on.

The assumption of a discrete state space Ω also simplifies the expressions for the MISE and total variation distance into forms that allow for tractable calculations. First, for the total variation distance we have

$$\delta_{\mathrm{TV}}[q,p] = \frac{1}{2}\|q-p\|_1 = \frac{1}{2}\sum_{y\in\Omega}|q(y)-p(y)|, \tag{2.30}$$

see for example [137, Proposition 4.2]. For the MISE we note that the integral converts into a sum which we combine with the linearity of expectation to give

$$\mathrm{MISE}[\hat{p} \parallel p] = \mathbb{E}\left[\|\hat{p}-p\|_2^2\right] = \sum_{y\in\Omega}\mathbb{E}\left[(\hat{p}(y)-p(y))^2\right], \tag{2.31}$$

which we can interpret as the integrated mean squared error (IMSE) being equal to the MISE. Note that this also provides a method to estimate the MISE by summing the pointwise MSE over Ω.

Example 2.3 (Statistical error for EDFs)**.**

For estimates $\hat{Q}$ of scalar summary statistics we know that the statistical error behaves like σ^2/N, where $\sigma^2 = \mathrm{Var}\left[\hat{Q}\right]$, e.g. see Example 2.2. If the scalar summary statistic, however, is of the form (2.23) it follows that $N\hat{p}(\omega) \sim \mathcal{B}(N, \mathbb{E}[\hat{p}(\omega)])$. Using known results for the variance of a binomial random variable we therefore immediately see

that

$$\mathbb{E}\left[\left(\hat{p}(\omega) - \mathbb{E}\left[\hat{p}(\omega)\right]\right)^2\right] = \frac{1}{N}\mathbb{E}\left[\hat{p}(\omega)\right]\left(1 - \mathbb{E}\left[\hat{p}(\omega)\right]\right), \tag{2.32}$$

which gives a useful decomposition for (2.31) of the form

$$\mathrm{MISE}\left[\hat{p} \parallel p\right] = \frac{1}{N}\left(1 - \sum_{y\in\Omega}\mathbb{E}\left[\hat{p}(y)\right]^2\right) + \sum_{y\in\Omega}\mathrm{Bias}\left[\hat{p}(y)\right]^2. \tag{2.33}$$

On the one hand this expression provides a means of calculating analytic values for the MISE for test problems which have a known distribution, i.e. if p and $\hat{p}$ are known *a priori*. It also shows that the statistical error component of the MISE for the EDF is bounded above by $1/N$. On the other hand it provides a means to estimate the statistical error component of the MISE for general problems by

$$\hat{S}^2_{\hat{p}} = \frac{1}{N-1}\left(1 - \sum_{y\in\Omega}\hat{p}(y)^2\right), \tag{2.34}$$

which has the property that $0 \leq \hat{S}^2_{\hat{p}} \leq 1/N$ and it is unbiased, i.e.

$$\mathbb{E}\left[\hat{S}^2_{\hat{p}}\right] = \frac{1}{N}\left(1 - \sum_{y\in\Omega}\mathbb{E}\left[\hat{p}(y)\right]^2\right). \tag{2.35}$$

2.3.2 Computational complexity and efficiency

MC methods often return increasingly accurate summary statistics in the limit of increasing number of samples, N, an illustration of which we saw earlier in Example 2.2. However, it is of course infeasible to generate an infinite number of sample paths and as a result MC estimators like equation (2.17) will always have a degree of uncertainty associated with them. The aim should therefore be *to have a MSE that is as small as possible given a fixed amount of computational resource.*

In order to work towards this goal one must define a measure of *computational*

complexity, which intuitively should quantify the computational cost associated to a specific method used to find approximations to equation (2.16). In this work we will denote such a complexity metric related to an approximate summary statistic $\hat{Q}$ as $\mathcal{C}[\hat{Q}]$. A common (and logical at first sight!) choice reported in the literature on SSAs is the CPU wall-clock time for specific methods. We note, however, that there are serious limitations to reporting such a metric, because it can depend heavily on the specific implementation details of a method, such as choices in algorithm, hardware, programming language and operating system. While method A might be reported to be computationally more efficient than method B *now* and *on this specific machine* it does not mean that the same will be true *two years from now* (even on the same machine!). These limitations thus form a significant hurdle for the reproducibility and portability of results on SSAs. We therefore opt for a different proxy for computational complexity in this work, namely *the (expected) number of random variables* used to generate a sample path (or ensemble of paths). This metric, though not perfect, is more robust to the failures of the CPU wall-clock time metric mentioned earlier and has recently been used to provide an extensive complexity analysis of some standard SSAs such as the DM and τ-leap method in the classical population scaling [8]. It also relates well to the number of steps that are needed in a SSA and thus could be seen as an indicator for (relative) practical CPU wall-clock time. However, we must preface that there is no perfect correlation between the CPU wall-clock time and the computational complexity in terms of random numbers utilised. Though the development of the NRM in [68] was partly stimulated by the observation that generating random variables is relatively slow, the results in [221] show CPU wall-time improvements of only up to 25% when reducing by half, via correct recycling of recycling random variates, the number of random numbers used in the DM. We do, however, note that many of the improvements in the literature that reduce run-time of SSAs can also be easily combined with the majority of the variance reduction methods that we discuss in this work. Finally, a notable shortcoming of our specific

complexity measure is that it ignores, amongst other effects, the memory requirements of a method. As a result performance on modern accelerated hardware such as GPUs [100, 110, 113, 114, 160], which is often memory limited, could be poorly predicted when solely focussing on the number of random variates used.

Example 2.4 (Complexity for Gillespie's DM and τ-leap)**.**
Here we show explicit expressions for the expected number of random variables complexity metric for both the DM and (fixed step) τ-leap algorithm. To do so we assume that we have a system composed of K reactions that we wish to simulate up until some known final time T. If we define $N_k(t)$ to be the random variable denoting the number of reactions that have fired in the system up until time t, it becomes clear from Algorithm 2.1 and Algorithm 2.3 that

$$\frac{\mathcal{C}\left[\hat{Q}_{\mathrm{DM}}\right]}{N} = 1 + 2\sum_{k=1}^{K} \mathbb{E}\left[N_k(T)\right], \tag{2.36a}$$

$$\frac{\mathcal{C}\left[\hat{Q}_{\tau\text{-leap}}\right]}{N} = KT\tau^{-1}, \tag{2.36b}$$

where N is the number of sample paths used. Note that even though $N_k(t)$ is a random variable it is possible to describe it in the same CTMC framework by extending all reactions in the system with the dummy product species N_k, i.e.

$$\alpha_{1,k}S_1 + \cdots + \alpha_{J,k}S_J \xrightarrow{c_k} \beta_{1,k}S_1 + \cdots + \beta_{J,k}S_J + N_k,$$

so that N_k increases by one every time reaction k fires. For some simple systems it is possible to derive analytic expressions for $\mathbb{E}\left[N_k(T)\right]$ via the CME, but alternatively one could keep track of it in any SSA via this extended reaction system.

In order then to compare two different estimates for (2.16) we need to take into account both the MSE and the computational complexity of the methods. A quantity which aims to balance the contribution from both is given by the (relative) *efficiency*,

see for example [162, Chapter 8]. For two estimators $\hat{Q}$ and $\hat{Q}_0$ the relative efficiency of $\hat{Q}$ with respect to $\hat{Q}_0$ is then defined as

$$\mathcal{E}\left[\hat{Q}, \hat{Q}_0\right] = \frac{\mathcal{C}\left[\hat{Q}_0\right] \text{MSE}\left[\hat{Q}_0\right]}{\mathcal{C}\left[\hat{Q}\right] \text{MSE}\left[\hat{Q}\right]}. \tag{2.37}$$

The definition of efficiency when estimating distributions is completely analogous to equation (2.37), replacing the MSE with the MISE. Note that the above definition is not dependent on the specific computational complexity measure. If the relative efficiency is larger than unity we can conclude that $\hat{Q}$ is the preferred estimator over $\hat{Q}_0$. Two examples illustrating the concept of efficiency in the context of chemical reaction networks are now given in Example 2.5 and Example 2.6.

Example 2.5 (MSE and efficiency for a simple linear system)**.**

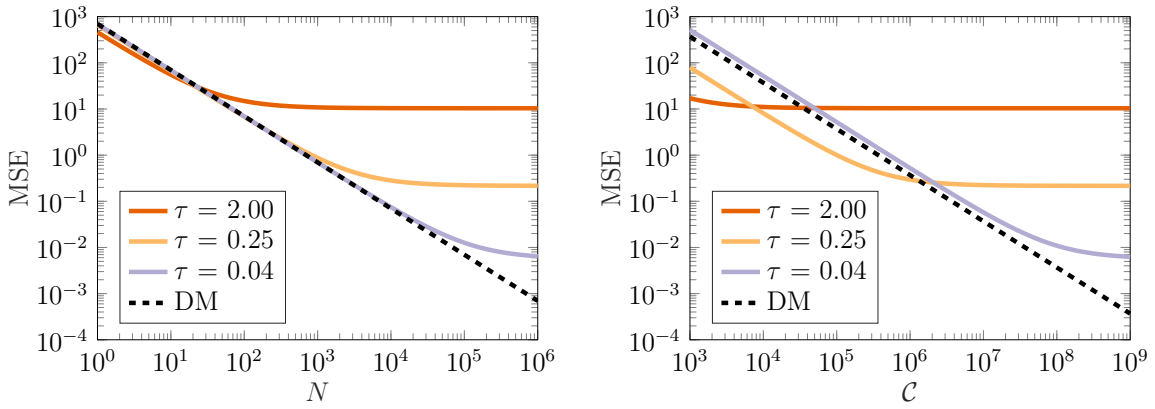
We consider a linear reaction system for a single species S_1 in a volume V of the following form:

$$\emptyset \xrightarrow{c_0} S_1, \tag{2.38a}$$

$$S_1 \xrightarrow{c_1} 2S_1, \tag{2.38b}$$

$$S_1 \xrightarrow{c_2} \emptyset. \tag{2.38c}$$

This system, a linear birth-death system with inflow, can be used to analytically study some properties of the τ-leap method and the DM due to its relative simplicity. If we take as a summary statistic the average value of S_1 at some time T, i.e. $Q = \mathbb{E}\left[X_1(T)\right]$, we can use the methods of Appendix 2.A to find exact expressions for $\hat{Q}$ and its variance for the DM and the τ-leap method in terms of system parameters c_0, c_1, c_2, the observation time, T, and the time step, τ. Combined with the explicit complexity expression as found in Example 2.4 we can evaluate the MSE for $\hat{Q}_{\text{DM}}$ and $\hat{Q}_{\tau\text{-leap}}$ analytically as a function of the number of samples, N, and the complexity, $\mathcal{C}$, as shown in Figure 2.1.



(a) Number of samples, N.

(b) Complexity, $\mathcal{C}$.

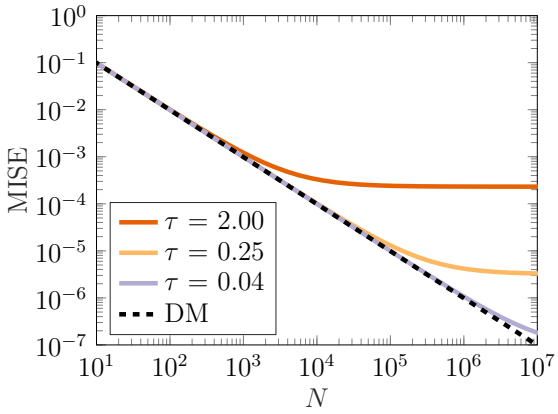
Figure 2.1: MSE as a function of the number of samples, N, used and computational complexity, $\mathcal{C}$, for the τ-leap method and the DM. Results are for the single species system (2.38) in a volume $V = 1$ with parameters $c_0 = 0.4, c_1 = 0.7, c_2 = 0.6$ and initial condition $\mathbf{X}_0 = 10$, run until $T = 10$.

We can make some general observations based on Figure 2.1. Firstly, it is clear that the MSE for the τ-leap method can be quickly dominated by the bias term if the step size is taken too large (as is evident from the curve for $\tau = 2.00$). Using more samples in this case will not result in more accurate estimates from $\sim 10^2$ samples onwards due to the bias. In addition we see that the bias clearly depends on the step size τ. We can also see in Figure 2.1(a) that, if we consider the MSE as a function of the number of samples, N, it seems as if the τ-leap method is better than Gillespie's DM for any τ (until the bias dominates). This, however, is misleading as we can see when we consider the MSE as a function of the complexity in Figure 2.1(b). Depending on the desired MSE target it can be orders of magnitude better to use the τ-leap method, but taking τ too small leads to a method which compares unfavourably to the DM based on efficiency.

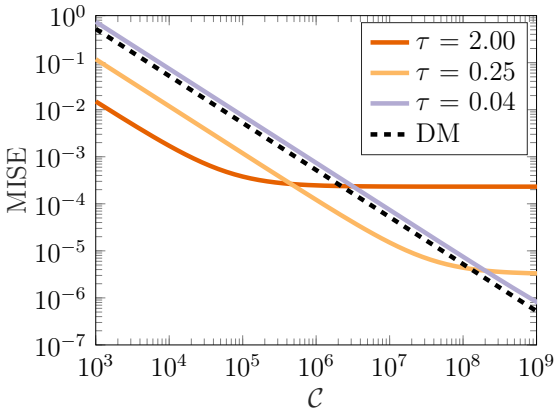
Example 2.6 (Distribution estimation for a simple linear system)**.**

For the simple linear system in the previous example, defined by (2.38), the exact transient distribution for S_1 can be derived, see Appendix 2.C. In addition we can

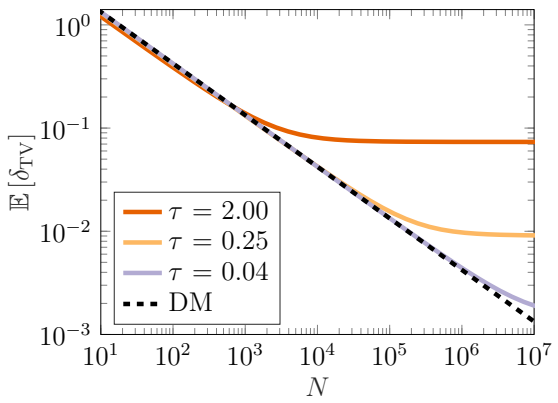
calculate the distribution for sample paths generated using the τ-leap method using the observation that its dynamics are governed by a time-homogeneous DTMC with known transition probabilities, which turn out to be related to the Skellam distribution. This allows us to illustrate some of the concepts introduced in Section 2.3.1 by calculating the statistical distance of the EDF to the true distribution, both as a result of using the DM and the τ-leap method, and the results are shown in Figure 2.2.



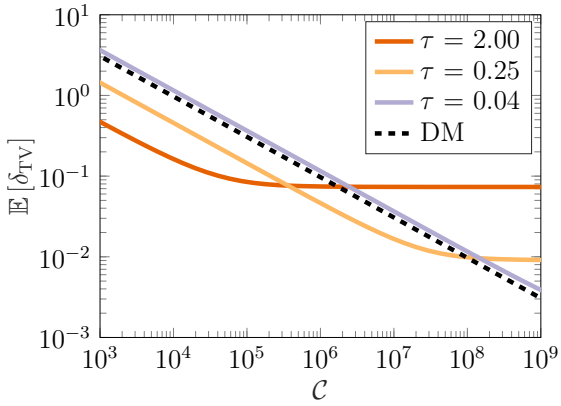
(a) MISE as a function of number of samples, N.



(b) MISE as a function of computational complexity, $\mathcal{C}$.



(c) Expected total variation distance as a function of number of samples, N.



(d) Expected total variation distance as a function of computational complexity, $\mathcal{C}$.

Figure 2.2: Comparing convergence of the error in the EDF from the τ-leap method and the DM with respect to the true distribution for single species system (2.38). Two types of statistical distance, the MISE and total variation distance, are used. Results are for the single species system (2.38) in a volume $V = 1$ with parameters $c_0 = 0.4, c_1 = 0.7, c_2 = 0.6$ and initial condition $\mathbf{X}_0 = 10$, run until $T = 10$.

Comparing Figure 2.1 and Figure 2.2 we can identify many similarities between the results for the estimation of the mean and the distribution, e.g. the bias contribution from applying the τ-leap method when enough samples, N, are used. Furthermore, the results for the MISE and total variation distance look qualitatively the same, though we note that, as we can expect, the total variation distance converges as $\mathcal{O}(N^{-1/2})$ rather than the $\mathcal{O}(N^{-1})$ rate for the MISE.

In contrast to moment estimation we see that, if we only consider the number of samples, N, used, the τ-leap method has a higher error, both for the MISE and the total variation distance, for any step size τ. Due to the lower computational complexity of the τ-leap method, however, we see that for a given computational complexity and MISE target it can still be advantageous to use the τ-leap method. This leads us to a similar conclusion for distribution estimation problems as at the end of Example 2.5.

From the decomposition of the MSE into statistical and systematic error and the definition of (relative) efficiency it becomes clear that there are two main ways of achieving an optimal method in a MC setting.

Firstly, we can aim to use a "better" SSA in terms of its complexity. This way one hopes to increase the number of sample paths, N, generated for a given computational budget and therefore reduce the statistical error (and hence the MSE or MISE). As we alluded to earlier, methods with the same complexity, defined in terms of number of random variables used, could have different CPU run-times. Therefore a related way to achieve the same goal in a practical scenario is to change the implementation of the SSA so as to keep the computational complexity constant, but reduce the CPU run-time for a specific instance. This approach has been most popular in the context of chemical reaction networks, both on a hardware, software or language implementation level, and on an algorithm design level. For example, a variety of (computational) modifications to the DM [1, 37, 68, 134, 148, 221] exist which aim to improve the runtime by optimising certain steps, predominantly steps 5 and 8, of

(variants of) Algorithm 2.1 without affecting the statistically exact sampling (and in many cases without changing the computational complexity). On a related note we have seen approximate methods such as τ-leap and R-leap which trade a (hopefully slight) increase in systematic error for a reduction in computation time per sample path.

The alternative, however, is the design of new methods which keep the run-time per sample path (nearly) equal whilst at the same time reduce either the systematic or statistical error in the computation of estimates to equation (2.16). Efforts in this direction in the more general MC framework are known as variance reduction techniques and will be discussed in Section 2.4. They also form the main inspiration for this work and their adaptation to the context of chemical reaction networks is the common thread in this thesis.

2.4 Variance reduction methodology

The focus of variance reduction techniques is to reduce the statistical error per unit of computational complexity in simulations and thereby increase the efficiency, as defined in equation (2.37), of the simulation procedure. Many different methods to achieve variance reduction have been proposed since the invention of MC simulations and their theoretical foundation is often stated in very generic terms, showing the versatility and general applicability of these approaches. In general, however, *"there is no such thing as a free lunch"* and the largest benefits from variance reduction techniques come from exploiting specific properties of the underlying model.

We will therefore next discuss three broad classes of general variance reduction techniques with relevance to the simulation of chemical reactions and, where possible, highlight how these methods can be used in the context of this thesis.

2.4.1 Correlation based techniques

To start we return to the common case of estimating a summary statistic $Q = \mathbb{E}[f(\mathbf{Y})]$ for some $\mathbf{Y} \sim p$. A generic MC method proceeds by sampling N sample paths, $\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(N)}$, to construct the estimate $\hat{Q}$ via equation (2.17). However, in contrast to the standard MC method, we now remove the independence assumption between these sample paths. In addition we generalise the estimate to allow non-equal weighting of the different sample paths via

$$\hat{Q} = \sum_{n=1}^{N} w_n f\left(\mathbf{Y}^{(n)}\right) \approx \mathbb{E}\left[f\left(\mathbf{Y}\right)\right] = Q, \tag{2.39}$$

which holds under the condition that $\sum_n w_n = 1$. Standard MC is recovered by letting $w_n = 1/N$ for all $n = 1, \ldots, N$ and requiring the sample paths to be independent. Note that these changes do not alter the systematic error of this method, but do change the statistical error, which is now given by

$$\mathrm{Var}\left[\hat{Q}\right] = \sigma^2 \left(\sum_{n=1}^{N} w_n^2\right) + 2 \sum_{1 \leq i < j \leq N} w_i w_j \mathrm{Cov}\left[f\left(\mathbf{Y}^{(i)}\right), f\left(\mathbf{Y}^{(j)}\right)\right], \tag{2.40}$$

where $\sigma^2 = \mathrm{Var}[f(\mathbf{Y})]$. The first term on the right in equation (2.40) can be easily identified as the statistical error for the standard MC method, i.e. assuming the sample paths $\mathbf{Y}^{(n)}$ are i.i.d. distributed. We therefore see that by removing the assumption of independent sample paths we can either increase or decrease the statistical error of $\hat{Q}$ relative to the standard MC method, depending on the resulting pairwise covariances between $f(\mathbf{Y}^{(i)})$ and $f(\mathbf{Y}^{(j)})$. The next three methods aim to reduce the estimator variance by exploiting structure in f and $\mathbf{Y}$.

Antithetic variables

The method of antithetic variables attempts to introduce pairwise negative correlations by generating for every sample path $\mathbf{Y}^{(n)}$ a complementary sample path $\tilde{\mathbf{Y}}^{(n)}$

such that, in some sense, $\tilde{\mathbf{Y}}^{(n)}$ and $\mathbf{Y}^{(n)}$ are opposite. By equation (2.40) this can lead to a reduction in the variance of the estimator $\hat{Q}$. In order to achieve this the antithetic method, in its most widespread form, assumes that we can write $\mathbf{Y} = \psi(\mathbf{u})$ where $\mathbf{u} \sim \mathcal{U}(0,1)^d$. Note that this is a very common situation in simulation methods, where we often generate generic random variates or decisions using uniform random variables on the hypercube. Next we define $\tilde{\mathbf{u}} = 1 - \mathbf{u}$ (element-wise) and note that $\tilde{\mathbf{u}} \sim \mathcal{U}(0,1)^d$ as well. This, in turn, implies that $\tilde{\mathbf{Y}} = \psi(\tilde{\mathbf{u}}) \sim p$, i.e. $\tilde{\mathbf{Y}}$ follows the correct distribution and yet is in some sense opposite to $\mathbf{Y}$.

To generate $2N$ antithetic samples we therefore first sample N standard uniform random variables $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(N)}$ on the hypercube $[0,1)^d$. These variates define N sample paths $\mathbf{Y}^{(n)} = \psi(\mathbf{u}^{(n)})$ and their antithetic counterparts $\tilde{\mathbf{Y}}^{(n)} = \psi(\tilde{\mathbf{u}}^{(n)})$, yielding a total of $2N$ sample paths. The antithetic estimator is then given by

$$\hat{Q}_{\text{anti}} = \frac{1}{N} \sum_{n=1}^{N} \frac{f\left(\mathbf{Y}^{(n)}\right) + f\left(\tilde{\mathbf{Y}}^{(n)}\right)}{2}, \tag{2.41}$$

which has statistical error, via equation (2.40), equal to

$$\text{Var}\left[\hat{Q}_{\text{anti}}\right] = \frac{1}{2N}\left(\sigma^2 + \text{Cov}\left[f\left(\mathbf{Y}\right), f\left(\tilde{\mathbf{Y}}\right)\right]\right) = \frac{\sigma^2}{2N}\left(1 + \rho\right), \tag{2.42}$$

where $\sigma^2 = \text{Var}\left[f(\mathbf{Y})\right]$ and $-1 \leq \rho \leq 1$ is the correlation between $f(\mathbf{Y})$ and $f(\tilde{\mathbf{Y}})$. The introduction of a dependency between the sample paths means that we can no longer use equation (2.21) to estimate the statistical error. Instead we have to define a new estimator

$$\hat{s}^2_{\text{anti}} = \frac{1}{N-1} \sum_{n=1}^{N} \left(\frac{f\left(\mathbf{Y}^{(n)}\right) + f\left(\tilde{\mathbf{Y}}^{(n)}\right)}{2} - \hat{Q}_{\text{anti}} \right)^2, \tag{2.43}$$

which has the property $\mathbb{E}\left[\hat{s}^2_{\text{anti}}/N\right] = \text{Var}[\hat{Q}_{\text{anti}}]$.

Equation (2.42) shows that the relative efficiency of the antithetic method com-

pared to standard MC is given by $2/(1+\rho)$, which can take values in $[1, \infty)$. This result indicates that we are guaranteed to improve the efficiency by blindly applying the antithetic method. We should stress, however, that this is under the assumption that the computational complexity is defined solely in terms of the random numbers used (see Section 2.3.2) and therefore any function calls, such as f, incur negligible cost. In any practical application this will not be strictly true and the efficiency gains of the antithetic method will depend on how much of the negative correlation between $\mathbf{u}$ and $\tilde{\mathbf{u}}$ is retained after applying the functions ψ and f to them (relative to computational overhead). For a special class of problems it can be proven that $\rho < 0$ under certain monotonicity conditions on f and ψ, see for example [129, Section 4.3].

Of particular interest in the context of this thesis is the recent resurfacing of different methods to generate Poisson processes, dating back originally to [63]. In [141] the authors show that applying the antithetic method to the simulation of a unit-rate Poisson process yields negatively correlated Poisson processes. These can then, in turn, be used to reduce the variance of chemical reaction sample paths if we view the Poisson processes as inputs to a simulation of the RTCR of the dynamics, e.g. using the NRM [68] or MNRM [1]. We will discuss an extension of this work in more detail in Chapter 5. An illustration of a more direct application of antithetic sampling to the simulation of chemical reactions can be found in Example 2.7 below. A study of direct application of antithetic sampling to the DM and a rejection based alternative can be found in [205].

Example 2.7 (Antithetic variables for a simple linear system)**.**

We consider again the simple linear system (2.38) that we first encountered in Example 2.5. Instead of focussing on the comparison with the exact answer according to the CME, as done in Example 2.5, we now consider purely the statistical error in a simulation using the τ-leap method. Inspection of Algorithm 2.3 shows that we can think of the τ-leap method to generate sample paths $\mathbf{X}$ as a function $\mathbf{X} = f(\mathbf{u})$ where $\mathbf{u} \sim \mathcal{U}(0,1)^d$. This link follows when we note that each component of $\mathbf{u}$ defines a

Poisson random variate in step 5 of Algorithm 2.3. The dimensionality of the random input is given by the computational complexity, see equation (2.36b) ($K = 3$ here).

This means that we can directly apply antithetic sampling on the random input $\mathbf{u}$ of the τ-leap method. An illustration of the resulting decay of the statistical error, compared to standard MC, is shown in Figure 2.3. It is clear from this figure that the antithetic method provides an improvement over the standard MC method, in this case a statistical error which is roughly ten times smaller for the same computational complexity, $\mathcal{C}$. As a sanity check we see that the estimated sample variance using equations (2.21) and (2.43) (solid lines) agrees well with the actual squared error (open circles, calculated using Appendix 2.A).

We note that this method was previously considered in [142, 143] and the authors proved in [142, Corollary 2] that it improves upon standard MC for a (sub)class of reaction systems with affine propensity functions (such as in this example).

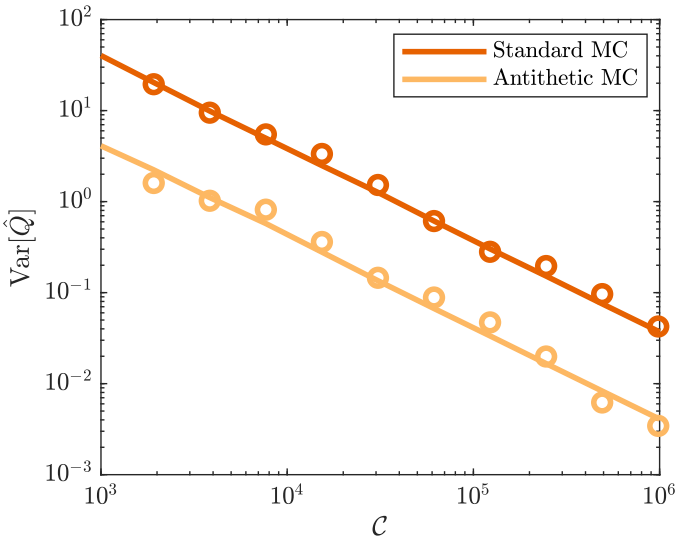


Figure 2.3: Statistical error, $\mathrm{Var}\left[\hat{Q}\right]$, as a function of the computational complexity, $\mathcal{C}$, for the τ-leap method ($\tau = 0.5$) using either standard MC sampling or antithetic sampling for the uniform random number input. The summary statistic is the mean species count for S_1 at time T. Results are for the single species system (2.38) in a volume $V = 1$ with parameters $c_0 = 0.4, c_1 = 0.7, c_2 = 0.6$ and initial condition $\mathbf{X}_0 = 10$, run until $T = 10$. Solid lines show the estimated sample variance and the open circles show the average of the exact squared error over 32 realisations with given complexity (calculated using Appendix 2.A).

Stratified sampling

Rather than directly sampling from p we first divide the state space Ω of $\mathbf{Y}$ into disjoint sets such that $\Omega = \cup_{j=1}^{J} \Omega_j$, where J can theoretically be infinite[6]. These sets will be denoted as strata. We will use the notation $p_{\mathcal{D}_j} = \mathbb{P}(\mathbf{Y} \in \Omega_j)$ and define $\mathcal{D}_j$ to be the event that $\mathbf{Y} \in \Omega_j$. In the case that the distribution of $\mathbf{Y}$ is time-dependent, i.e. $\mathbf{Y} \sim p(t)$, note that $p_{\mathcal{D}_j}(t)$ will also be time-dependent. With this notation, and using the fact that the sets are disjoint, we can use conditional expectation and the law of total expectation to write

$$Q = \mathbb{E}[f(\mathbf{Y})] = \sum_{j=1}^{J} p_{\mathcal{D}_j} \mathbb{E}[f(\mathbf{Y}) | \mathcal{D}_j] = \sum_{j=1}^{J} p_{\mathcal{D}_j} Q_{\mathcal{D}_j}, \tag{2.44}$$

where $Q_{\mathcal{D}_j} = \mathbb{E}[f(\mathbf{Y}) | \mathcal{D}_j]$ denotes the summary statistic conditional on the event $\mathcal{D}_j$. If the $p_{\mathcal{D}_j}$ are known this suggests a new way to construct an estimator of Q:

$$\hat{Q}_{\text{str}} = \sum_{j=1}^{J} p_{\mathcal{D}_j} \hat{Q}_{\mathcal{D}_j}, \tag{2.45}$$

where the $\hat{Q}_{\mathcal{D}_j}$ are estimators of $Q_{\mathcal{D}_j}$. To construct these conditional estimators we denote $\mathbf{Y}^{(i,j)}$ to be the i-th sample drawn from the conditional distribution of $(\mathbf{Y}|\mathcal{D}_j)$ which yields

$$\hat{Q}_{\mathcal{D}_j} = \frac{1}{N_j} \sum_{i=1}^{N_j} f\left(\mathbf{Y}^{(i,j)}\right). \tag{2.46}$$

Note that if conditional sampling from $(\mathbf{Y}|\mathcal{D}_j)$ can be carried out exactly, the $\hat{Q}_{\mathcal{D}_j}$ are unbiased estimators of $Q_{\mathcal{D}_j}$ and, as a result, $\hat{Q}_{\text{str}}$ is an unbiased estimator of the summary statistic Q. We can write the resulting stratified estimator in the form of equation (2.39) via

$$\hat{Q}_{\text{str}} = \sum_{j=1}^{J} \sum_{i=1}^{N_j} \frac{p_{\mathcal{D}_j}}{N_j} f\left(\mathbf{Y}^{(i,j)}\right). \tag{2.47}$$

[6]Note that taking $J = \infty$ in a stratified sampling strategy will lead to infinite sampling cost. The case $J = \infty$ is therefore mainly of use in asymptotic analysis of stratified sampling so that we find an upper bound to the variance reduction that can be achieved using this method.

The main benefit of this stratified sampling approach, however, lies in the fact that by a judicious choice of the number of samples, N_j, per stratum we can make sure that $\hat{Q}_{\text{str}}$ has a lower variance than the standard MC estimator $\hat{Q}$. To see this, note that the law of total variance decomposes the variance of our object of interest, $f(\mathbf{Y})$, as

$$\sigma^2 = \text{Var}\left[f\left(\mathbf{Y}\right)\right] = \mathbb{E}\left[\text{Var}\left[f\left(\mathbf{Y}\right)|\mathcal{D}_j\right]\right] + \text{Var}\left[\mathbb{E}\left[f\left(\mathbf{Y}\right)|\mathcal{D}_j\right]\right]. \tag{2.48}$$

If we introduce σ_j^2 as the conditional variance of $f(\mathbf{Y})$ given $\mathcal{D}_j$ we can rewrite (2.48) as

$$\sigma^2 = \sum_{j=1}^{J} p_{\mathcal{D}_j}\sigma_j^2 + \sum_{j=1}^{J} p_{\mathcal{D}_j}\left(Q - Q_{\mathcal{D}_j}\right)^2. \tag{2.49}$$

As a result we see that for the standard MC estimator, i.e. via equation (2.17) and i.i.d. samples, the sample variance is given by

$$\text{Var}\left[\hat{Q}\right] = \frac{1}{N}\sum_{j=1}^{J} p_{\mathcal{D}_j}\sigma_j^2 + \frac{1}{N}\sum_{j=1}^{J} p_{\mathcal{D}_j}\left(Q - Q_{\mathcal{D}_j}\right)^2, \tag{2.50}$$

where N again denotes the number of sample paths used. On the other hand, for the stratified estimator with independent sample paths in each stratum we have sample variance

$$\text{Var}\left[\hat{Q}_{\text{str}}\right] = \sum_{j=1}^{J} p_{\mathcal{D}_j}^2 \frac{\sigma_j^2}{N_j}. \tag{2.51}$$

Note that it is possible to deviate from independence within strata, e.g. using the antithetic method within the strata, which would alter the sample variance for $\hat{Q}_{\text{str}}$ accordingly. Equation (2.51) demonstrates the fact that, in order to reduce the sample variance of a stratified estimator, we need to carefully specify how many samples, N_j, will be used per stratum, $\mathcal{D}_j$. Perhaps the simplest and most common choice is *proportional allocation*, i.e. given a budget of N samples in total we set $N_j = p_{\mathcal{D}_j}N$. For this choice the stratified estimator has a sample variance that is guaranteed to

be at least as small as the standard MC sample variance:

$$\mathrm{Var}\left[\hat{Q}_{\mathrm{prop}}\right] = \sum_{j=1}^{J} p_{\mathcal{D}_j}^2 \frac{\sigma_j^2}{N p_{\mathcal{D}_j}} \leq \frac{1}{N}\sum_{j=1}^{J} p_{\mathcal{D}_j}\sigma_j^2 + \frac{1}{N}\sum_{j=1}^{J} p_{\mathcal{D}_j}\left(Q - Q_{\mathcal{D}_j}\right)^2 = \mathrm{Var}\left[\hat{Q}\right], \tag{2.52}$$

where $\hat{Q}_{\mathrm{prop}}$ is given by equations (2.45) and (2.46) with $N_j = p_{\mathcal{D}_j} N$. It is therefore clear that a judicious stratification strategy, such as proportional allocation, can in fact be a variance reduction technique. Other choices of sample allocation can be made, such as optimal allocation and post-stratification, leading to different sample variances. We will not discuss such strategies further here, but refer the reader to [129, Section 4.7].

In order to estimate the sample variance of a stratified estimator, just as for the antithetic method, we need to modify equation (2.21), which only holds for the standard MC method. Given the conditional estimators as in equation (2.46), we can write down the unbiased conditional sample variance estimator

$$\hat{s}_j^2 = \frac{1}{N_j - 1}\sum_{i=1}^{N_j}\left(f\left(\mathbf{Y}^{(i,j)}\right) - \hat{Q}_{\mathcal{D}_j}\right)^2. \tag{2.53}$$

The unbiased estimator of the sample variance of the stratified estimator of the summary statistic, $\hat{Q}_{\mathrm{str}}$, is then given by

$$\hat{s}_{\mathrm{str}}^2 = \sum_{j=1}^{J} p_{\mathcal{D}_j}^2 \frac{\hat{s}_j^2}{N_j}. \tag{2.54}$$

Stratification has its roots in the field of statistical surveys, see for example [43], but has since seen application to problems outside the survey literature [84, Section 4.3]. Relevant in the context of this thesis, and in particular to both stratified and antithetic sampling, is the problem of generating (unit rate) Poisson processes. We can, for example, stratify with respect to the number of times these Poisson processes fire over a fixed time interval $[0, T)$, an idea which seems to date back to [63]. An

application of stratification using this idea in the context of chemical reactions is discussed in more detail in Section 3.3.

Latin hypercube sampling

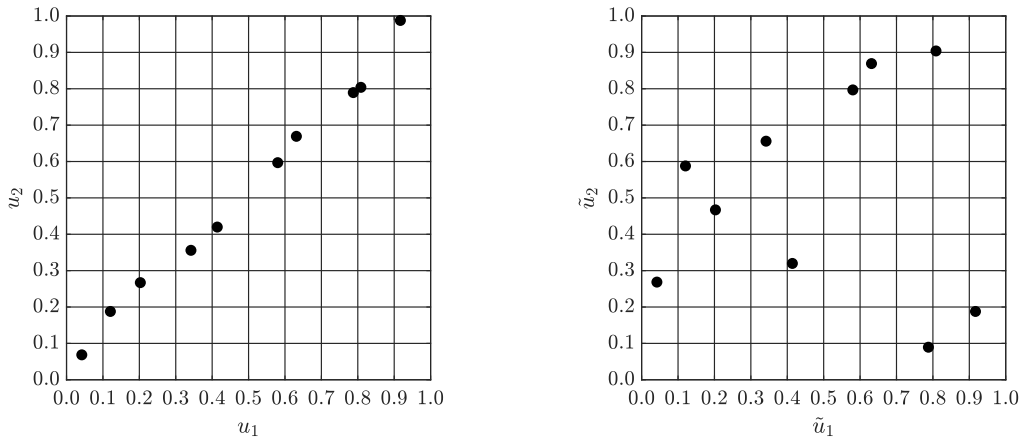
The stratified sampling strategy attempts to spread out the sample paths over a (fixed) number of strata. This, however, becomes problematic if the number of strata grows very large. A classical example of this issue is the sampling of points uniformly on the unit hypercube $[0,1)^d$ using the method of grid-based stratification [162, Section 10.1]. In this method we want to achieve stratification into equisized hypercubes with sides of length $1/N$. This results in N^d strata, in each of which we need at least two samples. As the number of dimensions d grows this method therefore quickly becomes infeasible. The same observation can be made for Cartesian products of numerical quadrature schemes, which are used to approximate integrals over the hypercube. One common approach to circumvent exponential growth of the number of strata with d is to restrict the stratification to a small subset of variables of dimension $\tilde{d} \ll d$, see Section 3.3 for an example where we only stratify with respect to one dimension of the problem.

It is, however, possible to achieve stratification of the samples onto all the one-dimensional projections of the points by using Latin hypercube sampling [149]. We work again on the premise that $\mathbf{Y} = \psi(\mathbf{u})$ where $\mathbf{u} \sim \mathcal{U}(0,1)^d$ and start with N samples $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(N)}$. We define $u_j^{(n)}$ to be the projection of the n-th sample point $\mathbf{u}^{(n)}$ onto the j-th dimension and note that in general $u_j^{(n)} \sim \mathcal{U}(0,1)$. Though this means that in the limit $N \to \infty$ the one-dimensional projections provide good cover, for finite N we can improve the equidistribution. To do so we enforce the condition that for each dimension j and each interval $[(i-1)/N, i/N)$, where $i = 1, \dots, N$, we have exactly one (and only one) $u_j^{(n)}$ contained in the interval. Note that this can be easily achieved by sampling points along the diagonal of the hypercube, see Figure 2.4(a). This does provide good cover for the one-dimensional projections, but

does not yield a uniform cover of the whole hypercube. We therefore apply an extra transformation and define $\tilde{\mathbf{u}}^{(n)}$ via

$$\tilde{u}_j^{(n)} = \frac{\pi_j(n) - 1 + u_j^{(n)}}{N}, \tag{2.55}$$

where π_j are independent random permutations of $\{1, \ldots, N\}$. The resulting Latin hypercube point set retains the good coverage for the one-dimensional projections, but at the same time is uniformly spread out over the hypercube, see Figure 2.4(b).



(a) $\pi_1 = \pi_2 = \mathrm{id}$.

(b) $\pi_1 = \mathrm{id}$, $\pi_2 = [3; 6; 5; 7; 4; 8; 9; 1; 10; 2]$.

Figure 2.4: Latin hypercube sample construction on $[0, 1]^2$ ($N = 10$). First points are sampled in a stratified manner along the hypercube diagonal in Figure 2.4(a). Applying the permutation transformation (2.55) yields a uniform covering point set with good one-dimensional projection properties in Figure 2.4(b). Note that id is the identity permutation.

In addition we note that we can view Latin hypercube sampling as a way to induce (component-wise) negative correlations between the random points $\tilde{\mathbf{u}}^{(n)}$ and $\tilde{\mathbf{u}}^{(m)}$, as we can easily derive the identity

$$\mathrm{Cov}\left[\tilde{u}_i^{(n)}, \tilde{u}_j^{(m)}\right] = \frac{1}{12}\delta_{ij} \begin{cases} 1, & m = n, \\ -\frac{N+1}{N^2}, & m \neq n. \end{cases} \tag{2.56}$$

It is possible to generalise the concept of the Latin hypercube sampling method to

yield better coverage of two or higher dimensional projections. The most straightforward extension along this line is orthogonal array sampling and we refer the reader to [162, Section 10.4] and references therein for more information on this approach. Alternatively the Latin hypercube can be seen as a first step in the QMC methodology, which we discuss in more detail in Chapter 4.

Using the Latin hypercube samples $\tilde{\mathbf{u}}^{(1)}, \ldots, \tilde{\mathbf{u}}^{(N)}$ we then define the Latin hypercube estimator

$$\hat{Q}_{\mathrm{LH}} = \frac{1}{N} \sum_{n=1}^{N} f\left(\psi\left(\tilde{\mathbf{u}}^{(n)}\right)\right). \tag{2.57}$$

Note that this estimator is unbiased due to the observation that $\tilde{\mathbf{u}} \sim \mathcal{U}(0,1)^d$ [162, Theorem 10.1]. The sample paths, however, are not independent which will make the sample variance deviate from the standard MC method variance. It was shown in [164] that for Latin hypercube sampling with N samples we have

$$\mathrm{Var}\left[\hat{Q}_{\mathrm{LH}}\right] \leq \frac{1}{N-1}\sigma^2 = \frac{N}{N-1}\mathrm{Var}\left[\hat{Q}\right], \tag{2.58}$$

where $\sigma^2 = \mathrm{Var}\left[f(\mathbf{Y})\right]$. In addition it can be proved that for models which are to first order described by the independent addition of d random contributions, Latin hypercube sampling forms a significant improvement over the standard MC approach [162, Proposition 10.1]. These results guarantee that, whilst Latin hypercube sampling can be a marked improvement on standard MC, in a worst case scenario its statistical error is only as large as using a standard MC method with one fewer sample point.

One of the downsides of the Latin hypercube sampling methodology is the lack of a simple estimator for the statistical error given N sample paths. To get a grip on the sample variance for *a posteriori* analysis we replicate M independent realisations of N sample paths, $\mathbf{Y}^{(n,m)}$, and from these construct $\hat{Q}_{\mathrm{LH}}^{(m)}$ for $m = 1, \ldots, M$. Using the M independent estimates we define the overall estimator $\hat{Q}_{\mathrm{LH}} = (1/M)\sum_m \hat{Q}_{\mathrm{LH}}^{(m)}$. An unbiased estimate for the sample variance of this estimator $\hat{Q}_{\mathrm{LH}}$ is given by the

usual MC sample variance

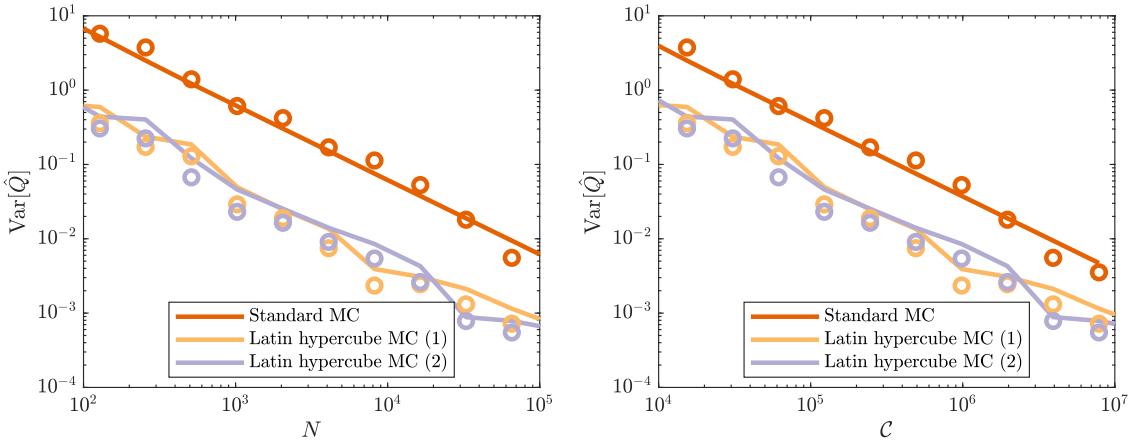
$$\hat{s}^2_{\mathrm{LH}} = \frac{1}{M-1}\sum_{m=1}^{M}\left(\hat{Q}^{(m)}_{\mathrm{LH}} - \hat{Q}_{\mathrm{LH}}\right)^2. \tag{2.59}$$

Note that this sample variance should be compared against other methods using NM sample paths.

Due to its simplicity and guaranteed efficiency Latin hypercube sampling is widely used in (high-dimensional) parameter sampling problems, see for example [213] and references therein. However, so far few direct use cases have been reported in the context of chemical reaction simulations. An example application of Latin hypercube sampling in the context of chemical reaction simulations is shown in Example 2.8 below.

Example 2.8 (Latin hypercube sampling for a simple linear system)**.**
Identically to Example 2.7 we directly identify the τ-leap method with a function $\mathbf{X} = f(\mathbf{u})$, where $\mathbf{u} \sim \mathcal{U}(0,1)^d$ with d given by the computational complexity in equation (2.36b). Replacing the i.i.d. uniform random input for an ensemble of N sample paths using τ-leap with a Latin hypercube sample of N points in $[0,1)^d$ we see a clear improvement in terms of the statistical error in Figure 2.5.

Two observations are in place here; to begin with we note that the standard Latin hypercube samples are not trivial to extend, i.e. given a Latin hypercube sample of N points in $[0,1)^d$ it is not trivial to create a Latin hypercube sample of $2N$ points in $[0,1)^d$. Two different approaches can therefore be taken, reflected by options (1) and (2) in Figure 2.5, respectively. First, we can simply sample a full Latin hypercube sample set of size N and if we need more samples not re-use the previous N samples, but instead generate new samples with a larger N. This seems of course an incredibly wasteful procedure, but it could in principal be beneficial due to the fact that increasing the number of points in a Latin hypercube set increases the one-dimensional stratification properties of the samples, which could further reduce



(a) Number of samples, N.

(b) Computational complexity, $\mathcal{C}$.

Figure 2.5: Statistical error, Var $\left[\hat{Q}\right]$, for the τ-leap method ($\tau = 0.5$) using either standard MC sampling or Latin hypercube sampling for the uniform random number input. The summary statistic is the mean species count for S_1 at time T. Results are for the single species system (2.38) in a volume $V = 1$ with parameters $c_0 = 0.4, c_1 = 0.7, c_2 = 0.6$ and initial condition $\mathbf{X}_0 = 10$, run until $T = 10$. Solid lines show the estimated sample variance and the open circles show the average of the exact squared error over 32 realisations (calculated using Appendix 2.A). Latin hypercube (1) uses full size N Latin hypercube samples, whereas Latin hypercube (2) joins independent Latin hypercube samples of size 2^6 until sample size N is reached.

the statistical error. Such a situation does arise in practice when simulating models that can be accurately described by d independent one-dimensional contributions. The alternative method samples a size N Latin hypercube set and, when needing to increase the sample size, we simply add the results of another independent Latin hypercube sample, for example again of size N (resulting in a total of $2N$ samples). For the example system in Figure 2.5 it is clear that option (2) is roughly on par with option (1) in terms of the variance reduction, and because option (2) is extensible, in the sense that it re-uses computations, it is the preferred method in this case. The final observation that we make is that Figure 2.5 reports the results both in terms of the number of sample paths, N, and the computational complexity, $\mathcal{C}$. This is done under the standard assumption that generating the Latin hypercube samples uses an extra $\mathcal{O}(dN)$ random numbers for the random permutations. The variance reduction as a function of the complexity is therefore lower (roughly four-fold), than when only

considering the number of sample paths (roughly nine-fold).

We note that a related approach previously appeared in a different form in [143], although it was considered as a stratification approach, rather than Latin hypercube sampling. Though equation (2.58) shows that the statistical error for Latin hypercube sampling is at worst only marginally larger than using standard MC, we can prove that Latin hypercube sampling is superior to standard MC when using the τ-leap method for a class of reaction systems with affine propensity functions. A direct proof of this fact follows from the observation that Latin hypercube sampling induces negative correlations on the level of the uniform random input, see equation (2.56), which proves a variant of [142, Lemma 4] for Latin hypercube sampling by [127, Theorem 2]. As a result [142, Corollary 2], which guaranteed improvement of antithetic sampling over standard MC sampling, is also valid for Latin hypercube sampling.

2.4.2 Coupling based techniques

A different scenario one often encounters in the literature on stochastic simulation methods is the estimation of a summary statistic related to the difference between two random variables $\mathbf{Y}$ and $\mathbf{Z}$,

$$Q = Q^{(y)} - Q^{(z)} = \mathbb{E}\left[\mathbf{Y}\right] - \mathbb{E}\left[\mathbf{Z}\right]. \tag{2.60}$$

Using the assumption that the two random variables $\mathbf{Y}$ and $\mathbf{Z}$ can be generated using uniform random variables we write

$$\begin{aligned} Q^{(y)} &= \mathbb{E}\left[g^{(y)}(\mathbf{u}, \mathbf{v}_1)\right] \\ Q^{(z)} &= \mathbb{E}\left[g^{(z)}(\mathbf{u}, \mathbf{v}_2)\right], \end{aligned}$$

where $\mathbf{u}$ is the random input that can be shared between $g^{(y)}$ and $g^{(z)}$. Note that $\mathbf{u}$, $\mathbf{v}_1$ and/or $\mathbf{v}_2$ can be empty, but $\mathbf{v}_1$ and $\mathbf{v}_2$ are independent. This specific structure

of the problem can be leveraged to create lower sample variance estimates compared to a naïve MC estimator that uses $2N$ independent samples to yield $\hat{Q}^{(y)}$ and $\hat{Q}^{(z)}$, respectively. This is in general achieved by coupling the simulation of the $\mathbf{Y}$ and $\mathbf{Z}$ variables (using $g^{(y)}$ and $g^{(z)}$) and we will now discuss some common techniques related to this topic.

Common random numbers

Instead of using independent estimates for $Q^{(y)}$ and $Q^{(z)}$ we write by linearity of expectation

$$Q = \mathbb{E}\left[g^{(y)}(\mathbf{u}, \mathbf{v}_1) - g^{(z)}(\mathbf{u}, \mathbf{v}_2)\right]. \tag{2.61}$$

This means we can generate N coupled sample paths for $\mathbf{Y}$ and $\mathbf{Z}$ by first sampling $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(N)}$, $\mathbf{v}_1^{(1)}, \ldots, \mathbf{v}_1^{(N)}$ and $\mathbf{v}_2^{(1)}, \ldots, \mathbf{v}_2^{(N)}$ independently. Note that we share the randomness in $\mathbf{u}$ between the sample paths, which explains the name of the common random numbers (CRN) method. Using the coupled samples we find an unbiased estimate for Q via

$$\hat{Q}_{\text{CRN}} = \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{Y}^{(n)} - \mathbf{Z}^{(n)}\right) = \frac{1}{N}\sum_{n=1}^{N}\left(g^{(y)}(\mathbf{u}^{(n)}, \mathbf{v}_1^{(n)}) - g^{(z)}(\mathbf{u}^{(n)}, \mathbf{v}_2^{(n)})\right). \tag{2.62}$$

The sample variance of this coupled estimator is given by

$$\text{Var}\left[\hat{Q}_{\text{CRN}}\right] = \frac{1}{N}\left(\sigma_1^2 + \sigma_2^2 - 2\sigma_1\sigma_2\rho_{1,2}\right), \tag{2.63}$$

where $\sigma_1^2 = \text{Var}\left[\mathbf{Y}\right]$, $\sigma_2^2 = \text{Var}\left[\mathbf{Z}\right]$ and $\rho_{1,2} = \text{Corr}\left[g^{(y)}(\mathbf{u}, \mathbf{v}_1), g^{(z)}(\mathbf{u}, \mathbf{v}_2)\right]$. Estimating the sample variance of $\hat{Q}_{\text{CRN}}$ can be done effectively using the usual (unbiased) MC estimator

$$\hat{s}^2 = \frac{1}{N-1}\sum_{n=1}^{N}\left[\left(\mathbf{Y}^{(n)} - \mathbf{Z}^{(n)}\right) - \hat{Q}_{\text{CRN}}\right]^2. \tag{2.64}$$

We note that the standard MC sample variance (using $2N$ samples) would be given by $(\sigma_1^2 + \sigma_2^2)/N$ and therefore equation (2.63) shows that we might improve

efficiency using CRNs if we can simulate $\mathbf{Y}$ and $\mathbf{Z}$ such that their sample paths are positively correlated, i.e. $\rho_{1,2} > 0$. For certain classes of problems, just as for the antithetic method, we can prove that $\rho_{1,2} > 0$, see for example [129, Section 4.8]. In general, though, the CRN method, despite being perhaps the intuitive way to calculate difference summary statistics like (2.60), is not guaranteed to improve upon simple MC; the efficiency of the CRN approach, relative to simple MC, is problem-dependent and influenced by the complexity of the CRN sampling versus the simple MC sampling.

A common scenario in which the CRN method can be effectively employed is the estimation of parameter sensitivities. Suppose the model of interest depends on a variable θ, e.g. one of the reaction rate constants in a chemical reaction network, then one might want to know how quantities like $f(\mathbf{u}, \theta)$ vary as θ is changed. This can be quantified using a finite difference approximation of the derivative with respect to θ,

$$\begin{aligned}\left.\frac{\mathrm{d}\mathbb{E}\left[f\left(\mathbf{u};\theta\right)\right]}{\mathrm{d}\theta}\right|_{\theta=\theta^*} &\approx \frac{\mathbb{E}\left[f\left(\mathbf{u};\theta^*+\varepsilon\right)\right]-\mathbb{E}\left[f\left(\mathbf{u};\theta^*-\varepsilon\right)\right]}{2\varepsilon}\\ &= \frac{\mathbb{E}\left[f\left(\mathbf{u};\theta^*+\varepsilon\right)-f\left(\mathbf{u};\theta^*-\varepsilon\right)\right]}{2\varepsilon}.\end{aligned}$$

Note that this (local) measure of sensitivity takes the form of equation (2.61) and we can therefore use the CRN approach to find an accurate estimate of the derivative. Because we generally take ε small to suppress the bias in the finite-difference approximation we expect $f(\mathbf{u};\theta^*\pm\varepsilon)$ to have high correlation, and therefore by equation (2.63) there is a large benefit from using the CRN method. For an overview and more information on the problem of parameter sensitivity analysis in the context of chemical reaction networks we refer the reader to [11, 131, 203] and references therein. Later in this section we will discuss two further variance reduction techniques relevant to the context of chemical reaction networks, the multifidelity Monte Carlo (MFMC) and multilevel Monte Carlo (MLMC) methods, which also both heavily rely on coupling sample paths and this is predominantly achieved using some form of the CRN approach.

Control variates

Any technique derived for difference summary statistics, like those in equation (2.60), can also be used to create improved estimators for general summary statistics $Q = \mathbb{E}[\mathbf{Y}]$ where $\mathbf{Y} \sim p$. To do so we note that in general we can write by linearity of expectation

$$Q = \mathbb{E}[\mathbf{Y} - \alpha\mathbf{Z}] + \alpha\mathbb{E}[\mathbf{Z}]. \tag{2.65}$$

To make this representation useful we note that $\mathbf{Z}$ is completely arbitrary and we can therefore choose it so that we either know $\mathbb{E}[\mathbf{Z}] = \mu_Z$ exactly, see for example [84, Section 4.1] for examples in the context of SDEs in finance, or can estimate μ_Z more easily than Q [58, 169]. This turns the summary statistic Q again into the expectation of the difference of two quantities, $\mathbf{Y}$ and $\mathbf{Z}$, respectively. As a result we can use a variance reduced estimator for Q if we can find a suitable stochastic process $\mathbf{Z}$ so that i) we can generate it together with $\mathbf{Y}$ such that their sample paths are correlated and ii) its expectation is either known or inexpensive to compute. This control variate estimator is then defined by

$$\hat{Q}_{\text{cv}}(\alpha) = \alpha\mu_Z + \frac{1}{N}\sum_{n=1}^{N}\left(\mathbf{Y}^{(n)} - \alpha\mathbf{Z}^{(n)}\right), \tag{2.66}$$

where we recall that $\mathbf{Y}^{(n)}$ and $\mathbf{Z}^{(n)}$ are generally coupled. Note that the sample variance of $\hat{Q}_{\text{cv}}$ is given by

$$\text{Var}\left[\hat{Q}_{\text{cv}}(\alpha)\right] = \frac{1}{N}\left(\sigma^2 + \alpha^2\text{Var}[\mathbf{Z}] - 2\alpha\text{Cov}\left[\mathbf{Y}^{(n)}, \mathbf{Z}^{(n)}\right]\right),$$

where $\sigma^2 = \text{Var}[\mathbf{Y}]$. Using the freedom in the weight, α, we can determine its optimal value

$$\alpha^* = \frac{\text{Cov}\left[\mathbf{Y}^{(n)}, \mathbf{Z}^{(n)}\right]}{\text{Var}[\mathbf{Z}]}. \tag{2.67}$$

This optimal weight parameter, α^*, then yields the lowest variance estimator, $\hat{Q}_{\text{cv}}(\alpha^*)$, with variance given by

$$\text{Var}\left[\hat{Q}_{\text{cv}}(\alpha^*)\right] = \frac{1}{N}\left(\sigma^2 - \frac{\text{Cov}\left[\mathbf{Y}^{(n)}, \mathbf{Z}^{(n)}\right]^2}{\text{Var}\left[\mathbf{Z}\right]}\right) = \frac{\sigma^2}{N}(1-\rho^2), \tag{2.68}$$

where ρ is the correlation between $\mathbf{Y}^{(n)}$ and $\mathbf{Z}^{(n)}$. In general, however, it is not possible to take $\alpha = \alpha^*$, because the covariance between $\mathbf{Y}$ and $\mathbf{Z}$ is unknown. One therefore often estimates α via linear least squares [129, Section 4.4], yielding an approximate $\hat{\alpha}$. If we use our original samples used for the construction of $\hat{Q}_{\text{cv}}$ to also perform the linear least squares estimation then this complicates matters slightly by introducing a bias in the estimator $\hat{Q}_{\text{cv}}(\hat{\alpha})$. Even though this bias is ordinarily negligible if we consider enough samples, N, we can also remove the bias by either using a few independent pilot samples ($\mathcal{O}(\sqrt{N})$ is often optimal) to estimate $\hat{\alpha}$ or by splitting our samples into independent sets [14]. Using any of the methods described to estimate α^* means the control variate method becomes very versatile and the sample variance is often only marginally larger than the optimal result in equation (2.68)[7]. Generalisation of the method to multiple control variates $\mathbf{Z}_1, \ldots, \mathbf{Z}_L$ is also trivial in this framework; we simply estimate multiple weight parameters $\alpha_1, \ldots, \alpha_L$ using a single least squares approach.

Looking at the sample variance of a control variate estimator using N sample paths (for both $\mathbf{Y}$ and $\mathbf{Z}$) we see that it is always equal to or smaller than the standard MC estimator using N sample paths for $\mathbf{Y}$. Note that this shows that the efficiency of the control variate method relative to the standard MC method depends on the complexity of generating the $\mathbf{Z}^{(n)}$ control variates coupled to the $\mathbf{Y}^{(n)}$ samples, which in general is problem dependent.

Effective direct applications of the control variate method in the context of chem-

[7] This is generally true when $\hat{\alpha} \approx \alpha^*$ (which is guaranteed when N grows large) by the observation that the derivative of $\text{Var}\left[\hat{Q}_{\text{CV}}(\alpha)\right]$ with respect to α is zero at α^*.

ical reaction networks can be found in [15, 154]. In [154] an alternative τ-leap path driven by a solution of the RRE is coupled to a regular τ-leap sample path. This approach is effective due to the observation that the control variate has known (non-zero) expectation and is inexpensive to calculate. An alternative approach was taken in [15] where known constraints on the raw moments of the species, derived from the direct integration of the CME, are used to derive control variates with zero expectation. We note that this latter approach relies on mass action kinetics modelling of the reaction propensities.

A different extension of control variates, called Monte Carlo with least-squares (MCLS), was recently proposed in [158]. Using this method we can derive control variates for any black box simulator via function approximation theory using suitably chosen basis functions and random input into the simulator, often in the form of standard uniform random variates on the hypercube. An illustration of the efficacy of MCLS in the context of chemical reactions can be found in Example 2.9 below.

Example 2.9 (MCLS for a simple linear system)**.**

Again we consider the same set-up as in Example 2.7 and consider the τ-leap method as a function $\mathbf{X} = f(\mathbf{u})$, where $\mathbf{X}$ is a sample path and $\mathbf{u} \sim \mathcal{U}(0,1)^d$ with d given by the computational complexity in equation (2.36b). Using the monomial basis functions to approximate the unknown f yields the simplest form of the MCLS method and in Figure 2.6 we show the convergence of the statistical error when we restrict the basis functions to degree-one monomials. There is roughly a six-fold improvement of MCLS over standard MC.

Note that the number of basis functions in this case is equal to $d + 1$. Taking higher degree basis functions will yield larger variance reductions, but this comes at the price of a more complex least-squares system to solve. Though the MCLS does not use any extra random numbers, and therefore would have the same complexity as the standard MC method, we note that it involves the solution of a possibly large linear algebra problem (scaling with the number of samples and the number of basis

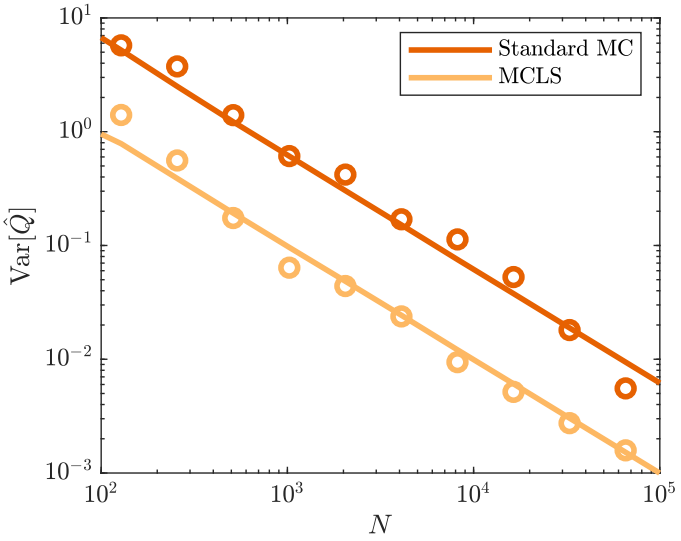


Figure 2.6: Statistical error, $\mathrm{Var}\left[\hat{Q}\right]$, as a function of the number of sample paths, N, for the τ-leap method ($\tau = 0.5$) using either standard MC sampling or MCLS (degree 1 monomial basis functions). The summary statistic is the mean species count for S_1 at time T. Results are for the single species system (2.38) in a volume $V = 1$ with parameters $c_0 = 0.4, c_1 = 0.7, c_2 = 0.6$ and initial condition $\mathbf{X}_0 = 10$, run until $T = 10$. Solid lines show the estimated sample variance and the open circles show the average of the exact squared error over 32 realisations with a given number of sample paths, N.

functions), which will add significantly to the memory requirement and run-time of the simulations.

Multilevel and multifidelity Monte Carlo

A more recent and very successful variance reduction technique, also used in the context of chemical reaction networks, is the MLMC method, which was first proposed in [71]. We first consider the most popular variant of the MLMC method, geometric MLMC. For this method we consider a sequence of hierarchical models for $l = 0, \ldots, L$, which we will denote as the levels, with sample paths $\mathbf{Z}_l$. Generally speaking we assume that the models for small l, the coarse levels, are inexpensive to simulate, but provide inaccurate, or biased, results; whereas for large l, the fine levels, the situation is reversed in the sense that the models are more accurate, but

it is expensive to generate sample paths from them. Suppose then that we want to estimate $Q = \mathbb{E}[\mathbf{Z}_L]$, which by standard MC would be expensive to do. Instead we note that we can use the hierarchical models to write

$$Q = \mathbb{E}[\mathbf{Z}_L] = \mathbb{E}[\mathbf{Z}_{L-1}] + \mathbb{E}[\mathbf{Z}_L - \mathbf{Z}_{L-1}] \tag{2.69a}$$

$$= \mathbb{E}[\mathbf{Z}_{L-2}] + \mathbb{E}[\mathbf{Z}_{L-1} - \mathbf{Z}_{L-2}] + \mathbb{E}[\mathbf{Z}_L - \mathbf{Z}_{L-1}] \tag{2.69b}$$

$$\dots$$

$$= \underbrace{\mathbb{E}[\mathbf{Z}_0]}_{Q_0} + \sum_{l=1}^{L} \underbrace{\mathbb{E}[\mathbf{Z}_l - \mathbf{Z}_{l-1}]}_{Q_l}. \tag{2.69c}$$

Using equation (2.69) to estimate Q would in general use sample paths from all hierarchical models via

$$\hat{Q}_{\text{MLMC}} = \hat{Q}_0 + \sum_{l=1}^{L} \hat{Q}_l \tag{2.70a}$$

$$= \frac{1}{N_0} \sum_{n=1}^{N_0} \mathbf{Z}_0^{(n)} + \sum_{l=1}^{L} \frac{1}{N_l} \sum_{n=1}^{N_l} \left(\mathbf{Z}_l^{(n)} - \mathbf{Z}_{l-1}^{(n)} \right). \tag{2.70b}$$

This expansion of Q onto a coarse level Q_0 and a series of corrections Q_l might initially seem like a counterproductive approach to estimate Q, due to the increased number of summary statistics that now need computation as well. However, if we note the similarity between equation (2.69) and the control variate formalism we encountered in the previous section, it should be clear that if we can couple the generation of $\mathbf{Z}_l^{(n)}$ and $\mathbf{Z}_{l-1}^{(n)}$ sample paths, e.g. using the CRN approach, the variance of the corrections $\hat{Q}_l$ could be much lower than if they were estimated using standard MC. This way one can reduce the estimator complexity by using many inexpensive simulations for the coarse level combined with a few expensive fine level corrections to reduce the bias. This was first realised in the context of SDE simulations in [71], where hierarchical models follow from the Euler-Maruyama numerical discretisation of the SDE with different step sizes. It was shown that a judicious choice of the number of samples

used to construct each of the $\hat{Q}_l$ can yield a variance reduction method that improves upon standard MC by several orders of magnitude. For an excellent overview and more details on the MLMC method in general we refer the reader to [72].

In the context of chemical reaction networks it was shown in [5] how to construct a multilevel method using a hierarchy of models stemming from using the τ-leap method with different step sizes. Rather than using the more standard CRN approach with inverse transform sampling to generate coupled paths, the authors use the thinning and thickening properties of Poisson processes to provide a good coupling approach. In addition they show how to remove the bias stemming from using the τ-leap method for the levels in the MLMC approach. This is done by using as the finest level correction, Q_{L+1}, the difference between sample paths from the exact SSA and the τ-leap method with the smallest step size used. The complexity of the resulting MLMC method for chemical reaction networks is studied in [7, 8] and it is shown that a correct implementation of the MLMC method will be at least as efficient as standard MC. For a practical guide on how to implement the MLMC method and choose the number of samples for each level we refer the reader to [133]. Several extensions and refinements of the original algorithm in [5] have been proposed over the last few years, which we will point out next. Firstly, in [132] the use of an adaptive step size in the τ-leap levels is introduced in order to efficiently simulate models for which reaction activity changes strongly over the course of a sample path of interest, e.g. in stiff systems. In a similar vein the use of implicit τ-leap methods allows one to take larger time steps for stiff systems and an MLMC variant was developed in [22]. An MLMC method using a hybrid between τ-leap and the MNRM for the correction levels in order to ensure that τ-leap simulations do not yield negative species counts can be found in [154]. An extension of this hybrid method is described in [153] and is used to adaptively speed up the simulation of fast and slow reaction channels by allowing one to switch between the τ-leap method and an exact SSA for each individual reaction channel. Even though most MLMC methods have used the τ-leap method for the

coarse level simulations it is also possible to use the R-leap method in an MLMC approach [135]. This latter work also describes a different coupling technique for τ-leap sample paths, which can improve upon the standard coupling in [5]. Though most of the MLMC methods consider the problem of estimating moments of chemical species it is possible to extend the MLMC approach to estimate densities [220].

Finally, a recent variant of the MLMC method is the MFMC method [172], which, unlike the most common MLMC variant, geometric MLMC, does not assume a hierarchical sequence of models. This method, in a similar vein to the MLMC method, provides a way to combine (cheaper) low-fidelity surrogate models, e.g. based on model approximations, with (expensive) high-fidelity models in an optimal control variate framework. We refer the reader to [171, 172] for mathematical details of the MFMC method and [173] for a comprehensive review of multifidelity methods in general. Though the MFMC method does not yet appear widely in the context of chemical reaction networks, the final example in [5, Section 9] shows that multifidelity approaches, in that case based on a quasi-steady state approximation, have large potential benefits. Another example of this methodology, relevant for multiscale systems in terms of species abundance, can be found in [136]. Future example usages could be the combination of methods based on the full CME, such as the DM, with reduced and biased models based on multiscale arguments, e.g. see [193, Section 4.6] and references therein, or other reduction techniques, e.g. [29, 61].

2.4.3 Importance sampling

Importance sampling is the final category of variance reduction methods we discuss in this thesis and has perhaps been the most explored variance reduction technique in the context of the simulation of chemical reaction networks. We start with the summary statistic $Q = \mathbb{E}[f(\mathbf{Y})]$ where $\mathbf{Y} \sim p$, but rather than sampling directly

from the distribution p we note that we can write

$$Q = \mathbb{E}\left[f(\mathbf{Y})\right] = \int_{\Omega} f(\omega)p(\omega)\,\mathrm{d}\omega = \int_{\Omega} f(\omega)\frac{p(\omega)}{q(\omega)}q(\omega)\,\mathrm{d}\omega, \tag{2.71}$$

for some function q with the property that $q(\omega) \neq 0$ when $f(\omega)p(\omega) \neq 0$. Due to the generality of equation (2.71) we note that we can take q to be a new distribution, i.e. $q \geq 0$ and normalised to unity. Defining the new function $r(\omega) = p(\omega)/q(\omega)$, called the likelihood ratio, we then see via equation (2.71) that if we let $\tilde{\mathbf{Y}} \sim q$

$$\mathbb{E}\left[f(\mathbf{Y})\right] = \mathbb{E}\left[f(\tilde{\mathbf{Y}})r(\tilde{\mathbf{Y}})\right]. \tag{2.72}$$

The significance of this result is that we can estimate Q, which is related to the distribution p, via sampling from an alternative distribution, q, called the importance distribution. Using N sample paths, $\tilde{\mathbf{Y}}^{(n)}$, drawn from the distribution q we construct the importance sampling estimate

$$\hat{Q}_{\mathrm{IS}} = \frac{1}{N}\sum_{n=1}^{N} f\left(\tilde{\mathbf{Y}}^{(n)}\right) r\left(\tilde{\mathbf{Y}}^{(n)}\right), \tag{2.73}$$

which is an unbiased estimate for Q. The variance of $\hat{Q}_{\mathrm{IS}}$ in general, however, is different from the usual MC estimator and given by

$$\mathrm{Var}\left[\hat{Q}_{\mathrm{IS}}\right] = \frac{1}{N}\mathrm{Var}\left[f\left(\tilde{\mathbf{Y}}\right) r\left(\tilde{\mathbf{Y}}\right)\right]. \tag{2.74}$$

A variant of the usual MC sample variance estimator, in equation (2.21), gives an unbiased estimate for the sample variance of the importance sampling estimator

$$\hat{s}^2 = \frac{1}{N-1}\sum_{n=1}^{N}\left[f\left(\tilde{\mathbf{Y}}^{(n)}\right) r\left(\tilde{\mathbf{Y}}^{(n)}\right) - \hat{Q}_{\mathrm{IS}}\right]^2. \tag{2.75}$$

One can show that a judicious choice of importance distribution, q, yields lower variance estimators, whilst a bad choice for q can make matters significantly worse as well. Importance sampling is therefore not guaranteed to improve upon standard MC, though its potential can be large, see e.g. [84, Sections 4.6 and 4.7]. The best choice of q is problem-dependent, relating to both the type of summary statistic via f and the underlying distribution p, and its construction can require numerical optimisation and/or expert knowledge[8]. For more background information and a discussion of the subtleties of importance sampling we refer the reader to [162, Chapter 9].

In the context of chemical reaction networks importance sampling is predominantly used for rare event simulation, i.e. the exploration of an event, e.g. extinction of a specific species, or system state when its occurrence probability is very small. By sampling from an importance distribution we can force the dynamics of an altered system to encounter such rare events of interest more frequently, whilst keeping the sampling correct for the original system of interest by weighting via the likelihood ratio. Importance sampling is achieved by changing the (relative) reaction propensities, either increasing or decreasing them depending on the rare event of interest. The resulting weighted stochastic simulation algorithm (wSSA) was first proposed in [118] and further developed in the following years, see for example [46, 74, 82, 189, 190, 191]. Finally, the use of a variant of the wSSA was recently proposed in the context of the MLMC method for chemical reaction simulations in [21]. Rather than using the wSSA to focus on more traditional rare event simulation they employ importance sampling via changing reaction propensities in order to reduce the phenomenon of catastrophic coupling. This occurs when sample paths are almost too tightly coupled and appear identical in all but a few of the simulations. Such a tight coupling naturally arises at the fine levels of an MLMC algorithm, where the two different levels become increasingly similar. Though in expectation this is the desired behaviour, the variance estimation in this circumstance becomes troublesome due to

[8] On a theoretical level the condition $q \propto fp$ yields the optimal results, i.e. the variation of q is proportional to variation in the product of f and the original distribution p.

the high kurtosis. By forcing the difference between the two sample paths to increase in a controlled manner via scaling of the reaction propensities the authors show they can alleviate the issue of high kurtosis.

2.5 Outlook

In this review chapter we introduced the framework for this thesis, namely the efficient simulation of CTMC models of chemical reactions using MC methods. Historically much of the progress in increasing efficiency has been made by either introducing approximate SSAs or by making the implementation of currently available SSAs faster. An orthogonal approach, which has received less attention, however, is the use of variance reduction techniques. These methods directly tackle the major bottleneck for many stochastic simulation scenarios, namely the statistical error. A general overview of the common trends among these methods was given at the end of this chapter. Theory and background information were supplemented with illustrations on how these techniques can be used to improve simulations of chemical reaction networks; either by referring to relevant literature or by showcasing the methods' effectiveness on tractable test problems.

Many of these variance reduction techniques are not only orthogonal to previous efforts to increase efficiency, but also to each other. As a result there is scope to combine variance reduction techniques with each other and with known efficient (approximate) SSAs, without the need to drastically change them. This rest of this thesis therefore explores the use of the variance reduction approach in the context of chemical reaction simulations. Using these ideas we discuss new methods with improved efficiency, both by building on existing SSAs and by introducing new simulation methodologies.

Chapter appendix

2.A Analytic moments for the τ-leap method in systems with affine propensity functions

Inspired by convergence studies for τ-leap methods such as [103, 182], we derive expressions for the first two central moments when applying the (naïve) τ-leap method to a system for which the propensity functions are affine functions of the state variables, i.e. $a_k(\mathbf{X}) = a + \mathbf{b}^\intercal \mathbf{X}$. The availability of exact moments can be useful in testing correct implementation of algorithms and to study properties of the τ-leap method on a more theoretical level, for example convergence and consistency [10, 103, 181, 182]. Exact expressions for the first moment of systems with purely linear propensity functions using both the explicit and implicit τ-leap simulation method can also be found in [182].

To see how these systems arise in practice consider a linear reaction system, which is defined by having K reactions of the form

$$\alpha_{1,k} S_1 + \cdots + \alpha_{J,k} S_J \xrightarrow{c_k} \beta_{1,k} S_1 + \cdots + \beta_{J,k} S_J,$$

where the restriction is that $\alpha_{j,k} \in \{0, 1\}$ for all j, k and at most one $\alpha_{j,k} = 1$ for a given reaction channel k. In addition we assume mass-action kinetics which by the previous restriction implies that we can write the propensity function as an affine function of the state vector, i.e. $a_k(\mathbf{X}) = \gamma_k V + \mathbf{c}_k^\intercal \mathbf{X}$, where $\mathbf{c}_k = c_k(\alpha_{1,k}, \ldots, \alpha_{J,k})^\intercal$ and $\gamma_k = c_k$ if $\alpha_{j,k} = 0$ for all j and 0 otherwise. For future reference we further define $\mathcal{R}$ as the set of all reactions in the system, i.e. $\mathcal{R} = \{R_1, \ldots, R_K\}$. We then note that $\mathcal{R} = \mathcal{R}_0 \cup \mathcal{R}_1$, where $\mathcal{R}_1$ denotes the set of all reactions with one reactant and $\mathcal{R}_0$ is the set of all zeroth order reactions, i.e. those of the form

$$\emptyset \xrightarrow{\gamma_k} \beta_{1,k} S_1 + \cdots + \beta_{J,k} S_J.$$

We note that for a large class of systems within the set of all linear reaction systems it is possible to make analytic progress with the CME, as shown in [104, 184]. Here, however, we derive analytic results when applying the τ-leap method to the system. For the remainder of this appendix we abuse notation and let $\mathbf{X}_n = \mathbf{X}(n\tau)$. Using this notation we find using equation (2.9) the recurrence relation for $\mathbf{X}$ when applying the τ-leap method[9]

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \sum_{\mathcal{R}_0} Y_k\left(\gamma_k V \tau\right) \boldsymbol{\zeta}_k + \sum_{\mathcal{R}_1} Y_k\left(\mathbf{c}_k^{\intercal} \mathbf{X}_n \tau\right) \boldsymbol{\zeta}_k. \tag{2.76}$$

We therefore see that to compute expectations of $\mathbf{Z}_{n+1}$ we need to use known results on the expectation of Poisson random variables, e.g.

$$\begin{aligned} \mathbb{E}\left[Y_k\left(\lambda_k\right)\right] &= \lambda_k, \\ \mathbb{E}\left[Y_k(\lambda_k) Y_{k'}\left(\lambda_{k'}\right)\right] &= \lambda_k \lambda_{k'} + \delta_{kk'} \lambda_k. \end{aligned}$$

We note that the above equalities, whilst true for Poisson random variables, do in fact also hold if we replace $Y_k(\lambda_k)$ with its normal approximation $\mathcal{N}_k(\lambda_k, \lambda_k)$. Therefore the results for the first two moments derived in this appendix for the τ-leap method are actually also equal to the first two moments obtained when applying the Euler-Maruyama scheme to the CLE approximation to the system, which can be thought of as the τ-leap equivalent for the CLE, see for example (2.10). Note that a similar equivalence holds in the limit $\tau \to 0$ between the first two moments of the CLE and the CME for linear reaction systems [89].

[9]Here we implicitly assume that the τ-leap method yields feasible sample paths, i.e. we assume the absence of negative molecule numbers when applying the τ-leap method. This is typically the case in parameter regimes for which the τ-leap method is applicable, but refer the reader to [182] for a more detailed discussion.

Mean

We start by taking an expectation of equation (2.76) conditional on $\mathbf{X}_n$ and use the mean of the Poisson random variables to find

$$\mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right] = \mathbf{X}_n + \underbrace{\left(\sum_{\mathcal{R}_0} \boldsymbol{\zeta}_k \gamma_k V\right)}_{\mathbf{b}} \tau + \underbrace{\left(\sum_{\mathcal{R}_1} \boldsymbol{\zeta}_k \mathbf{c}_k^{\intercal}\right)}_{\mathbf{A}} \mathbf{X}_n \tau.$$

Then, by the law of total expectation, we find the recurrence relation for the mean of the system

$$\begin{aligned}\mathbb{E}\left[\mathbf{X}_{n+1}\right] &= \mathbb{E}\left[\mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right]\right] \\ &= \underbrace{(I + \mathbf{A}\tau)}_{\mathcal{L}_0} \mathbb{E}\left[\mathbf{X}_n\right] + \mathbf{b}\tau.\end{aligned}$$

Using the recurrence relation we find a general solution

$$\mathbb{E}\left[\mathbf{X}_n\right] = \mathcal{L}_0^n \mathbf{X}_0 + \left(\sum_{i=0}^{n-1} \mathcal{L}_0^i\right) \mathbf{b}\tau = (I + \mathbf{A}\tau)^n \mathbf{X}_0 + \left(\sum_{i=0}^{n-1} (I + \mathbf{A}\tau)^i\right) \mathbf{b}\tau \tag{2.77}$$

As a sanity check we see that in the limit of small step size, τ, we recover the solution given by solving the CME directly:

$$\lim_{\tau \to 0, n\tau \to t} \mathbb{E}\left[\mathbf{X}_n\right] = \exp\left(\mathbf{A}t\right) \mathbf{X}_0 + \left(I - \exp\left(\mathbf{A}t\right)\right) \mathbf{A}^{-1} \mathbf{b}, \tag{2.78}$$

if $\mathbf{A}$ is invertible. When $\mathbf{A}$ is not invertible we can use the Jordan normal form of $\mathbf{A}$ to recover the standard solution in the same limit

$$\lim_{\tau \to 0, n\tau \to t} \mathbb{E}\left[\mathbf{X}_n\right] = \exp\left(\mathbf{A}t\right) \mathbf{X}_0 + \left(\int_0^t \exp\left(\mathbf{A}s\right) \, \mathrm{d}s\right) \mathbf{b}. \tag{2.79}$$

Covariance

For the covariance the calculations become more involved, but we can proceed as before if we use the law of total covariance, i.e.

$$\mathrm{Cov}\left[\mathbf{X}_{n+1}, \mathbf{X}_{n+1}\right] = \mathbb{E}\left[\mathrm{Cov}\left[\mathbf{X}_{n+1}, \mathbf{X}_{n+1} \mid \mathbf{X}_n\right]\right] + \mathrm{Cov}\left[\mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right], \mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right]\right]. \tag{2.80}$$

We start by considering the matrix $\mathbf{X}_{n+1}\mathbf{X}_{n+1}^{\intercal}$ and its conditional expectation with respect to $\mathbf{X}_n$ to first calculate the conditional covariance

$$\begin{aligned}\mathrm{Cov}\left[\mathbf{X}_{n+1}, \mathbf{X}_{n+1} \mid \mathbf{X}_n\right] &= \mathbb{E}\left[\mathbf{X}_{n+1}\mathbf{X}_{n+1}^{\intercal} \mid \mathbf{X}_n\right] - \mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right]\mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right]^{\intercal} \\ &= \mathcal{L}_2\mathbf{X}_n + \mathbf{B}\tau,\end{aligned}$$

where $\mathcal{L}_2$ is a deterministic linear operator and $\mathbf{B}$ given by

$$\begin{aligned}\mathcal{L}_2\mathbf{v} &= \underbrace{\left(\sum_{\mathcal{R}_1} \boldsymbol{\zeta}_k\boldsymbol{\zeta}_k^{\intercal}\mathbf{c}_k^{\intercal}\right)}_{\mathcal{A}}\mathbf{v}\tau, \\ \mathbf{B} &= \left(\sum_{\mathcal{R}_0} \boldsymbol{\zeta}_k\gamma_k V\boldsymbol{\zeta}_k^{\intercal}\right).\end{aligned}$$

The covariance of the conditional expectation is easily shown to be equal to

$$\mathrm{Cov}\left[\mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right], \mathbb{E}\left[\mathbf{X}_{n+1} \mid \mathbf{X}_n\right]\right] = \mathcal{L}_1\mathrm{Cov}\left[\mathbf{X}_n, \mathbf{X}_n\right],$$

where $\mathcal{L}_1$ is another deterministic linear operator given by

$$\mathcal{L}_1\mathbf{M} = \left(I + \mathbf{A}_1\tau\right)\mathbf{M}\left(I + \mathbf{A}_1\tau\right)^{\intercal}.$$

Because the linear operators are deterministic and taking the expectation is also linear operation we can interchange them. We therefore find, by the law of total

expectation, a generalised recurrence relation for $\mathrm{Cov}\left[\mathbf{X}_{n+1}, \mathbf{X}_{n+1}\right]$:

$$\mathrm{Cov}\left[\mathbf{X}_{n+1}, \mathbf{X}_{n+1}\right] = \mathcal{L}_1 \mathrm{Cov}\left[\mathbf{X}_n, \mathbf{X}_n\right] + \mathcal{L}_2 \mathbb{E}\left[\mathbf{X}_n\right] + \mathbf{B}\tau. \tag{2.81}$$

This then yields a general solution of the form

$$\mathrm{Cov}\left[\mathbf{X}_n, \mathbf{X}_n\right] = \mathcal{L}_1^n \mathrm{Cov}\left[\mathbf{X}_0, \mathbf{X}_0\right] + \sum_{i=0}^{n-1} \mathcal{L}_1^i \mathcal{L}_2 \mathbb{E}\left[\mathbf{X}_{(n-1)-i}\right] + \left(\sum_{i=0}^{n-1} \mathcal{L}_1^i\right) \mathbf{B}\tau. \tag{2.82}$$

Using the results in the previous section for the mean of $\mathbf{X}_n$ and assuming deterministic initial conditions we find the covariance matrix for the τ-leap method in symbolic form

$$\mathrm{Cov}\left[\mathbf{X}_n, \mathbf{X}_n\right] = \left(\sum_{i=0}^{n-1} \mathcal{L}_1^i \mathcal{L}_2 \mathcal{L}_0^{(n-1)-i}\right) \mathbf{X}_0 + \left(\sum_{i=0}^{n-1} \mathcal{L}_1^i \mathcal{L}_2 \sum_{j=0}^{n-2-i} \mathcal{L}_0^j\right) \mathbf{b}\tau + \left(\sum_{i=0}^{n-1} \mathcal{L}_1^i\right) \mathbf{B}\tau. \tag{2.83}$$

This covariance formula is explicit and can therefore be used to calculate the covariance for a given set of parameters by applying all the linear operators. However, it is also insightful to consider some examples for certain special classes of reaction networks in which the expressions can be simplified. These will be used throughout this thesis.

Example (Single species)**.** A single species S_1 system is any system of the form

$$\begin{aligned} \emptyset &\xrightarrow{\gamma_j} \beta_j S_1, \\ S_1 &\xrightarrow{c_{j'}} \beta_{j'} S_1, \end{aligned}$$

where we assume $\beta_j, \beta_{j'} \in \mathbb{N}_{\geq 0}$. In this case we can further simplify the algebra by noting that $\mathbf{X} = X$. This means that we have a one-dimensional state vector.

Furthermore we have

$$\mathbf{A} = a_1, \quad \boldsymbol{\mathcal{A}} = a_2, \quad \mathbf{b} = b_1 V, \quad \mathbf{B} = b_2 V, \tag{2.84}$$

where we have used

$$\begin{aligned} a_1 &= \sum_{\mathcal{R}_1} \zeta_k c_k, \\ a_2 &= \sum_{\mathcal{R}_1} \zeta_k^2 c_k, \\ b_1 &= \sum_{\mathcal{R}_0} \zeta_k \gamma_k, \\ b_2 &= \sum_{\mathcal{R}_0} \zeta_k^2 \gamma_k. \end{aligned}$$

We therefore find the following expressions for the mean

$$\mathbb{E}\left[X_n\right] = \begin{cases} (1 + a_1\tau)^n \left(X_0 + V b_1/a_1\right) - V b_1/a_1, & a_1 \neq 0, \\ X_0 + b_1 V n \tau, & a_1 = 0. \end{cases} \tag{2.85}$$

Note that from this expression for the mean we recover the asymptotic stability result from [182], i.e. we need $|1 + a_1\tau| < 1$ for asymptotic stability when $a_1 \neq 0$. For the variance the expression becomes slightly more cumbersome but is given by

$$\begin{aligned} \mathrm{Var}\left[X_n\right] = X_0 \frac{a_2}{a_1} (1 + a_1\tau)^{n-1} \left(-1 + (1 + a_1\tau)^n\right) \\ + V \frac{a_2 b_1}{a_1^2} \left(\frac{\left(1 - (1 + a_1\tau)^n\right)\left(1 + a_1\tau - (1 + a_1\tau)^n\right)}{(1 + a_1\tau)\left(2 + a_1\tau\right)} \right) \\ + V \frac{b_2}{a_1} \left(\frac{(1 + a_1\tau)^{2n} - 1}{2 + a_1\tau} \right), \end{aligned} \tag{2.86}$$

in the case where $a_1 \neq 0$. If we have $a_1 = 0$ this simplifies to

$$\mathrm{Var}\left[X_n\right] = X_0 a_2 n\tau + \frac{1}{2} b_1 V a_2 n(n-1)\tau^2 + b_2 V n \tau. \tag{2.87}$$

Example (Isomerisation)**.** The second example often used as a simple test system contains two species S_1 and S_2 and describes the isomerisation reactions between the two,

$$S_1 \xrightarrow{c_1} S_2,$$
$$S_2 \xrightarrow{c_2} S_1.$$

Note that, due to conservation of mass, the dynamics of this system can be effectively described by a one-dimensional system. To proceed we define $c = c_1 + c_2$ and $r = c_1/c$ and note that

$$\mathbf{A} = c \begin{pmatrix} -r & 1-r \\ r & -(1-r) \end{pmatrix}, \quad \boldsymbol{\mathcal{A}} = c \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} (r, 1-r), \quad \mathbf{b} = \mathbf{0}, \quad \mathbf{B} = \mathbf{0}. \tag{2.88}$$

This yields the following expression describing the mean evolution of the τ-leap solution

$$\mathbb{E}[\mathbf{X}_n] = \left(\begin{pmatrix} 1-r & 1-r \\ r & r \end{pmatrix} + \begin{pmatrix} r & -(1-r) \\ -r & 1-r \end{pmatrix} (1-c\tau)^n \right) \mathbf{X}_0. \tag{2.89}$$

Note that conservation of mass is also valid for the mean, i.e. $(1,1)^\intercal \cdot \mathbb{E}[\mathbf{X}_n] = (1,1)^\intercal \cdot \mathbf{X}_0$. Asymptotic stability of the τ-leap method requires $|1 - c\tau| < 1$ in line with [182]. For the covariance we find

$$\mathrm{Cov}[\mathbf{X}_n, \mathbf{X}_n] = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \sigma_n^2, \tag{2.90}$$

where we have the scalar variance given by

$$\sigma_n^2 = \frac{1-(1-c\tau)^n}{(2-c\tau)}\left[2r(1-r)\begin{pmatrix}1\\1\end{pmatrix} + (1-c\tau)^{n-1}\begin{pmatrix}r(2r-c\tau)\\(1-r)(2(1-r)-c\tau)\end{pmatrix}\right]\cdot \mathbf{X}_0. \tag{2.91}$$

Note that the covariance structure clearly shows the effectively one-dimensional nature of the isomerisation system.

2.B Histograms and self-distance

A commonly used alternative to the EDF from Section 2.3 when estimating distributions is the histogram. To construct a histogram we pick a pre-selected number of bins B and divide the state space of interest Ω into B bins ω_b so that $\Omega \supseteq \cup_{b=1}^{B}\omega_b$ and $\omega_i \cap \omega_j = \emptyset$ when $i \neq j$. Note that, unlike for the EDF, for a histogram we do not necessarily need to consider the whole state space Ω. For the one-dimensional problems often encountered in the context of chemical reaction networks, the usual bins are intervals on the positive half-line. Given N sample paths we can then define histogram function h_B by

$$h_B(\omega_b|\mathbf{Y}) = \frac{1}{|\omega_b|N}\sum_{n=1}^{N}\mathbb{1}_{\mathbf{Y}\in\omega_b}\left(\mathbf{Y}^{(n)}\right) \approx \frac{p(\omega_b)}{|\omega_b|}. \tag{2.92}$$

Note that the histogram function approximates the average probability density inside the bins ω_b. If we keep all bin sizes of the same order of magnitude we observe that as $B \to \infty$ the size of the bins tends to zero and the histogram function converges to the EDF.

To quantify the convergence of simulations to a ground truth on a distribution level we introduced the concept of a statistical distance in Section 2.3. Next we aim to define the distance between two sets of samples $\{\mathbf{Y}^{(n)}\}_{n=1}^{N}$ and $\{\mathbf{X}^{(m)}\}_{m=1}^{M}$ of not

necessarily the same sizes N and M, respectively. Note that a typical example in the context of this thesis would be the comparison of a histogram based on samples $\mathbf{Y}^{(n)}$ from an approximate SSA, such as the τ-leap method, with a histogram based on samples $\mathbf{X}^{(m)}$ coming from an exact SSA like the DM.

Similar to the MISE for the EDF, we define the MISE for histograms by

$$\text{MISE}\left[h_B(\,\cdot\,|\mathbf{Y}) \parallel h_B(\,\cdot\,|\mathbf{X})\right] = \sum_{b=1}^{B} |\omega_b|^2 \mathbb{E}\left[\left(h_B(\omega_b|\mathbf{Y}) - h_B(\omega_b|\mathbf{X})\right)^2\right]. \tag{2.93}$$

For histograms one can also define a new statistical distance similar to the total variation distance for the EDF, the histogram distance. This was introduced in [36] and is given by

$$D_{h_B}\left[h_B(\,\cdot\,|\mathbf{Y}) \parallel h_B(\,\cdot\,|\mathbf{X})\right] = \sum_{b=1}^{B} |\omega_b| \left|h_B(\omega_b|\mathbf{Y}) - h_B(\omega_b|\mathbf{X})\right|. \tag{2.94}$$

The dependency of D_{h_B} and the MISE on the number of samples N, M and the number of bins B is subtle. Taking $B = 1$ means all observations fall in the same bin and so the distance between the two sets will be zero. However, we will not be able to deduce anything from that trivial observation. Taking B larger will increase our ability to discriminate between the two sets of samples, but the accuracy of a comparison with B large will be severely hampered by the large number of samples that will be required due to the growing statistical error per bin as the bins become smaller in size.

As a result of the above property it is generally not true that for a finite number of samples and finite bin size the histogram distance between histograms generated by an exact SSA and a τ-leap method vanishes, even if we take $\tau \to 0$. Note that a similar observation holds when considering summary statistics like the mean. We can show convergence of the τ-leap mean to that of the exact SSA in expectation when $\tau \to 0$, but when considering the MSE we see that unless the number of samples $N \to \infty$

the MSE does not go to zero when $\tau \to 0$ due to the statistical error. This behaviour is reminiscent of the round-off error, e.g. for floating point arithmetic, in classical numerical analysis which prevents convergence to an answer in exact arithmetic.

We will now discuss the expected MC behaviour of these two often used statistical distances in the context of discrete state space models.

Self-distance

We start with two independent sets of samples $\{\mathbf{Y}^{(n)}\}_{n=1}^{N}$ and $\{\mathbf{X}^{(m)}\}_{m=1}^{M}$ of not necessarily the same sizes N and M. We will derive estimates for the self-distance, which describes the statistical distance between the two sets when the samples follow the same distribution. We denote the estimated (empirical) distributions and histograms for these sets by $\hat{p}_{\mathbf{X}}, \hat{p}_{\mathbf{Y}}$ and $h_{B,\mathbf{X}}, h_{B,\mathbf{Y}}$, respectively.

To consider the distance between sets not following the same distribution, e.g. when comparing the τ-leap method and exact SSA samples, we get the self-distance plus a (positive) bias contribution. The self-distance therefore forms a lower-limit of what one can expect the size of the statistical distance between two sets of a fixed number of samples N and M to be given. Sets with statistical distance less than the self-distance should be treated with caution and are likely to be due to random effects, similar to the case of answers below machine precision in classical numerical computations.

L_2-norm squared for histogram. We can use the observations from Example 2.3 and the fact that there is no bias between $\mathbf{X}$ and $\mathbf{Y}$ samples, because they follow from the same distribution, to find

$$\mathrm{MISE}\left[h_{B,\mathbf{X}} \parallel h_{B,\mathbf{Y}}\right] = \left(\frac{1}{N} + \frac{1}{M}\right)\left(1 - \sum_{b=1}^{B} p(\omega_b)^2\right) \leq \frac{1}{N} + \frac{1}{M}, \tag{2.95}$$

where p is the distribution for the $\mathbf{X}$ and $\mathbf{Y}$ samples. This shows the familiar MC error convergence behaviour, now $\mathcal{O}(N^{-1})$ because we are considering the squared error, and the upper bound is both independent of the underlying distribution for $\mathbf{X}$ and $\mathbf{Y}$ and the dimension of the underlying distribution (which is not the case for the histogram distance).

L_2-norm squared for EDF. Similar to the histogram case we find

$$\mathrm{MISE}\left[\hat{p}_{\mathbf{X}} \parallel \hat{p}_{\mathbf{Y}}\right] = \left(\frac{1}{N} + \frac{1}{M}\right)\left(1 - \sum_{y\in\Omega} p(y)^2\right) \leq \frac{1}{N} + \frac{1}{M}. \tag{2.96}$$

Again this shows the upper bound is both independent of the underlying distribution for $\mathbf{X}$ and $\mathbf{Y}$ and the dimension of the underlying distribution (which is not the case for the histogram distance).

L_1-norm for histogram. We note that this was covered in [38, Theorem 4.1] and we quote here the resulting upper-bound on the expected histogram distance

$$\mathbb{E}\left[D_{h_B}\left[h_{B,\mathbf{X}} \parallel h_{B,\mathbf{Y}}\right]\right] \lessapprox \sqrt{\frac{2B}{\pi}}\sqrt{\frac{1}{N} + \frac{1}{M}}, \tag{2.97}$$

where the approximation comes from using a large N, M sample size approximation. This bound is independent of the underlying distribution for $\mathbf{X}$ and $\mathbf{Y}$ and was shown to be in many cases a good estimate of the actual histogram distance in [38]. Note that this bound clearly shows the effect of increasing B, which intuitively should increase the error by enlarging statistical errors. In addition we see the familiar inverse square root error convergence behaviour, i.e. $\mathcal{O}(N^{-1/2})$, for standard MC methods. Finally we note that the estimate for the histogram self-distance has an implicit dependency on the dimension of the distribution via B, because for higher dimensional problems B should grow exponentially with the dimension to keep the same level of resolution.

L_1-norm for EDF. If instead of the histograms $h_{B,\mathbf{X}}, h_{B,\mathbf{Y}}$ we consider the EDFs $\hat{p}_{\mathbf{X}}, \hat{p}_{\mathbf{Y}}$ we could be tempted to simply use the fact that for $B \to \infty$ suitably defined histograms converge toward the EDF. Due to the dependency of the histogram distance on $\sqrt{B}$, however, this does not directly yield a useful finite bound showing the rate of convergence. We therefore adapt the proof laid out in [38] to derive an upper bound for the self-distance for the EDF for one-dimensional data (generalisation to higher-dimensional data is possible but cumbersome and therefore omitted). If we let the true distribution of X and Y be p then we note that for the EDF we have the distributions $M\hat{p}_X(n) \sim \mathrm{Bin}(M, p(n))$ and $N\hat{p}_Y(n) \sim \mathrm{Bin}(N, p(n))$. This yields the results

$$\begin{aligned}\mathbb{E}\left[\hat{p}_X(n) - \hat{p}_Y(n)\right] &= 0,\\ \mathrm{Var}\left[\hat{p}_X(n) - \hat{p}_Y(n)\right] &= p(n)(1-p(n))\left(\frac{1}{N} + \frac{1}{M}\right).\end{aligned}$$

For large enough $N, M \gg 1$ we can employ a normal approximation for the pointwise self distance via

$$(\hat{p}_{\mathbf{X}}(n) - \hat{p}_{\mathbf{Y}}(n)) \sim \mathcal{N}\left(0, p(n)(1-p(n))\left(\frac{1}{N} + \frac{1}{M}\right)\right). \tag{2.98}$$

Using the (pointwise) normal approximation and the expectation of its absolute value we then find that we can write the expected total variation distance as

$$\mathbb{E}\left[\delta_{\mathrm{TV}}\left[\hat{p}_X \parallel \hat{p}_Y\right]\right] \approx \frac{1}{2}\sqrt{\frac{2}{\pi}\left(\frac{1}{N} + \frac{1}{M}\right)}\sum_n \sqrt{p(n)(1-p(n))}. \tag{2.99}$$

Note that unlike in [38] the summation in equation (2.99) does not necessarily need to be finite. Existence of any moment of order higher than one, however, is enough

as we will now show:

$$\sum_n \sqrt{p(n)(1-p(n))} = \sqrt{p(0)(1-p(0))} + \sum_{n\neq 0} \sqrt{p(n)(1-p(n)} \tag{2.100}$$

$$\leq \frac{1}{2} + \sum_{n\neq 0} \sqrt{p(n)} \tag{2.101}$$

$$\leq \frac{1}{2} + \sqrt{\left(\sum_{n\neq 0} \frac{1}{|n|^\alpha}\right)\left(\sum_{n\neq 0} |n|^\alpha p(n)\right)} \tag{2.102}$$

$$\leq \frac{1}{2} + \sqrt{2\zeta(\alpha)\mathbb{E}\left[|Y|^\alpha\right]}, \tag{2.103}$$

where the second inequality follows from the Cauchy-Schwarz inequality and $\zeta(\alpha)$ is the Riemann zeta function evaluated at α. Note that in order to get a finite bound we need $\alpha > 1$ and therefore impose a regularity condition on the underlying distribution, namely that more than just the first moment of the distribution should be finite.

A counterexample testing the limits of this new bound is the discrete distribution law described by $p(n) = 0$ when $n < 2$ and $p(n) \propto 1/(n^2 \log^2 n)$ for $n \geq 2$. Note that all moments for this distribution apart from the first are infinite, and the above bound is thus not applicable. In addition, for this law we see that $\sum_n \sqrt{p(n)} \to \infty$ and thus by the limit comparison test $\sum_n \sqrt{p(n)(1-p(n))} \to \infty$. As a result the method described above cannot yield a finite estimate for the total variation self-distance.

In conclusion, we have derived an upper bound on the expected total variation distance for EDFs of the form

$$\mathbb{E}\left[\delta_{\mathrm{TV}}\left[\hat{p}_X \parallel \hat{p}_Y\right]\right] \lessapprox \frac{1}{2}\sqrt{\frac{2}{\pi}}\left(\frac{1}{2} + \sqrt{2\zeta(\alpha)\mathbb{E}\left[|Y|^\alpha\right]}\right)\sqrt{\frac{1}{N}+\frac{1}{M}}, \quad \alpha > 1. \tag{2.104}$$

We note that if we take $\alpha = 2$, i.e. the assumption of finite variance, this becomes

$$\mathbb{E}\left[\delta_{\mathrm{TV}}\left[\hat{p}_X \parallel \hat{p}_Y\right]\right] \lessapprox \frac{1}{2}\sqrt{\frac{2}{\pi}}\left(\frac{1}{2} + \frac{\pi}{\sqrt{3}}\sqrt{\mathbb{E}\left[Y^2\right]}\right)\sqrt{\frac{1}{N}+\frac{1}{M}}. \tag{2.105}$$

This result implies that when constructing the EDF via MC samples we get the

standard MC error convergence behaviour (under mild regularity conditions of the underlying distribution), even for discrete distributions with (semi-)infinite support. However, this new upper bound, unlike for the MISE and the histogram distance, does depend on the underlying distribution via its moments.

Finally we point out that this new bound is not tight in many practical examples and serves mainly as a theoretical justification for the expected convergence rate. In practice the histogram self-distance, where for example we take B to be equal to the number of unique data points[10], provides a better pragmatic estimate for the total variation self-distance.

2.C Probability generating functions for test systems

The availability of exact solutions to the CME is crucial when testing new methods or implementations. Here we therefore provide explicit expressions for the transient distribution of molecules in some simple test systems. These results are used in this thesis to test exact SSAs, just like the results in Appendix 2.A form a benchmark for τ-leap method testing.

We state the results in terms of the probability generating function (PGF), which is defined for a system of J species by

$$G(z_1, \ldots, z_J, t) = \sum_{n_1,\ldots,n_J} \left(\prod_{j=1}^{J} z_j^{n_j} \right) P(n_1, \ldots, n_J, t), \tag{2.106}$$

where $P(n_1, \ldots, n_J, t)$ is the time-dependent solution to the CME describing the probability to be in a state $\mathbf{X}(t) = (n_1, \ldots, n_J)$. We can use the PGF to get explicit expressions for the moments of the underlying distribution by considering its series expansion around $z_i = 1$, and to find the distribution values we consider the series expansion around $z_i = 0$. Though the analytic series expansion to derive the distri-

[10] In practice the number of unique data points in finite-precision arithmetic is necessarily finite, which means that in practice B is bounded above.

bution from the PGF can be cumbersome, the PGF can be used to quickly evaluate the probability distribution numerically using the discrete fast Fourier transform, see for example [184].

Example 2.10 (Isomerisation system)**.** In Appendix 2.A we encountered the simple test system describing the isomerisation between two species S_1 and S_2, i.e.

$$\begin{aligned} S_1 &\xrightarrow{c_1} S_2, \\ S_2 &\xrightarrow{c_2} S_1. \end{aligned}$$

The solution to the CME is known to be the convolution between two multinomial distributions, see for example [104, Example 5.2]. For simplicity we provide here the PGF $G(z_1, z_2, t)$ for the transient probability distribution, assuming deterministic initial conditions $S_1(0)$ and $S_2(0)$

$$\begin{aligned} G(z_1, z_2, t) = &\left(z_1 - (z_1 - z_2)\left(1 - e^{-(c_1+c_2)t}\right)\left(\frac{c_1}{c_2 + c_1}\right)\right)^{S_1(0)} \\ &\times \left(z_2 + (z_1 - z_2)\left(1 - e^{-(c_1+c_2)t}\right)\left(\frac{c_2}{c_2 + c_1}\right)\right)^{S_2(0)}. \end{aligned} \tag{2.107}$$

For a derivation we refer the reader to [65, Chapter 11]. Note that the marginal distributions for S_1 and S_2 can be found by considering $G(z_1, 1, t)$ and $G(1, z_2, t)$, respectively.

Example 2.11 (Linear birth-death with inflow)**.** The second example we discuss is another linear system, but now for a single species S_1 in a volume V

$$\begin{aligned} \emptyset &\xrightarrow{c_0} S_1, \\ S_1 &\xrightarrow{c_1} 2S_1, \\ S_1 &\xrightarrow{c_2} \emptyset. \end{aligned}$$

Note that this is Example 2.5, as we encountered earlier in Section 2.3. Despite many advances in the understanding of solutions for linear reaction systems the exact solution for the transient distribution of S_1 was, to the best of our knowledge, not reported in the literature until recently a new approach using path integrals [212] was shown to be successful in finding an explicit expression. Here, however, we show that this result can also be found using more elementary methods directly using generating functions. Incidentally this allows for easier interpretation of the final result in terms of well-known distributions and forms the basis for fast numerical evaluation of the distribution.

It can easily be shown that the PGF $G(z,t)$ for this system with deterministic initial conditions satisfies the following partial differential equation (PDE)

$$\frac{\partial G}{\partial t} = c_0 V(z-1)G - (z-1)(c_2 - c_1 z)\frac{\partial G}{\partial z}, \tag{2.108}$$

subject to the initial condition $G(z,0) = z^{S_1(0)}$. It is relatively straightforward to solve this PDE via the method of characteristics , i.e. we aim to find the solution to the following system of ODEs, also known as the characteristic equations,

$$\begin{aligned}
\frac{\mathrm{d}g(s)}{\mathrm{d}s} &= c_0 V(z(s)-1)g(s), && \text{subject to } g(0) = z_0^{S_1(0)},\\
\frac{\mathrm{d}z(s)}{\mathrm{d}s} &= (z(s)-1)(c_2 - c_1 z(s)), && \text{subject to } z(0) = z_0,\\
\frac{\mathrm{d}t(s)}{\mathrm{d}s} &= 1, && \text{subject to } t(0) = 0,
\end{aligned}$$

where we have defined $g(s) = G(z(s), t(s))$. We first solve for the characteristic curves $(t(s), z(s))$ by using that their respective ODEs are separable. The characteristic curves are given by

$$t(s) = s, \tag{2.109}$$

$$z(s) = \frac{c_1 z_0 e^{s(c_1-c_2)} - c_2 e^{s(c_1-c_2)} - c_2 z_0 + c_2}{c_1 z_0 e^{s(c_1-c_2)} - c_2 e^{s(c_1-c_2)} - c_1 z_0 + c_1}. \tag{2.110}$$

Upon substitution of $z(s)$ into the ODE for $g(s)$ we arrive at a separable first-order linear homogeneous ODE with solution

$$g(s) = \left(\frac{c_2 - c_1 \left(z_0 + (1 - z_0) e^{(c_2 - c_1)s} \right)}{c_2 - c_1} \right)^{\frac{c_0 V}{c_1}} z_0^{S_1(0)}. \tag{2.111}$$

Expressing s and z_0 in terms of t and z we find the final PGF solution to equation (2.108)

$$G(z,t) = \underbrace{\left(\frac{1}{1 + c_1 \frac{1 - e^{-(c_2 - c_1)t}}{c_2 - c_1} (1 - z)} \right)^{\frac{c_0 V}{c_1}}}_{\text{negative binomial}} \underbrace{\left(\frac{z + c_2 \frac{1 - e^{-(c_2 - c_1)t}}{c_2 - c_1} (1 - z)}{1 + c_1 \frac{1 - e^{-(c_2 - c_1)t}}{c_2 - c_1} (1 - z)} \right)^{S_1(0)}}_{\text{initial condition}}. \tag{2.112}$$

We note that equation (2.112) is the exact PGF for the distribution described by equation (134) in [212, Section 9], though our result can be interpreted in terms of common distributions more easily. To interpret equation (2.112) we note that we can split the PGF into two independent contributions, from the molecules initially present at $t = 0$ and molecules created when $t > 0$, respectively. The latter contribution can be described by a negative binomial distribution, $\mathrm{NB}(c_0 V / c_1, (c_2 - c_1)/(c_2 - c_1 \exp(-(c_2 - c_1)t)))$. We therefore find that the long-time limit distribution, valid in the limit $t \to \infty$ where the contribution of the initial conditions vanishes, is given by a negative binomial distribution. A similar result for the stationary distribution can be found in [200], but it should be noted that this result is only valid for $c_2 > c_1$. Alternatively, when the system initially contains zero S_1 molecules, the distribution is again a negative binomial distribution. Finally we discuss simplifications of the system which yield results found in the literature.

For $c_1 = 0$, i.e. no autocatalytic reactions, we find that the initial condition contribution becomes a binomial distribution with parameters $S_1(0)$ (number of trials) and $\exp(-c_2 t)$ (success probability per trial). The negative binomial contribution turns into a Poisson distribution with rate parameter $c_0 V (1 - \exp(-c_2 t))/c_2$. As a

result the distribution can be characterised by the sum of a binomial and a Poisson random variable in this case, see for example [104, Example 5.1].

For $c_2 = 0$, i.e. no decay reactions, we find that the initial condition contribution becomes a shifted negative binomial distribution, i.e. the sum of $S_1(0)$ and a negative binomial random variable, $\mathrm{NB}(S_1(0), \exp(c_1 t) - 1)$. This was also observed in [104, Proposition 7]. We therefore find that the overall distribution is given by a shifted negative binomial distribution, i.e. the sum of $S_1(0)$ and a $\mathrm{NB}(S_1(0)+c_0 V/c_1, \exp(c_1 t)-1)$ random variable.

Chapter 3

Uniformisation methods

In this chapter we discuss the use of the uniformisation technique in the context of chemical reaction network models. This technique has appeared in various forms before in this context [11, 95, 214, 223], but here we provide extra insight and applications of the method. Firstly we provide a new way to simulate a uniformised system efficiently. We also show how uniformisation can be used as a variance reduction technique if combined with stratification. Finally we show how uniformisation can be used for the transient analysis of Markov chains and how this ties in with the estimation of time-dependent summary statistics.

Comment on originality This chapter is (partially) reproduced with the permission of AIP Publishing from the following publication:

Beentjes, C. H. L. & Baker, R. E. Uniformization techniques for stochastic simulation of chemical reaction networks. *The Journal of Chemical Physics* **150,** 154107 (Apr. 2019).

3.1 Uniformisation

The uniformisation method is a well-known method in probability theory, and it can be used to convert a CTMC into a DTMC. First introduced in 1953 [105] it was

popularised in [88] as an efficient tool to compute transient solutions to CTMCs via matrix multiplications. As a more general computational tool for CTMCs it seems to trace back to a 1990 paper by Fox and Glynn [64] and some ideas in this chapter parallel that paper. In the context of simulating jump-diffusion models the method is more widely known as "thinning of Poisson random measures", e.g. [85]. For a recent review of uniformisation from a probability perspective we refer to [51]. Here, however, we describe how to employ the uniformisation technique in the context of the simulation of exact sample paths according to the RTCR, which has previously appeared, albeit in slightly different forms and with different motivation, in [11, 95, 192, 206, 214, 223].

For simplicity we make the assumption that the reaction propensities, a_k, have no explicit time-dependence and only depend on time through the state vector $\mathbf{X}(t)$. This assumption restricts the class of problems to time-homogeneous Markov chains and thus excludes, for example, time-dependent reaction rates. It is, however, possible to extend most of the results in this work by relaxing this condition, as we show in an appendix to this chapter, Appendix 3.B.

We start with a generic reaction system containing K reaction channels, as given in Section 2.1 in equation (2.1), and consider an extension of this system by adding a new reaction channel R_{K+1} which takes the trivial form

$$\emptyset \to \emptyset, \tag{3.1}$$

and we denote this as a virtual reaction. It should be clear that the addition of this virtual reaction does not change the dynamics of $\mathbf{X}$ because the new channel has none of the species as reactant or product. We note that, as a result, the statistics from this new extended system are equal to that of the original system and therefore sample paths for the extended system are exact realisations of the original RTCR (2.7). This observation is independent of the reaction propensity $a_{K+1}(\mathbf{X}(t))$ that we

choose for the virtual reaction. Given some $\bar{a} > 0$ we therefore use this liberty to set

$$a_{K+1}(\mathbf{X}(t)) = \bar{a} - a_0(\mathbf{X}(t)) = \bar{a} - \sum_{k=1}^{K} a_k(\mathbf{X}(t)). \tag{3.2}$$

In order for (3.2) to constitute a well-defined reaction propensity we need $\bar{a} \geq a_0(t)$ for all $t \in [0, T)$, the time interval of interest. We will come back to discuss this assumption later in this section.

The choice (3.2) for the reaction propensity of the reaction channel R_{K+1} might seem peculiar at first, but we note that in the new extended system the total propensity of a reaction happening is given by $a_0 + a_{K+1} = \bar{a}$, which is therefore independent of the particular state $\mathbf{X}(t)$ the system is in. This is in contrast to the original system where the total reaction propensity, a_0, is generally state dependent. We therefore have a uniform total reaction propensity $\bar{a}$ which we will call the uniformisation rate, and we will denote the extended system consisting of reactions $R_1, \ldots, R_K, R_{K+1}$ from now on as the uniformised system. This uniformisation of the system has a few implications which we will discuss next.

Firstly, in the uniformised system (trivially) at least as many reactions fire (on average) as in the original system, because of the addition of the independent virtual reaction channel, as illustrated in Figure 3.1. Naïve application of the DM to the uniformised system means we explicitly simulate every virtual reaction firing in addition to the "real" reactions present in the original system. As a result of the extra time taken to simulate the virtual reaction channel firing, such naïve application of the DM to the uniformised system yields slower run-times for exactly the same level of statistical accuracy compared to the DM for the original system.

Secondly, the reaction times in the uniformised system are i.i.d. distributed exponential random variables with parameter $\bar{a}$. As a simple consequence of this we observe that the number of reactions, M, firing in a time interval $[t, t + \tau)$ is Poisson distributed with parameter $\bar{a}\tau$, which we will denote as $M \sim \mathcal{P}(\bar{a}\tau)$. Note that this is

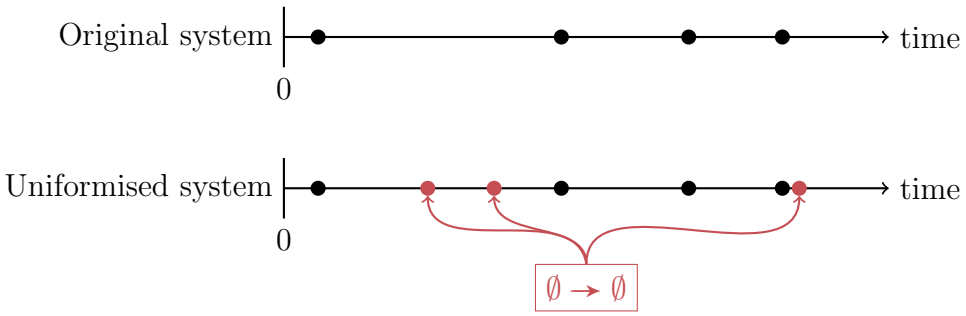


Figure 3.1: Illustration of reactions $R_1, \ldots, R_K$ firing in the original system and the uniformised/extended system, indicated by dots (•). We observe extra virtual reactions firing (•) in the uniformised system.

similar to the τ-leap method, where one makes the approximation that the reaction propensities of the original system stay constant within the time interval $[t, t+\tau)$. For the uniformised system, however, this expression for the number of firing reactions is instead exact and leads to a uniformised version of the DM, which we name the naïve uniformised direct method (NUDM), as shown in Algorithm 3.1.

Algorithm 3.1 Naïve uniformised direct method (NUDM).
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Uniformisation rate $\bar{a}$
Input: Final time T
$\mathbf{X} \leftarrow \mathbf{X}_0$
Generate $M \sim \mathcal{P}(\bar{a}T)$ ▷ Total number of reactions that fire in $[0, T)$.
$a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate real reaction propensities.
$a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total real reaction propensity.
for $m = 1, \ldots, M$ **do**
    Generate $u_1 \sim \mathcal{U}(0, 1)$
    if $\bar{a}u_1 > a_0$ **then** ▷ Check if a virtual reaction fires.
        continue ▷ If a virtual reaction fires skip to next iteration.
    end if
    Find p such that $\sum_{k=1}^{p-1} a_k < \bar{a}u_1 \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
    $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$ ▷ Update state vector.
    $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate real reaction propensities.
    $a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total real reaction propensity.
end for

Note that steps 7-9 in Algorithm 3.1 are an optimisation of the DM applied to the uniformised system which relies on the observation that firing a virtual reaction does not change the state vector and therefore also does not alter the reaction propensities. We therefore first check in a single comparison whether a virtual reaction fires so that we can potentially skip the current iteration. In the NUDM there is no explicit computation of the next reaction time (c.f. step 7 in Algorithm 2.1). For every reaction firing the NUDM only needs to generate a single uniform random variate to determine which reaction takes place and, therefore, involves fewer computations. As a result the computational complexity of the NUDM will be smaller than the DM applied to the uniformised system. It is possible to generate the reaction times, if required, by noting that we can condition on the fact that M reactions happen in $[0, T)$ with uniform rate. The reaction times are then distributed as the order statistics of M uniform random variables in $[0, T)$.

We observe, however, that in the current form the NUDM still suffers from two drawbacks. Firstly, the simulation will in general involve the firing of virtual reactions that do not contribute to the dynamics of the original system. This can therefore be thought of as computational waste and, as a result, the computational complexity of the NUDM in its current form will still be comparatively larger than that of the DM applied to the original system for time-homogeneous systems. Secondly, we mentioned that (3.2) needs to be a well-defined propensity function, i.e. $\bar{a} \geq a_0(t)$ needs to hold for all $t \in [0, T)$. It is not clear *a priori* whether such a uniformisation rate $\bar{a}$ exists[1] or what happens when a_0 becomes greater than $\bar{a}$ in the course of a simulation. These two issues are discussed next and we will show that the NUDM can be adapted to be at least as fast as the DM applied to the original system.

[1] In the case of a system with a bounded state space it is (theoretically) possible to find a uniformisation rate by taking the maximum of the total propensity over all allowed states. Note, however, that the size of the state space, albeit finite, could be prohibitively large for such an approach to be practically feasible.

3.2 Efficient simulation

3.2.1 Firing virtual reactions

In order to get around the issue of potentially slowing down the simulation by having to fire many virtual reactions we look at the distribution of the number of virtual reactions firing in between real reactions in the uniformised system.

Suppose the system is in a state with propensity a_0 for the real reactions to fire. If the system is uniformised with rate $\bar{a} > a_0$ this means that the probability that the next reaction firing belongs to one of the K real reaction channels is given by $a_0/\bar{a}$ and, equivalently, the probability a virtual reaction will fire next is $1 - a_0/\bar{a}$. Note that when a virtual reaction fires none of the propensities of the real reactions change because none of the copy numbers of the species S_i are changed in the reaction. The repeated firing of the virtual reaction channel before a real reaction fires can therefore be viewed as a series of Bernoulli trials with probability $a_0/\bar{a}$ of success (firing a real reaction) and $1 - a_0/\bar{a}$ of failure (firing a virtual reaction). In this scenario we are interested in the number of failures until the first success, i.e. the number of consecutive virtual reactions firing before a real reaction fires. This quantity has a well-known distribution, namely the geometric distribution, so that we have

$$\mathbb{P}\left(r \text{ consecutive virtual reactions before next real reaction fires}\right) = \left(1 - \frac{a_0}{\bar{a}}\right)^r \frac{a_0}{\bar{a}}. \tag{3.3}$$

As a result it is possible to fire all consecutive virtual reactions at once by sampling a single geometric random variable. This can be done efficiently for $a_0/\bar{a} < 1/3$ by generating a uniform random variable $u \sim \mathcal{U}(0,1)$ and calculating $\lfloor \ln(u)/\ln(1 - a_0/\bar{a}) \rfloor$, akin to the sampling of an exponential random variable. For $a_0/\bar{a} \geq 1/3$ we can use a direct search strategy [48, Chapter 10], which is similar to steps 6-9 in Algorithm 3.1. As a result we can improve upon steps 6-9 of Algorithm 3.1

whenever the (expected) number of consecutive reactions firing is approximately 3 or more. This observation leads to an improved version of the NUDM, as depicted in Algorithm 3.2. Note that this version is always at least as efficient as the NUDM if we use a geometric random variable generator as described previously.

Algorithm 3.2 (Improved) uniformised direct method (UDM).
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Uniformisation rate $\bar{a}$
Input: Final time T

$\mathbf{X} \leftarrow \mathbf{X}_0$
Generate $M \sim \mathcal{P}(\bar{a}T)$ ▷ Total number of reactions that fire in $[0, T)$.
$m \leftarrow 0$ ▷ Counter for number of reactions that have fired.
while $m < M$ **do**
    Generate $u_1 \sim \mathcal{U}(0, 1)$
    $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate real reaction propensities.
    $a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total real reaction propensity.
    Generate $m_{\text{virtual}} \sim \text{Geom}(a_0/\bar{a})$ ▷ Number of virtual reactions firing consecutively.
    $m \leftarrow m + m_{\text{virtual}}$
    if $m \geq M$ **then**
        $m \leftarrow M$
        break
    end if
    Find p such that $\sum_{k=1}^{p-1} a_k < a_0 u_1 \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
    $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$
    $m \leftarrow m + 1$
end while

In this form the uniformised direct method (UDM) is comparable with the DM applied to the original system, i.e. the gold standard in the field of simulation chemical reaction networks. Both methods now fire an equal number of reactions, and need two random numbers per reaction firing. In addition to this we note that many improvements that have been made to the DM, related to speed-ups in the choice of the next reaction firing and the update of the propensities such as in [37, 68, 138, 148, 198, 208] and/or the re-use of random numbers [221], can be equally well applied to the improved UDM. Although it is not possible to use the τ-leap approach in combination

with the (improved) UDM, it is trivial to apply the R-leap approximation [13] to the (naïve) UDM. The complexity of the improved version of the UDM is also insensitive to the uniformisation rate, $\bar{a}$, as can be seen in Figure 3.2. This can be intuitively understood from the observation that increasing or decreasing $\bar{a}$ only changes the number of consecutive virtual reactions firing which is taken care of in a single step in the UDM.

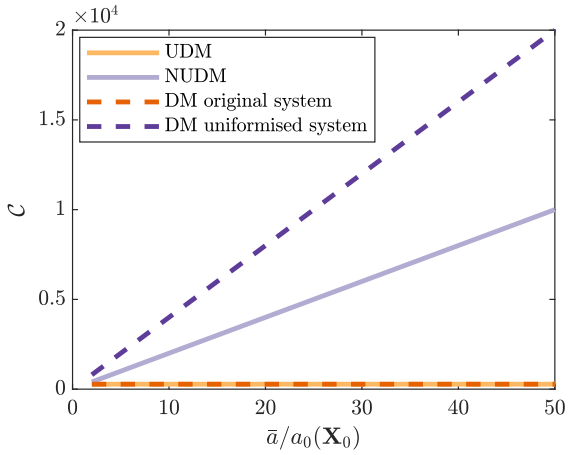


Figure 3.2: Computational complexity in terms of (expected) number of random variates needed to generate a sample path with the NUDM and improved UDM compared with the DM for different uniformisation rates $\bar{a}$ relative to the initial total propensity a_0. Results are for the isomerisation system, Example 3.3.2, with $c_1 = 0.2, c_2 = 0.1$ and $\mathbf{X}_0 = (20, 0)^\intercal$, run until $T = 50$.

We note from Figure 3.2 and the description of the UDM that the computational complexity of simulations of the uniformised system can be brought back to the that of the DM applied to the original system. However, there is no intrinsic reason to expect the UDM to be faster than the DM for the original system[2] and this is also observed in Figure 3.2. A computational speed-up, as such, is therefore not a sufficient motivation to employ the uniformisation technique in the stochastic simulation of chemical reaction networks. However, as mentioned earlier, the usage of uniformisation allows one to consider new applications, such as variance reduction methods, that are not possible under the standard SSAs for the original system.

[2]This observation is mainly true for time-homogeneous systems, as considered in this section. For time-inhomogeneous systems, methods based on the DM for the original system can often compare unfavourably with uniformisation-based methods, as discussed in [214] and the appendix to this chapter, Appendix 3.B.

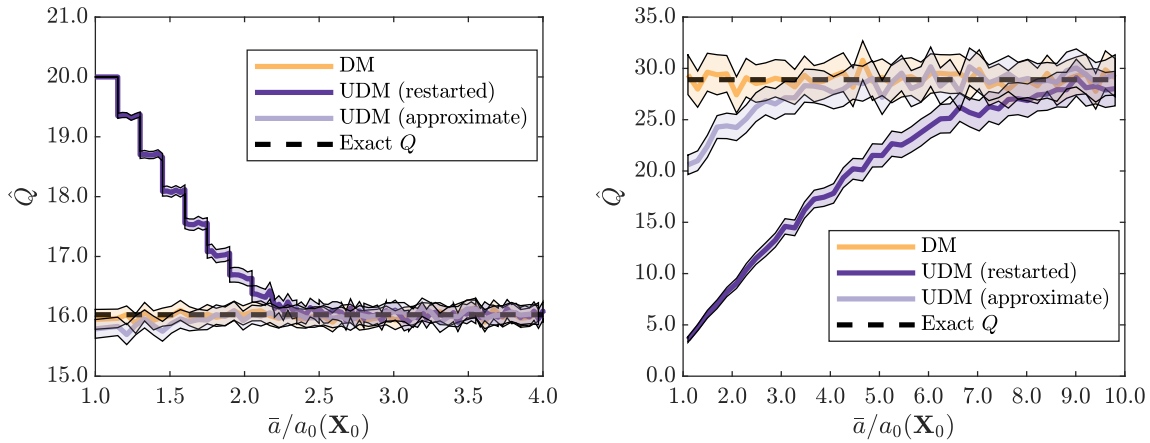
3.2.2 Adapting the uniformisation rate

In the exposition so far the uniformisation rate has been a free parameter and we mentioned the caveat that, by expression (3.2), $\bar{a} \geq a_0(t)$ should hold for all $t \in [0, T)$ in order for the system to remain a well-defined chemical reaction model. We therefore discuss here the ramifications of a simulation in which the total propensity $a_0(t)$ exceeds $\bar{a}$.

A first approach to the situation $a_0 > \bar{a}$ in the course of a simulation might be to classify such sample paths as invalid and generate a new, independent, sample path for each of the invalid paths until we end up with the desired number of sample paths that all satisfy $a_0 \leq \bar{a}$ across $[0, T)$. This approach, which we call restarted UDM, comes at a cost, because it introduces a bias due to the new sampling strategy. Effectively, this approach only samples from a subset of the distribution defined by the CME and it is clear from Figure 3.3 that the effect of the resulting bias can be dramatic and should be avoided if possible. A second alternative is approximate uniformisation, which was first described in [50], albeit not in a pathwise form. For the pathwise representation we consider in this chapter, approximate uniformisation in practice only changes step 7 in Algorithm 3.2; if $a_0 > \bar{a}$ the geometric distribution used in step 7 is ill-defined and therefore we take instead $m_{\text{virtual}} = 0$ in the approximate approach. Effectively this means that when $a_0 > \bar{a}$ we recover the standard DM (minus the reaction time selection) until $a_0 \leq \bar{a}$, at which point we return to the standard UDM. In this case the error due to approximation is solely due to the fact that the number of reactions in the system, M, that are sampled is too low. Again Figure 3.3 shows that the systematic error can be significant for a uniformisation rate, $\bar{a}$, that is too low. Despite the fact that approximate uniformisation is biased Figure 3.3 does show that it is certainly better than the restarted UDM approach we described earlier and in certain scenarios can be a viable method.

To circumvent systematic errors we therefore propose a way to keep samples for

which the uniformisation rate is breached, whilst keeping the estimator unbiased at the same time.



(a) Isomerisation system, see Example 2.10, with parameters $c_1 = 0.1$, $c_2 = 0.4$ and initial condition $\mathbf{X}_0 = (20, 0)^\intercal$.

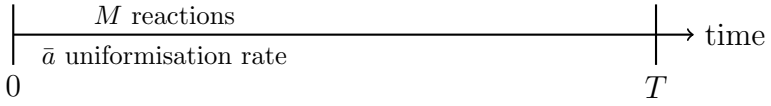
(b) Single species linear system, see Example 2.11, with parameters $c_0 V = 0.1$, $c_1 = 0.5$, $c_2 = 0.4$ and initial condition $\mathbf{X}_0 = 10$.

Figure 3.3: Bias effect due to invalid samples of the uniformised system for different uniformisation rates $\bar{a}$ relative to the initial total propensity a_0. The summary statistic in both cases is the number of S_1 molecules at final time $T = 10$ and estimators are shown with 99.5% confidence interval for $N = 10^3$ samples.

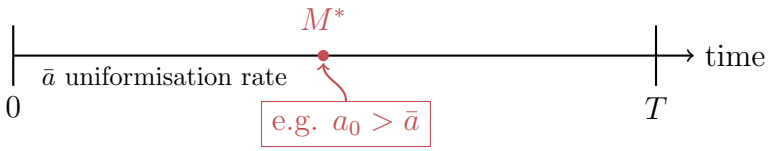
To do so, we start with a uniformised system with uniformisation rate $\bar{a}$ and sample the number of reactions $M \sim \mathcal{P}(\bar{a}T)$ firing in $[0, T)$. Now suppose that we observe that $a_0 > \bar{a}$ after $M^* \leq M$ reactions have fired. Rather than discarding this sample path completely we can ask at what time $T^* \in [0, T)$ reaction M^* took place, i.e. at what time did a_0 become larger than $\bar{a}$? As mentioned earlier, given that M reactions fire in $[0, T)$, the reaction times $t_1, \ldots, t_M$ are uniformly distributed on $[0, T)$. The reaction time $T^* = t_{M^*}$ of the M^*-th reaction out of M reactions therefore follows the distribution of the M^*-th order statistic of a collection of M uniform random variables on $[0, T)$. This is a well-known distribution and leads to $T^*/T \sim \text{Beta}(M^*, (M - M^*) + 1)$.

We can therefore sample T^* and use the Markov property to restart the simulation from the state after M^* reactions for the remaining time interval $[T^*, T)$ with a new uniformisation rate, $\bar{a}_{\text{new}}$. This procedure to describe T^* conditional on M reactions

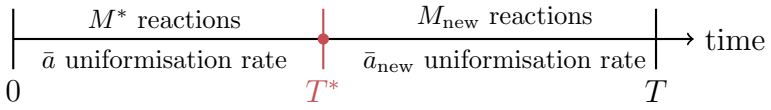
in $[0, T)$ leads to the correct unconditional distribution for T^* and therefore does not introduce a bias, see the appendix to this chapter, Appendix 3.A, for more detail. If we choose this new uniformisation rate such that $\bar{a}_{\text{new}} \geq a_0(T^*)$, the simulation for the remaining $t \in [T^*, T)$ can be done in exactly the same manner with the (improved) UDM, and this approach is illustrated in Figure 3.4. The sampling of the number of reactions remaining in $[T^*, T)$ can be done either by generating a new, independent, Poisson variable with rate $\bar{a}_{\text{new}}(T - T^*)$, relying on the memoryless property, or in a similar fashion to the post-leap check variant of the τ-leap method [2].



(a) With uniformisation rate $\bar{a}$ we sample M reactions to fire in $[0, T)$.



(b) After M^* reactions we want to adapt the uniformisation rate, for example because the uniformisation rate is breached, $a_0 > \bar{a}$.



(c) We sample the time T^* at which to adapt the uniformisation rate and restart the simulation at T^* with a new uniformisation rate $\bar{a}_{\text{new}}$ and $M_{\text{new}} \sim \mathcal{P}(\bar{a}_{\text{new}}(T - T^*))$.

Figure 3.4: Illustration of the procedure to adapt the uniformisation rate.

The approach of changing the uniformisation rate, $\bar{a}$, as described in Figure 3.4, can be applied throughout the simulation and this yields a method that is effectively a pathwise version of adaptive uniformisation [152]. We note that the adaptation of the uniformisation rate is not restricted to time points T^* at which $a_0(T^*) > \bar{a}$ is observed, adapting the uniformisation rate can be done at any point in the simulation, as shown in Algorithm 3.3, and yields unbiased sample paths. For example, one could have a system that initially has a high total propensity, a_0, requiring a high uniformisation

rate, $\bar{a}$, but after M^* reactions a_0 dips considerably. In this scenario one might want to adapt the uniformisation rate to make it smaller. Such an approach is similar to the adaptive fluctuation intervals as used in the rejection-based SSA algorithms [207, 208, 210]. An even more extreme choice might be to update the uniformisation rate after every reaction, which yields a procedure akin to the Extrande-method [214].

Algorithm 3.3 Adaptive (improved) UDM.
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Uniformisation rate $\bar{a}$
Input: Final time T

$\mathbf{X} \leftarrow \mathbf{X}_0$
Generate $M \sim \mathcal{P}(\bar{a}T)$ ▷ Total number of reactions that fire in $[0, T)$.
$m \leftarrow 0$ ▷ Counter for number of reactions that have fired.
while $m < M$ **do**
    Generate $u_1 \sim \mathcal{U}(0, 1)$
    $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate real reaction propensities.
    $a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total real reaction propensity.
    Set `adapt_flag` ▷ For example, when $a_0 > \bar{a}$ we set `adapt_flag=true`.
    if `adapt_flag` **then** ▷ Option to adapt the uniformisation rate.
        Generate $T^*/T \sim \text{Beta}(m, (M - m) + 1)$
        Adapt the uniformisation rate $\bar{a}$
        Generate $\bar{M} \sim \mathcal{P}(\bar{a}(T - T^*))$ ▷ Total number of reactions that fire in $[T^*, T)$.
        $M \leftarrow m + \bar{M}$
    end if
    Generate $m_{\text{virtual}} \sim \text{Geom}(a_0/\bar{a})$ ▷ Number of virtual reactions firing consecutively.
    $m \leftarrow m + m_{\text{virtual}}$
    if $m \geq M$ **then**
        $m \leftarrow M$
        break
    end if
    Find p such that $\sum_{k=1}^{p-1} a_k < a_0 u_1 \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
    $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$
    $m \leftarrow m + 1$
end while

We finally mention two observations regarding the adaptive uniformisation method. Firstly, the sample paths in this framework are again exact samples according to the

distribution described by the CME and can therefore be used to create unbiased summary statistics. Secondly, from the formulation of the improved UDM in the previous section we can see that the choice of $\bar{a}$ ultimately does not influence the run-time of the simulation. One might therefore be tempted to take $\bar{a}$ very large in the hope that $\bar{a} > a_0(t)$ for all $t \in [0, T)$ and we never have to adapt the uniformisation rate $\bar{a}$. However, for some choices of the reaction propensities, such as mass action kinetics, it is not always possible to select such an upper bound for a_0 as a uniformisation rate, $\bar{a}$, because the reaction propensities are unbounded. This is, for example, the case in a system where at least one of the species involved in mass action kinetics is unbounded itself, a situation that is often encountered. In [95] it was suggested to run a few pre-simulations to determine an empirical upper bound to a_0 on $[0, T)$ as a workaround if we do not know an analytical expression for an upper bound. Alternatively, one might have other motives to not choose the uniformisation rate much higher than a_0, an example of such a situation will be given in Section 3.3. To create unbiased estimators with the (improved) UDM it can therefore be necessary to employ an adaptive uniformisation method, as described in this section.

In conclusion we can create an (improved) UDM that is equally as versatile and applicable as the DM. The main benefit of uniformisation, however, lies in the fact that it opens the door to new applications that are not possible in the DM framework, as we will see next.

3.3 Variance reduction via stratification

Using the (improved) UDM will yield exact sample paths from the distribution defined by the CME, just like the DM. The resulting MSE of the standard MC estimates based on these sample paths will therefore behave exactly like the MSE for the DM. In this section, however, we will show that it is possible to create a new (unbiased) estimator that has a lower sample variance, and therefore MSE, by using a stratification strategy in combination with uniformisation.

3.3.1 Stratification of the number of reactions

Stratification, as discussed in Section 2.4.1, is a technique that can be used to incorporate some exact knowledge of part of the stochastic process into the required estimator so as to lower the sample variance. In the case of chemical reaction networks we can apply stratification ideas if we first uniformise the system with some rate $\bar{a}$. We can think of the uniformised system in terms of an extended state space of the form $\mathbf{Y} = (\mathbf{X}, M)$ with $\mathbf{X}$ the state of the species after M reactions have fired. The distribution of M is known *a priori* (as shown in Section 3.1) and therefore we can stratify with respect to the number of reactions, M, that fire in the time interval of interest, $[0, T]$. Because $M \in \mathbb{N}_{\geq 0}$ all we need to specify is a division of the non-negative integers into disjoint sets. For this work we consider the following stratification strategy: $\mathcal{D}_j = \{M_j \leq M < M_{j+1}\}$, for some collection of non-negative integers $M_1 < M_2 < \cdots < M_{J+1}$. This divides the state space of $\mathbf{Y} = (\mathbf{X}, M)$ by $\Omega = \cup_{j=1}^{J} \Omega_j$ with $\Omega_j = \{(\mathbf{X}, M) \text{ s.t. } M_j \leq M < M_{j+1}\}$, i.e. Ω_j represents all sample paths where the total number of reactions that have fired lies in the prescribed range $[M_j, M_{j+1})$. Note that for this choice the strata probabilities are given by a simple sum of Poisson probabilities

$$p_{\mathcal{D}_j}(t) = \sum_{m=M_j}^{M_{j+1}-1} \frac{(\bar{a}t)^m}{m!} e^{-\bar{a}t}. \tag{3.4}$$

To construct the conditional estimators $\hat{Q}_{\mathcal{D}_j}$ we need to be able to construct sample paths conditional on $\mathcal{D}_j$. This can be easily achieved by replacing step 1 of the (improved) UDM with drawing M from the truncated Poisson distribution on $[M_j, M_{j+1})$ rather than drawing $M \sim \mathcal{P}(\bar{a}T)$, as illustrated in Algorithm 3.4[3]. Therefore there

[3]Sampling from the truncated Poisson distribution can be easily carried out via the inverse transform sampling method combined with the truncated CDF, $F_{\text{truncated}}(x) = (F(x) - F(M_j))/(F(M_{j+1}) - F(M_j))$, where $F(x)$ is the standard Poisson CDF. In practice this means we use the Poisson inverse CDF on a transformed uniform random variable, see for example [162, Example 4.10].

is no extra cost associated with this stratification strategy compared to the original (improved) UDM and, as a result, the computational cost for the stratified estimator is comparable to that of the (improved) UDM and the DM. However, at the same time it is guaranteed to have a sample variance that is at least as small as that of the standard estimator from the DM and (improved) UDM. Before applying this new estimator to some examples we mention a few caveats and observations.

Algorithm 3.4 Stratification with the (improved) UDM.
This simulates a (stratified) ensemble of N sample paths.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Uniformisation rate $\bar{a}$
Input: Final time T
Input: Number of samples N
Input: Stratum definitions $M_1, \ldots, M_{J+1}$
for $j = 1, \ldots, J$ **do**
  $N_j \leftarrow \lceil \omega_{\mathcal{D}_j}(T) N \rceil$ ▷ Proportional allocation for number of samples per stratum.
  for $n = 1, \ldots, N_j$ **do**
    $\mathbf{X} \leftarrow \mathbf{X}_0$
    Generate M from the truncated distribution $\mathcal{P}(\bar{a}T)$ on $[M_j, M_{j+1})$
    $m \leftarrow 0$
    while $m < M$ **do**
      Generate $u_1 \sim \mathcal{U}(0, 1)$
      $a_k \leftarrow a_k(\mathbf{X})$ ▷ Calculate real reaction propensities.
      $a_0 \leftarrow \sum_k a_k$ ▷ Calculate the total real reaction propensity.
      Generate $m_{\text{virtual}} \sim \text{Geom}(a_0/\bar{a})$
      $m \leftarrow m + m_{\text{virtual}}$
      if $m \geq M$ **then**
        $m \leftarrow M$
        break
      end if
      Find p such that $\sum_{k=1}^{p-1} a_k < a_0 u_1 \leq \sum_{k=1}^{p} a_k$
      $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$
      $m \leftarrow m + 1$
    end while
  end for
end for

Firstly, in order for the stratification method to yield a correct unbiased estimate

it is important that the uniformisation rate $\bar{a}$ is in fact a valid uniformisation rate over the interval of interest $[0, T)$, i.e. $\bar{a} \geq a_0(t)$ for all $t \in [0, T)$. If this is not the case and we are required to adapt the uniformisation rate, the samples $(\mathbf{X}, M)$ are not all drawn from the same distribution and cannot be combined to yield the conditional estimators $\hat{Q}_{\mathcal{D}_j}$. Alternatively, we can choose to not adapt the uniformisation rate, but instead resort to approximate uniformisation, as described in Section 3.2.2. In this case we can continue to use the stratification method, at the cost of introducing a bias due to the use of the approximate uniformisation method, which we note is $\mathcal{O}(\bar{a}^{-1})$ [51, Section 7]. For accurate results it is therefore necessary to either know an upper bound to the total propensity *a priori* or to run some pre-simulations to generate an empirical (approximate) upper bound.

Secondly, the effectiveness of stratification will hinge on the choice of strata. If we use the proportional allocation strategy we see that the variance reduction benefit of stratification over standard MC comes from the $\frac{1}{N}\sum_{j=1}^{J} p_{\mathcal{D}_j}\left(Q - Q_{\mathcal{D}_j}\right)^2$ term that is lacking in the sample variance. This term represents the inter-strata variance, i.e. how much the conditional summary statistics deviate from the overall summary statistic. The larger this inter-strata variance, the bigger the variance reduction gain is. Unfortunately, it is in general not possible to know how to choose the strata so as to attain a relatively large inter-strata variance.

In choosing the strata we have to make a choice as to the number of strata. One might be tempted to take as many strata as possible as this will in theory yield the largest variance reduction. Note that theoretically we could use an infinite number of strata if we use the strata boundaries $M_j = (j-1)$ for all $j \in \mathbb{N}_{N\geq 0}$. However, in order to get accurate estimators for every stratum we need at least two samples per stratum, and ideally more. Therefore, increasing the number of strata will also increase the number of sample paths needed to get accurate estimates of, for example, the sample variance. It has been observed that increasing the number of strata beyond six, under some mild assumptions, yields very little extra benefit [43, Section 5A.8].

We will show that this is also true for the simulation of chemical reaction networks in an example in the next section.

Finally there is the choice of the strata boundaries. Without knowing the joint distribution of $(\mathbf{X}, M)$ we simply choose to use the quantiles of the distribution of M to define the strata. For example, if we require four strata we use the quartiles q_1, q_2, q_3 to define $M_1 = 0$, $M_2 = q_1$, $M_3 = q_2$, $M_4 = q_3$ and $M_5 = \infty$. Note that this choice roughly allocates an equal proportion of the total N samples to each stratum because the weights $\omega_{\mathcal{D}_j}$ are equal[4]. An alternative choice, reminiscent of work in [95], is to define M_1 and M_{J+1} such that $\mathbb{P}(M < M_1 \text{ or } M > M_{J+1}) \leq \varepsilon$. We can either use $M_j = M_1 + (j-1)$ or some other stratification for $M_1 \leq M \leq M_{J+1}$. This will yield a biased estimator, because the ε-tails of the distribution of M are neglected. However, if ε is sufficiently small this bias will be negligible by construction.

3.3.2 Examples

In this part of the work we consider two examples to test the variance reduction effects of using stratification in combination with uniformisation for chemical reaction networks. The first example is a linear bi-molecular system and therefore it is possible to show some analytical results to complement numerical results. The second example is a MAPK cascade model comprising eight species interacting via ten reactions with non-linear reaction propensities. We will use the following notation to denote the variance reduction factor (VRF)

$$\text{VRF} = \frac{\text{Var}\left[\hat{Q}\right]}{\text{Var}\left[\hat{Q}_{\text{str}}\right]}. \tag{3.5}$$

Note furthermore that the complexity of the stratified UDM and the DM are equal and that both methods are unbiased. As a result the relative efficiency, $\mathcal{E}$, of the stratified

[4]Due to the discrete nature of the Poisson distribution it is not possible to define strata that have exactly equal weights.

UDM compared to the DM, as defined in equation (2.37), is equal to the VRF. Additionally we can perform a similar exact analysis of the benefits of stratification when the summary statistic of interest is the distribution using Section 2.3.1.

Isomerisation

In this example we consider the simple reversible isomerisation of species S_1 into S_2:

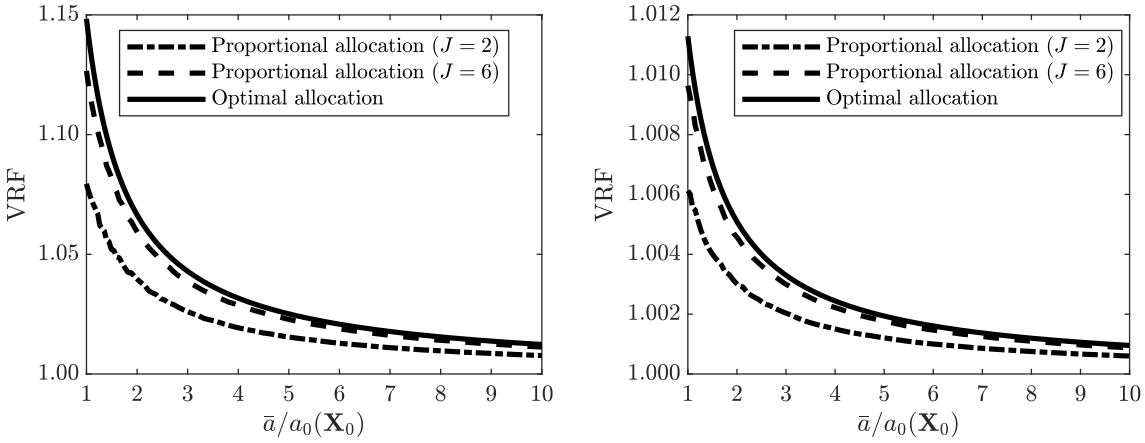
$$S_1 \xrightarrow{c_1} S_2, \tag{3.6a}$$

$$S_2 \xrightarrow{c_2} S_1. \tag{3.6b}$$

Due to the linear nature of the reactions we can write down an analytical expression for the probability distribution $\mathbb{P}(\mathbf{X}, t)$ [104]. We note that the state space of $\mathbf{X}$ is bounded and as a result there exists a uniformisation rate valid for all $T > 0$, e.g. $\bar{a} = \max(c_1, c_2) \cdot (S_1(0) + S_2(0))$. Furthermore, because it is a simple linear system, we can find the joint distribution of $(\mathbf{X}, M)$ numerically. This is achieved by writing down the transition matrix for the DTMC of the uniformised system. This allows one to numerically evaluate equation (2.49) and therefore give an analytic value for the variance reduction that can be attained.

We first look at the influence of the uniformisation rate, $\bar{a}$, on the VRF. Because the system is linear and effectively monomolecular we can calculate the VRF for both proportional allocation and optimal allocation. The latter method maximises the VRF, but for general systems is not a practical method. It does, however, provide an upper bound for the VRF for any stratification strategy and is therefore included here. As can be seen in Figure 3.5 the VRF is moderate and decreases monotonically as a function of $\bar{a}$. Furthermore, the VRF depends on the number of strata, J, used and, as expected, more strata entails a larger VRF. Note, however, that as expected the VRF is always larger than unity, meaning that regardless of $\bar{a}$ there is always a (slight) variance reduction. One might be tempted to take a lower value of $\bar{a}$ than

drawn in the hope of getting larger VRFs. Unfortunately, there is a balance between the uniformisation rate needing to be large enough so as to not bias the simulations and keeping $\bar{a}$ small enough to see a more significant VRF. The benefit from using stratification also depends on the summary statistic of interest as can be seen from comparing Figure 3.5(a) and Figure 3.5(b). When estimating moments, rather than a distribution, it appears that the improvement is significantly larger.



(a) Moment estimation (MSE). (b) Distribution estimation (MISE).

Figure 3.5: VRF of the stratified UDM compared to the DM, see equation (3.5), for different uniformisation rates $\bar{a}$ relative to the initial total propensity a_0. Isomerisation system (3.6) was considered with $c_1 = 0.3$, $c_2 = 0.1$ and $\mathbf{X}_0 = (20, 0)^\intercal$ until $T = 5$. The summary statistic is the average copy number of species S_1 at final time T for (a) and the distribution of S_1 copy numbers at final T for (b).

To see the effect of varying the number of strata we look at the same system that was used to generate Figure 3.5, but fix the uniformisation rate $\bar{a}$ whilst we vary the number of strata, J. The results are presented in Table 3.1 and clearly show that increasing the number of strata beyond a moderate number like $J = 6$ has only a marginal effect. We can also observe that there seems to be an inverse relationship between the uniformisation rate, $\bar{a}$, and the variance reduction factor of the form $\text{VRF} = 1 + A(J)/\bar{a}$ for some constant $A(J)$ depending on the number of strata. Similar observations hold when considering distribution estimation, rather than moment estimation.

Number of strata	Relative uniformisation rate $\bar{a}/a_0(\mathbf{X}_0)$		
J	1	10	100
2	$7.9 \cdot 10^{-2}$	$7.7 \cdot 10^{-3}$	$7.7 \cdot 10^{-4}$
4	$1.2 \cdot 10^{-1}$	$1.1 \cdot 10^{-2}$	$1.0 \cdot 10^{-3}$
6	$1.3 \cdot 10^{-1}$	$1.1 \cdot 10^{-2}$	$1.1 \cdot 10^{-3}$
12	$1.4 \cdot 10^{-1}$	$1.2 \cdot 10^{-2}$	$1.2 \cdot 10^{-3}$
∞ prop.	$1.5 \cdot 10^{-1}$	$1.2 \cdot 10^{-2}$	$1.2 \cdot 10^{-3}$
∞ opt.	$1.5 \cdot 10^{-1}$	$1.2 \cdot 10^{-2}$	$1.2 \cdot 10^{-3}$

Table 3.1: Effect of the number of strata, J, and the uniformisation rate, $\bar{a}$, on the VRF when estimating the mean number of molecules S_1 at final time T. Tabulated is VRF $- 1$ (higher is better) for the isomerisation model (3.6) with $c_1 = 0.3$, $c_2 = 0.1$ and $\mathbf{X}_0 = (20, 0)^\intercal$, run until $T = 5$. The last two rows, $J = \infty$, indicate the maximal VRF that can be achieved for both proportional allocation and optimal allocation, respectively.

We can understand the relatively small VRF that we observe based on the decomposition of the estimator variances (2.50) and (2.51). The difference between the two variances is due to the inter-strata variance and the VRF therefore increases with increasing inter-strata variance. However, by increasing the uniformisation rate, $\bar{a}$, the virtual reactions become more prevalent. This has the result that most of the reactions in a stratum become virtual reactions and therefore the strata distributions become more similar, diminishing the inter-strata variance.

Finally, we note that numerical verification of the VRF is computationally intensive, because the VRF is small. We used $N = 2^{14}$ sample paths from the uniformised system with uniformisation rate $\bar{a} = a_0(\mathbf{X}_0) = 6$ and $J = 6$ strata, parameters $c_1 = 0.3$, $c_2 = 0.1$ and $\mathbf{X}_0 = (20, 0)^\intercal$ at time $T = 5$ to find the estimator variances for the summary statistic $Q = \mathbb{E}[S_1(T)]$ using the standard MC sample variance estimator and (2.54). By repeating this procedure 256 times, yielding roughly $4 \cdot 10^6$ sample paths in total, we can construct confidence intervals for the variances of both estimators. This yields the 99.7% confidence intervals for the

standard MC method, $\mathrm{Var}\left[\hat{Q}\right] \in (4.549, 4.568)/N$, and for the stratified estimator, $\mathrm{Var}\left[\hat{Q}_{\mathrm{prop}}\right] \in (4.035, 4.052)/N$. Both of these results agree with the theoretical values of $\mathrm{Var}\left[\hat{Q}\right] = 4.559/N$ and $\mathrm{Var}\left[\hat{Q}_{\mathrm{prop}}\right] = 4.047/N$, respectively, and show that numerically there is a variance reduction visible, albeit a small one.

MAPK-cascade with feedback

The second example is a mitogen-activated protein kinase (MAPK) cascade model from [109]. It consists of eight species linked by ten reaction channels with Michaelis-Menten kinetics and Hill functions. A schematic representation of the network structure is shown in Figure 3.6.

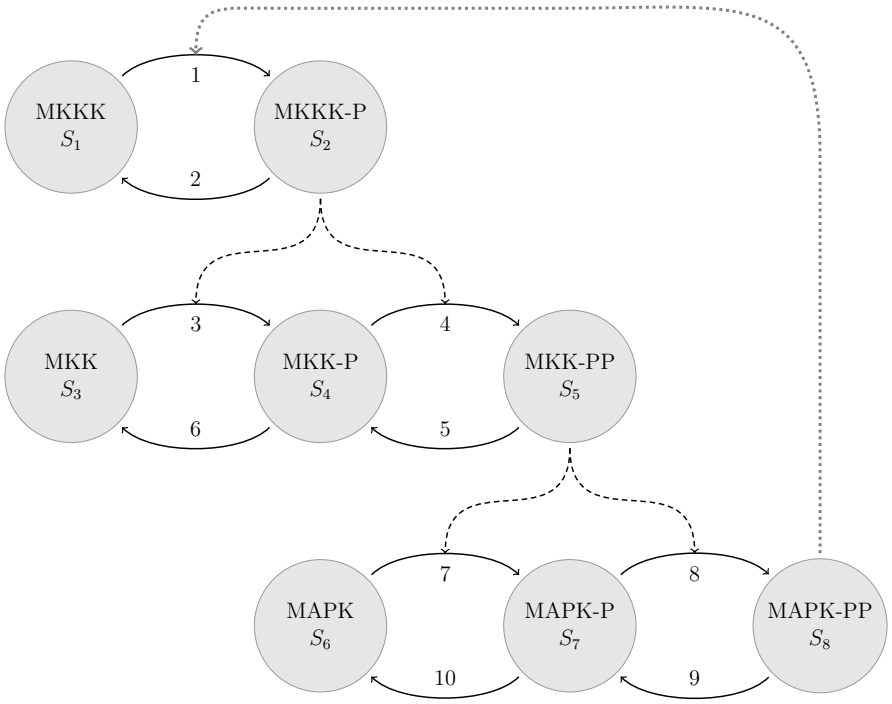


Figure 3.6: MAPK-cascade network, adapted from [109]. Reaction channels are depicted by the numbered horizontal arrows. Vertical dashed lines in black and the grey dotted line denote positive and negative feedback effects, respectively.

The MAPK-cascade network describes phosphorylation of MAPK by a layered process involving several kinases. MAPK, the terminal kinase, is phosphorylated by

MAPK kinases (MKKs), depicted by the interaction between the second and third layer in Figure 3.6. In turn the MKKs are phosphorylated by the MAPK kinase kinases (MKKKs), which is visible as the interaction between the first and second layer in Figure 3.6. The activation of MKKK is thought to be inhibited by MAPK via membrane-bound Ras and is modelled as a negative feedback loop, as can be seen in Figure 3.6.

For this system there exist three elementary conservation laws and every chemical species is involved in one of them:

$$S_1(t) + S_2(t) = S_1(0) + S_2(0); \tag{3.7a}$$

$$S_3(t) + S_4(t) + S_5(t) = S_3(0) + S_4(0) + S_5(0); \tag{3.7b}$$

$$S_6(t) + S_7(t) + S_8(t) = S_6(0) + S_7(0) + S_8(0). \tag{3.7c}$$

It is therefore possible to bound the total propensity uniformly. In this case we uniformise the system with uniformisation rate $\bar{a} = 15$ and apply stratification using $J = 6$ strata, i.e. the 6-quantiles. We use the parameter values and initial conditions ($\mathbf{X}_0 = (100, 0, 300, 0, 0, 300, 0, 0)^\intercal$) for the model as given in [109, Table 2]. Because of the non-linearity of the reaction dynamics it is not possible to find analytical values for the estimator sample variances and VRF. We therefore perform 256 repeated stochastic simulations using $N = 2^{14}$ sample paths per simulation to get estimates for the sample variances. As summary statistics we use the number of molecules at the final time $T = 200$ for the eight different species in the system and the results are tabulated in Table 3.2.

As we can see in Table 3.2 the VRF depends heavily on the chosen summary statistic. The MAPK-cascade is a hierarchical system in the sense that the conversion of the downstream species S_6, S_7 and S_8 depends on other upstream species, in particular S_5. The inter-strata variation will therefore likely be larger. The lower quantile strata observe fewer reactions and therefore will not see much activation of the downstream

Sample variance	Summary statistic $Q = \mathbb{E}[S_i]$			
	S_1	S_2	S_3	S_4
$N \cdot \mathrm{Var}\left[\hat{Q}\right]$	(22.6, 22.7)	(22.6, 22.7)	(202.4, 203.2)	(225.3, 226.2)
$N \cdot \mathrm{Var}\left[\hat{Q}_{\mathrm{prop}}\right]$	(22.2, 22.3)	(22.2, 22.3)	(193.7, 194.5)	(225.0, 225.9)
VRF	1.016 ± 0.003	1.016 ± 0.003	1.045 ± 0.003	1.004 ± 0.003

(a) First four species summary statistics

Sample variance	Summary statistic $Q = \mathbb{E}[S_i]$			
	S_5	S_6	S_7	S_8
$N \cdot \mathrm{Var}\left[\hat{Q}\right]$	(259.6, 260.7)	(199.8, 201.2)	(347.4, 348.7)	(777.1, 779.9)
$N \cdot \mathrm{Var}\left[\hat{Q}_{\mathrm{prop}}\right]$	(246.8, 247.9)	(180.0, 181.3)	(312.3, 313.6)	(670.6, 673.4)
VRF	1.052 ± 0.003	1.110 ± 0.006	1.112 ± 0.003	1.159 ± 0.003

(b) Last four species summary statistics

Table 3.2: Sample variance and estimated VRF for the MAPK cascade for both the standard MC estimator and the stratified estimator. The system is run until $T = 200$ with parameter values and initial conditions as given in [109, Table 2] and using a uniformisation rate $\bar{a} = 15$. Values tabulated are the 99.7% confidence intervals.

species, in contrast to the higher quantile strata, where more reactions fire and we see more conversion of the downstream species. The conditional summary statistics will therefore differ significantly between the different strata, which we know leads to a larger VRF by the decomposition of the variance (2.49).

If instead we consider the marginal distribution of the species at time $T = 200$ we find a VRF between 1.000 and 1.005, which is smaller than when computing moments of the species, just as we observed for the isomerisation example. Larger improvements are, again, achieved for the downstream species S_6, S_7 and S_8. Considering marginal bivariate joint distributions of the species yields similar results.

We can therefore conclude that the VRF using stratification and uniformisation can vary strongly depending on the model system, the summary statistic of choice, the uniformisation rate and the number of strata used. It is, however, also true that

for the same amount of computational complexity a variance at least as small as for the standard DM is observed, at no extra cost. The stratified UDM is therefore an improvement for models which would normally be simulated with (an optimised version of) the DM.

3.4 Transient analysis

Up to this point we were primarily concerned with summary statistics that depend on the value of the paths at some (fixed) time T, a commonly encountered scenario in practice. However, one might also be interested in path-dependent summary statistics, such as the transient evolution or mean value of a species over a series of time points $T_1, \ldots, T_n$. A straightforward approach to generate such summary statistics with the standard SSAs would be to run the paths over $[0, T_1]$ and record the state, then over $[T_1, T_2]$ and record the state, etc. Such an approach applies equally well to the (improved) UDM, where we can either generate the total number of reactions M on each interval individually or use the fact that given that M reactions occur in $[0, T_n]$ the numbers of reactions in the intervals defined by $T_1, \ldots, T_n$ follow a multinomial distribution, $\mathcal{M}(M, \boldsymbol{\pi})$ where $\pi_i = (T_i - T_{i-1})/T_n$. Using uniformisation, however, there is also an alternative way to estimate transient summary statistics as we will discuss next.

3.4.1 A weighted uniformisation method

We start with a general time-dependent summary statistic $Q(t) = \mathbb{E}[f(\mathbf{X}(t))]$ for some functional of the (time-dependent) sample paths $\mathbf{X}(t)$. Using uniformisation we recall that we can then expand this summary statistic using the state of the system after a fixed number of m (uniformised) reactions, denoted as $\mathbf{X}_m$, via a weighted average

$$Q(t) = \mathbb{E}[f(\mathbf{X}(t))] = \sum_{m=0}^{\infty} p_m(t) \mathbb{E}[f(\mathbf{X}_m)] = \sum_{m=0}^{\infty} p_m(t) Q_m, \tag{3.8}$$

where $p_m(t)$ is the probability that exactly m (uniformised) reactions have fired at time t. Importantly this shows that the time-dependence of the summary statistic $Q(t)$ is completely encoded in the (known) probabilities $p_m(t)$. Earlier in this chapter we proposed the UDM as a way to generate a MC estimate for the time-dependent summary statistic $Q(t)$ from the representation in equation (3.8). In the UDM we sample first $\tilde{m} \sim p_m(t)$ and then generate a single sample estimate for $Q_{\tilde{m}}$ by running the system for exactly $\tilde{m}$ reactions. Repeating this with N samples then yields an unbiased estimate for $Q(t)$.

However, we now note that generating a sample path with $\tilde{m}$ reactions also yields information about the system state after $0, \ldots, \tilde{m}-1$ reactions, crucially at no extra (simulation) cost. As an alternative method we can therefore successively approximate $Q_0, Q_1, \ldots$ if we run N (uniformised) sample paths in parallel and record their state after each reaction. This then directly yields a series of (unbiased) MC estimates $\hat{Q}_0, \hat{Q}_1, \ldots$ which we can combine to yield

$$\hat{Q}(t) = \sum_{m=0}^{\infty} p_m(t) \hat{Q}_m. \tag{3.9}$$

Note, however, that the $\hat{Q}_0, \hat{Q}_1, \ldots$ are not independent under this construction, which makes the analysis of the sample variance more complicated.

Generalised weights

More generally we note that using this formalism we can construct estimates for any time-dependent summary statistic of the form

$$\tilde{Q}(t) = \sum_{m=0}^{\infty} \mathcal{W}(m,t)\, Q_m, \tag{3.10}$$

where we now allow generalised weights $\mathcal{W}(m,t)$. Note that generalised weights, unlike the canonical weights, $\mathcal{W}(m,t) = p_m(t)$, do not necessarily need to be positive

or sum to unity. We do, however, require them to be regularised in such a way that equation (3.10) is well defined and $\tilde{Q}(t)$ is finite.

To see how such generalised weights could arise in practice we consider the problem of estimating the time-derivative of an expectation $\mathbb{E}\left[f(\mathbf{X}(t))\right]$. Employing the uniformisation approach we can simply use

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[f(\mathbf{X}(t))\right] = \frac{\mathrm{d}}{\mathrm{d}t}\left(\sum_{m=0}^{\infty} p_m(t)\mathbb{E}\left[f(\mathbf{X}_m)\right]\right) = \sum_{m=0}^{\infty}\left(\frac{\mathrm{d}}{\mathrm{d}t}p_m(t)\right)\mathbb{E}\left[f(\mathbf{X}_m)\right]. \quad (3.11)$$

This means that if we define the weight function $\mathcal{W}(m,t) = \mathrm{d}p_m(t)/\,\mathrm{d}t = \bar{a}(p_{m-1}(t) - p_m(t))$, where $\bar{a}$ is the uniformisation rate, we have the identity

$$\tilde{Q}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}\left[f(\mathbf{X}(t))\right] = \sum_{m=0}^{\infty}\mathcal{W}\left(m,t\right)Q_m. \quad (3.12)$$

More generally, for any linear operator, $\mathcal{L}_t$, depending only on time, t, we see that by similar logic we can find the action of $\mathcal{L}_t$ on $\mathbb{E}\left[f(\mathbf{X}(t))\right]$ via the relation

$$\mathcal{L}_t\left[\mathbb{E}\left[f(\mathbf{X}(t))\right]\right] = \sum_{m=0}^{\infty}\left(\mathcal{L}_t\left[p_m(t)\right]\right)\mathbb{E}\left[f(\mathbf{X}_m)\right] = \sum_{m=0}^{\infty}\mathcal{W}\left(m,t\right)Q_m. \quad (3.13)$$

For example, if $\mathcal{L}_t$ is the integral operator we find a weight function $\mathcal{W}(m,t) = \bar{a}^{-1}(1 - \sum_{i\leq m} p_i(t))$.

Computational approach

To estimate such weighted summary statistics in practice we must approximate the infinite sum via

$$\tilde{Q}(t) = \sum_{m=0}^{\infty}\mathcal{W}\left(m,t\right)Q_m \approx \sum_{m=M_L}^{M_R}\mathcal{W}\left(m,t\right)Q_m, \quad (3.14)$$

where we choose M_R and M_L such that the (relative) error we commit by truncating the sum is small. This can, for example, by done by specifying a tolerance level ε_R

from where we calculate M_R as the smallest integer such that $\sum_{m \geq M_R} |\mathcal{W}(m,T)| < \varepsilon_R$. A similar construction via a tolerance level ε_L yields M_L as the largest integer such that $\sum_{m \leq M_L} |\mathcal{W}(m,T)| < \varepsilon_L$. The resulting estimator takes the form

$$\hat{Q}_{\text{wUDM}}(t) = \sum_{m=M_L}^{M_R} \mathcal{W}(m,t)\, \hat{Q}_m, \tag{3.15}$$

where we recall that the $\hat{Q}_m$ are summary statistic estimates after m reactions. Note that due to the trunctation of the infinite sum this estimator is biased, though we can directly control its bias via the truncation points. Alternatively one could introduce a random truncation point for the infinite series to remove the bias following ideas in [186] and we provide more detail on this approach in Appendix 3.C.

To construct an algorithmic method built upon equation (3.15) we simulate N sample paths in parallel and calculate after each step $\hat{Q}_m = N^{-1} \sum_n f(\mathbf{X}_m^{(n)})$, where the $\mathbf{X}_m^{(n)}$ are the sample path states after m reactions. This method, which we call the weighted uniformised direct method (wUDM), Algorithm 3.5, has complexity $\mathcal{C}[\hat{Q}_{\text{wUDM}}] = N \cdot M_R$. A earlier variant of the wUDM, which solely considered the standard weight functions $\mathcal{W}(m,t) = p_m(t)$, appeared in [95] and we will discuss that version in more detail in Section 4.3.3.

Due to the fact that the estimators $\hat{Q}_m, \hat{Q}_n$ used in the wUDM are not independent we find that the sample variance for the wUDM estimator in equation (3.15) satisfies

$$\text{Var}\left[\hat{Q}_{\text{wUDM}}(t)\right] = \sum_{m=M_L}^{M_R} \sum_{n=M_L}^{M_R} \mathcal{W}(m,t)\, \mathcal{W}(n,t)\, \text{Cov}\left[\hat{Q}_m, \hat{Q}_n\right] \tag{3.16a}$$

$$= \frac{1}{N} \left(\sum_{m=M_L}^{M_R} \sum_{n=M_L}^{M_R} \mathcal{W}(m,t)\, \mathcal{W}(n,t)\, \text{Cov}\left[Q_m, Q_n\right] \right). \tag{3.16b}$$

Note that the terms $\text{Cov}[Q_n, Q_m]$ are related to the autocorrelations of the underlying DTMC (at lag $|m-n|$). Although we can estimate all the covariances via

Algorithm 3.5 Weighted uniformised direct method (wUDM).
This simulates an ensemble of N sample paths.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Final time T
Input: Uniformisation rate $\bar{a}$
Input: Algorithm parameters ε_L and ε_R.
Input: Weight function $\mathcal{W}(m, t)$

$\mathbf{X} \leftarrow \mathbf{X}_0$
Compute largest M_L s.t. $\sum_{m \leq M_L} |\mathcal{W}(m, T)| < \varepsilon_L$.
Compute smallest M_R s.t. $\sum_{m \geq M_R} |\mathcal{W}(m, T)| < \varepsilon_R$.
$m \leftarrow 0$
while $m < M_R$ **do**
    Sample $\tilde{v}^{(n)}$ pseudo-random points in $[0, 1)$ for $n = 1, \ldots, N$
    for $n = 1, \ldots, N$ **do**
        $a_k \leftarrow a_k(\mathbf{X}^{(n)})$ ▷ Calculate real reaction propensities.
        $a_{K+1} \leftarrow \bar{a} - \sum_{k=1}^{K} a_k$ ▷ Calculate virtual reaction propensity.
        Find p such that $\sum_{k=1}^{p-1} a_k < \bar{a}\tilde{v}^{(n)} \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
        if $p \in \{1, \ldots, K\}$ **then** ▷ Only need to fire real reactions.
            $\mathbf{X}^{(n)} \leftarrow \mathbf{X}^{(n)} + \boldsymbol{\zeta}_p$ ▷ Update state vector.
        end if
    end for
    if $m \geq M_L$ **then**
        Compute summary statistic after m reactions, $\hat{Q}_m$, using $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(N)}$ and add to weighted estimator via equation (3.15).
    end if
    $m \leftarrow m + 1$ ▷ Update reaction count.
end while

their usual unbiased estimators, this will generally be a computationally very intensive method to estimate the sample variance for $\hat{Q}_{\text{wUDM}}(t)$. We therefore treat the sample variance formula, equation (3.16), in practice as intractable and instead, just as, for example, for Latin hypercube sampling, generate M independent replicates[5], $\hat{Q}^{(1)}_{\text{wUDM}}, \ldots, \hat{Q}^{(M)}_{\text{wUDM}}$. Using these replicates we define $\hat{Q}_{\text{wUDM}} = M^{-1} \sum_m \hat{Q}^{(m)}_{\text{wUDM}}$ and

[5]Note that if we use the wUDM in combination with standard pseudo-random points, i.e. regular MC, we can take $N = 1$ in Algorithm 3.5 and generate M replicates as usual to get an estimator for the sample variance using a total of M samples. In Chapter 4 we will see an example where we can not directly take $N = 1$.

then note that the sample variance for this averaged estimator is given by

$$\hat{s}^2 = \frac{1}{M-1}\sum_{m=1}^{M}\left(\hat{Q}^{(m)}_{\text{wUDM}} - \hat{Q}_{\text{wUDM}}\right)^2. \tag{3.17}$$

Note that from equation (3.16) it is not clear whether the wUDM improves upon the (U)DM and thus could be seen as a variance reduction method. We will therefore explore the performance of the wUDM method on a tractable example in Section 3.4.2.

A posteriori transient calculation

Before looking at the wUDM method in practice we highlight a different way to analyse output from the wUDM. Suppose that we simulate the (uniformised) system up until M^* reactions have fired and we are willing to store the estimates $\hat{Q}_0, \ldots, \hat{Q}_{M^*}$ in memory. In that case we can analyse *a posteriori* transient summary statistics $\tilde{Q}(t)$, defined by equation (3.10), via equation (3.15), which shows that uniformisation can therefore yield an (unbiased) offline analysis method.

Using a (U)DM approach one would only be able to store $\hat{Q}(T_1), \ldots, \hat{Q}(T_n)$ and then need to interpolate, which incurs a bias, for any $\hat{Q}(t)$ where t is not equal to the saved time-points, $T_1, \ldots, T_n$. On the other hand, this new offline approach does not rely on such *a posteriori* interpolation and is accurate as long as the truncation error in equation (3.14) remains small. However, this approach is only practical for a given problem and time-range if it is actually feasible to store $\hat{Q}_0, \ldots, \hat{Q}_{M^*}$ in memory, which will depend on M^* and the type of summary statistic, $\hat{Q}_m$, used.

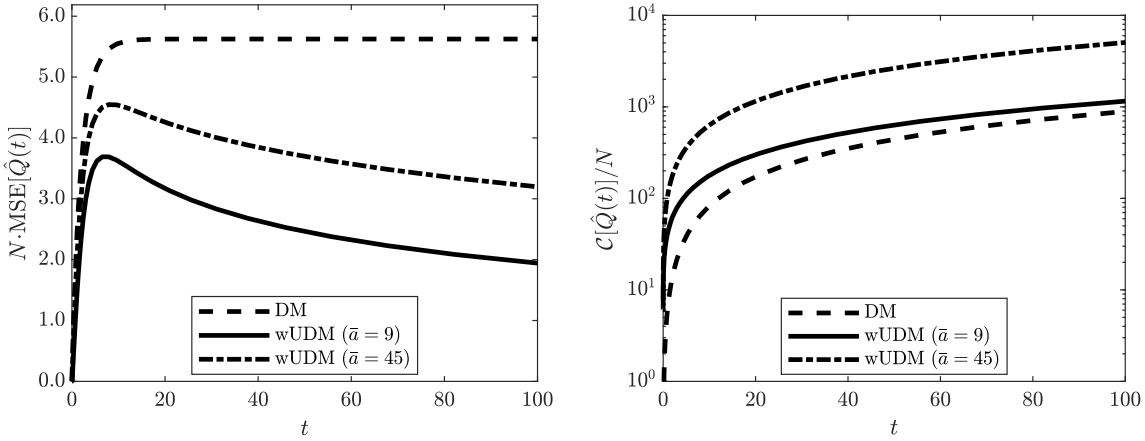
3.4.2 Examples

We consider again the isomerisation example from Section 3.3.2 and recall that this system, given in equation (3.6), can be uniformised exactly with uniformisation rate $\bar{a} = \max(c_1, c_2)N_0$, where N_0 is the total number of molecules in the system. Due to its relative simplicity we can analytically calculate both the results for the original

system, governed by the CME, and the uniformised system, which is now a DTMC. Note that throughout this example we use the wUDM method with truncation parameters $\varepsilon_L = \varepsilon_R = 10^{-16}/2$. This results in a (relative) bias due to truncation on the order of machine precision.

Moment estimation

First we examine the case when the summary statistic of interest is the average copy number of S_1 molecules, i.e. $Q(t) = \mathbb{E}[X_1(t)]$. Results for the evolution of the normalised statistical error of the DM and wUDM are shown in Figure 3.7(a).



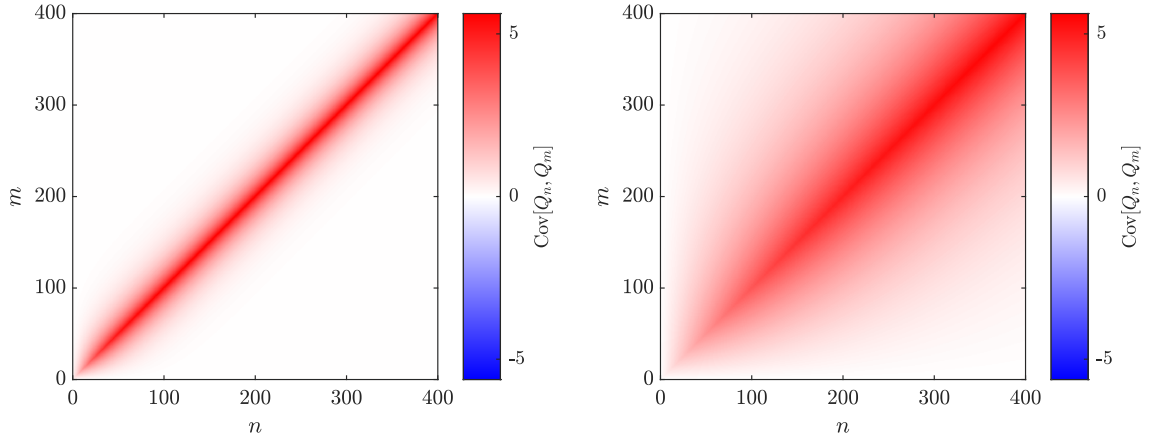
(a) Normalised MSE. (b) Normalised computational complexity $\mathcal{C}$.

Figure 3.7: Comparison between the wUDM and the DM for the isomerisation system, see Section 3.3.2, with parameters $c_1 = 0.3$, $c_2 = 0.1$ and initial condition $\mathbf{X}_0 = (0, 30)^\intercal$. The summary statistic $Q(t)$ is the average number of S_1 molecules at time t.

We can see that in terms of the statistical error the wUDM clearly outperforms the DM, even when we take a uniformisation rate which is five times larger ($\bar{a} = 45$) than strictly needed ($\bar{a} = 9$). On the other hand we also see in Figure 3.7(b) that the wUDM has a higher computational complexity than the DM. This can be understood from observing that in the uniformised system we will have to simulate more reactions due to the addition of the virtual reaction channel.

For this simple system we analytically compute the statistical error for the wUDM using equation (3.16) and the covariance between the different states in the uni-

formised systems is depicted in Figure 3.8. Comparing Figures 3.8(a) and 3.8(b) we can see the effect of increasing the uniformisation rate; effectively this increases the virtual reaction propensity leading to a smaller difference between states with similar numbers of reactions.



(a) $\bar{a} = 9$. (b) $\bar{a} = 45$.

Figure 3.8: Covariance between the states after m and n reactions in the uniformised system when the summary statistic Q is the average number of S_1 molecules at time t. Model is the isomerisation system, see Section 3.3.2, with parameters $c_1 = 0.3$, $c_2 = 0.1$ and initial condition $\mathbf{X}_0 = (0, 30)^\intercal$.

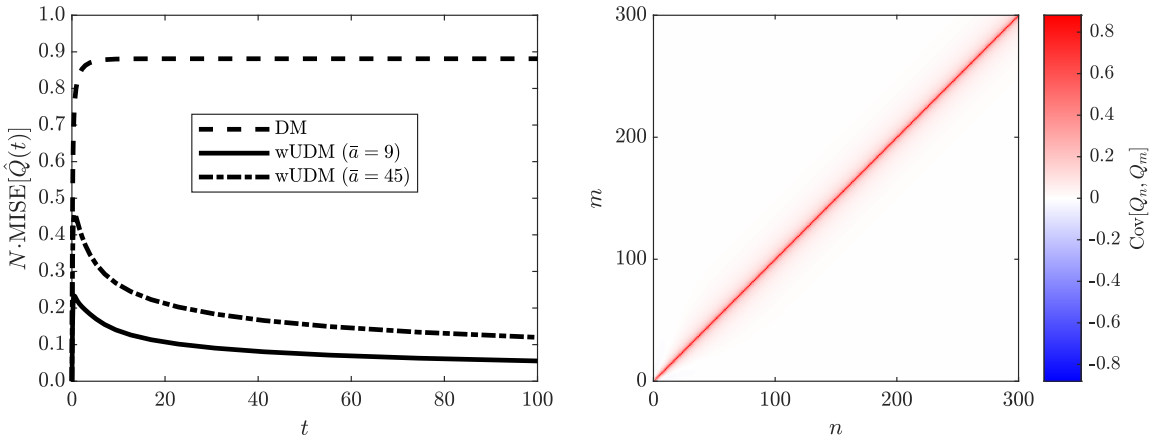
Combining the results on computational complexity and statistical error we find for the parameters in Figure 3.7(a) at time $t = 20$ a relative efficiency (see equation (2.37)) of the wUDM with respect to the DM of $\mathcal{E} \approx 1.02$ when $\bar{a} = 9$ and $\mathcal{E} \approx 0.20$ when $\bar{a} = 45$. In fact when $t \gtrsim 20$ we find $\mathcal{E} > 1$ when $\bar{a} = 9$ making the wUDM more efficient than the DM. The wUDM, however, for $t \lesssim 20$ is less efficient (for any choice of uniformisation rate, $\bar{a}$) than the DM when estimating the mean number of S_1 molecules, as can be seen in Figure 3.10.

Taking the uniformisation rate larger than strictly necessary decreases the variance reduction that is achieved by using the wUDM relative to the DM and increases the computational complexity of the wUDM, both of which reduce the efficiency of the wUDM. In fact, comparing the wUDM and the DM, the relative efficiency is such that $\mathcal{E} < 1$ when the uniformisation rate $\bar{a} = 45$. It is therefore clear that a good choice

for the uniformisation rate is paramount for the performance of the wUDM. A more robust strategy could be the use of adaptive uniformisation so that the uniformisation rate is close to the total reaction propensity throughout the time domain of interest.

Distribution estimation

Next we aim to estimate the distribution of species S_1 at time t via the EDF. We take the MISE as a measure for the statistical error and show the results for the same isomerisation system in Figure 3.9.



(a) Normalised MISE.

(b) Covariance between the states after m and n reactions in the uniformised system when $\bar{a} = 9$, integrated over the state space of S_1.

Figure 3.9: Comparison between the wUDM and the DM for the isomerisation system, see Section 3.3.2, with parameters $c_1 = 0.3$, $c_2 = 0.1$ and initial condition $\mathbf{X}_0 = (0, 30)^\intercal$. The summary statistic Q is the distribution of S_1 molecules at time t.

Changing the summary statistic can have a large effect on the decay of the statistical error as we can see by comparing Figures 3.7(a) and 3.9(a). We can see that the benefit from using the wUDM relative to the DM is larger if we aim to estimate the distribution of S_1 molecules, rather than solely its mean. Because the computational complexity is independent of the summary statistic of interest we find that when the uniformisation rate $\bar{a} = 9$, for all times $t > 0$ the relative efficiency of the wUDM with respect to the DM is larger than one (and up to an order of magnitude larger)

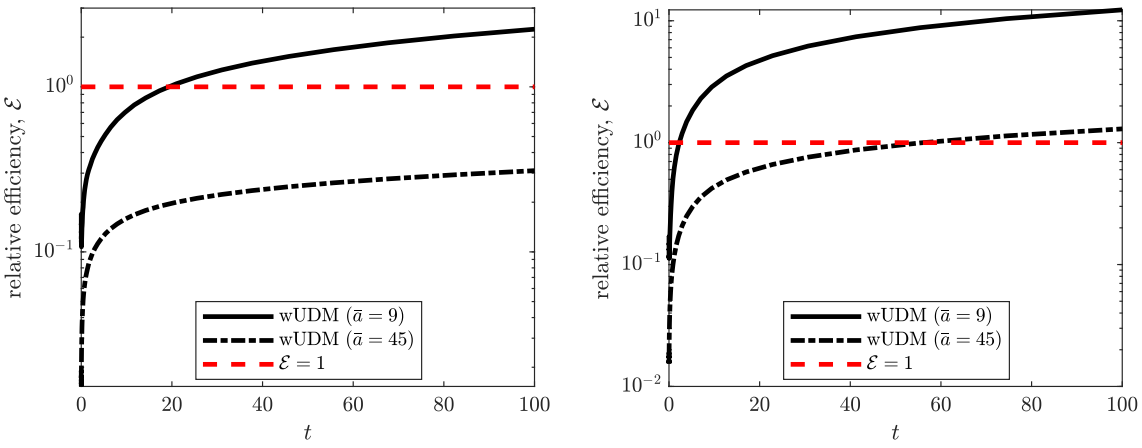
as can be seen in Figure 3.10. Even if we increase the uniformisation to $\bar{a} = 45$ we see that for $t \gtrsim 60$ the relative efficiency is still $\mathcal{E} > 1$. This seems to indicate that the wUDM is better suited to scenarios in which we aim to estimate distribution of species rather than (raw) moments.

We can understand this observation by considering the wUDM as a type of low-pass filter. Note that over an effective window, defined by the weighting functions $\mathcal{W}(m, t)$, we average the contributions of subsequent states in the DTMC. As a result we suppress high-frequency noise stemming from the firing of single reactions. If a summary statistic is influenced by this high-frequency noise the wUDM will thus smooth the output summary statistic. Note that the firing of a single reaction will only marginally change the copy number of a species (by an amount $\boldsymbol{\zeta}_k$) and the wUDM is thus less effective in this case. However, when we estimate distributions the effect of a single reaction is much larger due to the use of the indicator function, as described in Section 2.3.1, because a single reaction can completely change the state for which the indicator function returns a non-zero value, thereby effectively removing information on the previously occupied state if we use a conventional SSA. The wUDM, however, will retain this information by the averaging over subsequent states. We will explore this filter property of the wUDM in more depth in Section 4.3.

Time-derivative estimation

Finally, we consider the problem of estimating $\tilde{Q}(t) = \mathrm{d}\mathbb{E}\left[X_1(t)\right] / \mathrm{d}t$, i.e. the rate of change of the average copy number of S_1 molecules at a time t. We can use the wUDM with general weights (see equation (3.11)) to construct an efficient estimator of $\tilde{Q}(t)$. As a first alternative to the wUDM approach we consider a finite difference approach by using

$$\tilde{Q}(t) = \frac{\mathrm{d}\mathbb{E}\left[X_1(t)\right]}{\mathrm{d}t} \approx \frac{\mathbb{E}\left[X_1(t+\varepsilon)\right] - \mathbb{E}\left[X_1(t-\varepsilon)\right]}{2\varepsilon} = \frac{Q(t+\varepsilon) - Q(t-\varepsilon)}{2\varepsilon}. \quad (3.18)$$



(a) Efficiency when estimating mean species number, based on MSE.

(b) Efficiency when estimating species distribution, based on MISE.

Figure 3.10: Efficiency comparison between the wUDM and the DM for the isomerisation system, see Section 3.3.2, with parameters $c_1 = 0.3$, $c_2 = 0.1$ and initial condition $\mathbf{X}_0 = (0, 30)^\intercal$. The summary statistics Q are (a) the mean number and (b) the distribution, respectively, both of S_1 molecules at time t.

If we define $\hat{Q}_\varepsilon(t) = (2\varepsilon)^{-1}(\hat{Q}(t+\varepsilon) - \hat{Q}(t-\varepsilon))$ as the corresponding MC estimator we see that the bias of this estimator is of the order $\mathcal{O}(\varepsilon^2)$ via

$$\mathbb{E}\left[\hat{Q}_\varepsilon(t) - \tilde{Q}(t)\right] = \mathbb{E}\left[\frac{\hat{Q}(t+\varepsilon) - \hat{Q}(t-\varepsilon)}{2\varepsilon}\right] - \frac{\mathrm{d}\mathbb{E}\left[X_1(t)\right]}{\mathrm{d}t} \tag{3.19}$$

$$= \frac{Q(t+\varepsilon) - Q(t-\varepsilon)}{2\varepsilon} - \frac{\mathrm{d}\mathbb{E}\left[X_1(t)\right]}{\mathrm{d}t} \tag{3.20}$$

$$= \varepsilon^2 \left(\frac{1}{3!}\frac{\mathrm{d}^3\mathbb{E}\left[X_1(t)\right]}{\mathrm{d}t^3}\right) + \mathcal{O}(\varepsilon^4). \tag{3.21}$$

The variance of the finite difference approach, however, depends on how we couple the sample paths at times $t+\varepsilon$ and $t-\varepsilon$, respectively. Assuming a CRN type approach one can show that in fact $\mathrm{Var}[\hat{Q}_\varepsilon] = \mathcal{O}(\varepsilon^{-1})$. Note that this means that taking ε small, so as to reduce the bias, does increase the statistical error in $\hat{Q}_\varepsilon$. These two sources of error, systematic and statistical, respectively, must therefore be balanced in practice by a judicious choice of ε. We show results for the resulting normalised statistical error component and total MSE in Figure 3.11.

A second alternative[6] is the use of conditional Monte Carlo [162, Section 8.7]. We note that for a infinitesimal time interval $[t, t + \mathrm{d}t)$ using equation (2.3) we can write

$$\begin{aligned}\mathbb{E}\left[\mathbf{X}(t + \mathrm{d}t) \mid \mathbf{X}(t)\right] &= \mathbb{P}\big(\text{no reaction fires in } [t, t + \mathrm{d}t)\big)\mathbf{X}(t) \\ &\quad + \sum_{k=1}^{K} \mathbb{P}\big(R_k \text{ fires in } [t, t + \mathrm{d}t)\big)\left(\mathbf{X}(t) + \boldsymbol{\zeta}_k\right) \\ &= \big(1 - a_0(\mathbf{X}(t))\,\mathrm{d}t\big)\mathbf{X}(t) + \sum_{k=1}^{K}\left(\mathbf{X}(t) + \boldsymbol{\zeta}_k\right) a_k(\mathbf{X}(t))\,\mathrm{d}t \\ &= \mathbf{X}(t) + \sum_{k=1}^{K} \boldsymbol{\zeta}_k a_k(\mathbf{X}(t))\,\mathrm{d}t.\end{aligned}$$

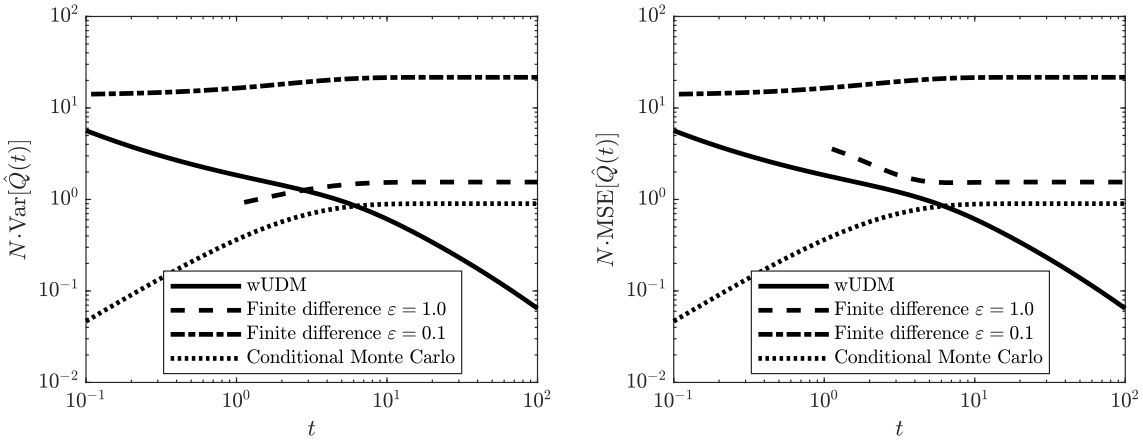
Using the limit definition of the derivative and integrating out the conditioning on $\mathbf{X}(t)$ we therefore arrive at

$$\frac{\mathrm{d}\mathbb{E}\left[\mathbf{X}(t)\right]}{\mathrm{d}t} = \sum_{k=1}^{K} \boldsymbol{\zeta}_k \mathbb{E}\left[a_k(\mathbf{X}(t))\right]. \tag{3.22}$$

This expression means that we can derive unbiased estimators for the time-derivatives by finding unbiased estimators for the value of the individual reaction propensities at a given time t, $a_k(\mathbf{X}(t))$. This can simply be achieved by evaluating the propensity functions, a_k, over the course of N independent sample paths and we denote this approach by "conditional Monte Carlo" in Figure 3.11.

From Figure 3.11(a) we can see that the wUDM approach is superior to the finite difference approach when $\varepsilon = 0.1$. For $\varepsilon = 1.0$ we see that for small t the finite difference approach can have smaller statistical error than the wUDM. This does, however, come at a price, because the bias contribution is larger when $\varepsilon = 1.0$ and in Figure 3.11(b) we see that the overall MSE already for $N = 10^3$ samples is dominated by this bias for small t. The comparison with conditional Monte Carlo is less clear-cut. We note that for small t the conditional Monte Carlo approach is preferable,

[6] Proposed in personal communication by Michael B. Giles who attributes the original idea to David F. Anderson. It also bears similarity to the smoothed perturbation analysis method [83].



(a) Normalised variance.

(b) Normalised MSE when using $N = 10^3$ sample paths.

Figure 3.11: Comparison between the wUDM and the DM-based finite difference approach for the isomerisation system, see Section 3.3.2, with parameters $c_1 = 0.3$, $c_2 = 0.1$ and initial condition $\mathbf{X}_0 = (0, 30)^\intercal$. The summary statistic Q is the rate of change of the average copy number of S_1 molecules at time t, i.e. $\mathrm{d}\mathbb{E}\left[X_1(t)\right] / \mathrm{d}t$.

but for moderate values of t its performance degrades and becomes worse than that of the wUDM. We can understand this degradation of performance by observing the increase in the variance of estimates of the mean of $\mathbf{X}(t)$ in Figure 3.8 which will subsequently increase the variance of estimators of the quantities $\mathbb{E}\left[a_k(\mathbf{X}(t))\right]$ needed for the conditional Monte Carlo method. For the wUDM, however, we note that the filter property mentioned in the previous section is stronger for larger t, i.e. we incorporate more information in the filter, which boosts its performance. We therefore conclude that in the case of estimating the time-derivative the wUDM is also a competitive method and both conditional Monte Carlo and the wUDM should be preferred over the finite difference approach.

3.5 Discussion

Uniformisation techniques are well-known tools in probability theory and have previously appeared in the context of chemical reaction networks in various forms [11, 95, 206, 214, 223]. In this chapter we revisited the uniformisation technique in the

context of pathwise simulation of realisations of said chemical reaction networks and focussed on the question of whether pathwise simulations under uniformisation can be carried out efficiently and whether there are any benefits of employing uniformisation methods.

To simulate a uniformised system one will introduce computational overhead by virtue of virtual reactions firing, and *a priori* it might therefore be expected that uniformisation-based methods are outperformed by more standard approaches like Gillespie's DM. However, the performance benefits of approaches building on uniformisation have been observed in the literature, albeit mainly in the context of time-inhomogeneous reaction networks [214]. We therefore, affirmatively, answer the question of whether in the case of time-homogeneous systems, which is still the predominant modelling choice of practitioners, it is possible to find a stochastic simulation approach based on uniformisation that is as at least efficient as the aforementioned golden standard, Gillespie's DM. By addressing the issues of efficiency and adaptiveness of the uniformised systems we arrived at Algorithm 3.3 which, complexity-wise, is on par with Gillespie's DM. We reiterate that many of the improvements that have been made to the basic implementation of Gillespie's DM, such as those leading to the sorting DM and optimised DM [39, 148], are equally applicable to the uniformised approaches discussed in this chapter, because the approach to choose the reaction channel that is to fire next is the same as for Gillespie's DM. The work in this chapter therefore opens up the exploration of new approaches built on uniformisation by removing a previously perceived complexity problem.

Extensions of Gillespie's DM for time-homogeneous models to include delays have been proposed previously [18, 34, 208, 209]. It is also possible to apply uniformisation to systems including delay reactions. In order to simulate such uniformised systems a first approach would be to explicitly generate the reaction times of the reactions as done for the time-inhomogeneous case, see the appendix to this chapter, Appendix 3.B. Thereafter we can proceed as in the delayed-SSA algorithm in [18].

We note that this approach also explicitly generates the reaction times of the virtual reactions and therefore has an extra overhead cost relative to the original system. We leave exploration of complexity, delays and uniformisation for future work.

As mentioned in Section 3.3, one can use stratification to improve the estimation of summary statistics for a wide variety of functionals of the sample path solutions. In the examples of Section 3.3 we focused on the mean species numbers and (empirical) distribution at a fixed time T, but reiterate that the stratification method is not restricted to such use cases only. Following the arguments from Section 3.3 it is clear that for a general class of summary statistics we can expect a performance at least as good as implementations using Gillespie's DM. At the same time, however, we note that when applying stratification to estimate, for example, the one-dimensional marginal distributions of the species in the MAPK-cascade or isomerisation system, a smaller variance reduction, now defined in terms of the MISE, was observed in Section 3.3.2 than when looking at the mean number of molecules as the summary statistic of interest. This shows that the improvement from using stratification depends on the summary statistic one is interested in. An issue appears when one is interested in using a summary statistic that inherently depends on lower order moments of the random variable $\mathbf{Y}$, such as the population variance. To use (unbiased) stratified sampling for such summary statistics is a non-trivial task and a variance reduction is no longer guaranteed, see for example [215]. This raises the question as to what kind of problems, both in terms of the chemical reaction network and in terms of the employed summary statistic, result in a substantial variance reduction when combining uniformisation and stratification. Though to conclude we reiterate again that in many common scenarios stratification will always yield a lower statistical error for the same computational complexity.

Finally we discussed in Section 3.4 the use of the wUDM, a weighted variant of the UDM. Effectively this method averages the states after consecutive reactions in the uniformised system, whilst maintaining an unbiased overall estimate. Note that we

can view this procedure as a type of low-pass filter in the following sense; by averaging subsequent states we suppress high-frequency noise stemming from the firing of single reactions and thereby smooth the output summary statistic. It was shown that this method can outperform the standard DM (and UDM), especially when the summary statistic of interest is a species distribution. In addition, the wUDM can yield an unbiased estimator for a wider class of (time-dependent) summary statistics, such as the rate of change of species moments. We note that the wUDM does not rely on a fixed uniformisation rate and can therefore also be used in combination with adaptive uniformisation.

Chapter appendix

3.A Unbiased adapting of the uniformisation rate

Here we provide an alternative view on the sampling of the time T^* at which one adapts the uniformisation rate, as discussed in Section 3.2.2. At the same time we show that the conditional sampling of T^* does in fact yield unbiased samples from the correct distribution.

Derivation of the conditional distribution of T^*

For completeness we first show that the M^*-th order statistic of M uniform random variables follows a beta distribution. We start with the assumption that we know that M reactions fire in the time interval $[0, T)$ as sampled from the Poisson distribution with parameter $\bar{a}T$. Their reaction times are then uniformly distributed on $[0, T)$. To find the distribution of the time T^* of the M^*-th reaction we note that in order to have $T^* \in [t, t + \Delta t)$ we need exactly $M^* - 1$ reactions in $[0, t)$, one reaction in $[t, t + \Delta t)$ and $(M - M^*)$ reactions in $[t + \Delta t, T)$. This yields the following expression

$$\mathbb{P}\left(T^* \in [t, t + \Delta t)\right) = \frac{M!}{(M - M^*)!(M^* - 1)!} \left(\frac{t}{T}\right)^{M^*-1} \frac{\Delta t}{T} \left(1 - \frac{t + \Delta t}{T}\right)^{M-M^*}, \tag{3.23}$$

where the pre-factor stems from the indistinguishable nature of the reactions in $[0, t)$ and $[t + \Delta t, T)$. By letting $\Delta t \to 0$ we get the probability density function $f(t)$ for T^*

$$f(t) = \frac{\Gamma(M - 1)}{\Gamma(M - M^* - 1)\Gamma(M^*)} \left(\frac{t}{T}\right)^{M^*-1} \frac{1}{T} \left(1 - \frac{t}{T}\right)^{M-M^*}. \tag{3.24}$$

We now let $u = t/T$, i.e. we consider the distribution of T^*/T, and recognise the pre-factor as $1/B(M - M^* + 1, M^*)$, where B is the beta-function. This leads to the

following probability density function $g(u)$ for T^*/T

$$g(u) = \frac{1}{B(M - M^* + 1, M^*)} (u)^{M^*-1} (1-u)^{M-M^*}, \tag{3.25}$$

which is readily seen to be equivalent to the beta distribution on $[0, 1)$ with parameters M^* and $(M - M^*) + 1$.

Unconditional distribution of T^*

In our approach to adapt the uniformisation rate we sample a time T^* conditional on the fact that M reactions within the interval $[0, T)$ were sampled at the uniformisation rate $\bar{a}$. For the unconditional distribution of T^*, the time of the M^*-th reaction and the point at which we adapt the uniformisation rate, as in Algorithm 3.3, we start with the following observation

$$\begin{aligned}
\mathbb{P}(T^* \leq t) &= \sum_{M=0}^{\infty} \mathbb{P}(T^* \leq t | M \text{ reactions in } [0,T)) \, \mathbb{P}(M \text{ reactions in } [0,T)) \\
&= \sum_{M=M^*}^{\infty} \mathbb{P}\left(\frac{T^*}{T} \leq \frac{t}{T} \middle| M \text{ reactions in } [0,T) \right) \mathbb{P}(M \text{ reactions in } [0,T)),
\end{aligned}$$

where we now recognise the first term in the sum to be described by a beta distribution (with parameters M^* and $(M-M^*)+1$) and the second term by a Poisson distribution (with parameter $\bar{a}T$). Substitution of the relevant expressions for these distributions then yields

$$\begin{aligned}
\mathbb{P}(T^* \leq t) &= \sum_{M=M^*}^{\infty} \left\{ \left(\frac{1}{B(M^*, (M-M^*)+1)} \int_0^{t/T} s^{M^*-1} (1-s)^{M-M^*} \, \mathrm{d}s \right) \right. \\
&\qquad \left. \frac{(\bar{a}T)^M}{M!} e^{-\bar{a}T} \right\} \\
&= \int_0^{t/T} \frac{s^{M^*-1} (\bar{a}T)^{M^*} e^{-aTs}}{\Gamma(M^*)} \left(\sum_{M=M^*}^{\infty} \frac{(\bar{a}T(1-s))^{M-M^*}}{(M-M^*)!} e^{-\bar{a}T(1-s)} \right) \mathrm{d}s \\
&= \frac{1}{\Gamma(M^*)} \int_0^{\bar{a}t} u^{M^*-1} e^{-u} \, \mathrm{d}u,
\end{aligned}$$

which is the distribution function for the Gamma($M^*, \bar{a}$) distribution. This is exactly the distribution of M^* i.i.d. exponential random variables with parameter $\bar{a}$. As a result, T^* is distributed as the M^*-th reaction time in a system with constant propensity $\bar{a}$. This proves that the conditional sampling of T^* as described in Section 3.2.2 yields in fact samples from the correct distribution and therefore makes the construction of sample paths with an adaptive uniformisation rate unbiased.

3.B Time-inhomogeneous models

The results presented so far in this chapter are valid for time-homogeneous Markov chains, i.e. models where the propensity functions do not have an inherent time-dependency and only depend on the state of the system $\mathbf{X}$. This assumption is generally sufficient to model intrinsic noise in a system, but whenever extrinsic noise effects are to be taken into account one has to relax this assumption. Extrinsic noise can be used to model the influence of external environments on the model behaviour and has been shown to influence the model dynamics [27, 99, 204, 214].

Stochastic model and simulation

The most common modelling change to account for extrinsic noise is to assume that the reaction rate constants are allowed to be time varying functions, i.e. $c_k(t)$. This dependency can be prescribed as a simple function, e.g. $c_k(t) = 10(1 + \sin(t))$, or be more elaborate, e.g. rates of the form $c_k(t) = 2\exp(Z(t))$ with $Z(t)$ some other stochastic process, which can, for example, be described by an SDE. The result is a time-inhomogeneous Markov chain model and currently the best-known SSA for such systems is the Extrande method [214] which uses the uniformisation method to avoid having to (numerically) integrate the propensity functions, $c_k(t)$. Methods built on the (numerical) integration of the propensity functions such as the modified next-reaction method [1] can also be constructed, but these methods do not compare favourably with methods based on uniformisation in general [214]. The difference

between the Extrande method and the UDM described in Section 3.1 is that the former method, like the DM, generates all the reaction times in a serial fashion. In contrast to the case of time-homogeneous models, where it is not necessary to generate the reaction times explicitly for the UDM, in the case of a time-inhomogeneous model one has to know the reaction times to account for the explicit time dependence of the reaction propensities. We therefore have to adapt the UDM (Algorithm 3.1) to account for this. As mentioned in Section 3.1 we can sample the reaction times, t_i, conditional on the knowledge that the number of reactions firing in $[0, T)$ is M. We note that the reaction times $t_1, \ldots, t_M$ are then distributed as the order statistics of M uniform random variables in $[0, T)$. To generate these one can simply generate M uniform random variables in $[0, T)$ and sort them in ascending order. Alternative methods to generate sorted uniform random numbers exist with better complexity properties, see, e.g. [48, Chapter 3]. This extra step can be done prior to the reaction dynamics loop as depicted in Algorithm 3.6.

Algorithm 3.6 UDM for time-inhomogeneous models.
This simulates a single sample path.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X}, t)$
Input: Uniformisation rate $\bar{a}$
Input: Final time T

$\mathbf{X} \leftarrow \mathbf{X}_0$
Generate $M \sim \mathcal{P}(\bar{a}T)$ ▷ Total number of reactions that fire in $[0, T)$.
Generate $t_1, \ldots, t_M \sim \mathcal{U}(0, T)$ s.t. $t_1 < t_2 < \cdots < t_M$ ▷ Reaction times.
for $m = 1, \ldots, M$ **do**
    Generate $u_1 \sim \mathcal{U}(0, 1)$
    $a_k \leftarrow a_k(\mathbf{X}, t_m)$ ▷ Calculate real reaction propensities.
    $a_{K+1} \leftarrow \bar{a} - \sum_{k=1}^{K} a_k$ ▷ Calculate virtual reaction propensity.
    Find p such that $\sum_{k=1}^{p-1} a_k < \bar{a}u_1 \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
    if $p \in \{1, \ldots, K\}$ **then** ▷ Only need to fire real reactions.
        $\mathbf{X} \leftarrow \mathbf{X} + \boldsymbol{\zeta}_p$ ▷ Update state vector.
    end if
end for

We note that if $a_0 > \bar{a}$ is observed in the course of a sample path the procedure

to adapt the uniformisation rate is largely the same as described in Section 3.2.2. However, when using Algorithm 3.6 the time T^* at which we adapt the uniformisation rate does not have to be sampled, as was done in step 10 in Algorithm 3.2. We can simply set $T^* = t_{M^*}$, with M^* the number of reactions that have fired up until the point at which $a_0 > \bar{a}$, which we note is available after completing step 3 in Algorithm 3.6.

The extra computational gain that was achieved by firing virtual reactions consecutively cannot be realised in this framework. For time-homogeneous models we used the fact that the distribution of the number of consecutive virtual reactions firing was geometric, but this is no longer valid for time-inhomogeneous models. Due to the explicit dependence on time of the propensity functions we get an expression for the distribution of the number of consecutive virtual reactions of the form

$$\mathbb{P}\left(r \text{ consecutive virtual reactions before next real reaction fires}\right) = \frac{a_0(t_{r+1})}{\bar{a}} \prod_{i=1}^{r} \left(1 - \frac{a_0(t_i)}{\bar{a}}\right). \quad (3.26)$$

This is no longer a single parameter distribution and, although it is analytically tractable, sampling from it will generally involve evaluating the total propensity a_0 at the time points of the virtual reactions. We describe such a procedure, which is effectively the inverse sampling transform, in Algorithm 3.7.

This approach is efficient in terms of its use of the random number generator as only a single random number is needed to fire consecutive virtual reactions. In addition, we could re-use this random number using methods described in [221] to find the next real reaction firing, therefore effectively requiring a single random number per real reaction fired in the interval of interest just like for the time-homogeneous methods.

However, as we previously noted, this process of firing consecutive virtual reactions requires the evaluation of the reaction propensities at the times of the virtual

Algorithm 3.7 Firing virtual reactions in UDM for time-inhomogeneous models.

Input: Propensity functions $a_k(\mathbf{x}, t)$
Input: Uniformisation rate $\bar{a}$
Input: Current state $\mathbf{X}$
Input: Reaction times $t_i, t_{i+1}, \ldots$

Generate $u \sim \mathcal{U}(0, 1)$
$r \leftarrow 0$ ▷ Count for number of virtual reactions to fire.
$b \leftarrow 1$
$a_k \leftarrow a_k(\mathbf{X}, t_i)$ ▷ Calculate real reaction propensities.
$a_0 \leftarrow \sum_{k=1}^{K} a_k$ ▷ Calculate the total real reaction propensity.
$s \leftarrow (a_0/\bar{a})b$ ▷ CDF for consecutive virtual reactions.
while $s < u$ **do**
    $r \leftarrow r + 1$
    $b \leftarrow b(1 - a_0/\bar{a})$
    $a_k \leftarrow a_k(\mathbf{X}, t_{i+r})$ ▷ Calculate real reaction propensities.
    $a_0 \leftarrow \sum_{k=1}^{K} \mathbf{A}_k$ ▷ Calculate the total real reaction propensity.
    $s \leftarrow s + (a_0/\bar{a})b$
end while
return r

reactions, which can become time-intensive if many virtual reactions are to be fired. As a result we do not want to have to choose a uniformisation rate $\bar{a}$ that is much larger than a_0 over $[0, T)$ as every virtual reaction will incur a propensity evaluation for the time-inhomogeneous case in Algorithm 3.6 (combined with Algorithm 3.7). If the total propensity, a_0, varies strongly over the course of $[0, T)$ it might be beneficial to regularly adapt the uniformisation rate, $\bar{a}$, which in the extreme case of adapting the rate after every reaction yields the Extrande method [214].

We therefore conclude that both the Extrande method and the UDM incur a propensity evaluation cost per virtual reaction fired, but that the Extrande method, by continuously adapting the uniformisation rate, can try to minimise the number of virtual reactions firing. The benefit of the UDM, however, is that it allows one to use extra variance reduction techniques such as stratification because the theory for stratification still holds in the case of time-inhomogeneous models. If one can therefore feasibly generate sample paths using Algorithm 3.6 with a fixed uniformisation rate, $\bar{a}$, it would be wise to apply stratification with respect to the number of reactions

firing, M, to achieve a variance reduction at no extra cost. If, however, the propensity a_0 changes noticeably over $[0, T)$ one might want to use an adaptive uniformisation rate, $\bar{a}$, or the Extrande method, in which case the stratification method is no longer valid because the reaction times in the interval $[0, T)$ are no longer drawn from a single parameter joint distribution that is the same for each sample path.

3.C Unbiasing the wUDM via randomisation

The standard wUDM estimator, see equation (3.15), is biased due to the truncation of the infinite sum in equation (3.14). Despite the fact that this bias can be accurately controlled *a priori* we present here an alternative variant of the wUDM leading to an unbiased estimator. This is based on a random truncation of infinite sums, an idea that seems to have first appeared in related context in [150, 185], but has its foundations as *Russian Roulette* in the physics literature. A more rigorous theoretical framework for this method is provided in [186]. Here we only discuss the case of using an upper-truncation point of the infinite sum, though we note that a lower-truncation point can also be incorporated in an analogous manner.

Rather than using the truncated estimator in equation (3.15) we define a new estimator

$$\hat{Q}^*(t) = \sum_{m=0}^{M^*} \mathcal{W}^*(m, t)\, \hat{Q}_m, \tag{3.27}$$

where the truncation point, M^*, is now a random variable, i.e. $M^* \sim p^*$ for some truncation distribution p^*, and $\mathcal{W}^*(m, t)$ is a new custom weight function that we will define later. Note that once the weight function $\mathcal{W}^*$ and the truncated distribution p^* are defined, it is straightforward to generate MC estimates of the summary statistic using this new estimator; we sample a truncation point M^* from p^* and subsequently run the wUDM approach up until M^* reactions have fired. Rather than using the update equation (3.15) in step 16 of Algorithm 3.5 we construct the estimator via equation (3.27).

By the law of total expectation we find the expectation of this new estimator

$$\mathbb{E}\left[\hat{Q}^*(t)\right] = \sum_{n=0}^{\infty} p_n^* \mathbb{E}\left[\hat{Q}^*(t) \mid M^* = n\right] \tag{3.28a}$$

$$= \sum_{n=0}^{\infty} p_n^* \left(\sum_{m=0}^{n} \mathcal{W}^*(m,t)\, Q_m \right) \tag{3.28b}$$

$$= \sum_{m=0}^{\infty} \left(\sum_{n=m}^{\infty} p_n^* \right) \mathcal{W}^*(m,t)\, Q_m. \tag{3.28c}$$

This shows that the estimate $\hat{Q}^*(t)$ can be made unbiased, i.e. $\mathbb{E}[\hat{Q}^*(t)] = Q(t)$, if we choose the custom weight function $\mathcal{W}^*(m,t) = \mathcal{W}(m,t)/(\sum_{n=m}^{\infty} p_n^*)$. We also note that the construction above holds for any truncation distribution p^*.

To analyse its efficacy we observe that $\mathcal{W}(m,t) \to 0$ and $\sum_{n=m}^{\infty} p_n^* \to 0$ as $m \to \infty$. This does not, however, mean that $\mathcal{W}^*(m,t) \to 0$ as $m \to \infty$, which is required[7] to have a finite variance estimator $\hat{Q}^*(t)$. In order to achieve a finite variance estimator one must therefore choose the truncation distribution p^* to decay slower to zero as $m \to \infty$ than the weight function $\mathcal{W}(m,t)$. For example, if the weight function $\mathcal{W}(m,t) = \mathcal{P}(m;\lambda t)$, i.e. a Poisson distribution with parameter λt, we can take $p^* = \mathcal{P}(m;\lambda^* t)$ with $\lambda^* > \lambda$. Note, however, that the expected complexity of this unbiased approach is $\mathbb{E}[M^*]$, which can be (significantly) higher than the biased wUDM method.

[7] This assumes that the covariances $\mathrm{Cov}[Q_n, Q_m]$ do not all vanish as $m, n \to \infty$, which for a general reaction system is the case.

Chapter 4

Quasi-Monte Carlo methods

In this chapter we discuss the use of low-discrepancy sequences (also known as quasi-random sequences) as input for SSAs in the context of chemical reaction networks. In other fields that rely heavily on MC computations, such as computational finance, the use of QMC techniques is common practice to decrease the statistical error. However, in the context of the simulation of chemical reaction networks this idea has received very little attention. Inspired by the only available work in this area [95] we first explore the implications of the combination of QMC methods with the τ-leap method in more depth. We show that the benefits of using QMC methods in this case are perhaps less striking than anticipated based on the success of QMC methods in the numerical solution of SDEs. We provide a detailed explanation for why this is the case, which serves as an explanation for the observations made in [95] as well. In line with [95] we also consider the use of the array-RQMC method, an extension of the more traditional QMC approach, and its combination with the τ-leap or the uniformisation method, which we discussed in Chapter 3.

Comment on originality This chapter is (partially) reproduced from the following publication:

Beentjes, C. H. L. & Baker, R. E. Quasi-Monte Carlo methods applied to tau-

leaping in stochastic biological systems. *Bulletin of Mathematical Biology* **81,** 2931–2959 (May 2019).

4.1 Quasi-Monte Carlo methodology

One of the drawbacks of general MC methods is their slow convergence rate, the root mean squared error (RMSE) is often of the order $\mathcal{O}(N^{-1/2})$ when using N samples. As discussed in Section 2.4, we can improve upon plain MC methods by using variance reduction techniques. Related to such techniques are QMC methods, which are based on the idea of attaining more uniform point distributions than standard (pseudo-) random numbers, thereby extending for example the Latin hypercube sampling method discussed previously in Section 2.4.1. Originally QMC methods were developed to approximate multidimensional integrals of the form

$$I = \int_{[0,1)^d} f(\mathbf{u})\,\mathrm{d}\mathbf{u}, \tag{4.1}$$

where d is the dimension of the problem. In a standard MC approach we generate a sequence $\mathbf{u}^{(1)}, \ldots, \mathbf{u}^{(N)}$ of d-dimensional uniform random variates and calculate an estimate of the integral via

$$\hat{I} = \frac{1}{N}\sum_{n=1}^{N} f(\mathbf{u}^{(n)}) \approx \int_{[0,1)^d} f(\mathbf{u})\,\mathrm{d}\mathbf{u}, \tag{4.2}$$

which in essence is the same as equation (2.17). The convergence of $\hat{I} \to I$ as $N \to \infty$ for MC methods is based on the law of large numbers (LLN), but this is not necessary for convergence of an approximation to the true integral I. For example, deterministic quadrature rules such as the midpoint-rule exist and have no relation to the LLN, yet provide the exact integral in the limit of $N \to \infty$. It turns out that, by virtue of the Koksma-Hlawka inequality, we can link the rate of convergence of $\hat{I}$ to I as $N \to \infty$ to the uniformity of the points $\left\{\mathbf{u}^{(n)}\right\} \subset [0,1)^d$ used. Uniformity in this context is

defined by the discrepancy, or the star-discrepancy D_N^* in particular. This measures the greatest deviation of a point set from a perfect uniform distribution on $[0,1)^d$, which is illustrated in Figure 4.1. Taking the supremum over all the boxes $\boldsymbol{B}$ with one corner at the origin we measure the difference between the expected number of points in the perfect uniform case and reality.

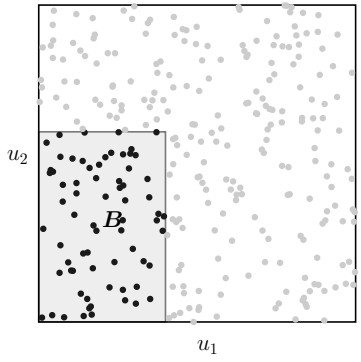


Figure 4.1: Illustration of the discrepancy concept in $[0,1)^2$. Shown are $N = 300$ points $\left\{\mathbf{u}^{(n)}\right\}$ scattered at random. If a perfect uniform distribution was attained by these points the number of points in $\boldsymbol{B}$ would be equal to $\text{Vol}(\boldsymbol{B}) \cdot N$ and this would hold true for every box $\boldsymbol{B} \subseteq [0,1)^2$.

More precisely then, the Koksma-Hlawka inequalities link the (star-) discrepancy D_N^* of the point set $\left\{\mathbf{u}^{(n)}\right\}$ and convergence of the approximate integral, $\hat{I}$, see for example [162, Section 15.4]. It is given in the most common form by considering the absolute error inequality

$$\left|\hat{I} - I\right| \leq V[f] D_N^*, \tag{4.3}$$

where $V[f]$ is the total variation of the integrand f in the sense of Hardy and Krause. This approximation error inequality can be thought of as the equivalent of equation (2.19) for MC methods. Note that equation (2.18) is an equality and holds in probability whereas equation (4.3) is a deterministic, worst-case, inequality. Comparing the two error bounds we see that $V[f]$ takes the place of the variance, σ^2, as both quantities depend on the integrand f. Furthermore we see that, rather than

having an error decay rate of $\mathcal{O}(N^{-1/2})$, we now have a factor D_N^* determining the behaviour as N increases. The total variation $V[f]$ of the integrand is for all practical purposes impossible to calculate and harder to estimate than the actual integral I. Furthermore in practical applications one can encounter functions with infinite $V[f]$, which voids the practical use of equation (4.3).

It turns out that it is possible to construct low-discrepancy sequences that will cover the integration domain more uniformly than random numbers, i.e. their discrepancy decays quicker than for equivalent random sequences, which have $D_N^* = \mathcal{O}(N^{-1/2})$. An example comparison between a (pseudo-) random sequence and a low-discrepancy sequence is depicted in Figure 4.2, which shows that low-discrepancy sequences can attain a much better spread over the integration domain $[0, 1)^2$.

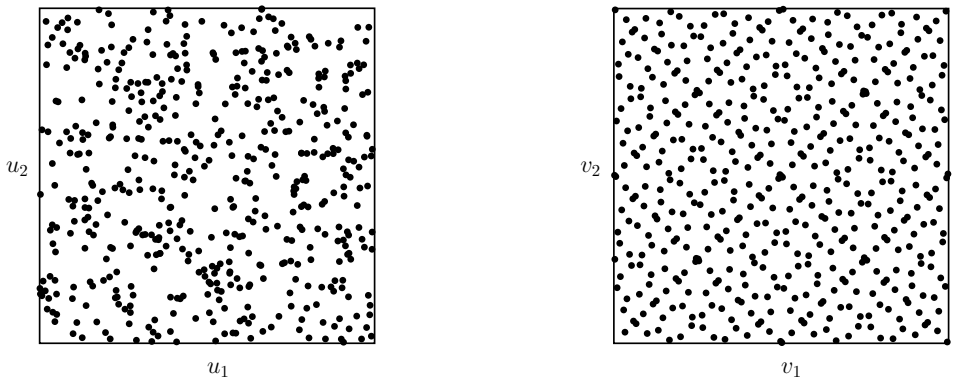


Figure 4.2: Comparison between a (pseudo-) random point set (*left*) and a low-discrepancy Sobol' point set (*right*) in $[0, 1)^2$, both of size $N = 2^9$.

For the QMC method we replace the (pseudo-) random sequence $\{\mathbf{u}^{(n)}\}$ by a deterministic sequence of low-discrepancy numbers $\{\mathbf{v}^{(n)}\}$ [129, Chapter 5]. Many different types of low-discrepancy point sequences exist and their fast generation is a subject of active research, see [49] and references therein for a recent review. By their deterministic construction these sequences can attain convergence orders like $\mathcal{O}((\ln N)^d N^{-1})$ for a wide range of integrands f by virtue of equation (4.3). This $\mathcal{O}((\ln N)^d N^{-1})$ convergence rate in the limit of $N \to \infty$ will always be better than that which can be attained with any regular MC method (even if we employ a variance reduction

technique), but if the dimension, d, is large and N is not very large it is not clear, based on theoretical results, whether QMC will provide an improvement. There are, however, various reports in the literature, albeit some without a theoretical justification, of QMC methods seemingly outperforming MC methods. Nowadays, QMC finds its application in many more areas than just integration, such as computational finance, see for example [129, Chapter 7], and Bayesian inference [67].

Uniformity and dimensionality using Sobol' sequences

A common low-discrepancy point set of choice is the Sobol' sequence and we will use this particular choice throughout this thesis. One of the advantages of this point set is that it is (quite) widely implemented in mathematical software libraries and extensible both in dimensions and in length. The latter means that to generate $N+1$ Sobol' points we can simply take the first N points and add a single new point to this set. The extensibility in dimensions means that if we have a Sobol' point in $[0,1)^d$ we can create a Sobol' point in $[0,1)^{d+1}$ by adding an extra (carefully chosen) variable to the point in $[0,1)^d$. Both of these properties make this particular type of low-discrepancy point set very suitable for stochastic simulations of a sequential nature, i.e. for which the number of sample points is not fixed *a priori* as is often the case in the context of chemical reaction simulations. Though Sobol' points can be extended one point at a time, this is in fact not optimal. Due to their construction the Sobol' points are best balanced when the number of points, N, is taken to be a power of 2. We will see an example of this later in Section 4.2.2.

By construction the Sobol' sequences have very uniform one-dimensional projections, similar to Latin hypercube sampling from Section 2.4.1. Though they fill the unit hypercube more uniformly than MC point sets (including Latin hypercube point sets), the same projection property does not necessarily hold for higher-dimensional projections, not even two-dimensional projections. In fact, due to its extensible construction the lower dimensions of the Sobol' sequence enjoy better equidistribution

properties compared to the dimensions following. As a result the choice of which dimension of the point set is used for which step in the SSA can have a large influence on the variance reduction that is achieved and in general one should try to use the lower dimensions of the points for the steps in the SSA that are most important in terms of the output variance. This means that the performance of QMC methods can be strongly dependent on the effective dimension of the model problem. The effective dimension quantifies this notion of importance of the input variables for the overall output variance, so that problems of low effective dimension are mainly governed by just a few input variables. QMC methods have been shown to be very well suited to such problems with low effective dimension, but at the same time often provide little improvement for problems that have an intrinsic high effective dimension. For more detail on the theory and construction of Sobol' sequences and the effective dimension we refer the reader to [162, Sections 15.7 and 17.2].

4.1.1 Randomised quasi-Monte Carlo

As discussed, a weakness of QMC methods compared to other quadrature rules is the lack of a measure of error. For MC methods we can use the LLN to estimate the variance and obtain confidence intervals. However, for QMC methods the points used follow from a deterministic construction and therefore do not allow the application of the LLN. The Koksma-Hlawka inequality (4.3) does provide deterministic error bounds, but for all practical purposes the quantities involved, $V[f]$ and D_N^*, cannot be calculated or efficiently computed. Furthermore we note that, because the low-discrepancy numbers are a deterministic set, the QMC estimator is not unbiased.

We can, however, consider a hybrid of MC and QMC methods. This type of approach introduces randomness into QMC methods in such a way that we keep their good convergence properties whilst at the same time allowing for error estimation with the LLN. The resulting methods are also known as randomised quasi-Monte Carlo (RQMC) methods.

The common approach in such RQMC methods is to take a low-discrepancy point set $\{\mathbf{v}^{(n)}\}$ and apply a randomised transformation to get a new set $\{\tilde{\mathbf{v}}^{(n)}\}$. Good randomisations (specific for the low-discrepancy sequence used) exist such that this new set is still a low-discrepancy point set but, at the same time, for all points in this set $\tilde{\mathbf{v}}^{(n)} \sim \mathcal{U}(0,1)^d$ holds. As a result of such a randomisation $\hat{I}$ using $\{\tilde{\mathbf{v}}^{(n)}\}$ will be an unbiased estimator of I. We refer to [129, Section 6.2] and references therein for more information on such randomisations.

To construct a measure of the statistical error in the estimate $\hat{I}$ we create M different randomised low-discrepancy point sets $\{\tilde{\mathbf{v}}^{(n,1)}\}, \ldots, \{\tilde{\mathbf{v}}^{(n,M)}\}$ which each will yield an unbiased estimator $\hat{I}^{(m)}$ of the objective I if we use equation (4.2). Combining these M randomisations using a standard MC approach gives rise to a new estimator

$$\hat{I}_{M,\mathrm{RQMC}} = \frac{1}{M}\sum_{m=1}^{M}\hat{I}^{(m)} = \frac{1}{M}\sum_{m=1}^{M}\left(\frac{1}{N}\sum_{n=1}^{N} f\left(\tilde{\mathbf{v}}^{(n,m)}\right)\right), \tag{4.4}$$

which we note again is an unbiased estimator of I. At the same time, we can now estimate the variance like we can for MC methods, because we effectively have M independent unbiased estimates of I. First we compute

$$\hat{s}^2_{\mathrm{RQMC}} = \frac{1}{M-1}\sum_{m=1}^{M}\left(\hat{I}^{(m)} - \hat{I}_{M,\mathrm{RQMC}}\right)^2, \tag{4.5}$$

which is an unbiased estimate for $\mathrm{Var}\left[\hat{I}^{(m)}\right]$. We can then incorporate this into the MC framework to find an unbiased empirical estimator of $\mathrm{Var}\left[\hat{I}_{M,\mathrm{RQMC}}\right]$

$$\hat{s}^2_{M,\mathrm{RQMC}} = \frac{\hat{s}^2_{\mathrm{RQMC}}}{M} = \frac{1}{M(M-1)}\sum_{m=1}^{M}\left(\hat{I}^{(m)} - \hat{I}_{\mathrm{M,RQMC}}\right)^2. \tag{4.6}$$

Note that this procedure is in many ways similar to the procedure to estimate the sample variance using Latin hypercube sampling as described in Section 2.4. In general this batched procedure to create unbiased estimators allows one to find an unbiased

sample variance estimator for any MC method and summary statistic, whether an explicit estimator for the sample variance of a single batch exists or not.

Due to its construction there are two ways one can reduce the variance of an RQMC estimator; either by taking more samples, N, per randomisation or by taking more randomisations, M. It is not always clear what choice one should make in this regard, but we can make some general observations. We note that increasing N means that within each randomisation more points of the low-discrepancy set will be used. This will therefore take advantage of the better spread of low-discrepancy point sets by lowering $\text{Var}\left[I^{(m)}\right]$, possibly at a rate faster than $\mathcal{O}(N^{-1/2})$. On the other hand, M only controls the number of randomisations, which ties in with the standard MC framework. Therefore M has a more limited influence on the statistical error convergence ($\mathcal{O}(M^{-1/2})$ for the RMSE). However, the number of randomisations, M, should be large enough to make the variance estimation via equation (4.6) sufficiently accurate, which can often already happen for $M \geq 10$ [129, Section 6.2]. Note that to get an RQMC estimator and sample variance we use MN sample points and thus for a fair comparison an RQMC method should be compared to standard MC with MN sample points.

Up to this point RQMC has been introduced as a variation on standard QMC methods by adding MC style randomisations. However, an alternative perspective of RQMC starts from a MC method and then adds the low-discrepancy points to make it into a variance reduction method for standard MC methods as we will briefly describe next.

RQMC as a MC variance reduction technique

This alternative look on RQMC as a variance reduction technique within the standard MC framework was first noted in [121]. After randomisation of the low-discrepancy point set the estimator $\hat{I}^{(m)}$ becomes an unbiased estimator of the integral I in equa-

tion (4.1). The variance of the estimator can, by linearity, be written as

$$\mathbb{V}\left[\hat{I}^{(m)}\right] = \frac{\sigma^2}{N} + \frac{2}{N^2} \sum_{1 \leq i < j \leq N} \mathrm{Cov}\left[f\left(\tilde{\mathbf{v}}^{(i,m)}\right), f\left(\tilde{\mathbf{v}}^{(j,m)}\right)\right], \tag{4.7}$$

which we note is similar to equation (2.40). In standard MC methods the points $\left\{\hat{\mathbf{v}}^{(n,m)}\right\}$ used are independent and therefore the covariances are zero. For an RQMC method, however, this is not the case because of the deterministic construction of the points used. Note that this remains true despite the randomisation, because the point set as a whole is still a low-discrepancy point set. As mentioned earlier in Section 2.4.1, to reduce the variance one wants the contribution of the sum of covariances to be as negative as possible. For the antithetic method, for example, this is achieved by a pairwise coupling of samples. For RQMC methods, however, its construction couples all the N sample paths in the hope to create an overall negative correlation effect, meaning a potentially larger variance reduction.

4.1.2 Array-RQMC

An interesting alternative RQMC method exists for the special case of simulating sample paths from DTMCs. Progress for this particular case was made in [119, 120, 126] with the introduction of a variant of RQMC specific for DTMCs, known as array-RQMC. The array-RQMC method is applicable to Markov chains of the form

$$\mathbf{Y}_{t+1} = \varphi_t(\mathbf{Y}_t, \mathbf{u}_t), \tag{4.8}$$

where the subscripts t denote the step in the DTMC, the φ_t are transition functions and $\mathbf{u}_t \in [0,1)^s$ depict uniform random variables. Suppose we are interested in the average value (or distribution) of $\mathbf{Y}_\mathcal{T}$, i.e. the state after $\mathcal{T}$ steps. Note that it is

possible to write this problem as the estimation of a standard expectation

$$\mathbb{E}\left[\mathbf{Y}_{\mathcal{T}}\right] = \int_{[0,1)^{s\mathcal{T}}} \Phi\left(\mathbf{Y}_0, \mathbf{U}\right) \mathrm{d}\mathbf{U} \tag{4.9}$$

$$= \int_{[0,1)^s} \cdots \int_{[0,1)^s} \varphi_{\mathcal{T}}\left(\varphi_{\mathcal{T}-1}\left(\ldots \varphi_1\left(\mathbf{Y}_0, \mathbf{u}_1\right)\ldots\right), \mathbf{u}_{\mathcal{T}}\right) \mathrm{d}\mathbf{u}_1 \ldots \mathrm{d}\mathbf{u}_{\mathcal{T}}, \tag{4.10}$$

where now Φ denotes the overall effect of applying the $\mathcal{T}$ sequential transitions φ_t. Note that this is an $s\mathcal{T}$-dimensional integral and to use a standard MC or RQMC method would require the use of point sets in $[0,1)^{s\mathcal{T}}$. Though this does not necessarily form a problem for the MC method it can severely reduce the variance reduction achieved by RQMC methods because the (effective) dimensionality of this problem will likely be of substantial size when we take many time steps, i.e. $\mathcal{T} \gg 1$.

Instead the array-RQMC method generates N sample paths, $\mathbf{Y}_t^{(1)}, \ldots, \mathbf{Y}_t^{(N)}$, in parallel by using arrays of length N filled with low-discrepancy sequences of size s for each time step. It therefore combines $\mathcal{T}$ low-discrepancy point sets of lower dimension, s, rather than using a single $s\mathcal{T}$-dimensional low-discrepancy point set. Note that without care this would not result in an improved estimate, because the trivial combination of, for example, two low-discrepancy point sets of dimension s does not yield a $2s$-dimensional low-discrepancy point set, see Figure 4.3 for an illustration when $s = 1$. Firstly, a correct implementation requires $\mathcal{T}$ independently randomised low-discrepancy point sets of dimension s. By independently randomising the low-discrepancy points at each time step we can ensure that there is no correlation between the quasi-random numbers used in different time steps and as a result the paths will be unbiased realisations of the underlying DTMC [119, Proposition 1].

To prevent the scenario of Figure 4.3 one needs a better method to combine these s-dimensional point sets. For example, we can independently and randomly re-order the points at each step, akin to Latin hypercube sampling in Section 2.4.1, and this method is an example of the Latin supercube construction [161]. The array-RQMC method, however, uses different strategies. One possibility is the use of an importance

function, h, mapping Ω, the state space of $\mathbf{Y}$, to $\mathbb{R}$. Given this importance function we then, at each step of the simulation, sort and re-number the N simulations such that

$$h\left(\mathbf{Y}_t^{(1)}\right) \leq h\left(\mathbf{Y}_t^{(2)}\right) \leq \cdots \leq h\left(\mathbf{Y}_t^{(N)}\right). \tag{4.11}$$

Note that the choice of the specific importance function is arbitrary and yet it can have a profound influence on the performance of the array-RQMC method, even making the convergence worse than standard MC. A study of sorting methods, including alternatives beyond importance functions, can be found in [122].

Finally we note a downside of the array-RQMC method; similarly to the Latin hypercube sampling discussed in Section 2.4.1 the array-RQMC method is not extensible. That means that if we have a set of N sample paths and we wish to add more sample paths, for example to further reduce the statistical error, we need to create a new ensemble of $\tilde{N}$ independent sample paths. In order to ensure that the combined estimator from both ensembles has a lower sample variance than the estimate from the first N sample paths we can take $\tilde{N} = N$. This effectively means we create a new batch just as we did with the RQMC method. This means the statistical error rate does not decay as if we increased N, but rather becomes the statistical error from N samples divided by the number of ensembles that we simulate. This latter contribution only happens at the MC rate. If we use Sobol' point sets as well then we need to take N to be a power of two. These factors mean that the array-RQMC method is slightly less flexible than the standard RQMC and MC methods.

4.2 RQMC applied to the τ-leap method

RQMC methods were introduced in the previous sections in the context of classical quadrature problems, but the framework applies equally well to many stochastic simulation approaches. This is due to the fact that the object of interest often takes the form of an expectation, which can also be written as an integral. Therefore it can

be sufficient for stochastic simulations, just as for quadrature problems, to substitute pseudo-random numbers in a MC simulation method with low-discrepancy numbers to get an equivalent RQMC method.

A crucial difference, however, is that for many standard methods of generating low-discrepancy numbers we need to know the dimensionality of the problem *a priori*. This is due to the fact that one cannot make a low-discrepancy point set in two dimensions by simply combining two one-dimensional point sets (note that this does work for pseudo-random numbers and is frequently used), which can be clearly seen in Figure 4.3. This difference between the two types of points is caused by the way low-discrepancy point sets are generated, in a well-defined deterministic manner, which introduces correlation between the individual points.

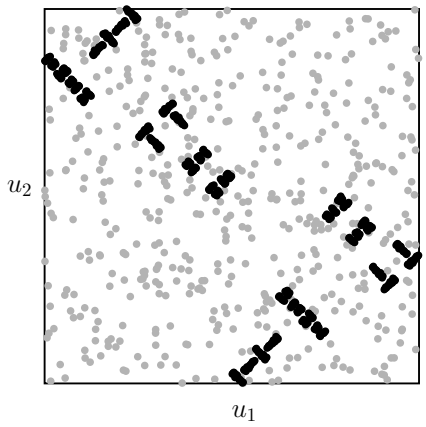


Figure 4.3: Illustration of the combination of two one-dimensional point sets into a two-dimensional set, both for randomised Sobol' sets (•) and pseudo-random sets (•). This approach for pseudo-random numbers results in a new two-dimensional pseudo-random number set, but this is not true for low-discrepancy numbers.

It is therefore not straightforward to combine QMC methods with for instance Gillespie's DM, as it is not clear, *a priori*, how many random numbers will be used in the simulation of a single path, i.e. the dimension is unknown and possibly infinite. There do exist ways to deal with possibly infinite integration problems in the context of QMC using (extensible) lattice rules and sequences, see for example [49, Section 5] for an overview and [121] for a software implementation of such constructions. For

chemical reactions a workaround for the simulation of CTMCs was presented in [95], using uniformisation of the CTMC combined with array-RQMC.

In this section, however, we focus instead on the (approximate) τ-leap method, which in its simplest form (fixed τ) does allow for an *a priori* determination of the dimension of the problem, which is equal to the computational complexity given in equation (2.36b). The low discrepancy numbers are then used in step 5 of Algorithm 2.3 to generate the Poisson random variables Y_k by applying an inverse transformation. Note that if this is done using a fast inverse transform, such as in [70], the process is not slower than direct methods for generating Poisson random variables in the recent implementations of MATLAB and Python (R2018b and Numpy 1.14.0, respectively).

Numerical experiments

We now test the effect of the combination of RQMC and the τ-leap method on a set of example chemical reaction systems. We compare the results using the τ-leap method to the results from numerically solving the CLE (2.11) using the Euler-Maruyama discretisation as QMC methods have proven to be very effective for numerical simulation of SDEs in the past [84]. We note that the two computational methods are based on different models, the RTCR (2.7) and the CLE (2.10), respectively. As a result, the bias of the methods will be different in general and we therefore do not directly compare the summary statistics computed. Instead, we ignore bias and only measure the convergence rate of statistical errors for both methods. For work on the bias error incurred from using the τ-leap approach we refer the reader to [4, 9, 181].

All numerical results for RQMC methods used as input the Sobol' sequences [202], with a random linear scramble combined with a random digital shift [144] to create randomised low-discrepancy points.

4.2.1 Monomolecular reaction networks

First we look at some elementary test systems to be able to closely compare the CLE-based method and the τ-leap method. The benefit of these systems is that the first two moments of the sample paths can be calculated analytically for both the τ-leap method and the Euler-Maruyama discretisation scheme, see Appendix 2.A. Combining this information with the exact expressions for the moments derived from the CME means that the bias due to the finite step size τ is exactly known as well.

Linear birth-death process

The first example is a single species linear birth-death system

$$S_1 \xrightarrow{c} \emptyset, \tag{4.12a}$$

$$S_1 \xrightarrow{c} 2S_1, \tag{4.12b}$$

which models auto-catalytic production and degradation of the species S_1. For simplicity we take the two reaction rates equal to each other so that we have $\mathbb{E}[\mathbf{X}(t)] = \mathbf{X}_0$ and $\mathrm{Var}[\mathbf{X}(t)] = 2ct\mathbf{X}_0$, i.e. the system will exhibit fluctuations around the steady state given by the initial state $\mathbf{X}_0$. Note that these identities also hold for the Euler-Maruyama discretisation of the CLE and the τ-leap scheme applied to the RTCR[1], both computational methods are thus unbiased with respect to the CTMC model. The complexity of the DM in this case is given by $\mathcal{C} = 2c\mathbf{X}_0 t$, and thus when $\tau^{-1} < c\mathbf{X}_0$ the complexity of the approximate τ-leap and CLE-based methods is lower than for the DM.

In Figure 4.4 we show the convergence results of the RMSE at time $T = 1.6$ for a system with $c = 1$ and $\mathbf{X}_0 = 10^3$. Both the Euler-Maruyama discretisation of the CLE and the τ-leap method use a time step $\tau = 0.2$, i.e. we take eight steps in both methods. The dimension of the problem is therefore 16 (two reaction channels and

[1]Under the assumption that no negative states occur while simulating sample paths.

eight time steps), which is generally thought to be within the realm of possibilities with (R)QMC methods.

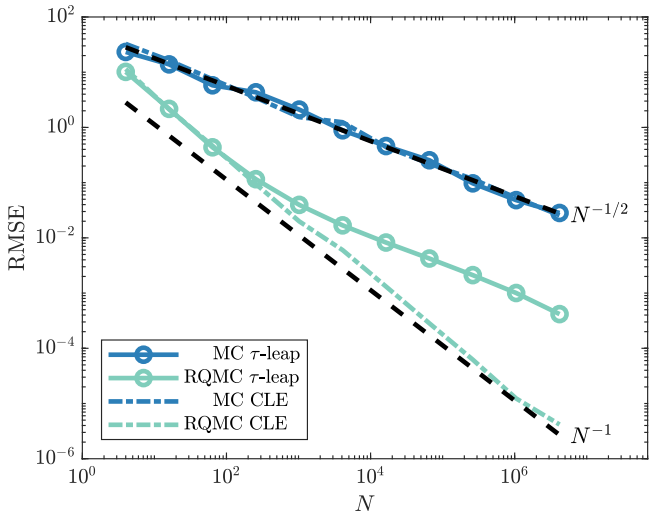


Figure 4.4: RMSE convergence for the mean number of S_1 molecules in the linear birth-death system, (4.12), with $c = 1$ and $\mathbf{X}_0 = 10^3$ at $T = 1.6$. The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC and $\mathcal{O}(N^{-1})$ for RQMC. To establish the RMSE for the RQMC method equation (4.5) was used with $M = 32$ randomisations.

We can clearly see that RQMC applied to both τ-leap and the CLE gives a strong improvement over the same method with standard pseudo-random numbers. However, it is also clear that, contrary to the MC method, where both the CLE-based discretisation and τ-leap method show equal convergence in terms of the RMSE, the RQMC methods differ in terms of their performance benefit. The SDE-based method has a convergence rate of roughly $\mathcal{O}(N^{-1})$ for all N. The same behaviour is not observed, however, for the τ-leap method which starts at an $\mathcal{O}(N^{-1})$ rate, but for $N \gtrsim 10^2$ seems to switch to the standard MC rate, i.e. $\mathcal{O}(N^{-1/2})$. This might come as a surprise, because in the regime of high molecule numbers and reaction propensities the CLE-derived methods are expected to form an excellent approximation to the RTCR and τ-leap method.

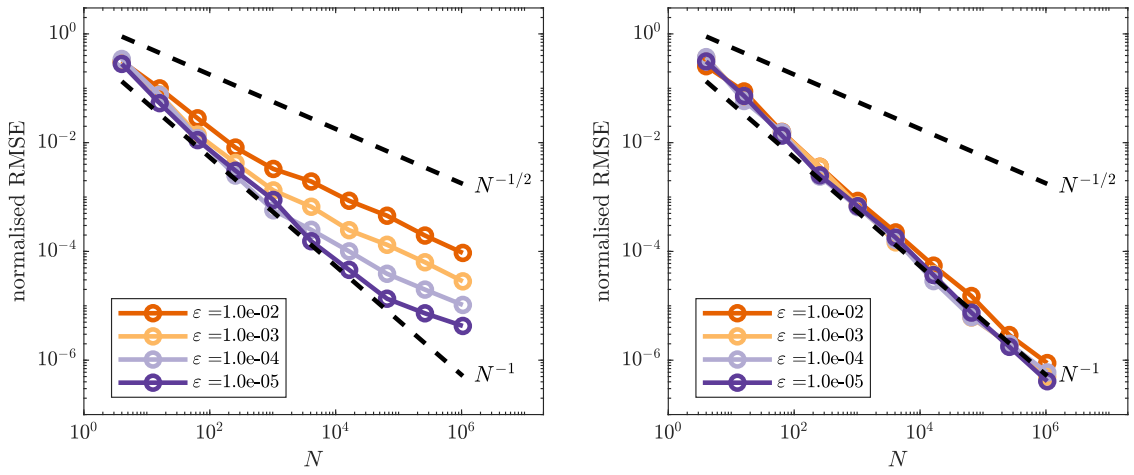
We note that the decrease in convergence rate is not due to sample paths reaching low molecule numbers, which could result in a discrepancy between CLE-based

methods and discrete molecule number methods such as the τ-leap method. With the initial conditions given above such sample paths are very unlikely to happen and were indeed not observed in the simulations used to produce Figure 4.4. This also means that a strategy to prevent negative molecule numbers, e.g. [2, 35, 42, 211], was not needed for this example.

A clear difference between the τ-leap method and CLE-based method stems from their respective update formulas, (2.15) and (2.10), which are related but not equal. Therefore the results from the two methods can differ subtly. By increasing the reaction rate parameters of the system the Poisson updates used for the τ-leap method are better approximated by normal random variables, which are what is used in CLE-based methods. However, as a result of the difference in updates, the state space of the variable $\mathbf{X}(t)$ is continuous for the CLE-based method and discontinuous, only taking integer values, for the RTCR-based τ-leap method. We now investigate what differences between the τ-leap method and CLE-based method exactly lead to the two contrasting convergence rate behaviours observed in Figure 4.4.

Firstly we test whether this observed behaviour of switching between convergence regimes changes when the difference between the τ-leap method and the equivalent discretisation of the CLE becomes smaller. This is done by running a similar set of simulations with varying initial conditions, and therefore molecule number regimes. We set $\mathbf{X}_0 = \varepsilon^{-1}$, so that as $\varepsilon \to 0$ we expect better agreement between the outputs of the τ-leap method and the CLE method. Note that as we vary ε the sample path variance for both methods has the form $\mathrm{Var}\left[\mathbf{X}(t)\right] = 2ct\varepsilon^{-1}$ and therefore grows as $\varepsilon \to 0$. In Figure 4.5 we show the resulting comparison between the two methods, with the RMSE rescaled by $\varepsilon^{-1/2}$. This is done to normalise the RMSE by the sample path variance as ε is changed. Note that this rescaling does not influence the convergence rate behaviour as a function of N.

It is clear from Figure 4.5 that for the Euler-Maruyama discretisation of the continuous CLE the value of ε does not influence the convergence rate of the RMSE,



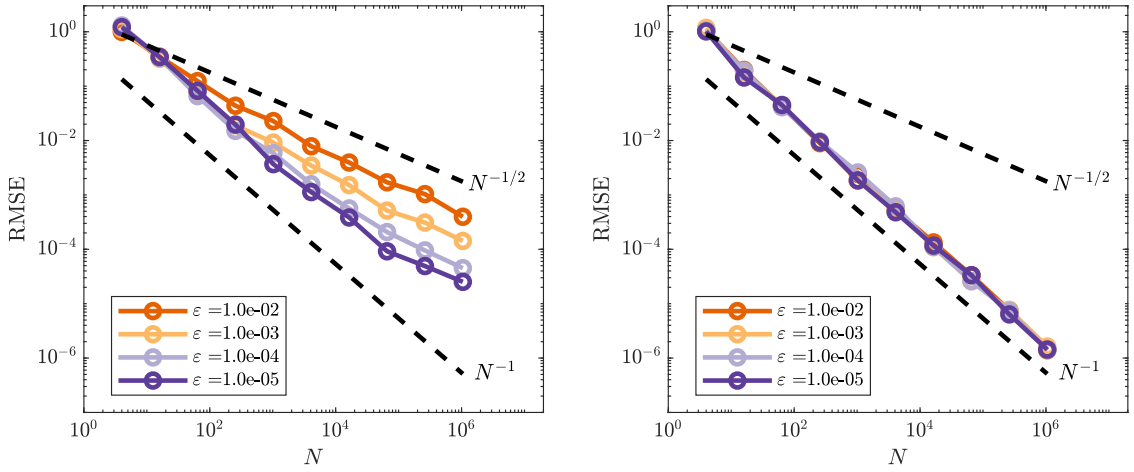
(a) τ-leap method. (b) Euler-Maruyama discretisation of CLE.

Figure 4.5: Comparison between the normalised RMSE convergence rates of the τ-leap method (a) and an Euler-Maruyama discretisation of the CLE (b) for the mean number of S_1 molecules in system (4.12) with $c = 1$ at $T = 1.6$ and varying initial condition $\mathbf{X}_0 = \varepsilon^{-1}$. The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC and $\mathcal{O}(N^{-1})$ for RQMC. The MC reference line shows the normalised MC sample variance for both methods, i.e. $\sqrt{\mathrm{Var}\left[\hat{Q}\right]/\mathbb{E}\left[\hat{Q}\right]} = \sqrt{2ct/N}$. To establish the RMSE, equation (4.5) was used with $M = 32$ randomisations.

i.e. it remains $\mathcal{O}(N^{-1})$ under changes in ε. The same cannot be said for the τ-leap method as now ε influences the transition between two different convergence regimes, fast $\mathcal{O}(N^{-1})$ and slow $\mathcal{O}(N^{-1/2})$ convergence, respectively. We observe that a smaller ε means that the transition takes place later, i.e. for higher N. Note that varying ε in the context of this system means changing the average copy number of S_1 encountered, and with that also the average reaction propensities. As a result, ε toggles how well the Poisson random variables in the τ-leap method can be approximated by normal variables, and therefore how good the CLE is as an approximation to the discrete dynamics. One might therefore think that RQMC performance depends on the 'closeness' of a discrete RTCR system is to its continuous CLE approximation. We now show that this is not necessarily the case.

We consider an additional rescaling of the reaction rate constant of the form $c = c_0\varepsilon$ in combination with the previous rescaling of the initial condition. Note

that now this rescaling keeps the reaction propensities on average constant and of the order $\mathcal{O}(c_0\tau)$ during a time step as $\varepsilon \to 0$. As a result the value of ε does not change whether the Euler-Maruyama discretisation of the CLE forms a good approximation to the τ-leap method. We perform a test to see what happens to the convergence rate if we let $\varepsilon \to 0$ in this case. Note that in this case we do not need to rescale the sample variance as it is already independent of ε. The results are shown in Figure 4.6 and show similar behaviour compared to the previous example, where c was fixed. It is therefore not the 'closeness' of the RTCR to the CLE which governs the convergence rate, as this is solely determined by the propensities of the reaction channels. Rather one could be tempted to conclude that the copy number of S_1 molecules rather than reaction propensities is crucial for this system. In the next sample we will show that this is also not the case and we will instead provide an alternative explanation at the end of Section 4.2.2.



(a) τ-leap method. (b) Euler-Maruyama discretisation of CLE.

Figure 4.6: Comparison between the RMSE convergence rates of the τ-leap method (a) and an Euler-Maruyama discretisation of the CLE (b) for the mean number of S_1 molecules in system (4.12) with $c = 10\varepsilon$ at $T = 1.6$ and varying initial condition $\mathbf{X}_0 = \varepsilon^{-1}$. The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC and $\mathcal{O}(N^{-1})$ for RQMC. The MC reference line shows the MC sample variance for both methods, i.e. $\sqrt{\text{Var}\left[\hat{Q}\right]} = \sqrt{2ct/N}$. To establish the RMSE equation (4.5) was used with $M = 32$ randomisations.

Isomerisation system

The previous example showed that in the case of molecule numbers in the system being not too small the standard RQMC method in combination with the τ-leap method performed well. In the following example we show that having large molecule numbers for some species in the system does not guarantee good convergence behaviour of RQMC in combination with the τ-leap method. We consider the two species system

$$S_1 \xrightarrow{c_1} S_2, \tag{4.13a}$$

$$S_2 \xrightarrow{c_2} S_1, \tag{4.13b}$$

and start with $\mathbf{X}_0 = (X_1, X_2)^\intercal$ initial molecules. We define $N_0 = X_1 + X_2$, $c = c_1 + c_2$, $r = c_1/c$ and note that this simple system is closed, which means that the sum of the numbers of S_1 and S_2 molecules at all times will be equal to N_0. This information can be used to decouple the dynamics of S_1 and S_2. Note that this system, under the CTMC model, converges to an equilibrium state of $(1-r, r)^\intercal N_0$. In order to ignore a transient regime in which the system goes to this equilibrium we start the simulations with $N_0 = \varepsilon^{-1}$ and $\mathbf{X}_0$ proportional to this equilibrium state, i.e. $\mathbf{X}_0 = \varepsilon^{-1}(1-r, r)^\intercal$.

We note that with this initial condition for both the τ-leap method and the Euler-Maruyama discretisation of the CLE we have $\mathbb{E}[\mathbf{X}(t)] = \mathbf{X}_0$ and $\mathrm{Var}[\mathbf{X}(t)] \propto \varepsilon^{-1}$, as in the previous example. This also means that both computational methods are unbiased for this system.

In Figure 4.7 we show results for a simulation until $T = 1.6$ with time step $\tau = 0.2$ and parameters $c = 1$, $r = 10^{-4}$ and $\varepsilon = 10^{-6}$. This means that S_1 has copy numbers of the order 10^6, which one might reasonably say is large. In particular, the S_1 copy number is an order of magnitude larger than any of the copy numbers encountered in the previous linear birth-death processes. We note again that there is a gain in performance in terms of RMSE if we compare RQMC and the equivalent MC method. However, we also observe that, despite S_1 having large copy numbers, the RMSE for S_1 from the τ-leap method quickly goes to $\mathcal{O}(N^{-1/2})$ convergence.

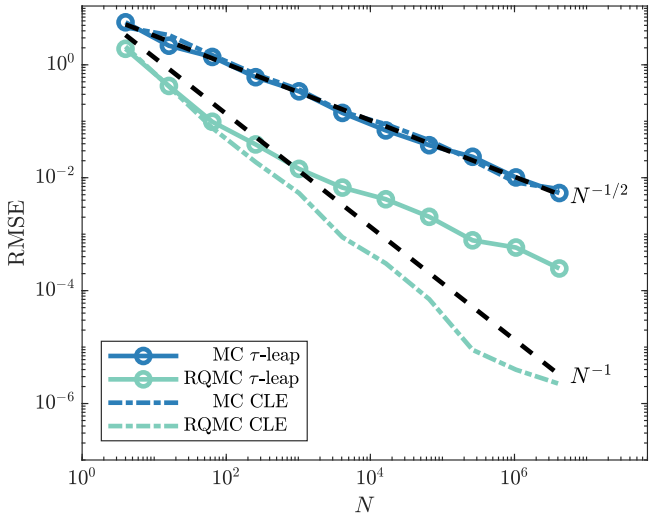


Figure 4.7: RMSE convergence for the mean number of S_1 molecules in the isomerisation system (4.13) with $c = 1$ and $r = 10^{-4}$ at $T = 1.6$ and initial conditions $\mathbf{X}_0 = 10^6(1 - r, r)^\intercal$. The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC and $\mathcal{O}(N^{-1})$ for RQMC. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 32$ randomisations.

This example therefore shows that high copy numbers for some of the reacting species are also no guarantee for RQMC method convergence rates faster than the standard $\mathcal{O}(N^{-1/2})$ rate. This is even the case when we use summary statistics that involve just those high copy number species (in our example the number of S_1 molecules). We must therefore find a different description for the observed convergence behaviour of the RQMC τ-leap method.

4.2.2 Discrete toy model

To explain the observations from the previous examples we consider a problem in traditional quadrature. We consider the integration of the following d-dimensional test functions over the domain $[0, 1)^d$:

$$f(\mathbf{u}) = \sqrt{12/d} \sum_{i=1}^{d} \left(u_i - \frac{1}{2} \right), \tag{4.14a}$$

$$f(\mathbf{u}) = \sqrt{12^d} \prod_{i=1}^{d} \left(u_i - \frac{1}{2} \right). \tag{4.14b}$$

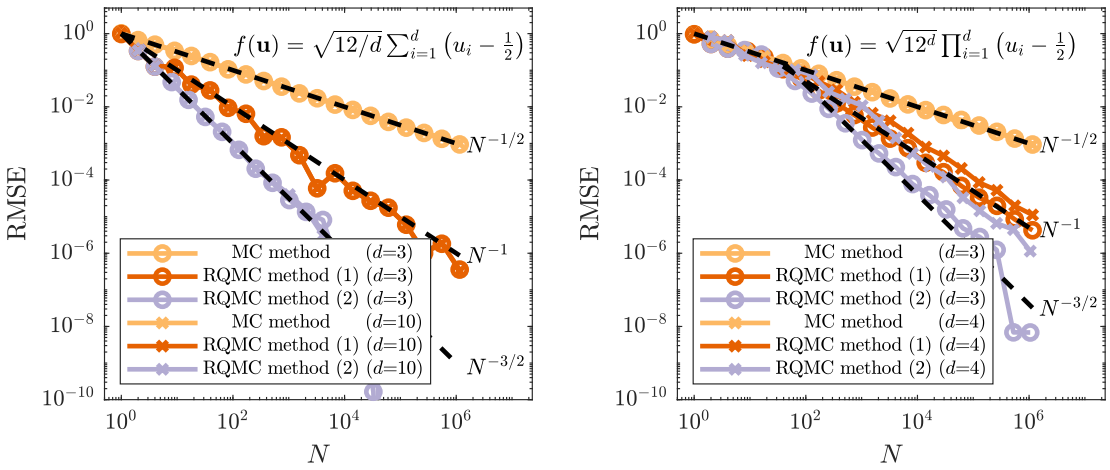
Both functions integrate to zero over the d-dimensional hypercube and have variance

$$\int_{[0,1)^d} f^2(\mathbf{u})\,\mathrm{d}\mathbf{u} - \left(\int_{[0,1)^d} f(\mathbf{u})\,\mathrm{d}\mathbf{u}\right)^2 = 1, \tag{4.15}$$

regardless of d. We note that (4.14a) is an easy test function for (R)QMC methods as it represents a linear combination of one-dimensional functions (for which (R)QMC methods perform well). The effective dimension in the superposition sense of these additive functions is equal to one [33] and the convergence rate for RQMC[2] is $\mathcal{O}(N^{-3/2})$ regardless of dimension d if we consider N to be a power of two (the base power for the Sobol' sequence). The second function (4.14b) was considered previously in [165] and is a much harder integrand for RQMC and MC methods. It has the property that RQMC methods for a low number of points have $\mathcal{O}(N^{-1/2})$ RMSE convergence which turns into $\mathcal{O}(N^{-3/2})$ if sufficiently many points are used (the definition of sufficient, which depends on d, is found in [165]). RMSE convergence for these test functions for some dimensions d is depicted in Figure 4.8. This shows that RQMC does indeed do a very good job at integrating (4.14a) and for N large enough the same holds for (4.14b). For (4.14a) we see that in terms of RMSE convergence there is no dependency on d. In addition we note that using Sobol' sequences means that we can get a better performance if we take N to be a power of 2, which we will do hereupon.

For the chemical test systems previously discussed there was a clear difference in performance for RQMC methods between the continuous CLE and the discrete RTCR. In terms of quadrature, the integrand f in the first case is continuous, whereas in the second case it is discontinuous. Most convergence results for RQMC are based on the assumption that the integrand is continuous and it has been observed before that discontinuities can have an adverse effect on the convergence rate, see for example [23, 94, 155, 156]. We now show that a certain type of discontinuity, closely resembling

[2]Provable results on the convergence rate for randomised Sobol' sequences are only available if Owen nested uniform scrambling is used [163], rather than the randomised matrix scrambling as used in this work. We refer the reader to [166] for a more in-depth discussion of scrambling methods for Sobol' sequences.



(a) Test function (4.14a). (b) Test function (4.14b).

Figure 4.8: Comparison between the RMSE convergence rates of MC and RQMC estimates for integrals of equation (4.14). RQMC method (2) has N equal to a power of 2, whereas RQMC method (1) does not have N equal to a power of 2. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 512$ randomisations and the dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC and $\mathcal{O}(N^{-1})$ and $\mathcal{O}(N^{-3/2})$ for RQMC.

the chemical reaction system case, can replicate the convergence behaviour that we have observed in the previous section.

We introduce the following transformation of the test functions f, which acts upon the input of the function f,

$$f_\varepsilon(\mathbf{u}) = f\left(\varepsilon\left\lfloor\frac{\mathbf{u}}{\varepsilon}\right\rfloor\right), \tag{4.16}$$

where ε is a parameter that tunes the level of discontinuity. Note that as $\varepsilon \to 0$ the function becomes smoother. In Figure 4.9 we show the effect of varying ε on the one-dimensional function (4.14a) and the filled contour plot for (4.14b) for $\varepsilon = 0.07$. Note that by applying transformation (4.16) we create a function which has all its discontinuities parallel to the axes of the integration domain $[0, 1)^d$. In [94] it was proven that such axes-parallel discontinuities have a relatively mild effect on the convergence of RQMC methods.

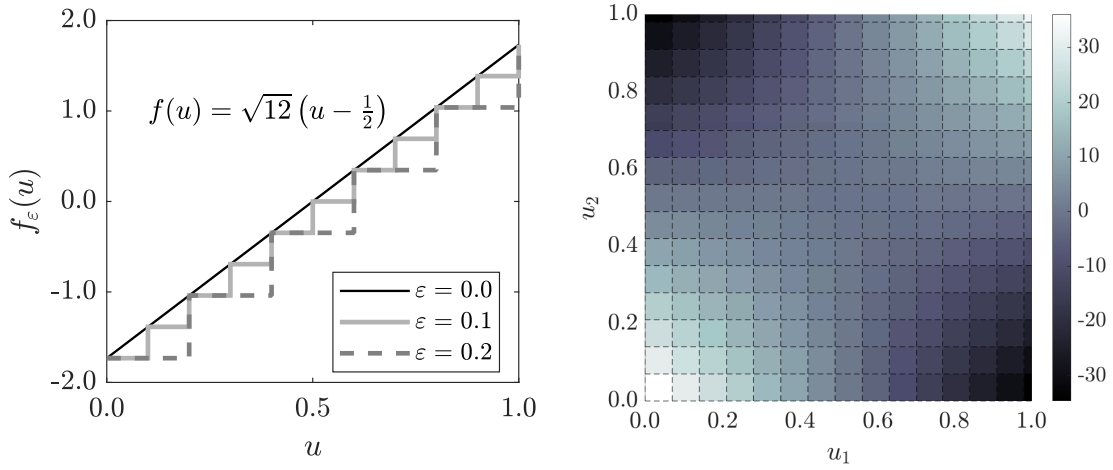


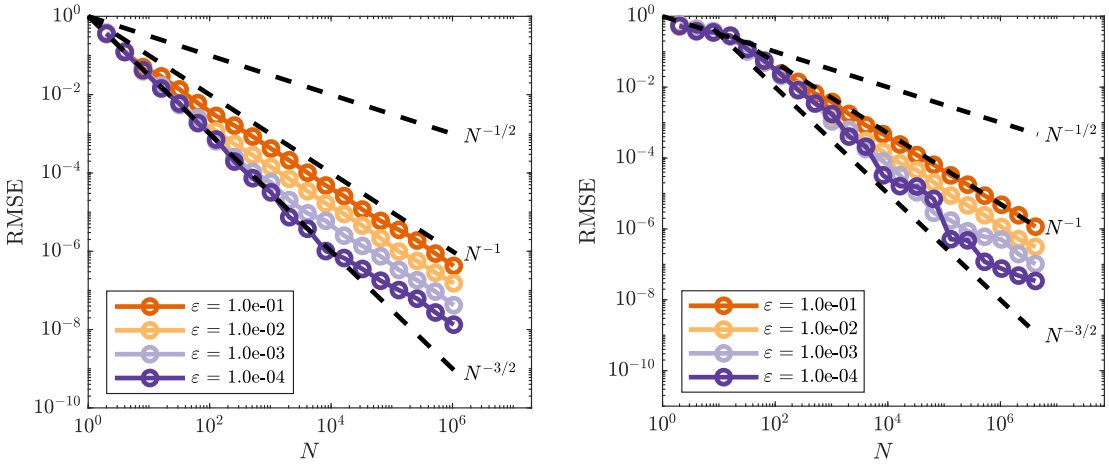
Figure 4.9: Result of the discontinuity transformation (4.16). For the one-dimensional function (4.14a) we plot the graph of $f_\varepsilon(\mathbf{u})$ (*left*). For the two-dimensional function (4.14b) we plot the filled contour plot for $\varepsilon = 0.07$ (*right*), clearly showing the discontinuities of $f_\varepsilon(\mathbf{u})$, depicted with dashed lines.

In Figure 4.10 we see the effect that the introduction of discontinuities by (4.16) has on the RMSE convergence. Where the continuous functions showed $\mathcal{O}(N^{-3/2})$ convergence (recall Figure 4.8), the discontinuous counterparts have, for large enough N, a slower $\mathcal{O}(N^{-1})$ convergence rate. The results in Figure 4.10 hold for a wide range of dimensions d. As expected, results for (4.14a) are not affected by d due the fact that the function after transformation is still one-dimensional in the superposition sense. On the other hand for (4.14b) the effect of transformation (4.16) only shows once enough points have been used to leave the $\mathcal{O}(N^{-1/2})$ initial convergence, and after that convergence rates seem to drop from $\mathcal{O}(N^{-3/2})$ to $\mathcal{O}(N^{-1})$ as well.

Next we introduce a different transformation that converts continuous functions into discontinuous ones,

$$f_\varepsilon(\mathbf{u}) = \varepsilon \left\lfloor \frac{f(\mathbf{u})}{\varepsilon} \right\rfloor . \tag{4.17}$$

Note that, in contrast to (4.16), this transformation acts upon the function output values. As a result, discontinuities introduced by (4.17) do not necessarily align with the axes of $[0, 1)^d$, as can be seen in Figure 4.11 in two dimensions.



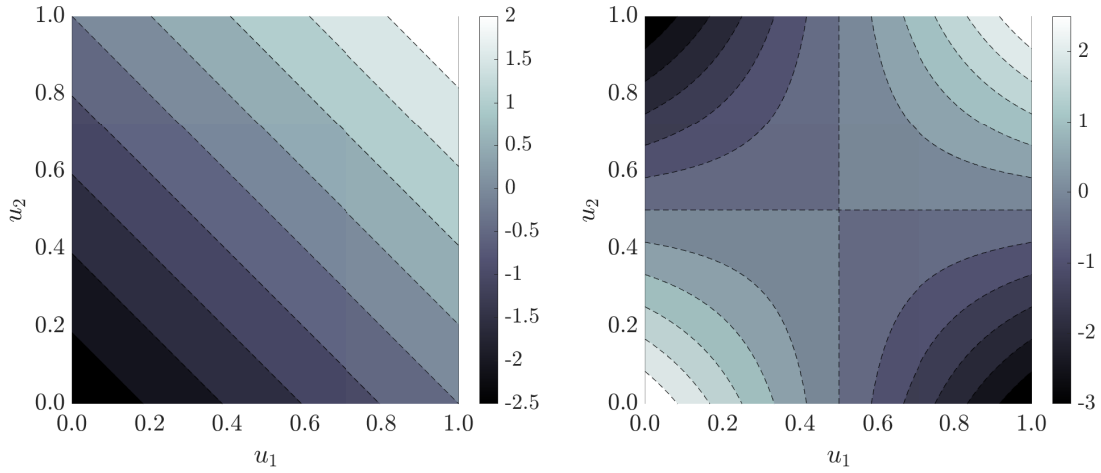
(a) Integrand (4.14a), $d = 10$.

(b) Integrand (4.14b), $d = 3$.

Figure 4.10: Effect of discontinuity transformation (4.16) on the RMSE convergence for (4.14a) ($d = 10$) and (4.14b) ($d = 3$). $M = 128$ randomisations were used and the dashed black lines show typical reference convergence rates, $\mathcal{O}(N^{-1})$ and $\mathcal{O}(N^{-3/2})$ for RQMC and $\mathcal{O}(N^{-1/2})$ for MC.

Results for the RMSE convergence for varying ε are shown in Figure 4.12. We observe again that for small values of N the convergence rate is $\mathcal{O}(N^{-3/2})$, similar to the continuous case. However, we see that with transformation (4.17), for N large enough, the convergence rate becomes $\mathcal{O}(N^{-1/2})$, rather than $\mathcal{O}(N^{-1})$ which was observed for transformation (4.16). This comes back to the fact that the discontinuities introduced by (4.17) do not align with the axes of the integration domain $[0, 1)^d$. One can understand the impact of this by considering the way many RQMC point sets are constructed (in particular digital nets, of which Sobol' point sets are a special case). For such sets the points are equidistributed with respect to axes-aligned hyperrectangles. If the discontinuities of the integrand do not align with the domain axes, such as for transformation (4.17), then the RQMC points will not be able to sample the integrand's different contributions uniformly. In [94] it was also shown that discontinuities that do not align with the domain axes are of a more detrimental type of discontinuity if one wants to use RQMC methods.

The limiting convergence rate is given by the MC rate $\mathcal{O}(N^{-1/2})$. This agrees with the result that RQMC methods will, in the worst case scenario, behave very much



(a) Integrand (4.14a). (b) Integrand (4.14b).

Figure 4.11: Result of the discontinuity transformation (4.17). Plots show the filled contour plot with $\varepsilon = 0.5$ for the integrands (a) (4.14a) and (b) (4.14b), respectively. The discontinuity lines of $f_\varepsilon(\mathbf{u})$ (dashed) do not align with the axes of $[0,1)^2$.

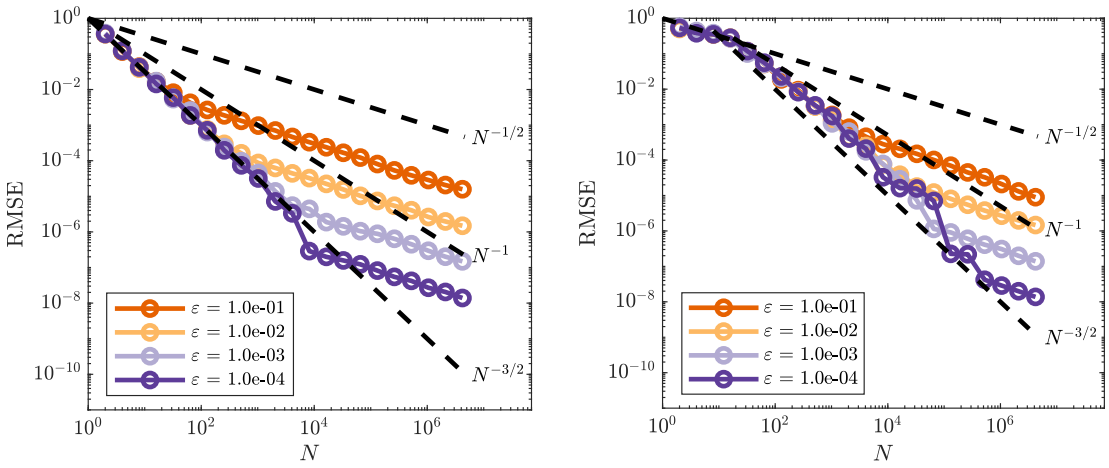
like a standard MC method and have a convergence rate which is not more than a constant times the MC rate [165].

To further explain the convergence behaviour we consider the decomposition of the discontinuous function into a continuous part, $F(\mathbf{u})$, and discontinuous part, $G(\mathbf{u})$, of the form

$$f_\varepsilon(\mathbf{u}) = \underbrace{f(\mathbf{u})}_{\text{continuous } F(\mathbf{u})} + \underbrace{(f_\varepsilon(\mathbf{u}) - f(\mathbf{u}))}_{\text{discontinuous } G(\mathbf{u})}. \tag{4.18}$$

Note that $|G(\mathbf{u})| \leq \varepsilon$ and as a result the variance of $G(\mathbf{u})$ over the hypercube is generally $\mathcal{O}(\varepsilon^2)$. We can then decompose the MSE of the estimator of the integral of $f_\varepsilon(\mathbf{u})$ using an unbiased RQMC rule as the sum of the MSE of the integration of $F(\mathbf{u})$ and $G(\mathbf{u})$. We note that the MSE for the continuous part, $F(\mathbf{u})$, behaves like $\mathcal{O}(N^{-3})$, as observed in Figure 4.8. In the case of transformation (4.17) the RQMC method achieves MC-like error rates for the discontinuous part, $G(\mathbf{u})$. We therefore have the following decomposition of the MSE

$$\text{MSE}\left(\varepsilon \left\lfloor \frac{f(\mathbf{u})}{\varepsilon} \right\rfloor\right) = C_1 N^{-3} + C_2 \varepsilon^2 N^{-1}. \tag{4.19}$$



(a) Integrand (4.14a), $d = 10$.

(b) Integrand (4.14b), $d = 3$.

Figure 4.12: Effect of discontinuity transformation (4.17) on the RMSE convergence for (4.14a) ($d = 10$) and (4.14b) ($d = 3$). $M = 128$ randomisations were used and the dashed black lines show typical reference convergence rates, $\mathcal{O}(N^{-1})$ and $\mathcal{O}(N^{-3/2})$ for RQMC and $\mathcal{O}(N^{-1/2})$ for MC.

This yields a switch from fast $\mathcal{O}(N^{-3})$ convergence to slow $\mathcal{O}(N^{-1})$ when $N = \mathcal{O}(\varepsilon^{-1})$, i.e. at this point the error made for the discontinuous component of the function starts to dominate the MSE. The same holds true for the RMSE and this scaling of the switch point as a function of ε is also observed in Figure 4.12.

In the case of transformation (4.16) the RQMC method does not perform like a standard MC method and instead achieves $\mathcal{O}(N^{-2})$ convergence for the MSE due to the mild effect of the discontinuity. Note that the scaling of the variance now does not come into play, because the convergence is not governed by the LLN. Instead we observe a rescaling of the switching point $N = \mathcal{O}(\varepsilon^{-1})$ as well in Figure 4.10. This leads to the following decomposition of the MSE

$$\mathrm{MSE}\left(f\left(\varepsilon\left\lfloor\frac{\mathbf{u}}{\varepsilon}\right\rfloor\right)\right) = C_1 N^{-3} + C_2 \varepsilon N^{-2}. \tag{4.20}$$

This shows that, even in the case of a discontinuous integrand, RQMC methods can achieve a lower MSE if the integrand can be decomposed in a continuous part and a discontinuous part with the latter relatively smaller in magnitude. RQMC

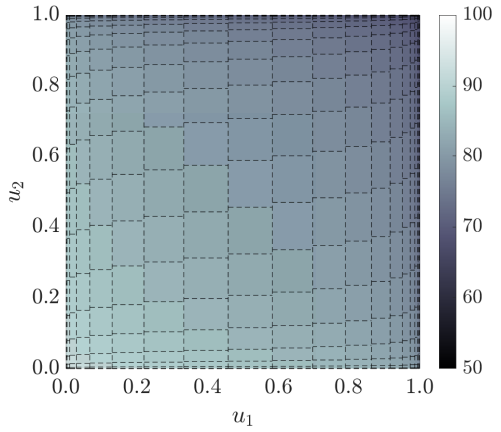
compared to MC performs superiorly on the continuous component of the integrand, giving fast error decay for a moderate number of points N. In the worst-case scenario a MC convergence rate is achieved by RQMC on the discontinuous part, which will dominate the convergence order for large N.

This observation can be linked to observations made in [32]. Caflisch notes that low-discrepancy point sets differ subtly from pseudo-random point sets in the sense that for a pseudo-random point set every point is an independent estimate of the integral. This is not true for a low-discrepancy point set, which has a deterministic structure. For these point sets the initial points sample the integration domain on a coarse scale, whereas the later points are used for progressively finer scales. Therefore initially RQMC will perform well on a function like f_ε, because on a coarse scale it is dominated by its continuous part, $F(\mathbf{u})$. If more points are used the fine, discontinuous, structure due to $G(\mathbf{u})$ starts to dominate and this is where the convergence stalls.

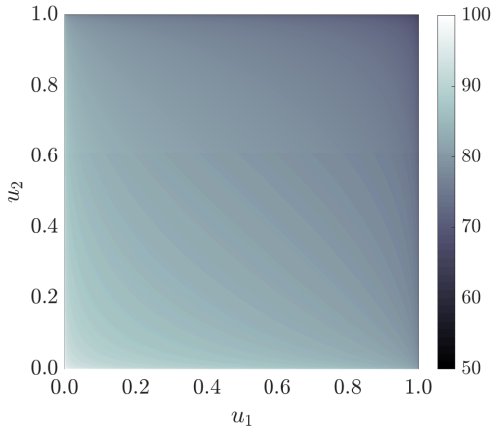
Implications for chemical reaction networks

To link the observations on the toy models to chemical reaction network simulations we use again the fact that we can write $\mathbf{X} = f(\psi(\mathbf{u}))$ with $\mathbf{u} \in [0,1)^d$, describing how we get a sample path $\mathbf{X}$ from a group of uniform random variates. For a simple decay system we show the contour plot for the output of two steps of the τ-leap method and its continuous counterpart, the Euler-Maruyama discretisation of the CLE, in Figure 4.13. We can clearly see that the smoothness of the output is the distinct difference between the two methods. The coarse scale structure of both methods aligns well and although the discontinuities for the τ-leap method are in some sense axis-aligned, the hypercubes formed by the discontinuities are not necessarily aligned or regularly shaped, which was the case for transformation (4.16). This means that we can expect RQMC methods to only achieve a RMSE of $\mathcal{O}(N^{-1/2})$ for large N, when the discontinuities dominate. To get a more precise understanding of the

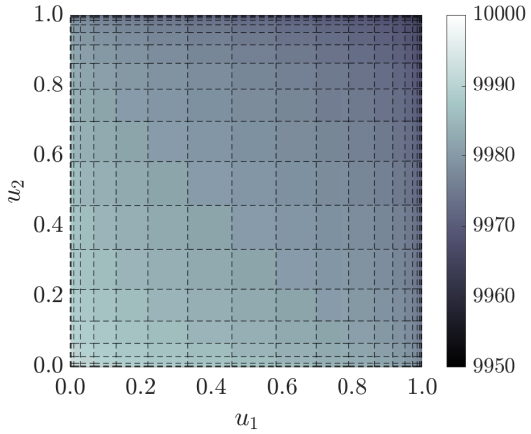
convergence behaviour of the RQMC τ-leap method we now look at the two effects of discontinuities in our discrete toy models.



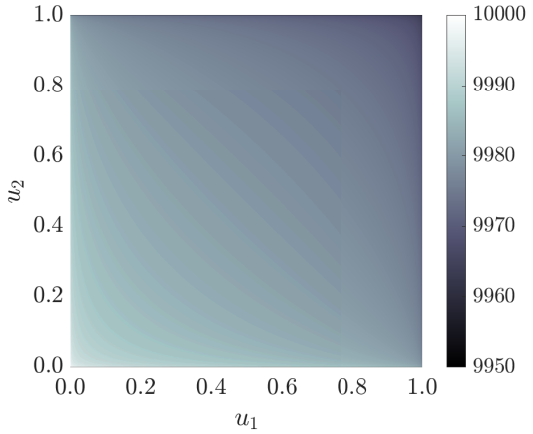
(a) τ-leap method.



(b) Euler-Maruyama discretisation of CLE.



(c) τ-leap method.



(d) Euler-Maruyama discretisation of CLE.

Figure 4.13: Stochastic simulation output for two time steps ($\tau = 0.1$) of the system, $S_1 \xrightarrow{c} \emptyset$, with (a,b) $\mathbf{X}_0 = 10^2$ and $c = 1$ and (c,d) $\mathbf{X}_0 = 10^4$ and $c = 10^{-2}$. The inputs u_1 and u_2 are used to generate Poisson and normal random variables, respectively, in both simulation methods. The CLE-based method output is a continuous function of u_1 and u_2; the τ-leap method output on the other hand is a discontinuous function of u_1 and u_2, dashed lines in (a,c) show discontinuities.

The first effect is the actual presence of discontinuities and, importantly, at which scale these discontinuities start to dominate the statistical error convergence. Many small discontinuities require more sample points to fully resolve than a few larger discontinuities and thus have a smaller effect, as illustrated in Figure 4.10. This

explains the difference between the systems in Figure 4.5, for which a decrease in ε results in a more stretched out distribution of the chemical species. A larger dispersion means that the discontinuities manifest themselves on a relatively fine scale compared to the overall continuous coarse scale and thus are only visible using a larger number of sample points. At that point the discontinuities start to dominate the statistical error and thus result in the asymptotic $\mathcal{O}(N^{-1/2})$ convergence rate.

This effect, however, does not explain the observations in Figure 4.6, where a change in ε does not change the dispersion of the resulting species distribution. The second effect to consider, therefore, is the alignment of the discontinuities with the hypercube axes. We note that as ε decreases in Figure 4.6 the change in the reaction propensities after each time step becomes smaller, effectively leading to (near) independence between the time steps. This same effect is visible when we compare Figure 4.13(a) and Figure 4.13(c). RQMC performs better on the model in Figure 4.13(c) as the discontinuities are better aligned the axes of the hypercube.

To conclude, we note that the statistical error in the RQMC method for chemical reaction networks can seemingly be decomposed as

$$\operatorname{Var}\left[\hat{Q}_{\mathrm{RQMC}}\right] = \underbrace{C_1 N^{-2}}_{\text{continuous contribution}} + \underbrace{C_2 N^{-1}}_{\text{discontinuous contribution}}, \tag{4.21}$$

with C_1 and C_2 constants that are determined by the model and the parameters used. The relative sizes of C_1 and C_2 determine the importance of the continuous and discontinuous parts, respectively. However, in general it is not clear *a priori* how the summary statistic of interest can be decomposed into a continuous part and a discontinuous part, and accordingly C_1 and C_2 are unknown. Or, in other words, it is not clear how important coarse scale continuous contributions are in relation to finer scale discrete ones. Therefore the performance benefit from using RQMC methods over MC methods can be hard to estimate *a priori*. We do, however, note that the implementation of low-discrepancy point sets is often relatively simple and

does not need to increase the runtime of the simulation procedures (Appendix 4.A). As a result, RQMC methods have the potential to provide computational savings over MC methods in the simulation of chemical reaction networks by attaining lower statistical errors for equal computational complexity and similar computational time.

4.2.3 Non-linear example

As a final example we look at the bistable Schlögl system, as encountered in [39], which incorporates non-linear interactions

$$2S_1 + S_2 \xrightarrow{c_1} 3S_1, \tag{4.22a}$$

$$3S_1 \xrightarrow{c_2} 2S_1 + S_2, \tag{4.22b}$$

$$S_3 \xrightarrow{c_3} S_1, \tag{4.22c}$$

$$S_1 \xrightarrow{c_4} S_3, \tag{4.22d}$$

where we assume that the copy numbers for S_2 and S_3 are constant and large. The initial condition for S_1 is 250 molecules. Non-dimensional parameters are given by $c_1 = 3 \cdot 10^{-7}$, $c_2 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 3.5$ and the copy numbers for S_2 and S_3 are taken as 10^5 and $2 \cdot 10^5$, respectively. The system is bistable for these parameters, with stable states around 100 copy numbers and 550 copy numbers for S_1.

We simulate the system up until final time $T = 4$ with time step $\tau = 0.25$. We take the approach in [5] to deal with sample paths with zero or fewer molecule numbers at a given time, i.e. we apply $\max(X_j, 0)$ after each time step to ensure sample paths do not become negative. First we consider the mean number of S_1 molecules, though more meaningful summary statistics can be constructed for bistable systems.

In Figure 4.14 we show results comparing the τ-leap method and Euler-Maruyama discretisation of the CLE using both pseudo-random points and low-discrepancy points. We see that, although the RQMC method does not attain a much better convergence rate than the standard MC rate of $\mathcal{O}(N^{-1/2})$, the RQMC method is su-

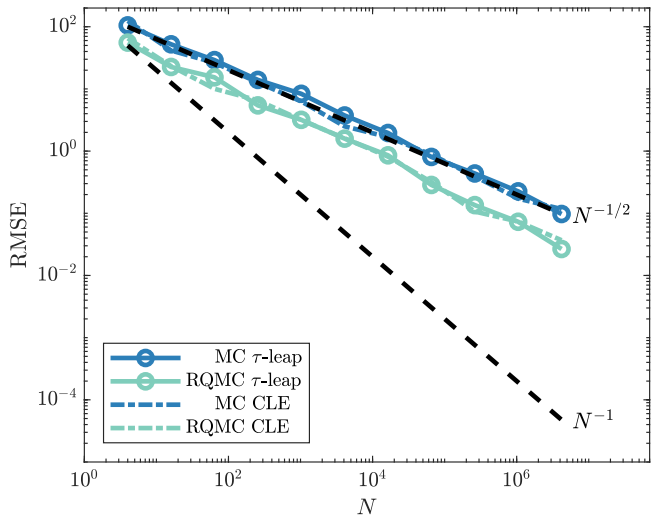


Figure 4.14: RMSE convergence for the mean number of S_1 molecules in (4.22) at time $T = 4.0$. The time step was $\tau = 0.25$ in all simulations. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 32$ randomisations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC and $\mathcal{O}(N^{-1})$ for RQMC.

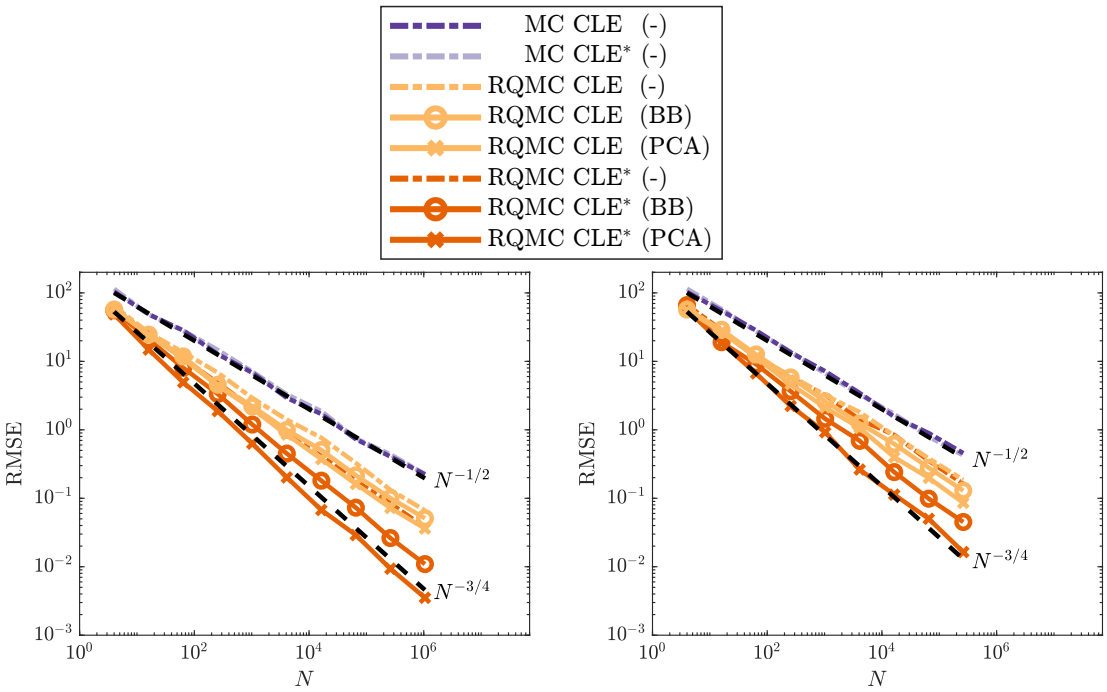
perior to the standard MC method. Numerical experiments (not shown) suggest that a similar situation as in Figure 4.14 holds for at least the first few raw moments of S_1 copy numbers.

We also observe that, even though the CLE is continuous, the convergence rate for its Euler-Maruyama discretisation is equal to that of the τ-leap method. This indicates that for this specific problem it might not be the discrete nature of the S_1 dynamics that causes the observed $\mathcal{O}(N^{-1/2})$ convergence rate. The behaviour is likely due to the fact that the system has four reaction channels and 16 time steps, leading to a dimensionality of 64 for this specific problem. Such a number of dimensions can be challenging for naïve QMC methods as applied here. One might benefit from applying a change of variables which transforms the effective dimension of the problem, and therefore improves the RQMC convergence. Techniques such as the Brownian bridge and construction via principal component analysis (PCA) are available for SDEs that can help in making RQMC methods effective even for high-dimensional problems [84, Chapter 5].

We briefly explore the use of these two methods, and of the alternative representation of the CLE as given in equation (2.12), as a means to reduce the effective dimension for the Schlögl system. We compare a coarse and a fine time-discretisation to see how the dimensionality reduction techniques perform when the (original) problem dimension changes. The PCA construction of the Wiener processes driving the CLE is done via a fast discrete sine transform [130], whereas we use the classical Brownian bridge construction [156]. A computational implementation of the alternative representation of the CLE can be achieved via a Cholesky or singular value decomposition of the relevant (species) covariance matrices, $\mathbf{C}\mathbf{C}^\intercal$ in equation (2.12). However, in the case of the Schlögl system the covariance matrix $\mathbf{C}\mathbf{C}^\intercal$ is in fact scalar and no decomposition is therefore required in practice. The results of the various combinations of these techniques is shown in Figure 4.15.

We can see that using the different methods to construct the Brownian motion driving the CLE sample paths can improve the decay rate of the statistical error for the CLE, both for small and large time steps. In addition, we see that using the alternative representation (denoted by CLE* in Figure 4.15) which results in a lower-dimensional SDE (computational complexity $\mathcal{C} = T/\tau$ for the alternative representation versus $\mathcal{C} = 4T/\tau$ for the standard representation), also yields improved convergence for this problem. The combination of the two methods, i.e. the alternative representation with either the Brownian bridge construction or PCA construction, performs best, though the convergence rate is at most $\mathcal{O}(N^{-3/4})$ rather than $\mathcal{O}(N^{-1})$ which we observed for examples in Section 4.2.1. Generally speaking the alternative representation of the CLE has lower computational complexity if the number of species is smaller than the number of reactions, but will involve extra computational overhead due to the extra operations needed, such as the computation of a Cholesky or singular value decomposition. It is therefore dependent on the specific problem and implementation which method should be preferred in practice.

In conclusion, problems with larger (effective) dimension see less benefit from



(a) $\tau = 0.25$ (2^4 time steps). (b) $\tau = 0.015625$ (2^8 time steps).

Figure 4.15: RMSE convergence for the mean number of S_1 molecules in (4.22) at time $T = 4.0$. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 128$ randomisations. Dashed black lines show the typical reference convergence rates. We use standard RQMC in combination with the regular CLE (2.11) or the alternative representation CLE* (2.12). Wiener processes driving the SDE are constructed using either the standard sequential sampling (-), via a Brownian bridge construction (BB) or via a PCA construction (PCA).

using standard RQMC methods. For the CLE model of chemical reactions we can leverage (effective) dimension reduction techniques to improve the convergence rate of such RQMC methods. Equivalent transformations for chemical reaction dynamics following the RTCR (2.7) are, however, not known. An interesting recent development is the Poisson bridge method that we will discus in more detail in Chapter 5. In many respects this method parallels the Brownian bridge method for SDEs and is therefore a good candidate transformation to reduce the effective dimension. We leave this direction for future research.

4.3 Array-RQMC for chemical reactions

The previous section purely considered standard RQMC applied to the τ-leap method and the Euler-Maruyama discretisation of the CLE. However, it is known that the performance benefit of standard RQMC methods can suffer badly as the dimensionality of the problem is increased, as was observed for the Schlögl system in Section 4.2.3. In the context of chemical reaction network models, and the τ-leap method in particular, the dimensionality is equal to the computational complexity and this will generally be large for many problems of interest.

One possible remedy for this problem in certain scenarios is to use the array-RQMC method [119]. As discussed in Section 4.1.2 we can use this method to simulate DTMCs. Even though the CME describes a CTMC we note that the τ-leap approach yields a DTMC and we therefore explore the combination of the τ-leap method and array-RQMC in Section 4.3.1. In addition, the array-RQMC method has already been applied to Gillespie's DM in combination with uniformisation [95] and this combination will be studied further in Section 4.3.3.

Note that in order for the array-RQMC method to work we need a sorting method for the sample paths at each stage of the DTMC. In this section we use a simple strategy, namely a component-wise sorting method, which implicitly defines an importance function. This means we sort the sample paths based on the value of S_1 first. Paths with equal values for S_1 are then further sorted based on S_2 etc. This approach was also taken in [95] where it was noted that the performance of this method can vary when changing the sorting order.

4.3.1 Array-RQMC and the τ-leap method

By using the array-RQMC method, rather than standard RQMC, in combination with the τ-leap method we reduce the dimensionality of the QMC point set needed for simulations from KT/τ to K, where we recall that K is the number of reaction

channels. In many realistic scenarios $T/\tau \gg 1$ and therefore using array-RQMC instead of standard RQMC can drastically reduce the dimension of the QMC point sets used. As the quality of QMC point sets is generally better for smaller dimensions [162, Section 15.8] this can potentially significantly improve the variance reduction that we achieve. Algorithm 4.1 depicts the combination of array-RQMC and the τ-leap method to generate an ensemble of N sample paths.

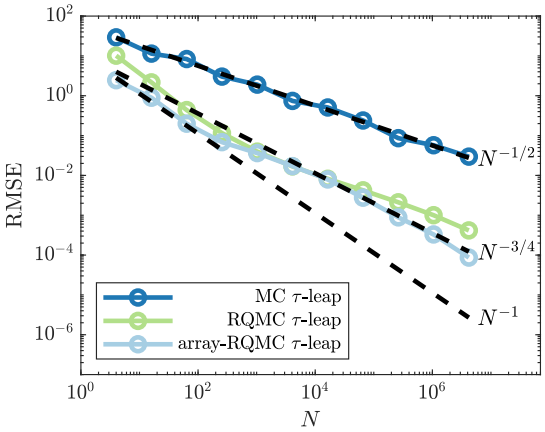
Algorithm 4.1 Array-RQMC τ-leap method.
This simulates an ensemble of N sample paths.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Time step τ
Input: Final time T
Input: Importance function $h(\mathbf{X})$

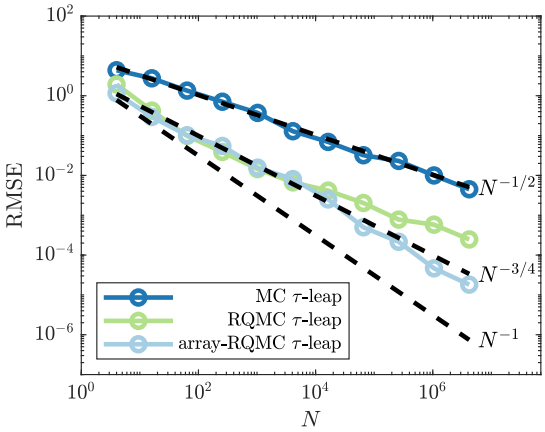
$\mathbf{X}^{(n)} \leftarrow \mathbf{X}_0$ for $n = 1, \ldots, N$ ▷ Initialise ensemble of sample paths.
$t \leftarrow 0$
while $t < T$ **do**
    Sort and renumber the sample paths based on their importance function value, s.t. $h(\mathbf{X}^{(1)}) \leq \cdots \leq h(\mathbf{X}^{(N)})$
    Sample $\tilde{\mathbf{v}}^{(n)}$ low-discrepancy points in $[0,1)^K$ for $n = 1, \ldots, N$
    for $n = 1, \ldots, N$ **do**
        $a_k \leftarrow a_k(\mathbf{X}^{(n)})$ ▷ Calculate reaction propensities.
        Generate $Y_1, \ldots, Y_K$ Poisson random variables via inverse transform sampling using $\tilde{\mathbf{v}}^{(n)}$, s.t. $Y_k \sim \mathcal{P}(a_k \tau)$
        $\mathbf{X}^{(n)} \leftarrow \mathbf{X}^{(n)} + \sum_{k=1}^{K} Y_k \boldsymbol{\zeta}_k$ ▷ Update state vector.
    end for
    $t \leftarrow t + \tau$ ▷ Update time.
end while

Monomolecular reaction networks

First we revisit the examples from Section 4.2.1, where we now use the τ-leap method both with standard MC, RQMC and the array-RQMC method depicted in Algorithm 4.1. The results are shown in Figure 4.16 and show that the array-RQMC method is superior compared to the more direct combination of RQMC methods and the τ-leap approach, which we previously considered in Section 4.2.



(a) RMSE convergence for the mean number of S_1 molecules in the linear birth-death system, (4.12), with $c = 1$ and initial condition $\mathbf{X}_0 = 10^3$ at $T = 1.6$.



(b) RMSE convergence for the mean number of S_1 molecules in the isomerisation system (4.13) with $c = 1$, $r = 10^{-4}$ and initial condition $\mathbf{X}_0 = 10^6(1 - r, r)^\intercal$ at $T = 1.6$.

Figure 4.16: Benefit from using array-RQMC over standard RQMC (compare with Figures 4.4 and 4.7). The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC, $\mathcal{O}(N^{-3/4})$ for array-RQMC and $\mathcal{O}(N^{-1})$ for RQMC. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 32$ randomisations.

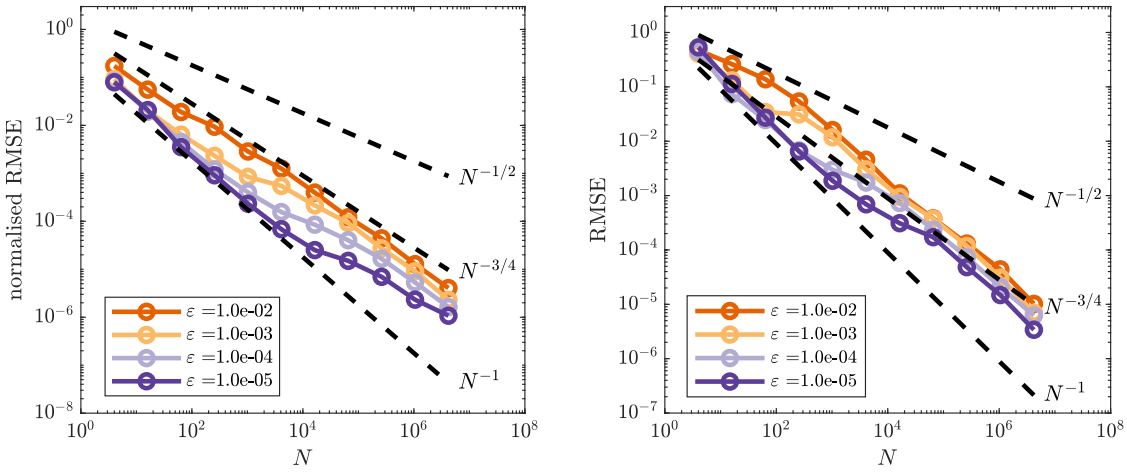
In particular, we note that for a moderate number of sample paths, N, the differences between the two methods are small and both methods initially converge at an $\mathcal{O}(N^{-1})$ rate. Whilst for both methods this convergence rate starts to drop when we consider more sample paths, N, the array-RQMC method clearly excels for large number of sample paths. In the regime of many sample paths, N, the RQMC convergence is hampered by the discontinuities due to the discrete state space of chemical species and therefore behaves like a standard (reduced variance) MC method, $\mathcal{O}(N^{-1/2})$, as discussed in the previous section. The array-RQMC method, however, is better suited to deal with such discontinuities and attains a $\mathcal{O}(N^{-3/4})$ convergence rate. This latter convergence rate is equal to the best current proven result for array-RQMC, see for example [119, Proposition 6] which holds only for a one-dimensional DTMC, though much better results for smooth problems have also been observed in practice [122].

As in Section 4.2.1 we also look at the effect of the discreteness of the state space of chemical species on the array-RQMC convergence. In a similar setup as Section 4.2.1

we tune the relative importance of the continuous and discontinuous components of the output and the results are shown in Figure 4.17. We can see that for these systems, similar to equation (4.21) for the standard RQMC method, we can write

$$\mathrm{Var}\left[\hat{Q}_{\text{array-RQMC}}\right] = \underbrace{\tilde{C}_1 N^{-2}}_{\text{continuous contribution}} + \underbrace{\tilde{C}_2 N^{-3/2}}_{\text{discontinuous contribution}}, \tag{4.23}$$

where $\tilde{C}_1$ and $\tilde{C}_2$ depend on the specific model and parameters. Comparing equations (4.21) and (4.23) we conclude that the standard RQMC method achieves similar results for the continuous contribution, but the array-RQMC method does a better job at reducing the statistical error for the discontinuous contribution.



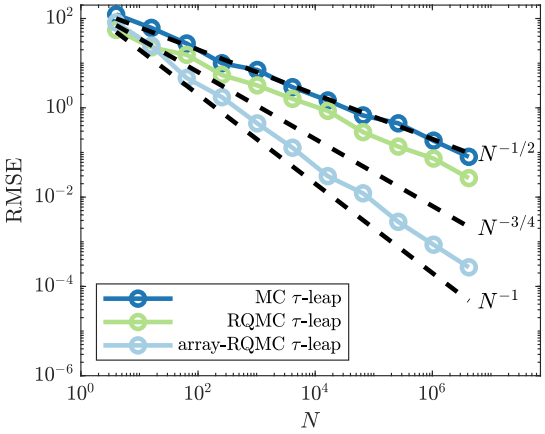
(a) Normalised RMSE convergence for the mean number of S_1 molecules in the linear birth-death system, (4.12), with $c = 1$ and $\mathbf{X}_0 = \varepsilon^{-1}$ at $T = 1.6$.

(b) RMSE convergence for the mean number of S_1 molecules in the linear birth-death system, (4.12), with $c = 10\varepsilon$ and $\mathbf{X}_0 = \varepsilon^{-1}$ at $T = 1.6$.

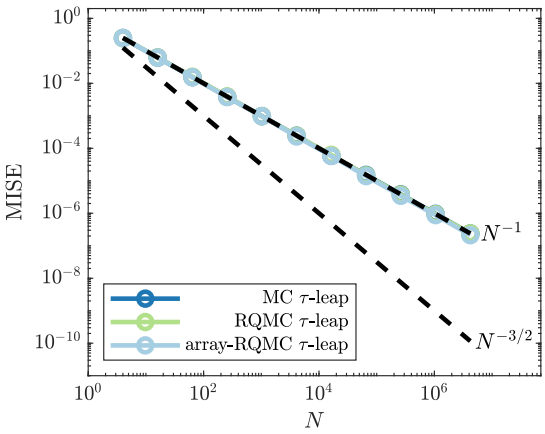
Figure 4.17: Benefits from using array-RQMC over standard RQMC (compare with Figures 4.5 and 4.6). The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates, $\mathcal{O}(N^{-1/2})$ for MC, $\mathcal{O}(N^{-3/4})$ for array-RQMC and $\mathcal{O}(N^{-1})$ for RQMC. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 32$ randomisations.

Schlögl system

Finally we revisit the non-linear example of the Schlögl system from Section 4.2.3. When estimating the mean number of S_1 molecules in the system using the τ-leap method we observe in Figure 4.18(a) again that the array-RQMC method yields a substantial improvement over both the standard RQMC and MC approaches. Part of the reason for the superior performance of the array-RQMC method over the standard RQMC method is the dimensionality of the point sets used in each method, four and 64, respectively. The dimensionality of the Schlögl system is small for the array-RQMC method, resulting in fast decay of the statistical error, but moderate for the standard RQMC method.



(a) RMSE convergence for the mean number of S_1 molecules in the Schlögl system, (4.22), at $T = 4.0$. Parameters described in Section 4.2.3.



(b) MISE convergence for the EDF of S_1 molecules in the Schlögl system, (4.22), at $T = 4.0$. Parameters described in Section 4.2.3.

Figure 4.18: Benefits from using array-RQMC over standard RQMC and MC methodologies (compare with Figure 4.14). The time step was $\tau = 0.25$ in all simulations. Dashed black lines show the typical reference convergence rates. To establish the RMSE and MISE for the RQMC methods equation (4.5) was used with $M = 32$ randomisations.

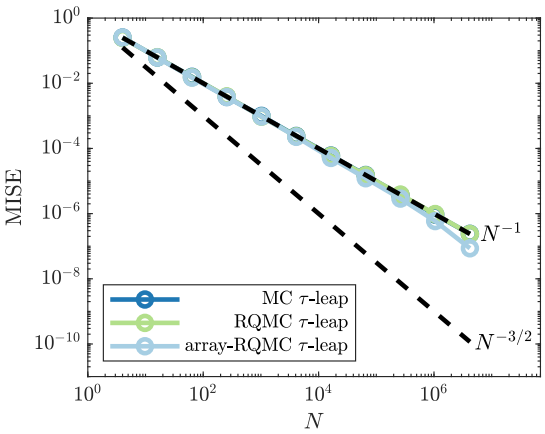
So far the observations have only been for raw moment summary statistics and we now consider what happens if instead we try to estimate the distribution of sample paths, either using the EDF or a histogram.

Distribution estimation

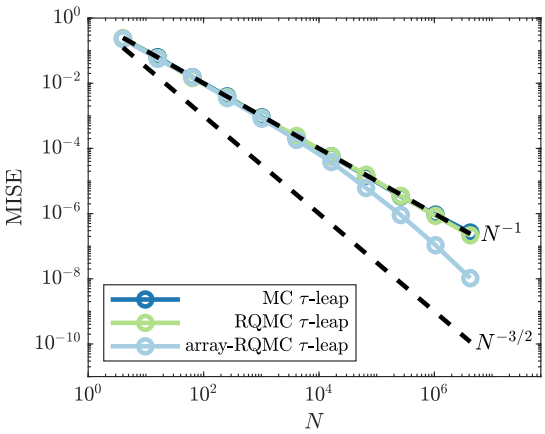
For the bistable Schlögl system it is more meaningful to look at the distribution of S_1 molecules than the mean copy number. Using Section 2.3.1 we construct the EDF for the output of the τ-leap method using standard MC, RQMC and the array-RQMC methods. If we solely consider the statistical error in the resulting EDFs we see in Figure 4.18(b) that there is no difference between the three methods. This could come as a surprise based on the results in Figure 4.18(a) which show that the array-RQMC method clearly outperforms the other two methods when looking at the mean of the distribution.

For comparison we also consider the problem of estimating the distribution of S_1 molecules in the monomolecular examples from Section 4.2.1 and the resulting decay of the statistical error is depicted in Figure 4.19. We see that for these two systems, at least for a large number of sample paths, N, the array-RQMC method outperforms the other two methods, though the benefit from using array-RQMC (or RQMC) seems to be smaller than when trying to estimate the mean. This signifies that observations on efficiency of a method based on a mean copy number summary statistic do not necessarily carry over directly when considering a different summary statistic, such as the probability distribution.

We can understand these observations by considering the width, in the sense of the (effective) support, of the actual distributions we are trying to estimate. The distribution for S_1 molecules in the Schlögl distribution is wider than its equivalent in the linear birth-death system, which in turn is wider than the distribution for S_1 molecules in the isomerisation system. Loosely speaking to get an accurate EDF we need at least a few observations for each likely value in the distribution. As a result we can only start to see the benefits from using the RQMC or array-RQMC method when we have a number of sample paths, N, which is at least an order of magnitude larger than the width of the distribution we are trying to estimate. If we use fewer points than this, the statistical error will be dominated by the effect of not being



(a) MISE convergence for EDF of S_1 molecules in the linear birth-death system, (4.12), with $c = 1$ and $\mathbf{X}_0 = 10^3$ at $T = 1.6$.

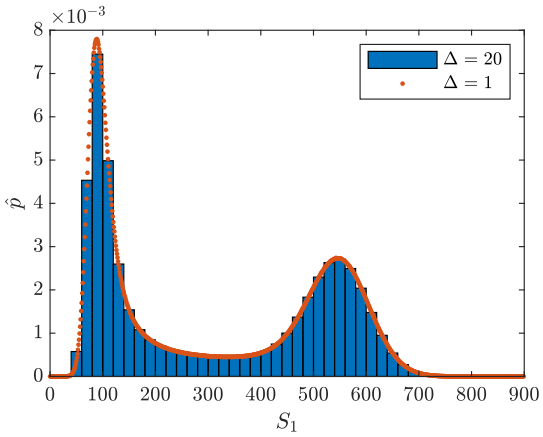


(b) MISE convergence for EDF of S_1 molecules in the isomerisation system (4.13) with parameters $c = 1$, $r = 10^{-4}$ and initial condition $\mathbf{X}_0 = 10^6(1 - r, r)^\intercal$ at $T = 1.6$.

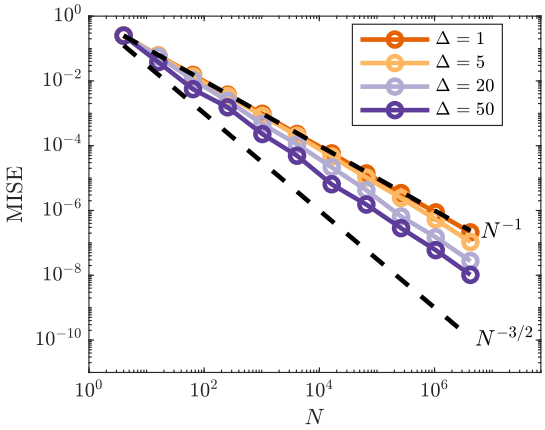
Figure 4.19: Benefits from using array-RQMC over standard RQMC and MC methodologies. The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates. To establish the MISE for the RQMC methods equation (4.5) was used with $M = 32$ randomisations.

able to generate a fully representative collection of samples. This is true for all three methods and their convergence rate will therefore not differ for small N, i.e. their MISEs converge at the standard MC rate of $\mathcal{O}(N^{-1})$.

To illustrate this point further we consider the problem of estimating the distribution of S_1 via an (equal bin width) histogram instead of the EDF. By increasing the width of the histogram bins, Δ, we note that the number of bins, whose height we aim to estimate, becomes smaller. This means that to generate a representative sample, i.e. proportional to the histogram bin heights, requires fewer sample paths. Therefore we can get an accurate estimate of the histogram bin heights using fewer samples, N. The EDF is recovered when we take $\Delta = 1$ and we therefore expect to see the benefit from using array-RQMC earlier if we take $\Delta > 1$. This is indeed the case as we can see for the Schlögl system in Figure 4.20. Note that the decay rate of the MISE for large numbers of sample paths, N, using the array-RQMC method is slower than $\mathcal{O}(N^{-3/2})$, in contrast to the raw moment summary statistic case discussed previously.



(a) Histogram with fixed bin width, Δ, using $N = 2^{22}$ samples for the array-RQMC method in combination with the τ-leap method.



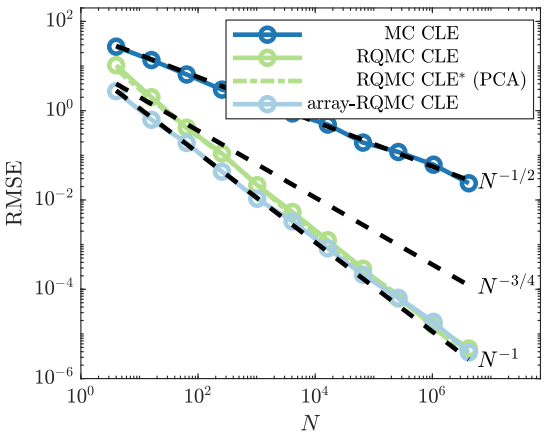
(b) MISE convergence for histograms with fixed bin width, Δ.

Figure 4.20: Histogram for the number of S_1 molecules in the Schlögl system, (4.22), and the accompanying statistical error convergence when combining array-RQMC with the τ-leap method. System parameters are as described in Section 4.2.3 and $T = 4.0$. The time step was $\tau = 0.25$ in all simulations. Dashed black lines show the typical reference convergence rates. To establish the MISE equation (4.5) was used with $M = 32$ randomisations.

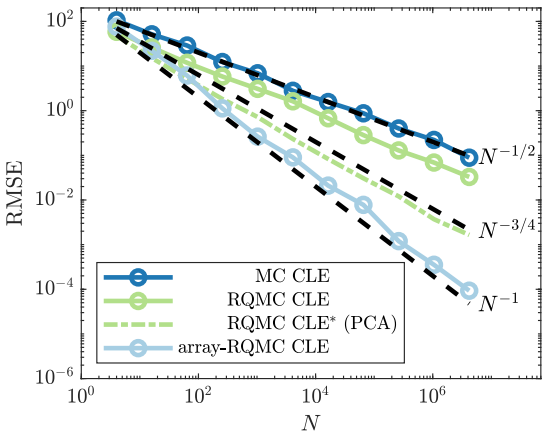
4.3.2 Array-RQMC and CLE discretisation

If we instead consider the use of the Euler-Maruyama discretisation of the CLE and its combination with the array-RQMC method we might expect further improvements due to the fact that the state space is now continuous. The algorithmic depiction of this approach follows simply by replacing step 8 in Algorithm 4.1 with the generation of $Y_1, \ldots, Y_K$ normal random variables so that $Y_k \sim \mathcal{N}(a_k\tau, a_k\tau)$. When the summary statistic is the mean number of S_1 molecules we indeed see in Figure 4.21 that the array-RQMC method in this case can attain $\mathcal{O}(N^{-1})$ RMSE decay, which is on par with the standard RQMC method for simple systems such as the monomolecular systems from Section 4.2.1.

In addition we see that for the Schlögl system the array-RQMC method is superior compared to standard RQMC methods applied to the CLE, both in the original



(a) RMSE convergence for the mean number of S_1 molecules in the linear birth-death system, (4.12), with $c = 1$ and $\mathbf{X}_0 = 10^3$ at $T = 1.6$.



(b) RMSE convergence for the mean number of S_1 molecules in the Schlögl system, (4.22), at $T = 4.0$. Parameters described in Section 4.2.3.

Figure 4.21: Benefits from using array-RQMC over standard RQMC and MC methodologies when using the Euler-Maruyama discretisation of the CLE to generate sample paths. The method RQMC CLE* (PCA) uses the alternative representation of the CLE given in equation (2.12) with the PCA construction for the Wiener processes driving the SDE (see Section 4.2.3 for more details). The time step was $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates. To establish the RMSE for the RQMC methods equation (4.5) was used with $M = 64$ randomisations.

formulation, given in equation (2.11), and the lower dimensional equivalent, given in equation (2.12). Even when using the PCA construction of the Wiener process driving the CLE, which we found to have the smallest error of the methods considered in Section 4.2.3, we find a $\mathcal{O}(N^{-3/4})$ decay rate, which is slower than what we observe for array-RQMC. We can attribute this to the fact that the dimensionality of the low-discrepancy point sets used for the array-RQMC method is lower ($d = 4$ in this case) compared to the standard RQMC method ($d = 80$ and $d = 20$ in this case for the CLE and CLE*, respectively). Note that the array-RQMC can also be used in conjunction with the alternative representation of the CLE in equation (2.12) to further reduce the dimensionality, though we do not explore this avenue further here.

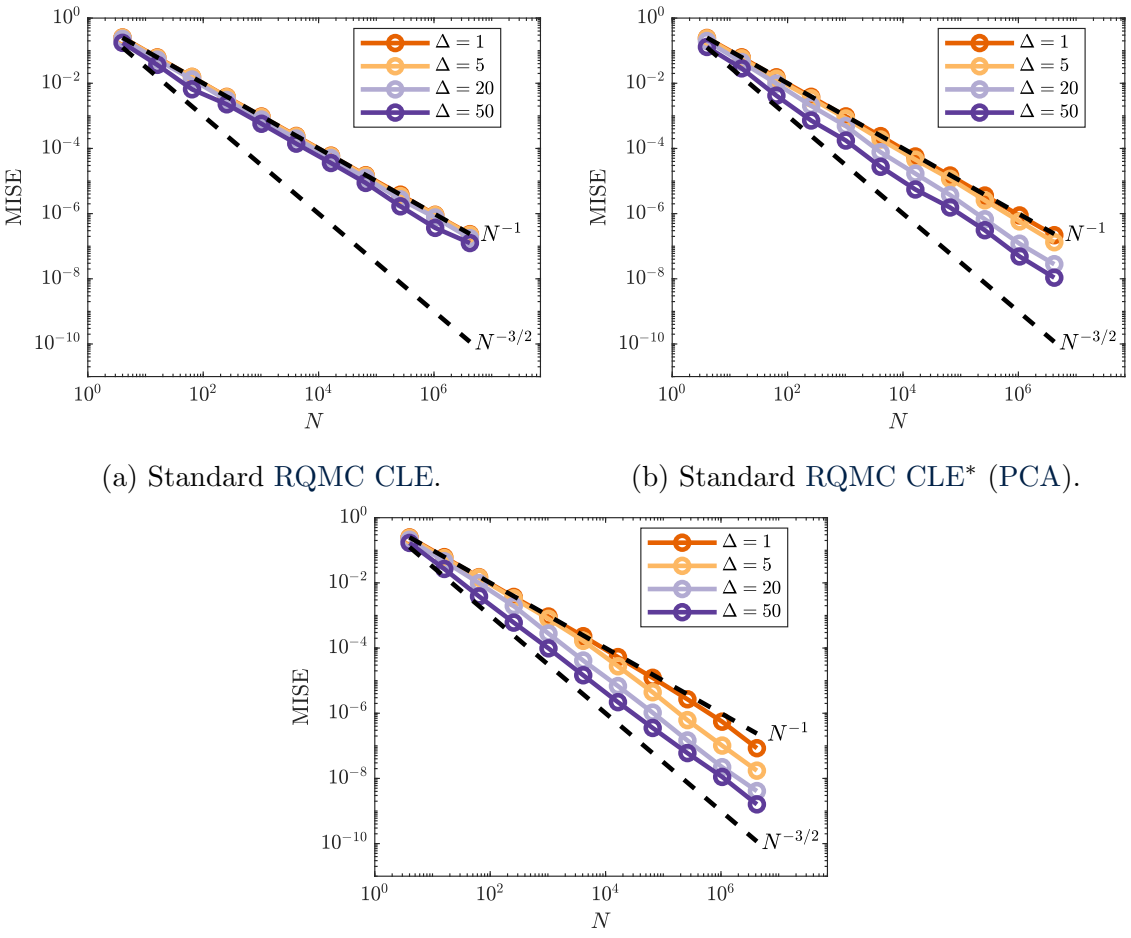
Density estimation

Based on the discussion for the mean summary statistic one might be tempted to consider the standard RQMC methods equally as effective as the array-RQMC method when the dimensionality of the low-discrepancy point set used is small. This, however, is not true if we consider the estimation of the density of the sample paths using histograms of fixed width, Δ. Note that in this case $\Delta = 1$ does not correspond to the EDF due to the fact that the state space for the CLE sample paths is continuous.

As we can see in Figure 4.22 there is a clear difference between the MISE decay rate for the array-RQMC and standard RQMC methods. The former attains a decay rate that is faster than $\mathcal{O}(N^{-1})$ for large enough number of sample paths, N. This is not true for the standard RQMC method applied to the original CLE, despite the fact that when considering the mean summary statistic both methods have an equal statistical error decay. Using the alternative formulation of the CLE (2.12) in combination with a PCA construction of the driving Wiener process also improves the convergence of the standard RQMC method when estimating densities, but is still inferior compared to array-RQMC. We also note that for all methods the MISE decay rate is slower than $\mathcal{O}(N^{-2})$, which is the MSE decay rate when estimating the mean, and this is likely to be caused by the discontinuous nature of the histogram summary statistic. This means that also when using the Euler-Maruyama discretisation of the CLE to generate sample paths, now from a continuous state space, there is a benefit in using the array-RQMC method over the standard RQMC method as described in Section 4.2.

4.3.3 Array-RQMC and uniformisation

One of the two QMC papers in the context of chemical reactions [95] uses uniformisation to convert the CTMC that describes the sample paths for Gillespie's DM into a DTMC in order to be able to use the array-RQMC method. It was observed in [95]



(a) Standard RQMC CLE.

(b) Standard RQMC CLE* (PCA).

(c) Array-RQMC CLE.

Figure 4.22: Statistical error convergence of histograms with fixed bin width, Δ, for the number of S_1 molecules at $T = 1.6$ in the linear birth-death system, (4.12), when using the Euler-Maruyama discretisation of the CLE to generate sample paths. The method RQMC CLE* (PCA) uses the alternative representation of the CLE given in equation (2.12) with the PCA construction for the Wiener processes driving the SDE (see Section 4.2.3 for more details). The parameter $c = 1$, initial condition $\mathbf{X}_0 = 10^3$ and the time step $\tau = 0.2$ in all simulations. Dashed black lines show the typical reference convergence rates. To establish the MISE equation (4.5) was used with $M = 64$ randomisations.

that a large variance reduction could be achieved using this approach, even without the addition of array-RQMC, i.e. by combining uniformisation with standard MC. This resulting method is, in essence, the wUDM introduced in Section 3.4 and in Section 3.4.2 we indeed found that it outperforms the DM in certain scenarios.

Here we revisit these methods and the examples provided in [95] to i) comment on their validity and ii) with the combined results from Chapter 3 and Chapter 4 provide

an explanation for the observed convergence rates. In [95] the typical application was the approximation of a probability distribution function (PDF) by use of an EDF, but we will also consider the problem of moment estimation and histogram construction. Just as in [95] we will compare three methods: i) standard MC with Gillespie's DM; ii) weighted uniformisation (see Section 3.4 for the wUDM); and iii) weighted uniformisation combined with array-RQMC (see Algorithm 4.2).

Following [95] we introduce an extra algorithm parameter, $\eta \in [0, 1]$, for the array-RQMC method which tunes how many steps in the wUDM are generated with low-discrepancy sequences[3]. Due to the overhead from i) sorting and re-numbering states based on an importance function and ii) the generation of low-discrepancy points, decreasing η means the run-time of the array-RQMC variant of the wUDM will increase. In all examples that follow we use the algorithm parameters $\varepsilon_L = \varepsilon_R = 10^{-10}/2$. Unless stated otherwise we also use the algorithm parameter $\eta = 0.9$ and we sort the states with a component-wise sorting method using the natural order $S_1, S_2, \ldots$.

Isomerisation

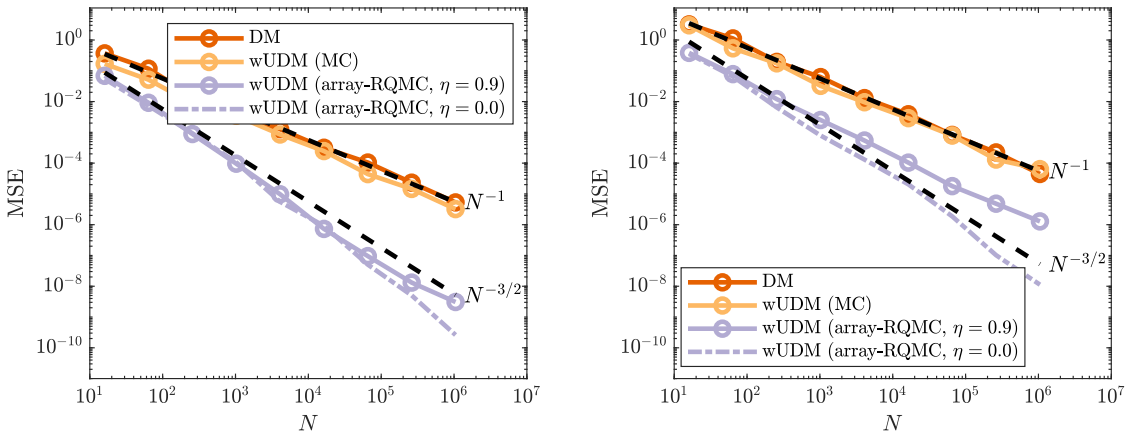
First we revisit a canonical example we encountered previously in Sections 3.3.2 and 3.4.2 when studying uniformisation techniques, namely the isomerisation system, (3.6). Note that due to the one-dimensional nature of this system the sorting and ordering of the states, needed for the array-RQMC method, is unambiguous. In Section 3.4.2 we already showed that the standard MC implementation of the wUDM can be more efficient than the standard DM. The benefit, however, was relatively small when the summary statistic of interest was the average copy number of S_1 molecules. In Figure 4.23 we see that this is indeed the case if we only consider the MC implementation of the wUDM. However, superior results are achieved when we combine array-RQMC and the wUDM.

[3] In [95] it seems $\eta \gtrsim 0.94$ is chosen, though no justification for this choice is given.

Algorithm 4.2 Array-RQMC weighted uniformised direct method (wUDM).
This simulates an ensemble of N sample paths.

Input: Initial data $\mathbf{X}_0$
Input: Stoichiometric matrix $\boldsymbol{\zeta}$
Input: Propensity functions $a_k(\mathbf{X})$
Input: Final time T
Input: Uniformisation rate $\bar{a}$
Input: Importance function $h(\mathbf{X})$
Input: Algorithm parameters $\varepsilon_L, \varepsilon_R$ and η.
Compute largest M_L s.t. $\sum_{m \leq M_L} \mathcal{P}(m, \bar{a}T) < \varepsilon_L$.
Compute smallest M_R s.t. $1 - \sum_{m \leq M_R} \mathcal{P}(m, \bar{a}T) < \varepsilon_R$.
$\mathbf{X}^{(n)} \leftarrow \mathbf{X}_0 \quad$ for $n = 1, \ldots, N$ ▷ Initialise ensemble of sample paths.
$m \leftarrow 0$
while $m < M_R$ **do**
    if $m \geq \eta M_L$ **then**
        Sort and renumber the sample paths based on their importance function value, s.t. $h(\mathbf{X}^{(1)}) \leq \cdots \leq h(\mathbf{X}^{(N)})$
        Sample $\tilde{v}^{(n)}$ low-discrepancy points in $[0, 1)$ for $n = 1, \ldots, N$
    else
        Sample $\tilde{v}^{(n)}$ pseudo-random points in $[0, 1)$ for $n = 1, \ldots, N$
    end if
    for $n = 1, \ldots, N$ **do**
        $a_k \leftarrow a_k(\mathbf{X}^{(n)})$ ▷ Calculate real reaction propensities.
        $a_{K+1} \leftarrow \bar{a} - \sum_{k=1}^{K} a_k$ ▷ Calculate virtual reaction propensity.
        Find p such that $\sum_{k=1}^{p-1} a_k < \bar{a}\tilde{v}^{(n)} \leq \sum_{k=1}^{p} a_k$ ▷ Choose next reaction to fire.
        if $p \in \{1, \ldots, K\}$ **then** ▷ Only need to fire real reactions.
            $\mathbf{X}^{(n)} \leftarrow \mathbf{X}^{(n)} + \boldsymbol{\zeta}_p$ ▷ Update state vector.
        end if
    end for
    if $m \geq M_L$ **then**
        Compute summary statistic $\hat{Q}_m$ after m reactions using $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(N)}$ and add to pooled estimator via equation (3.15).
    end if
    $m \leftarrow m + 1$ ▷ Update reaction count.
end while

Interestingly we see by comparing Figures 4.23(a) and 4.23(b) that increasing the total number of molecules in the system decreases the benefits from using the wUDM array-RQMC version with $\eta = 0.9$. We attribute this observation to the fact that if we increase the total number of molecules we also have to increase the uniformisation rate. As a result more steps in the wUDM are executed with pseudo-random points,



(a) $\mathbf{X}_0 = (0, 30)^\intercal$. (b) $\mathbf{X}_0 = (0, 300)^\intercal$.

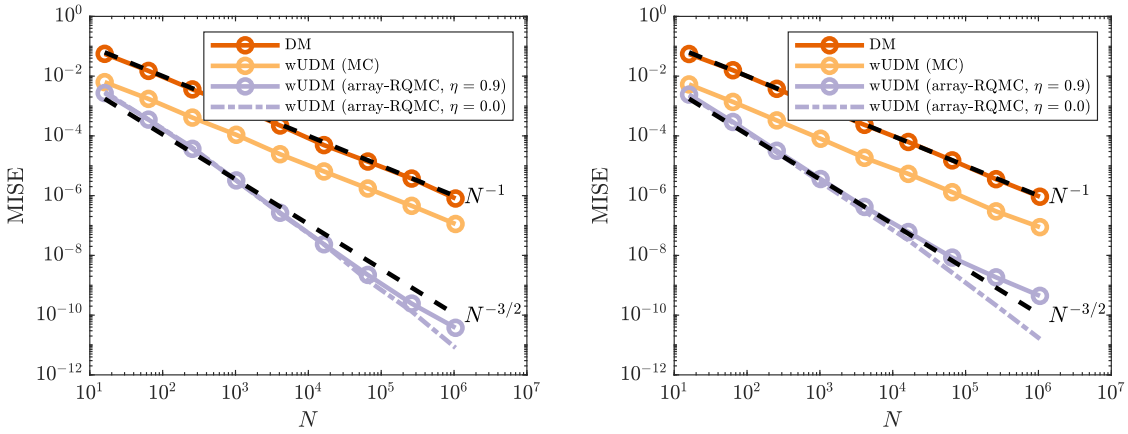
Figure 4.23: Comparison of the wUDM and the standard DM for the isomerisation system, (3.6), with parameters $c_1 = 0.3$ and $c_2 = 0.1$. We consider both the array-RQMC implementation and the standard MC implementation of the wUDM with uniformisation rates (a) $\bar{a} = 9$ and (b) $\bar{a} = 90$. Summary statistic is the average copy number of S_1 molecules at final time $T = 20$. Dashed black lines show the typical reference convergence rates. To establish the MSE equation (4.5) was used with $M = 64$ randomisations.

rather than low-discrepancy points (see steps 6-11 in Algorithm 4.2). This therefore yields a component of the error that converges at the standard MC error rate, $\mathcal{O}(N^{-1})$. Intuitively the dominant contribution to statistical error of the wUDM stems from when the number of reactions, m, in Algorithm 4.2 satisfies $M_L \leq m \leq M_R$, which is always handled with low-discrepancy points. However, as can be seen in Figure 4.23, the statistical error in the state after M_L reactions, the first state that is added to the filtered estimator in step 21, can be non-negligible, especially if M_L takes large values and $\eta \approx 1$. We note, however, that when $\bar{a} = 9$ and $\bar{a} = 90$ the computational complexity[4] $\mathcal{C}[\hat{Q}_{\text{wUDM}}] \approx 1.5 \cdot \mathcal{C}[\hat{Q}_{\text{DM}}]$ and $\mathcal{C}[\hat{Q}_{\text{wUDM}}] \approx 1.2 \cdot \mathcal{C}[\hat{Q}_{\text{DM}}]$, respectively. This means that in both scenarios the array-RQMC version of the wUDM, regardless of the choice of $\eta \in [0, 1]$, is still more efficient than the standard DM.

If we change the objective to the estimation of the distribution of S_1 molecules we see in Figure 4.24 that the wUDM yields a larger improvement over the standard

[4]The computational complexity of the DM is estimated via stochastic simulation as described in Example 2.4 whenever no analytic expression is available. The computational complexity of the wUDM is given by M_R in Algorithm 4.2 and can be determined using $\bar{a}$, T and the value of ε_R.

DM, confirming our earlier observations in Section 3.4.2. We also see that errors for the array-RQMC version of the wUDM decay at a higher rate than the MC methods considered and therefore conclude that for this problem the array-RQMC method is, by far, the most efficient method. The effect of the choice of the algorithm parameter η on the MISE is visible as well, albeit less pronounced than in the previous case of estimating the mean copy number.



(a) $\mathbf{X}_0 = (0, 30)^\intercal$. (b) $\mathbf{X}_0 = (0, 300)^\intercal$.

Figure 4.24: Comparison of the wUDM and the standard DM for the isomerisation system, (3.6), with parameters $c_1 = 0.3$ and $c_2 = 0.1$. We consider both the array-RQMC implementation and the standard MC implementation of the wUDM with uniformisation rates (a) $\bar{a} = 9$ and (b) $\bar{a} = 90$. Summary statistic is the distribution of S_1 molecules at final time $T = 20$. Dashed black lines show the typical reference convergence rates. To establish the MISE equation (4.5) was used with $M = 64$ randomisations.

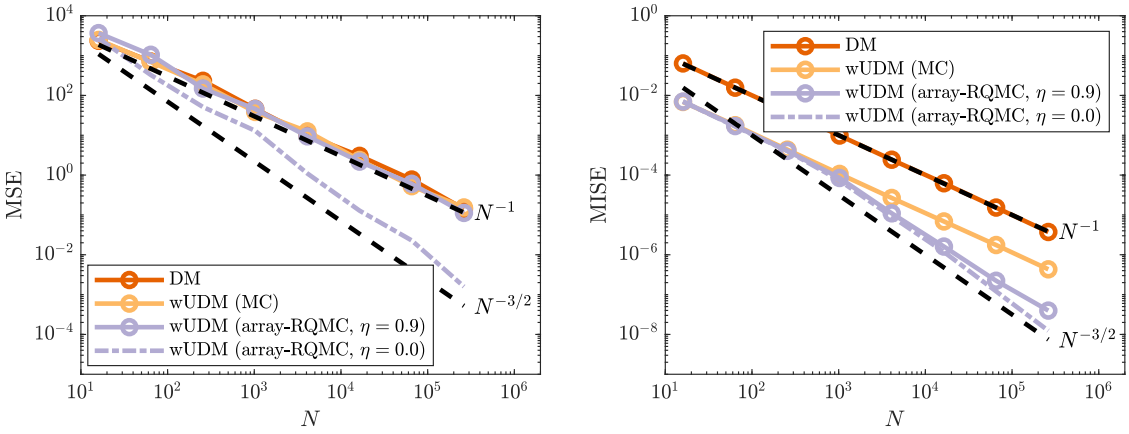
Schlögl system

Next we revisit the bistable Schlögl system from Section 4.2.3. This system may be less well-suited to an exact SSA approach due to the many reactions taking place in the system within a unit of time and the τ-leap method, considered in previous sections, is therefore an attractive alternative if one is willing to use a biased SSA.

For comparison, however, we consider here the unbiased estimation using two exact SSAs, the DM and wUDM, respectively. Note that, like the isomerisation system, the state space of the Schlögl system is one-dimensional and therefore sorting of the

states for an array-RQMC method is unambiguous. Taking the same parameters as in Section 4.2.3 we estimate that an (empirical) uniformisation rate $\bar{a} = 2.1 \cdot 10^4$ is sufficient. Using this uniformisation rate, $\bar{a}$, we furthermore estimate that the computational complexity of the wUDM is $\mathcal{C}[\hat{Q}_{\text{wUDM}}] \approx 3.2 \cdot \mathcal{C}[\hat{Q}_{\text{DM}}]$.

First we consider the estimation of the average copy number of S_1 molecules and the MSE decay as a function of the number of sample paths, N, used is depicted in Figure 4.25(a). It is clear that the wUDM is little better than the standard DM, even if we look at the array-RQMC variant with $\eta = 0.9$. For $N = 2^{18}$ sample paths we find that the MSE is only roughly 1.2 times smaller using the wUDM than when we use the DM, implying that the latter method is more efficient. Only if we use the wUDM with array-RQMC and $\eta = 0$, i.e. use low-discrepancy points for each reaction in the system, we see a clear benefit in terms of the MSE. Note, however, that this latter approach does have a larger run time due to the (significantly) increased number of times the states have to be sorted (step 7 in Algorithm 4.2).



(a) MSE when summary statistic is the average copy number of S_1 molecules.

(b) MISE when summary statistic is the distribution of S_1 molecules.

Figure 4.25: Comparison of the wUDM and the standard DM for the Schlögl system, (4.22), with parameters and initial condition as in Section 4.2.3. We consider both the array-RQMC implementation and the standard MC implementation of the wUDM with uniformisation rate $\bar{a} = 2.1 \cdot 10^4$. Summary statistics are (a) the average copy number of S_1 molecules at $T = 4$ and (b) the distribution of S_1 molecules at final time $T = 4$. Dashed black lines show the typical reference convergence rates. To establish the MSE and MISE equation (4.5) was used with $M = 64$ randomisations.

The situation is different when we consider the problem of estimating the distribution of S_1 molecules (which is a more relevant summary statistic for the Schlögl system) as can be seen in Figure 4.25(b). We can clearly see that in this scenario the wUDM forms a strong improvement over the DM, even if we do not use the array-RQMC method. Using the standard wUDM we observe a nine-fold decrease in the MISE. If we use the array-RQMC variant of the wUDM with sufficient number of samples this can grow even further, e.g. using $N = 2^{16}$ samples and $\eta = 0.9$ we observe a 68-fold improvement of the MISE compared to the standard DM. This superiority of the array-RQMC version of the wUDM is also visible to the eye in Figure 4.26, where we compare the actual empirical distributions from both methods.

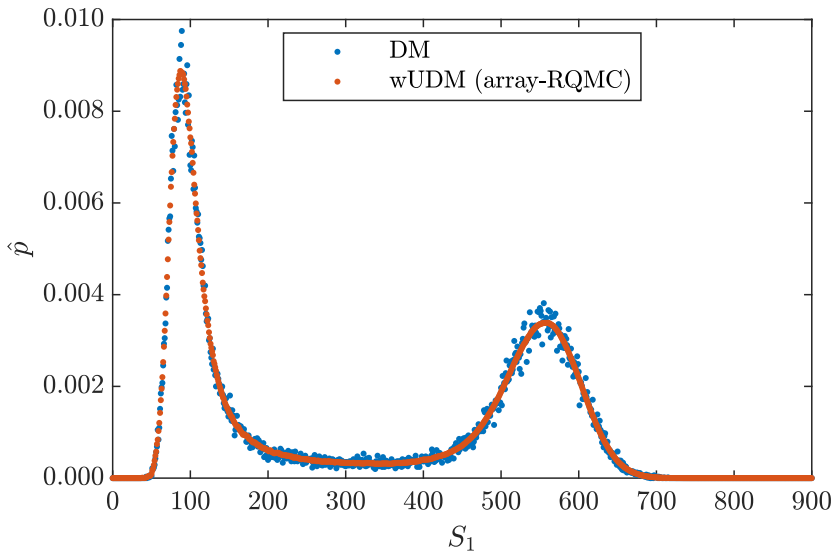
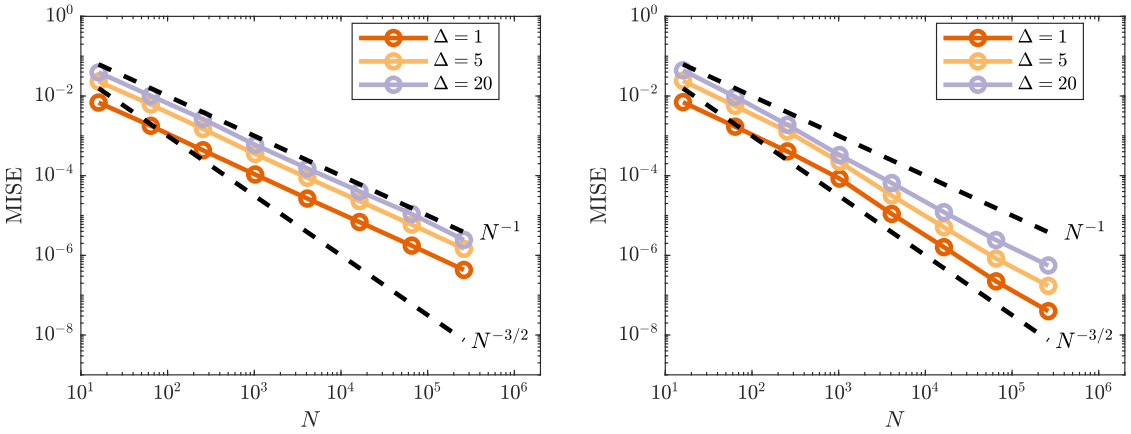


Figure 4.26: Empirical distribution for the number of S_1 molecules at $T = 4.0$ in the Schlögl system, (4.22), with parameters and initial condition as in Section 4.2.3. Distributions are constructed with $N = 2^{16}$ samples using the standard DM and array-RQMC wUDM ($\eta = 0.9$).

Based on the results in Section 4.3.1 one might be tempted to think that the array-RQMC method could provide even bigger performance gains if we attempt to estimate the distribution of S_1 molecules using histograms with fixed width Δ. This, however, is not necessarily true if we use the wUDM to construct such histograms as can be seen in Figure 4.27. The performance gain using the wUDM actually decreases if we

increase the bin width, Δ, of the histograms. We can understand this by noting that if we increase the histogram bin width, Δ, a histogram summary statistic becomes less influenced by high-frequency noise due to the firing of individual reactions. The wUDM, on the other hand, will perform better on systems with summary statistics that are subjected to high-frequency noise, because the wUDM acts as a low-pass filter. The largest benefits using the wUDM can therefore be seen when estimating the distribution using the actual EDF, rather than a histogram.



(a) wUDM (MC). (b) wUDM (array-RQMC).

Figure 4.27: Error convergence of histograms with fixed bin width, Δ, for the number of S_1 molecules at $T = 4.0$ in the Schlögl system, (4.22), with parameters and initial condition as in Section 4.2.3. Histograms are constructed using the wUDM with uniformisation rate $\bar{a} = 2.1 \cdot 10^4$. Dashed black lines show the typical reference convergence rates. To establish the MISE equation (4.5) was used with $M = 64$ randomisations.

Coupled flows

The final two examples are directly taken from [95]. We revisit these examples because the results reported in [95] cannot all be correct; for example, using the $1/N$ upper-bound for the MISE (see Example 2.3) one can see that Figures 1 and 3 in [95] must contain errors.

The first system, denoted as coupled flows, comprises four species, S_1, S_2, S_3 and

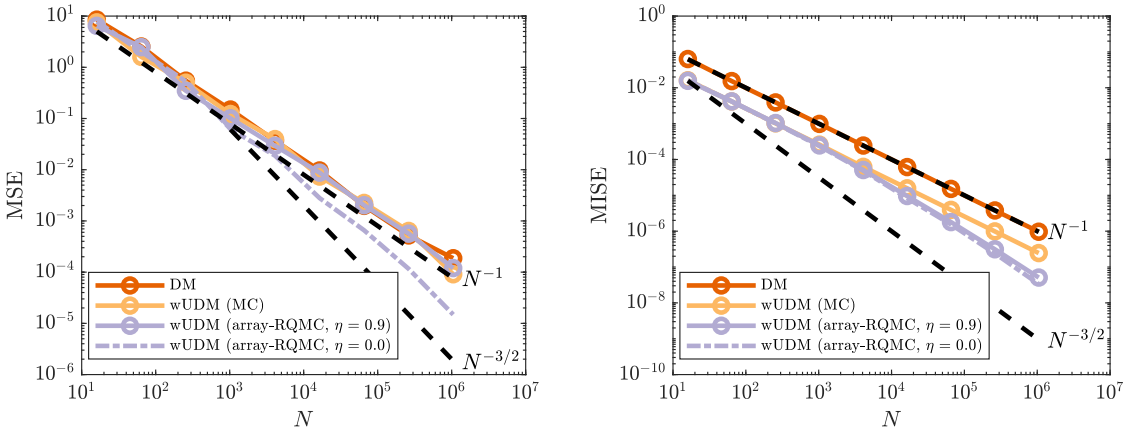
S_4, subject to the following nine reactions

$$\begin{aligned}
&\emptyset \xrightarrow{a_1(\mathbf{X})} S_1, \qquad \emptyset \xrightarrow{a_2(\mathbf{X})} S_2, \qquad S_1 + S_2 \xrightarrow{a_3(\mathbf{X})} \emptyset, \\
&S_1 \xrightarrow{a_4(\mathbf{X})} \emptyset, \qquad S_2 \xrightarrow{a_5(\mathbf{X})} \emptyset, \qquad \emptyset \xrightarrow{a_6(\mathbf{X})} S_3, \\
&\emptyset \xrightarrow{a_7(\mathbf{X})} S_4, \qquad S_3 \xrightarrow{a_8(\mathbf{X})} \emptyset, \qquad S_4 \xrightarrow{a_9(\mathbf{X})} \emptyset.
\end{aligned} \tag{4.24}$$

The reaction propensities (not all mass action kinetics) are given in Appendix 4.B together with the model parameters. We start the system with initially zero molecules, i.e. $\mathbf{X}_0 = (0, 0, 0, 0)^\intercal$.

It was noted in [95] that this system can then be uniformised with uniformisation rate $\bar{a} = 10$. We report the results, both for the case of estimating the average copy number of S_1 molecules and the joint distribution for S_1 and S_2 molecules, in Figure 4.28. It is again clear from Figure 4.28(a) that when estimating the average copy numbers the wUDM does not yield an advantage over the standard DM, even when combined with array-RQMC. The same observation holds if we attempt to estimate the average copy number of any of the other three species in the system.

We note that, though correct, the proposed uniformisation rate overestimates the number of reactions that fire on the interval $t \in [0, 100]$, in which case a uniformisation rate $\bar{a} = 4$ appears to be sufficient. We find that the computational complexity is $\mathcal{C}[\hat{Q}_{\text{wUDM}}] \approx 11 \cdot \mathcal{C}[\hat{Q}_{\text{DM}}]$ when $\bar{a} = 10$, again reflecting that the uniformisation rate is chosen too high if we are primarily interested in the system at $T = 100$. This means that if we take $\bar{a} = 10$, only for large number of sample paths does the wUDM with array-RQMC become more efficient than the standard DM if the objective is the joint distribution of S_1 and S_2 molecules. Without array-RQMC the wUDM is even less efficient than the DM. If we, however, repeat the simulation with a lower uniformisation rate, $\bar{a} = 4$, we find the performance of the wUDM significantly improves, both due to a lower complexity and a larger variance reduction, as can be seen in Table 4.1. Whereas with $\bar{a} = 10$ the MC version of the wUDM is not



(a) MSE when summary statistic is the average copy number of S_1 molecules.

(b) MISE when summary statistic is the joint distribution of S_1 and S_2 molecules.

Figure 4.28: Comparison of the wUDM and the standard DM for the coupled flows system, (4.24), with parameters and initial condition as in Section 4.2.3. We consider both the array-RQMC implementation and the standard MC implementation of the wUDM with uniformisation rate $\bar{a} = 10$. Summary statistics are (a) the average copy number of S_1 molecules at $T = 100$ and (b) the joint distribution of S_1 and S_2 molecules at final time $T = 100$. Dashed black lines show the typical reference convergence rates. To establish the MSE and MISE equation (4.5) was used with $M = 64$ randomisations.

more efficient than the DM, choosing a more suitable uniformisation rate makes the wUDM more efficient than the DM. This is in line with our earlier observations in Section 3.4.2. Using the array-RQMC wUDM version with $N = 2^{20}$ samples yields roughly a seven-fold or twelve-fold efficiency improvement over the standard DM using $\eta = 0.9$ or $\eta = 0$, respectively. Note that our results disagree (quantitatively) with those reported in [95, Table II].

From the combined results in this example and the two previous examples we see that the choice of $\eta \in [0, 1]$ has a more pronounced influence when the summary statistic of interest is a (raw) moment, i.e. scenarios in which the wUDM does not necessarily excel. For the case of estimating distributions, for which the wUDM is more suitable due to its low-pass filter property, we see that changing η has less effect and we therefore use $\eta = 0.9$ from this point on whenever we estimate a distribution.

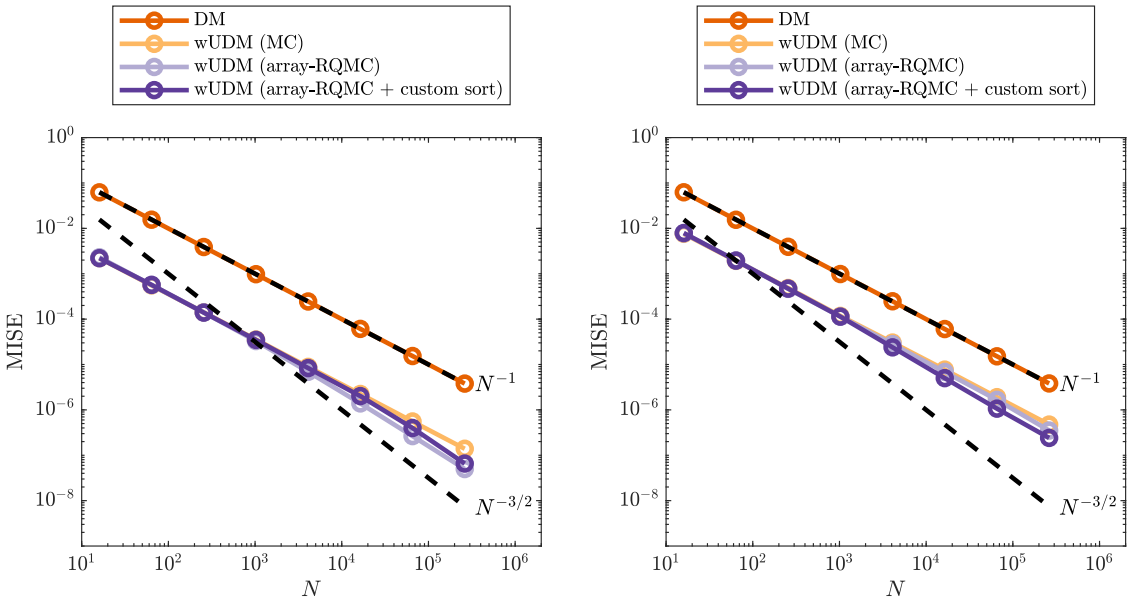
Finally we consider what happens if we run the simulations up to $T = 1000$. In this case we do need a uniformisation rate $\bar{a} = 10$, which yields a comparatively good

		Number of samples, N				
		2^{12}	2^{14}	2^{16}	2^{18}	2^{20}
wUDM (MC)	$\bar{a} = 10$	3.9	3.9	3.8	3.8	3.9
	$\bar{a} = 4$	5.5	5.5	5.4	5.3	5.6
wUDM (array-RQMC) $\eta = 0.9$	$\bar{a} = 10$	4.8	6.4	8.4	12	19
	$\bar{a} = 4$	8.6	11	15	23	33
wUDM (array-RQMC) $\eta = 0$	$\bar{a} = 10$	5.0	7.5	9.8	16	29
	$\bar{a} = 4$	8.8	12	18	32	61

Table 4.1: Variance reduction factor (in terms of the MISE) of the wUDM relative to the standard DM when estimating the joint distribution of S_1 and S_2 molecules at final time $T = 100$.

computational complexity for the wUDM, $\mathcal{C}[\hat{Q}_{\text{wUDM}}] \approx 2.5 \cdot \mathcal{C}[\hat{Q}_{\text{DM}}]$. As observed in [95] the state space of the species is larger when $T = 1000$ than when $T = 100$ and therefore estimating the species distributions accurately needs a large number of sample paths. As we can see in Figure 4.29 the benefit from using array-RQMC only shows when we use more than 10^4 sample paths. Because the state space for this system is four-dimensional we consider two different sorting orders, the canonical order $[S_1, S_2, S_3, S_4]$ versus the custom order $[S_3, S_4, S_1, S_2]$, for the array-RQMC method and look at its effect on the convergence behaviour for two different summary statistics. We note that if we consider the joint distribution of S_1 and S_2 molecules the canonical sorting order performs best (using $N = 2^{18}$ samples the MISE is roughly 1.5 times smaller compared to the custom order), whereas if we consider the joint distribution of S_1 and S_4 molecules the situation is exactly reversed. This shows that the choice of importance function or sorting order in the array-RQMC variant of the wUDM, used in step 7 of Algorithm 4.2, influences the method's performance.

In Figure 4.30 we explore a range of summary statistics for the system at time $T = 1000$. We see that the benefit from using the wUDM (both variants of the method) depends on the marginal distribution that we are attempting to estimate.



(a) MISE when summary statistic is the joint distribution of S_1 and S_2 molecules.

(b) MISE when summary statistic is the joint distribution of S_1 and S_4 molecules.

Figure 4.29: Comparison of the wUDM and the standard DM for the coupled flows system, (4.24), with parameters given in Appendix 4.B and zero molecules initially. We consider both the array-RQMC implementation and the standard MC implementation of the wUDM with uniformisation rate $\bar{a} = 10$. The custom sort method changes the sorting order for the states to $[S_3, S_4, S_1, S_2]$. Summary statistics are calculated at final time $T = 1000$. Dashed black lines show the typical reference convergence rates. To establish the MSE and MISE equation (4.5) was used with $M = 64$ randomisations.

The efficiency of the array-RQMC wUDM relative to the DM is larger than 20 for the S_1, S_2 joint distribution, whereas it is less than one for the S_3, S_4 joint distribution. The benefit is also smaller when estimating the one-dimensional marginal distributions (not shown). We can explain these observations by again considering the low-pass filter property of the wUDM; the reactions changing the species S_3 and S_4 counts fire less frequently compared to the reactions changing the S_1 and S_2 counts, implying that the summary statistics involving species S_1 and/or S_2 are subject to more high-frequency noise. The wUDM filters such high-frequency noise effectively, leading to a large variance reduction if S_1 and/or S_2 are involved in the summary statistic.

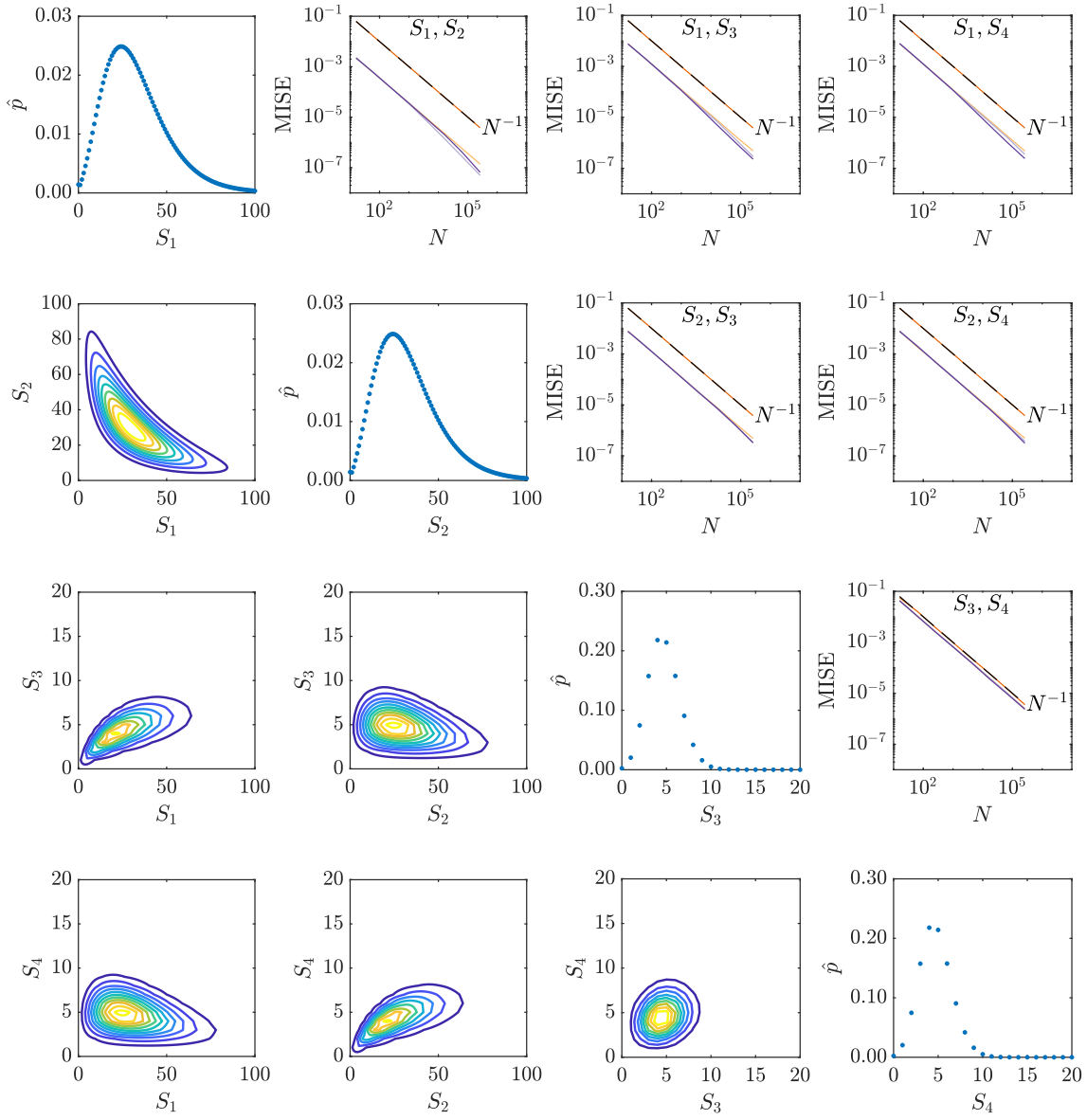


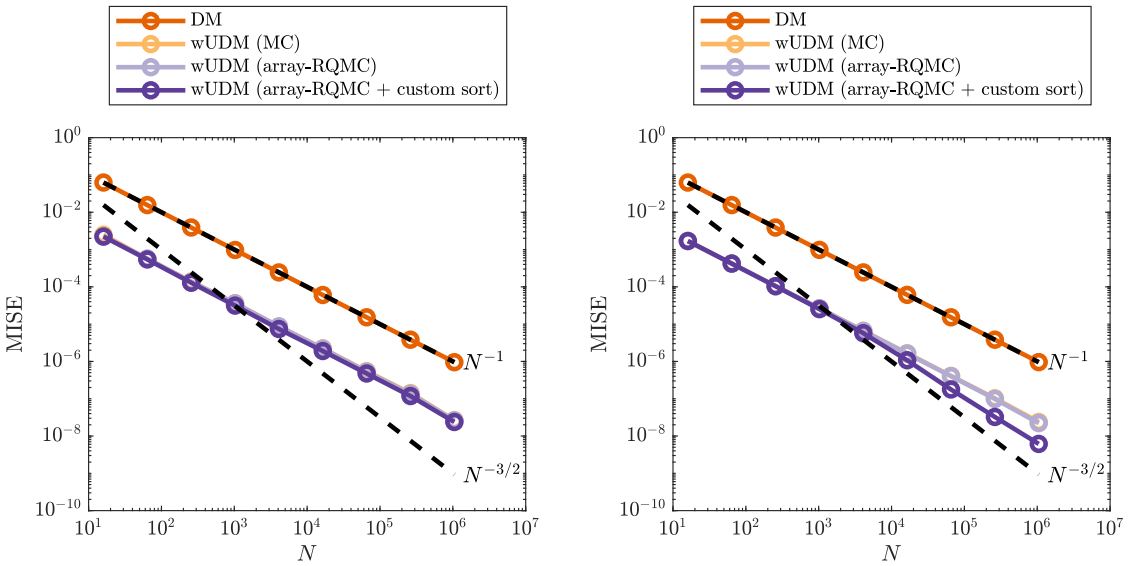
Figure 4.30: Marginal distributions and error convergence for the coupled flows system, (4.24), at time $T = 1000$ using parameters in Appendix 4.B. The diagonal tiles show the one-dimensional marginal distributions and the lower triangular tiles the bivariate marginal distributions (isolines). All distributions are calculated by averaging $M = 64$ independent runs of the array-RQMC wUDM with $N = 2^{18}$ samples. The upper triangular tiles show the error convergence for the bivariate marginal distributions. Legend and colour coding for the methods in the upper triangular tiles is the same as in Figure 4.29. To establish the MISE equation (4.5) was used with $M = 64$ randomisations.

MAPK-cascade with feedback

The final example is the MAPK-cascade system, previously studied in Section 3.3.2. Following [95] we uniformise the system with uniformisation rate $\bar{a} = 15$, which, perhaps surprisingly, leads to a computational complexity of $\mathcal{C}[\hat{Q}_{\text{wUDM}}] \approx 0.95 \cdot \mathcal{C}[\hat{Q}_{\text{DM}}]$, implying that the wUDM is roughly on par with the standard DM. Though not stated explicitly in either [95] or [109], we assume here an initial condition with each kinase in its inactive form, i.e. $\mathbf{X}_0 = (100, 0, 300, 0, 0, 300, 0, 0)^\intercal$. We found the results that follow next insensitive to small perturbations in these initial conditions.

Our primary interest is the behaviour of S_8 (MAPK-PP) and we compute its bivariate marginal distribution with the other seven species. We study two different sorting strategies for the array-RQMC variant of the wUDM, the canonical sorting order $[S_1, \ldots, S_8]$ and the custom order $[S_8, S_5, S_2]$, meaning that we only sort based on three out of eight species. The latter is chosen as an attempt to capture the cascading structure in the network whilst prioritising our species of interest, S_8.

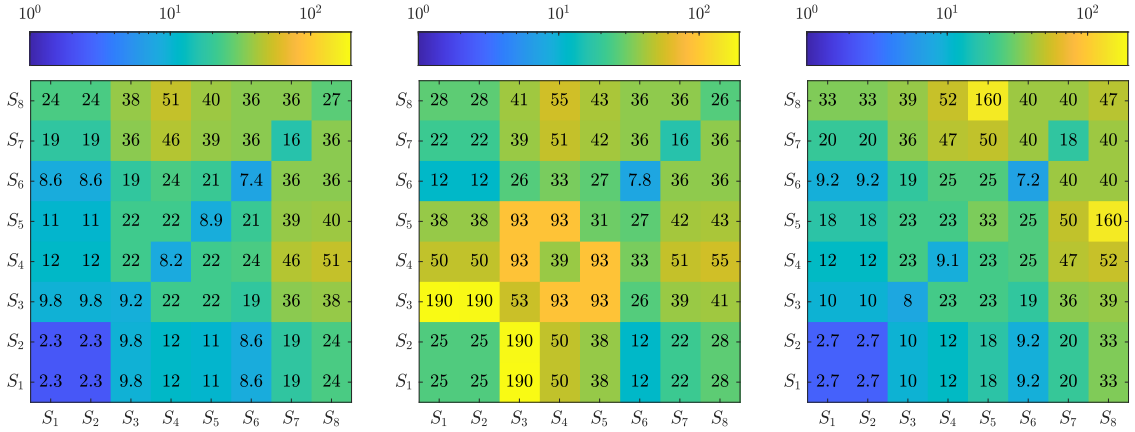
In Figure 4.31 we show the resulting error decay for two different bivariate marginal distributions. For the joint distribution of S_6 and S_8 all wUDM methods have a similar error decay, though we see that if we estimate the joint distribution of S_5 and S_8 the array-RQMC wUDM with the custom sorting strategy outperforms the other methods. Furthermore, in Figure 4.32 we see that the array-RQMC method rarely gives more than a factor of two actual improvement over the standard MC implementation of the wUDM for this system when using (up to) $N = 2^{20}$ sample paths. Some notable exceptions are the estimation of the joint distribution of S_5 and S_8 or S_3 and S_1/S_2, for which we see that the array-RQMC method can be two orders of magnitude more efficient than the standard DM. In addition Figure 4.32 shows again how the sorting step in the array-RQMC method can influence the overall benefit from using array-RQMC, and in particular how this benefit for a given sorting method can strongly depend on the summary statistic of interest. We draw the same conclusions when $\eta = 0$ (not shown here).



(a) MISE when summary statistic is the joint distribution of S_6 and S_8 molecules.

(b) MISE when summary statistic is the joint distribution of S_5 and S_8 molecules.

Figure 4.31: Comparison of the wUDM and the standard DM for the MAPK-cascade system, see Figure 3.6, with parameters given in Appendix 4.B. We consider both the array-RQMC implementation and the standard MC implementation of the wUDM with uniformisation rate $\bar{a} = 15$. The custom sort method uses the sorting order $[S_8, S_5, S_2]$ and the standard array-RQMC method the canonical sorting order $[S_1, \ldots, S_8]$. Summary statistics are calculated at final time $T = 200$. Dashed black lines show the typical reference convergence rates. To establish the MSE and MISE equation (4.5) was used with $M = 64$ randomisations.



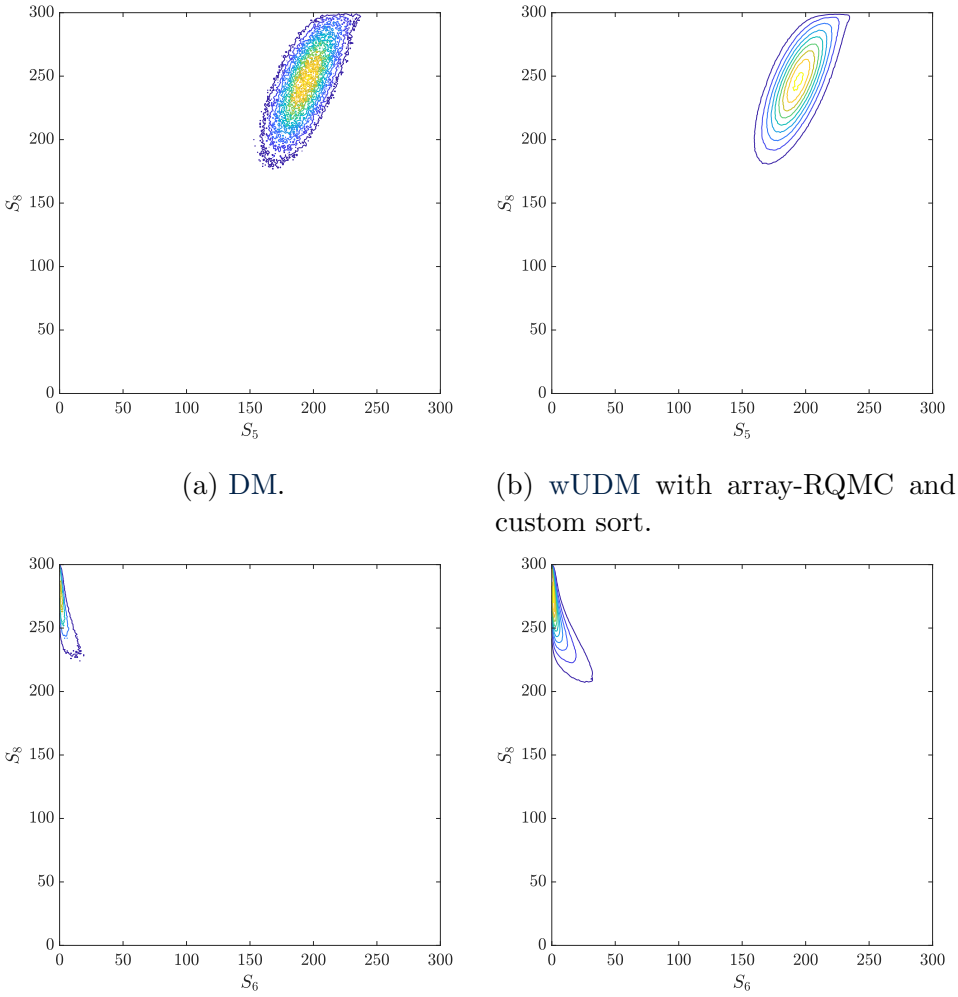
(a) wUDM (MC).

(b) wUDM (array-RQMC) (1).

(c) wUDM (array-RQMC) (2).

Figure 4.32: Variance reduction factor (in terms of the MISE) of the wUDM relative to the DM when estimating marginal joint distributions in the MAPK-cascade system at final time $T = 200$ using $N = 2^{20}$ sample paths. Array-RQMC method (1) uses the canonical sorting order $[S_1, \ldots, S_8]$, array-RQMC method (2) uses the custom sorting order $[S_8, S_5, S_2]$.

The improvement from using the wUDM combined with array-RQMC can also be clearly seen in Figure 4.33, which shows a comparison between the results from the DM and the wUDM, both using an equal number of sample paths, $N = 2^{18}$. Note also that Figures 4.33(c) and 4.33(d), showing the joint distribution of S_6 (MAPK) and S_8 (MAPK-PP), do not agree with the distributions shown in [95, Figure 4].



(a) DM.

(b) wUDM with array-RQMC and custom sort.

(c) DM.

(d) wUDM with array-RQMC and custom sort.

Figure 4.33: Marginal distributions (isolines) computed using $N = 2^{18}$ sample paths for the MAPK-cascade system, see Figure 3.6, with parameters given in Appendix 4.B. The custom sort method re-orders the states based on $[S_8, S_5, S_2]$ (in that order). Summary statistics are calculated at final time $T = 200$.

4.4 Discussion

It is known that the use of low-discrepancy numbers instead of pseudo-random numbers can greatly improve the convergence rate for problems involving traditional quadrature and SDEs. In this chapter we explored the application of RQMC methods in the framework of simulation of stochastic chemical reaction systems.

Standard RQMC methodology

In particular, we first looked at the combination of low-discrepancy numbers with the τ-leap method. For simplicity, the fixed step τ-leap method was considered so as to allow for a simple implementation of low-discrepancy points without negative effects on the runtime. We note that the question of whether this is a good procedure has been addressed in the literature before [2, 35, 36, 39]. This chapter, however, does not focus on the question of whether the τ-leap method forms a good approximation to the CTMC dynamics governed by the CME, which is the motivation therein for the discussion about time step selection. Rather, we focus on the question of how quickly statistical errors in desired summary statistics decay as a function of the number of sample paths simulated. We answer this question in the simplest possible case, namely using fixed time step τ-leap, though we expect our conclusions below to be general enough to hold for a large class of simulation procedures for stochastic biological systems.

Theory proves that in some cases the convergence rate for an RQMC method is not worse than for the equivalent MC method (up to a constant [165]). Evidence presented here shows that in case of chemical reaction networks standard RQMC is superior to plain MC, as demonstrated by numerical experiments in Section 4.2. As a result, if one chooses the fixed time step τ-leap approach to simulate a chemical reaction network, the use of RQMC methods gives better convergence behaviour when compared to the traditional MC implementation at no extra computational

complexity and negligible extra run-time cost.

However, the benefits from using low-discrepancy numbers are smaller than anticipated based on results seen in the simulation of SDEs. In particular, if one chooses instead to model chemical reaction systems using SDEs in the form of the CLE, one can see a greater advantage in the use of low-discrepancy numbers. This effect is caused by at least two factors.

Firstly, the inherently discrete nature of stochastic simulations of chemical reaction networks hinders RQMC convergence. It has been reported in the literature that discontinuous integrands experience less benefit from RQMC methods over standard MC methods [23, 94, 155, 156]. In Section 4.2 we showed through the use of a simplified test system that the behaviour observed in simulating chemical reactions can be replicated by introducing certain types of discontinuity in classical quadrature problems. The simple test systems in Section 4.2 allow for a detailed understanding of the RMSE convergence rate observed when applying RQMC. It is, however, not always possible to choose the biological model or its parameters such that the effect of discontinuities will be small. It would therefore be advantageous to have techniques that leave the desired summary statistic intact, but diminish the effect of discontinuities on the RMSE convergence. Smoothing techniques have previously been considered to mitigate the effects of discontinuities in other contexts [32, 63, 156] and could perhaps be of use for problems in the context of this thesis.

Secondly, it is known that the performance of (R)QMC methods can strongly depend on the dimension of the problem. As illustrated in Section 4.2.3, a higher dimension can lead to a much smaller performance benefit, regardless of the smoothness of the underlying problem. Methods to reduce the effective dimension of the problem by a change of variables have proven to be effective in other fields and it is an open question as to whether such transformations can be found for the simulation of biological systems.

We also point out that the original article introducing QMC methods in 1951 by

Richtmyer [187] considered a discrete linear birth process. He observed a smaller performance gain than expected and this might have impeded the further exploration of QMC methods in stochastic simulation for a few decades. Richtmyer's results can now be understood to be caused by the unfortunate choice of his chosen model problem, which is discontinuous in nature. In addition it was recently argued that the quality of the low-discrepancy point sets used by Richtmyer was low compared to modern standard construction methods [162, Chapter 15].

Array-RQMC methodology

Another related method which has proven to be fruitful in the simulation of DTMCs of potentially large dimension is array-RQMC, described in Section 4.1.2. In Section 4.3 we explored its use in the context of simulating chemical reaction network models. It was found that combining the array-RQMC method with approximate SSAs, such as the τ-leap method, can provide a significant improvement over both standard MC and RQMC variants, especially when many time steps are taken. This was true for two very different type of summary statistics, namely the mean species count and the species distribution. For the latter summary statistic, however, generally a larger number of sample paths, N, was needed for the benefit from using array-RQMC to become apparent due the more complex nature of the summary statistic. The array-RQMC method is less hindered by the discreteness and dimensionality of the problem considered than standard RQMC and therefore is likely to be more suited to the simulation of chemical reaction network models.

The array-RQMC method is also the cornerstone of the only other known QMC work in the area of stochastic biological systems [95]. In Section 4.3.3 we revisited this method and the results in [95]. We showed that, while the results in [95] appear to be quantitatively incorrect, there can be a benefit from combining uniformisation, in the form of the wUDM, and array-RQMC. In particular, it was observed through various examples that the wUDM with array-RQMC performs well only when the

summary statistic of interest is suited to the standard wUDM and sufficient sample paths can be used. We found the wUDM to be effective, for example, when estimating distributions of species that fluctuate on the fastest timescale in the system. If the object of interest involves species that fluctuate on slower timescales the wUDM becomes less effective and the combination of multiscale methods, such as the nested SSA [55, 56], with the wUDM could be a successful strategy.

Our overall results indicate that array-RQMC methods, in combination with a variety of widely used SSAs, work very well in the context of simulating chemical reaction networks. A remaining challenge we identified is how the performance of such array-RQMC methods depends on the importance function used. In order to automate the construction of effective array-RQMC methods in the context of chemical reaction networks more research like that in [122] will be needed to determine an effective strategy for choosing importance functions for models with multidimensional state spaces.

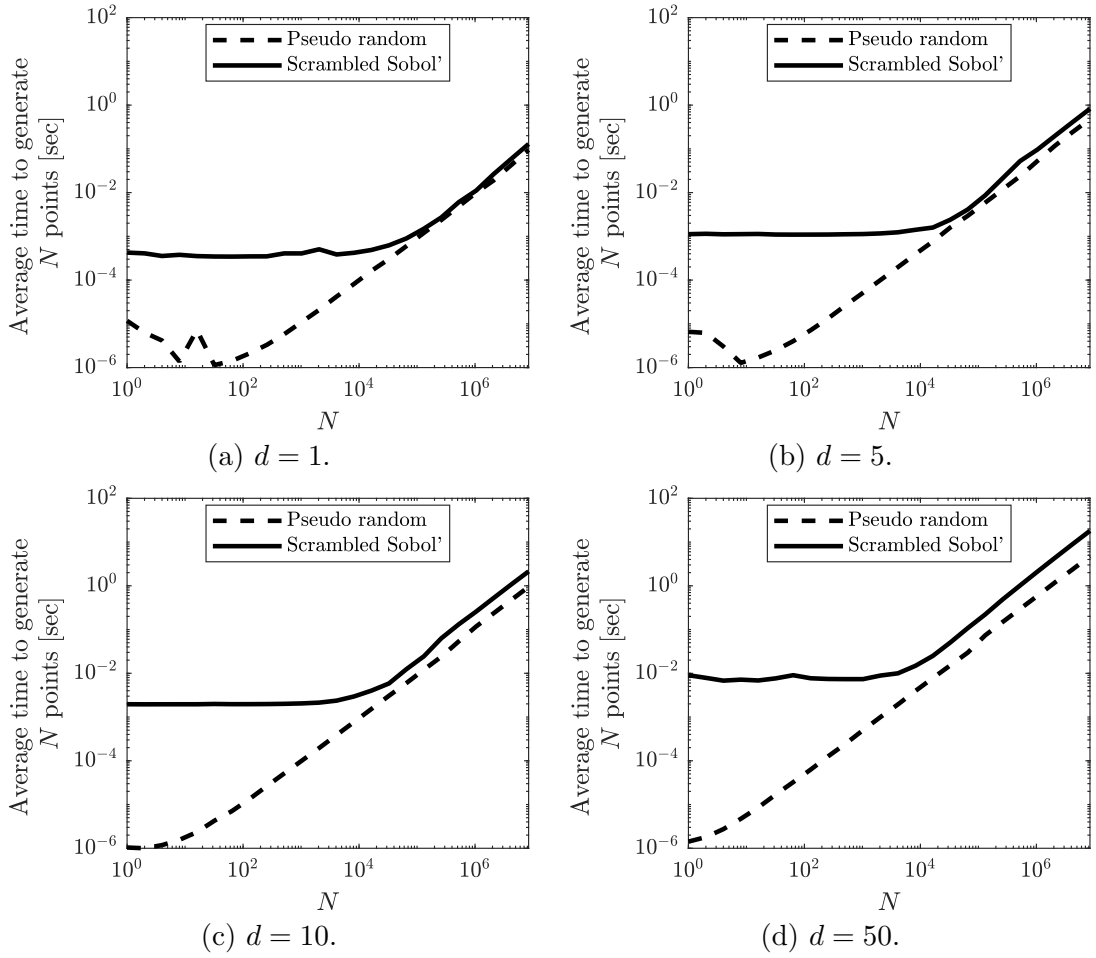
Chapter appendix

4.A Computational effort to generate quasi-random numbers

Whilst the efficiency gains from using RQMC methods over standard MC methods are clear in terms of computational complexity one should question whether the time taken to generate scrambled low-discrepancy sequences for an RQMC method is much greater than the time needed for pseudo-random numbers to be generated as this could void any observed efficiency gains. We therefore perform a small test to time the generation of the various random numbers. We time how long it takes to generate a point set of length N in d dimensions (averaged over 50 trials). Timing experiments were performed using MATLAB R2018b on an Ubuntu desktop PC with a 3.40 GHz Intel Core i7-2600K CPU and 16 GB of random access memory. We test the standard pseudo-random number generator (which uses the Mersenne Twister algorithm) versus Sobol' points with linear matrix scrambling and a random digital shift. The results are depicted in Figure 4.34 and show that only for relatively small point sets is the generation of pseudo-random numbers distinctly faster than the Sobol' points (on the order of milliseconds), presumably due to a larger overhead for the Sobol' points. For point sets of lengths not uncommon in simulations (10^5 or more points) the difference becomes negligible. Therefore the completion time for an algorithm which has replaced pseudo-random numbers with low-discrepancy numbers will not differ noticeably. These findings agree with practical timing results for simulations of various financial applications in [129].

4.B Model parameters

Here we provide the reaction propensities and parameters for the two models taken from [95]. Numerical results on these systems can be found in Sections 3.3.2 and 4.3.3.



(a) $d = 1$. (b) $d = 5$.

(c) $d = 10$. (d) $d = 50$.

Figure 4.34: Comparison between the time taken to generate N pseudo-random points and an equal number of scrambled Sobol' points in $[0, 1)^d$.

Coupled flows

For the coupled flows system, equation (4.24), we have the following reaction propensities

$$a_1(\mathbf{X}) = \frac{k_a X_3}{1 + X_1/K_i}, \qquad a_4(\mathbf{X}) = \mu X_1, \qquad a_7(\mathbf{X}) = \frac{k_{eB}}{1 + X_2/K_r},$$
$$a_2(\mathbf{X}) = \frac{k_b X_4}{1 + X_2/K_i}, \qquad a_5(\mathbf{X}) = \mu X_2, \qquad a_8(\mathbf{X}) = \mu X_3,$$
$$a_3(\mathbf{X}) = k_2 X_1 X_2, \qquad a_6(\mathbf{X}) = \frac{k_{eA}}{1 + X_1/K_r}, \qquad a_9(\mathbf{X}) = \mu X_4,$$

where the parameters can be found in Table 4.2. Unless stated otherwise we use the initial condition $\mathbf{X}_0 = (0, 0, 0, 0)^\intercal$.

parameter	k_a	k_b	k_2	K_i	μ	k_{eA}	k_{eB}	K_r
value	0.3	0.3	0.001	60	0.002	0.02	0.02	30

Table 4.2: Parameters for the coupled flows system, taken from [95].

MAPK-cascade with feedback

The interaction network for the MAPK-cascade can be found in Figure 3.6. The reaction propensities and model parameters are taken from [109]. The following reaction propensities, which are of the Hill function type, correspond to the numbered reactions in Figure 3.6 and [109],

$$
\begin{aligned}
a_1(\mathbf{X}) &= V_1 \frac{X_1}{(1+(X_8/K_I)^n)\cdot(K_1+X_1)}, & a_6(\mathbf{X}) &= V_6 \frac{X_4}{K_6+X_4},\\
a_2(\mathbf{X}) &= V_2 \frac{X_2}{K_2+X_2}, & a_7(\mathbf{X}) &= k_7 X_5 \frac{X_6}{K_7+X_6},\\
a_3(\mathbf{X}) &= k_3 X_2 \frac{X_3}{K_3+X_3}, & a_8(\mathbf{X}) &= k_8 X_5 \frac{X_7}{K_8+X_7},\\
a_4(\mathbf{X}) &= k_4 X_2 \frac{X_4}{K_4+X_4}, & a_9(\mathbf{X}) &= V_9 \frac{X_8}{K_9+X_8},\\
a_5(\mathbf{X}) &= V_5 \frac{X_5}{K_5+X_5}, & a_{10}(\mathbf{X}) &= V_{10} \frac{X_7}{K_{10}+X_7},
\end{aligned}
$$

and the parameters are given in Table 4.3. Unless stated otherwise we use the initial condition $\mathbf{X}_0 = (100, 0, 300, 0, 0, 300, 0, 0)^\intercal$.

parameter	V_1	V_2	k_3	k_4	V_5	V_6	k_7	k_8	V_9	V_{10}
value	2.5	0.25	0.025	0.025	0.75	0.75	0.025	0.025	0.5	0.5

parameter	n	K_I	K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8	K_9	K_{10}
value	1	9	10	8	15	15	15	15	15	15	15	15

Table 4.3: Parameters for the MAPK-cascade system, taken from [109, Table 2].

Chapter 5

Poisson bridge methods

In this chapter we discuss variance reduction techniques tailored to the simulation of unit-rate Poisson processes. In particular we investigate various Poisson bridge constructions, which can be thought of as discrete analogues of the well-known Brownian bridge construction [84, Section 3.1]. This construction forms a popular starting point for variance reduction methods for SDEs, perhaps most notably since the influential 1996 Moskowitz and Caflisch paper [156] showed how the Brownian bridge construction in combination with a low-discrepancy point set can lead to an incredible performance benefit. The core ideas for the Poisson bridge methods developed in this chapter date back to a report published in that same year [62] which, on the contrary, appears to have attracted very little attention. Though an expansion of this original report was included in an extensive and more widely used monograph [63], no adaptation of Poisson bridge methods in the context of chemical reaction networks existed until very recently when some ideas were rediscovered and backed up by theory in [141].

Here, we unify concepts found in [141] and [62] and, by expanding on these ideas, we show, in practice and proof, how we can achieve orders of magnitude improvement over standard MC approaches when constructing unit-rate Poisson processes. At the same time we provide practical guidance as to how to implement and tune Poisson

bridge methods to achieve, in some sense, (near) optimal performance. Though the resulting methods are interesting in their own right, their main role in the context of chemical reaction networks is in conjunction with exact SSAs, such as the NRM [68] and MNRM [1]. We therefore also consider Poisson bridge methods when used to generate sample paths of several example chemical reaction networks.

5.1 Standard sampling of Poisson processes

Throughout this chapter we consider the problem of generating a unit-rate Poisson process, $Y(\tau)$. Perhaps the most commonly used method to do so relies on the observation that the increments of a Poisson process are independent and that the interarrival epochs are exponentially distributed with unit-rate, i.e. Exp(1). Repeated sampling and summing of unit-rate exponential random variables therefore yields a space, i.e. memory, and time efficient method to sequentially generate both the arrival epochs and value of the Poisson process $Y(\tau)$, i.e. the number of arrival epochs at τ, as illustrated in Figure 5.1.

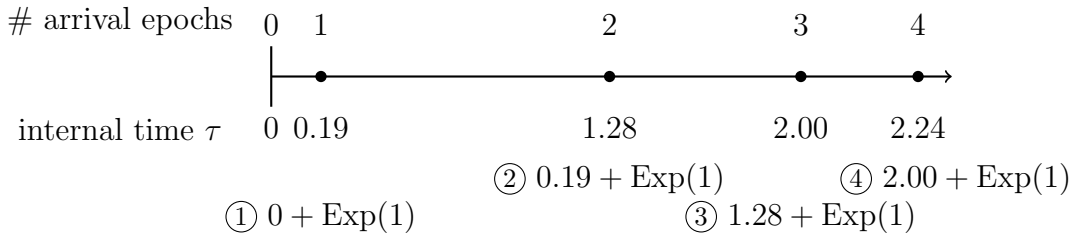
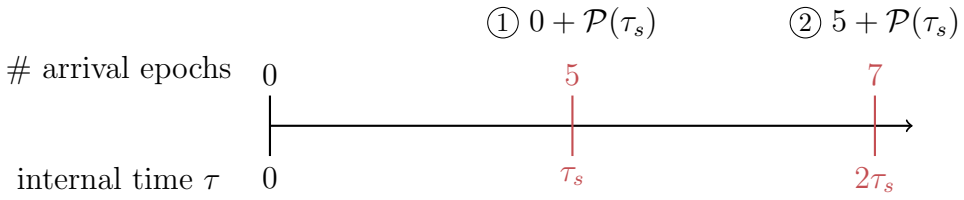


Figure 5.1: Sampling of a Poisson process using the characteristics of the interarrival epochs. Solid dots (•) indicate the arrival epochs of the Poisson process. Note that, by abuse of notation, $\text{Exp}(\lambda)$ represents random variables from the exponential distribution.

A widely used alternative to the above is the exponential spacings method [48, Chapter V], which generates a Poisson process, $Y(\tau)$, over a fixed time interval, say $[0, \tau_s)$. For this approach we first sample the number of arrival epochs of $Y(\tau)$ in $[0, \tau_s)$, $N(\tau_s) \sim \mathcal{P}(\tau_s)$, and then generate the actual arrival epochs as $N(\tau_s)$ uniforms on the interval $[0, \tau_s)$. This method is efficient in terms of time, but its naïve implementation

is not necessarily space efficient as it requires one to store all the arrival epochs over $[0, \tau_s)$ in memory. However, as shown in [48, Section V.3.1], we can efficiently generate n ordered uniform random variables sequentially in $\mathcal{O}(n)$ time and $\mathcal{O}(1)$ space, thus making this approach as efficient as the previously described method based on exponential random variables. Note that, by independence of the increments of a Poisson process, we can use repeated application of the exponential spacings method to generate a Poisson process on $[0, \tau_s)$, $[\tau_s, 2\tau_s)$, etc., as is illustrated in Figure 5.2.



(a) Sample the total number of arrivals over the intervals $[0, \tau_s)$ and $[\tau_s, 2\tau_s)$.



(b) Sample the arrival epochs conditioned on the Poisson skeleton.

Figure 5.2: Sampling of a Poisson process on $[0, 2\tau_s)$ via the exponential spacings method with step size τ_s. Solid dots (•) indicate the arrival epochs of the Poisson process. (a) shows the construction of a Poisson skeleton and (b) depicts the filling in of the Poisson skeleton using ordered uniforms. Note that, by abuse of notation, $\mathcal{P}(\lambda)$ represents random variables from the Poisson distribution.

We can view the exponential spacings method as a two-stage method, where first we generate the values of the Poisson process $Y(\tau)$ on a pre-defined time grid and then fill in the arrival epochs conditioned on the information from the first step. This naturally leads to the following definition of a Poisson skeleton.

Definition 1 (Poisson skeleton)**.** *A Poisson skeleton for a collection of N Poisson processes $\left\{Y^{(1)}(\tau), \ldots, Y^{(N)}(\tau)\right\}$ is a collection $\left\{\tau_i, Y^{(1)}(\tau_i^-), \ldots, Y^{(N)}(\tau_i^-)\right\}_{i=1}^{R}$ for some $R \in \mathbb{N}_{>0}$, where $Y(s^-) = \lim_{\tau \uparrow s} Y(\tau)$. The Poisson skeleton given by $\left\{\tau_i, Y^{(1)}(\tau_i^-), \ldots, Y^{(N)}(\tau_i^-)\right\}_{i=1}^{R}$ is* unbiased *if $\mathbb{E}\left[Y^{(n)}(\tau_i^-)\right] = \tau_i$ holds for all $i = 1, \ldots, R$ and $n = 1, \ldots N$.*

In the case of a single Poisson process, $Y(\tau)$, a Poisson skeleton is simply a collection of time points and the value that the Poisson process $Y(\tau)$ takes at those points. The output of the first stage of the exponential spacings method is thus an (unbiased) Poisson skeleton. The second stage of the exponential spacings method then fills in the arrival epochs conditioned on this Poisson skeleton.

Now that we have separated the exponential spacings method in two independent stages we consider two alternative methods to construct unbiased Poisson skeletons, which can replace the first stage of the exponential spacings method.

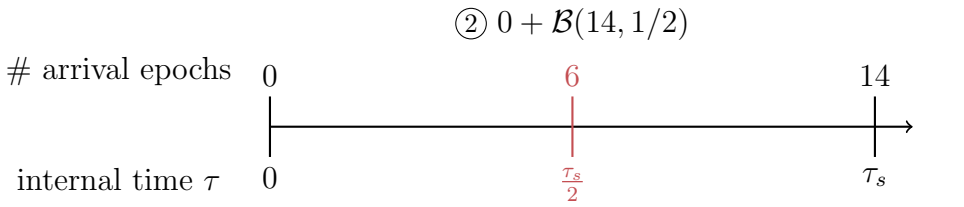
5.2 Midpoint-based Poisson bridge

The basic idea for the midpoint-based Poisson bridge, first introduced in [62], is to increase the time resolution of a Poisson skeleton generated by the exponential spacings method via conditional subsampling. For simplicity of our explanation we assume that we have a Poisson skeleton $\{\tau_s, Y(\tau_s^-)\}$. We then know that at the midpoint of the interval $[0, \tau_s)$ the Poisson process value is binomially distributed, i.e. $Y(\tau_s^-/2) \sim \mathcal{B}(Y(\tau_s^-), 1/2)$. Sampling from this binomial distribution then creates a new Poisson skeleton at the times $\{\tau_s/2, \tau_s\}$, and if the original Poisson skeleton was unbiased then the new Poisson skeleton is also unbiased if we use an unbiased sample from the binomial distribution. Of course, we can then apply the same technique to this newly created Poisson skeleton; repeating this subdivision and conditional subsampling at the midpoint(s) procedure L times we construct a Poisson skeleton on $\tau_1, \ldots, \tau_{2^L} \in [0, \tau_s)$, where the times are given by $\tau_l = l\tau_s/2^L$ for $l = 1, \ldots, 2^L$. This procedure represents a single step, of size τ_s, of the midpoint-based Poisson bridge with L levels. Just as for the exponential spacings method we can repeatedly use the midpoint-based Poisson bridge construction on $[0, \tau_s)$, $[\tau_s, 2\tau_s)$, etc., and this leads to the full midpoint-based Poisson bridge with L levels and step size τ_s. An illustration of a single step of the midpoint-based Poisson bridge with $L = 2$ levels is shown in Figure 5.3.

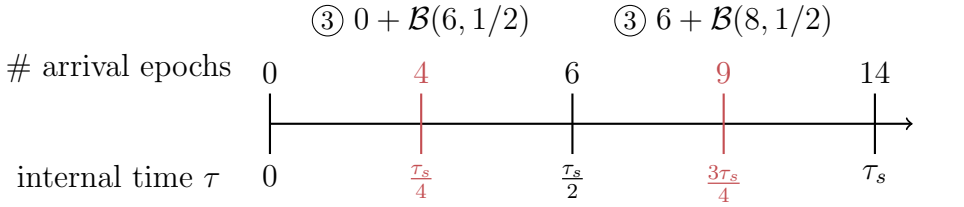
The name of the Poisson bridge method stems from its link to the Brownian bridge construction of a Wiener process, which builds a Brownian motion path by successively adding detail on finer time scales.



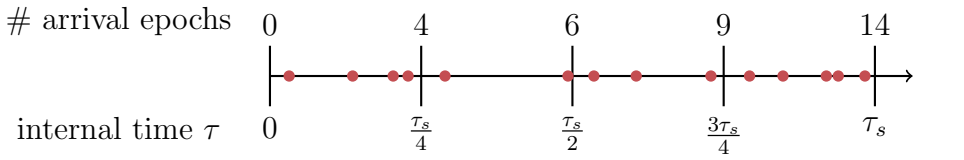
(a) Sample the total number of arrivals over the interval $[0, \tau_s)$.



(b) Sample the total number of arrivals over the interval $[0, \tau_s/2)$ conditioned on the number of arrivals at $\tau = \tau_s$.



(c) Sample the total number of arrivals over the intervals $[0, \tau_s/4)$ and $[\tau_s/2, 3\tau_s/4)$ conditioned on the number of arrivals at $\tau = \tau_s$ and $\tau_s/2$.



(d) Sample the arrival epochs conditioned on the Poisson skeleton.

Figure 5.3: Sampling of a Poisson process on $[0, \tau_s)$ via the midpoint-based Poisson bridge construction with $L = 2$ levels. Solid dots (•) indicate the arrival epochs of the Poisson process. (a-c) show the construction of a Poisson skeleton and (d) depicts the filling in of the Poisson skeleton using ordered uniforms. Note that, by abuse of notation, $\mathcal{P}(\lambda)$ and $\mathcal{B}(N, p)$ represent random variables from the Poisson and binomial distributions, respectively.

5.2.1 Implementation details

For an algorithmic description of a single step of the midpoint-based Poisson bridge construction of the Poisson skeleton and the subsequent filling in using exponential spacings we refer the reader to Appendix 5.B, which is based on [63, Section 15.3]. Multiple steps of the Poisson bridge can be sampled independently by the independent increment property of a Poisson process. Alternatively, [141, Algorithm 2] also describes the midpoint-based Poisson bridge method, but we note that the filling in of the Poisson skeleton should not be done by simulating n uniform random numbers and sorting them as is proposed in [141]. Instead we can efficiently generate n uniform order statistics in $\mathcal{O}(n)$ time using the one-pass method in [63, Section 15.6][1], which is a numerically stable version of the exponential spacings method [48, Section V.3.1].

In order to use variance reduction methods such as QMC or antithetic sampling in combination with the midpoint-based Poisson bridge the Poisson and binomial random variates need to be generated via inverse transform sampling. For the Poisson random variables we use the fast inverse Poisson CDF from [70]. For the binomial random variables to be sampled from an inverse CDF we need to solve $u = I_{1/2}(N - k, 1 + k)$, where $I_x(a, b)$ is the regularised incomplete beta function, for k when $u \in [0, 1]$, and this is a relatively costly operation. For small N this can be done efficiently using look-up tables, but for large N such an approach is inefficient. Instead, to invert the incomplete beta function, we use an adaptation of ideas in [70] that yield a fast approximation to the inverse of the incomplete gamma function.

5.3 Median-based Poisson bridge

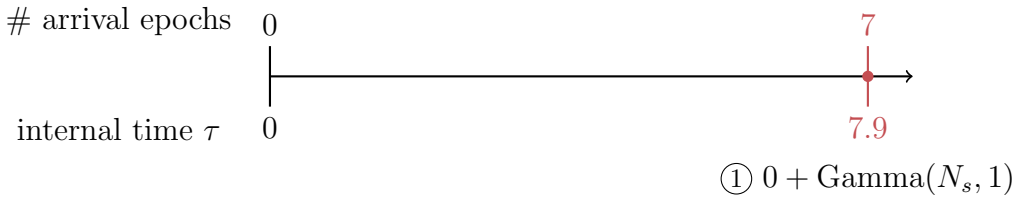
The original median-based Poisson bridge method, also first proposed in [62], is in some sense the dual to the midpoint-based Poisson bridge method. Rather than focussing on sampling the number of arrival epochs for a given internal time, τ, of

[1] We note that the original subroutine as described on p. 346 of [63] contains a typo, the fifth line should read $S \leftarrow S + (q - i + 1)^{-1} \log V_i$.

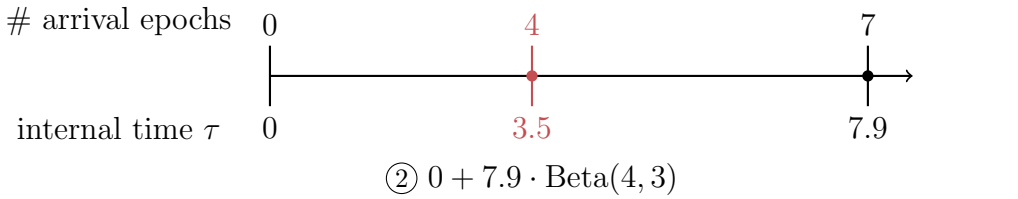
the Poisson process, $Y(\tau)$, in order to create a Poisson skeleton, this Poisson bridge method samples the internal time for a given number of arrival epochs to build a Poisson skeleton. To start we describe a single step of the median-based Poisson bridge construction, where now the step size is a number of arrival epochs, N_s, rather than a fixed internal time. Given that the Poisson process is unit-rate and the fact that the interarrival epochs are i.i.d. according to a unit-rate exponential distribution, we see that the time of the N_s-th arrival is Erlang distributed with parameters N_s and unity, or equivalently Gamma$(N_s, 1)$. This means we can directly sample the N_s-th arrival epoch, τ_{N_s}, from the gamma distribution, which yields a Poisson skeleton $\{\tau_{N_s}, N_s - 1\}$. Note in particular that $Y(\tau_{N_s}^-) = N_s - 1$ holds by observing that the N_s-th arrival epoch is exactly equal to τ_{N_s}. Rather than refining the Poisson skeleton by subdivision and conditional subsampling in the internal Poisson process time, we instead focus on sampling the arrival epoch of the $\lceil N_s/2 \rceil$-th arrival[2] conditional on the N_s-th arrival epoch, i.e. we perform subdivision and subsampling in the number of arrival epochs. This can be achieved by noting that the $N_s - 1$ arrival epochs in an interval $[0, \tau_{N_s})$ are distributed like the order statistics of $N_s - 1$ i.i.d. uniform random variables on $[0, \tau_{N_s})$. Thus, the $\lceil N_s/2 \rceil$-th order statistic of such a collection of uniform random variates is a scaled beta random variable[3], which allows us to directly sample from the beta distribution and apply a scaling to find $\tau_{\lceil N_s/2 \rceil}$. This then creates a new, finer, Poisson skeleton. Note that, contrary to the midpoint-based Poisson bridge, the Poisson skeleton resulting from this procedure has the property that an arrival occurs at each time point in the Poisson skeleton. By recursive subdivision and conditional subsampling according to this dual procedure we arrive at the median-based Poisson bridge, where, again, the number of levels, L, defines how many times we perform the subdivision and subsampling procedure. An illustration of a single step of the $L = 2$ median-based Poisson bridge is shown in Figure 5.4.

[2]Alternatively, we can use the $\lfloor N_s/2 \rfloor$-th arrival, which yields a similar median-based Poisson bridge construction, and in practice we found no difference in performance between them.

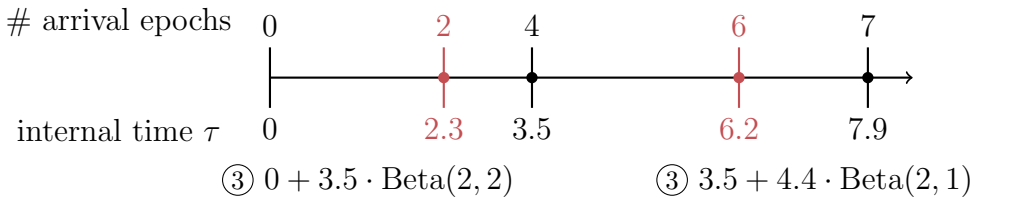
[3]In particular, given that exactly β arrival epochs fall in $[0, \tau)$, the α-th arrival epoch is of the form $\tau \cdot \text{Beta}(\alpha, \beta + 1 - \alpha)$.



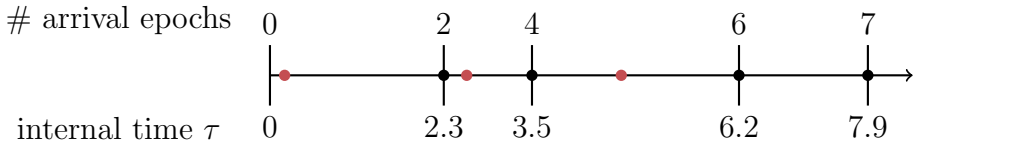
(a) Sample the arrival epoch of the N_s-th event, here we use $N_s = 7$ for illustration purposes.



(b) Sample the arrival epoch of the $\lceil N_s/2 \rceil$-th event, i.e. the median of the interval spanned by the N_s arrival epochs, conditional on the arrival epoch of the N_s-th event.



(c) Sample the arrival epochs of the median of each current interval (if non-empty) conditional on the arrival epochs of the already sampled events.



(d) Sample the remaining arrival epochs conditioned on the Poisson skeleton.

Figure 5.4: Sampling of a Poisson process with a total of N_s (here $N_s = 7$) number of events via the median-based Poisson bridge construction with $L = 2$ levels. Solid dots (•) indicate the arrival epochs of the Poisson process. (a-c) show the construction of a Poisson skeleton and (d) depicts the filling in of the Poisson skeleton using ordered uniforms. Note that, by abuse of notation, $\text{Gamma}(\alpha, \beta)$ and $\text{Beta}(\alpha, \beta)$ represent random variables from gamma and beta distributions, respectively.

We note that the dual relation between the two Poisson bridges can also be seen in the fact that the gamma and beta random variables are, in some sense, the continuous analogues to the Poisson and binomial random variables.

Contrary to the midpoint-based Poisson bridge, where we can increase the number

of levels, L, indefinitely in theory, the number of levels in the median-based Poisson bridge is necessarily bounded by $\lceil \log_2 N_s \rceil$. In addition we observe that the median-based Poisson bridge construction samples arrival epochs both in the second and in the first stage, i.e. in the construction of the Poisson skeleton, whereas the midpoint-based Poisson bridge method only generates arrival epochs in the second stage by filling in the Poisson skeleton. This means that the computational complexity, which we defined as the number of random variables used, of the median-based Poisson bridge method is technically lower than that of the midpoint-based Poisson bridge method. Finally, we mention that the length of the Poisson process $Y(\tau)$, in terms of the internal time, τ, in the case of the median-based Poisson bridge construction is a random variable itself. In terms of the number of reactions, however, the length is fixed and given by the number of steps times the step size N_s. The opposite holds for the midpoint-based Poisson bridge, again highlighting the dual relationship between the two methods, cf. the R-leap and the τ-leap methods.

5.3.1 Implementation details

To implement a single-step of the median-based Poisson bridge we can almost directly follow [63, Section 15.2]. Note, however, that [63, Section 15.2] describes a (hybrid) median-based Poisson bridge of fixed internal time length, i.e. for a given internal time τ one first samples $Y(\tau) \sim \mathcal{P}(\tau)$ as we did in the midpoint-based Poisson bridge method. Thereafter we can use the median-based procedure described in this section to construct a Poisson skeleton and fill this in via the exponential spacing method. We can, however, replace this initial step by fixing N_s and sampling the arrival epoch τ_{N_s}, as described above, from a gamma distribution. A complete description of this procedure can be found in Appendix 5.B.

In order to use variance reduction methods the gamma and beta random variates need to be generated via inverse transform sampling. For the beta random variates this means that we have to invert the regularised incomplete beta function, i.e. solve

$u = I_x(a, b)$ for x when $u \in [0, 1]$, and this is a relatively costly operation. However, as noted in [63, Section 3.3], the beta random variables used in the median-based Poisson bridge are always distributed according to Beta(α, α) or Beta$(\alpha, \alpha - 1)$ and we can therefore use a fast inversion method that is tailored to the symmetrical beta distribution, such as [123]. For the case Beta$(\alpha, \alpha - 1)$ we postulate that the identities relating $I_x(\alpha, \alpha)$ and $I_x(\alpha, \alpha - 1)$ in combination with a method for the symmetrical beta distribution can provide fast inversion too. For the gamma random variates we can, for example, use the efficient numerical inversion of the incomplete gamma function ratios [69] to sample via inverse transform sampling. Finally, after generating the Poisson skeleton we fill in the remaining reaction times using uniform order statistics, which we generate in exactly the same way as described in Section 5.2.

5.4 Antithetic construction of Poisson bridges

In many practical scenarios we are tasked with generating a population of N unit-rate Poisson processes $Y^{(1)}(\tau), \ldots, Y^{(N)}(\tau)$, for example to be used in a MC simulation. If we use the previously described Poisson bridge methods to construct these Poisson processes in the standard MC way, i.e. every Poisson skeleton is independently generated and filled in using standard MC sampling, there is no benefit relative to the standard approaches described in Section 5.1. We can, however, leverage the structure in the Poisson bridge methods via a correlation-based variance reduction technique (see Section 2.4.1) to reduce the overall population variance of $Y^{(1)}(\tau), \ldots, Y^{(N)}(\tau)$, whilst at the same time leaving each individual Poisson process $Y^{(n)}(\tau)$ unbiased. Crucially we will use variance reduction techniques solely in the first stage of the exponential spacings method, i.e. we only employ a variance reduction technique in the generation of the Poisson skeleton, not in the second stage where we fill in the arrival epochs conditional on the Poisson skeleton.

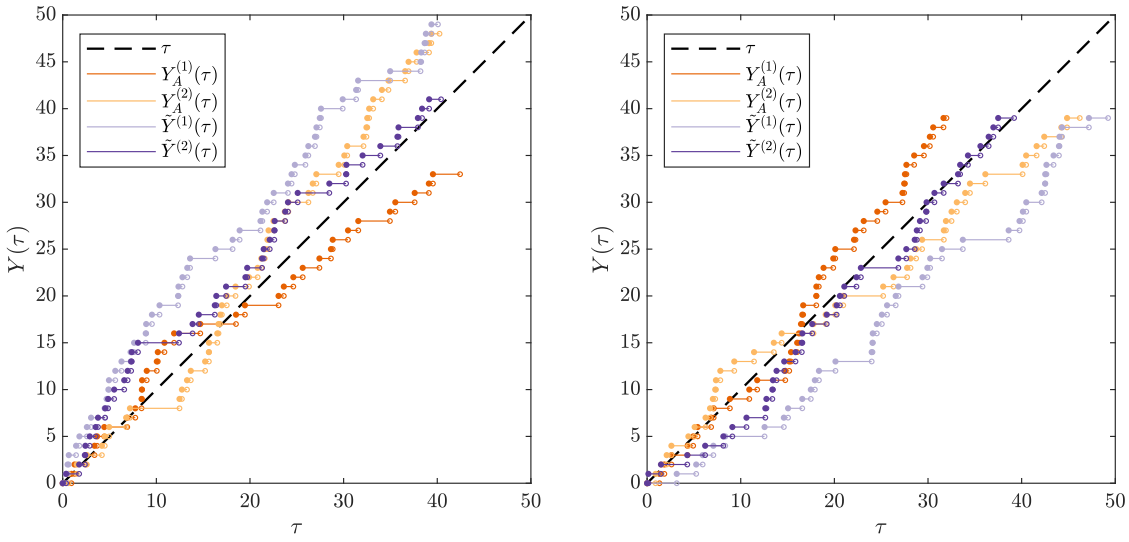
Though the Poisson bridge methods were originally proposed in [62] as a means to generate Poisson processes via QMC sampling, we consider here a simpler variant,

namely antithetic sampling. Note that in the context of antithetic sampling the midpoint-based Poisson bridge was recently rediscovered in [141]. Compared to the QMC version in [62] the antithetic version is, arguably, easier to implement and analyse, whilst at the same time being embarrassingly parallel.

For simplicity assume that we need to sample a population of two Poisson processes, $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$, for $\tau \in [0, \tau_f)$, where from now on we will use τ_f to denote the final internal time of interest of a Poisson process. If we use the midpoint-based Poisson bridge with step size τ_s and L levels we note that to generate the Poisson skeleton for one of the Poisson processes we need $s = \lceil \tau_f/\tau_s \rceil \cdot 2^L$ uniform random variables. We can thus sample $\mathbf{u}^{(1)} \sim \mathcal{U}(0,1)^s$ and $\mathbf{u}^{(2)} \sim \mathcal{U}(0,1)^s$ independently and use inverse transform sampling to construct a Poisson skeleton for $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$. However, rather than generating the second Poisson process by sampling $\mathbf{u}^{(2)}$ independent of $\mathbf{u}^{(1)}$, for the antithetic Poisson bridge we let $\mathbf{u}^{(2)} = 1 - \mathbf{u}^{(1)}$, i.e. the antithetic version of $\mathbf{u}^{(1)}$. Note that if we use this $\mathbf{u}^{(2)}$ to generate $Y^{(2)}(\tau)$, then $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$ will not be independent, but they will both be unbiased Poisson processes since $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ are still (marginally) uniformly distributed in $[0,1)^s$. In fact, it is the hope that, since $\mathbf{u}^{(1)}$ and $\mathbf{u}^{(2)}$ are negatively correlated, $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$ will now also be negatively correlated, which, as shown in Section 2.4.1, can reduce the variance of an estimator that combines $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$. A similar argument applies when we use the median-based Poisson bridge construction for two Poisson processes, but instead now we fix the total number of arrivals over these two Poisson processes to be N_f and take step size N_s and L levels in the Poisson bridge.

In Figure 5.5 we show example realisations of two Poisson processes using both types of Poisson bridge constructions and compare results using standard, independent MC sampling with antithetic MC sampling. It is immediately clear from comparing Figures 5.5(a) and 5.5(b) that the two Poisson bridges when used with standard MC sampling yield independent and uncorrelated Poisson processes. On the other hand, when the Poisson bridge constructions are combined with antithetic sampling

they do indeed produce negatively correlated Poisson processes. However, we also see that the two different Poisson bridge constructions do so in subtly different ways.



(a) Midpoint-based Poisson bridge construction with $\tau_f = 40$, $\tau_s = 2$ and $L = 0$.

(b) Median-based Poisson bridge construction with $N_f = 40$, $N_s = 2$ and $L = 0$.

Figure 5.5: Illustration of the Poisson bridge constructions when sampling two Poisson processes with antithetic sampling, $Y_A^{(1)}(\tau)$ and $Y_A^{(2)}(\tau)$, versus two independently sampled Poisson processes, $\tilde{Y}^{(1)}(\tau)$ and $\tilde{Y}^{(2)}(\tau)$.

The midpoint-based Poisson bridge controls the internal time, τ, of the Poisson processes and thus injects negative correlation at specific internal time points, which importantly are equal for both antithetic Poisson processes. The method thereby aims to have the average of the two antithetic Poisson processes at a given internal time, τ, be close to the true mean of a Poisson process, $\mathbb{E}[Y(\tau)] = \tau$. Note that this corresponds to taking the average along a vertical slice in Figure 5.5(a).

The median-based Poisson bridge, on the other hand, exerts its control via the number of arrival epochs, N, of the Poisson processes and instead injects negative correlation at points where the Poisson processes have a fixed number of arrival epochs, which, again, importantly are equal for both antithetic Processes. Note, however, that this does not mean that the internal times, τ, at which the median-based Poisson bridge attempts to inject negative correlation between the two antithetic Poisson

processes, is equal for both these two Poisson processes. Instead, this method aims to accurately give the arrival epoch τ_N of the N-th arrival when averaging between the two antithetic paths, which now corresponds to averaging across a horizontal slice in Figure 5.5(b). Note that the true expected N-th arrival epoch for a unit-rate Poisson process is given by $\mathbb{E}[\tau_N] = N$. As a by-product we might expect that the average of the two antithetic Poisson processes at a given internal time, τ, will be close to the true mean, as was achieved for the midpoint-based Poisson bridge, but this is by no means guaranteed because we control for a different component of the Poisson process. The same holds of course for the estimation of the N-th arrival epoch in the midpoint-based Poisson process, which we might expect to be more accurate as a by-product of negative correlation induced at fixed time points.

Two natural questions arise now: i) How should we choose the Poisson bridge parameters, τ_s and L, and N_s and L, for the midpoint-based and median-based constructions, respectively, so as to achieve optimal performance? ii) Which of the two Poisson bridge constructions, if any, is to be preferred in practice?

Our analysis of the first question will focus on the midpoint-based Poisson bridge method, but an analysis along similar lines can be used to derive similar results for the median-based Poisson bridge construction.

5.4.1 Midpoint bridge parameters

To answer the first question for the midpoint-based Poisson bridge construction we have to formalise the exact problem setup and specify how we measure performance of the different Poisson process constructions. Here we consider the accurate estimation of (the value of) a Poisson process $Y(\tau)$ over a (finite) time-interval of interest $[0, \tau_f)$. Note that we can do so by generating $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$ on $[0, \tau_f)$ and subsequently look at the mean-estimator

$$\hat{Y}(\tau) = \frac{1}{2}\left(Y^{(1)}(\tau) + Y^{(2)}(\tau)\right), \tag{5.1}$$

which is of course equation (2.17) for the mean summary statistic of two sample paths. We quantify the error of this mean estimator $\hat{Y}(\tau)$ in equation (5.1) via the MSE. To provide a fair comparison between different sampling methods for Poisson processes we assume that all methods generate unbiased Poisson processes, $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$. In addition we normalise the MSE to the number of samples used to calculate the mean estimator, which, under the assumptions made, is given by

$$2 \cdot \mathrm{MSE}(\tau) = \mathrm{Var}\left[Y^{(1)}(\tau)\right] + \mathrm{Cov}\left[Y^{(1)}(\tau), Y^{(2)}(\tau)\right] = \tau + \mathrm{Cov}\left[Y^{(1)}(\tau), Y^{(2)}(\tau)\right]. \tag{5.2}$$

This normalised MSE is invariant to the addition of more pairs of sample Poisson processes to construct $\hat{Y}(\tau)$ if we use standard MC or antithetic MC sampling and represents the population MSE of the respective methods. Note that for standard, independent, MC sampling we find that the normalised MSE is τ and thus grows linearly with the internal time of the Poisson process because $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$ are independent. Using antithetic sampling, however, we saw that a negative correlation could be induced between $Y^{(1)}(\tau)$ and $Y^{(2)}(\tau)$ which by equation (5.2) can be seen to reduce the normalised MSE.

If we were solely interested in accurately estimating the value of $Y(\tau^*)$ at some fixed $\tau^* \in [0, \tau_f)$ the best strategy is to simply sample $Y(\tau^*) \sim \mathcal{P}(\tau^*)$ via antithetic sampling and forgo the sampling of the whole Poisson process on $[0, \tau_f)$. However, this is not a very realistic scenario; more often than not we want to capture the Poisson process over the whole time interval of interest, $[0, \tau_f)$. To this end we consider the IMSE, which accounts for the statistical error throughout the whole time interval. Again we normalise the IMSE to the number of sample paths to find

$$2 \cdot \mathrm{IMSE}(\tau_f) = 2 \int_0^{\tau_f} \mathrm{Var}\left[\frac{1}{2}\left(Y^{(1)}(\tau) + Y^{(2)}(\tau)\right)\right] \mathrm{d}\tau. \tag{5.3}$$

Division of the IMSE by τ_f gives an average MSE over the interval $[0, \tau_f)$ and thus minimising the (normalised) IMSE can be seen as finding method parameters that

achieve the lowest average statistical error across the whole time interval of interest. Note that for standard MC sampling we find that the normalised IMSE is given by $\tau_f^2/2$, i.e. it grows quadratically with the internal time of the Poisson process.

As noted in [141] we must strike a balance in the choice of L and τ_s (or N_s for the median-based Poisson bridge) in order to achieve good performance, both in terms of the IMSE and in terms of the computational complexity. Good parameter choices should thus follow from solving a constrained optimisation problem and though some limited theory presented in [141] hints that there might be a range of suitable parameters for which antithetic sampling is very efficient, the results are not sufficient as a guide to estimating optimal parameter choices. To tackle this problem we therefore present here two strategies, optimal in terms of the normalised IMSE, that (approximately) solve the relevant optimisation problems and are also very easy to implement. For a detailed discussion and derivation of the results that follow we refer the reader to Appendix 5.A.

Strategy 1: optimal step size for a fixed number of levels

The first strategy covers the scenario in which we *a priori* fix the number of levels, L, used by the midpoint-based Poisson bridge. To generate two antithetic Poisson processes using the midpoint-based Poisson bridge construction with minimal IMSE we should in this case take a step size

$$\tau_s^* = \sqrt{2^L} \cdot \sqrt{3\gamma} \cdot \sqrt{\tau_f}, \tag{5.4}$$

where we have introduced the constant $\gamma = 7/36$ (see Appendix 5.A). Note that for equation (5.4) to hold we need $\tau_s^* < \tau_f$ and thus $2^L < \tau_f$. This optimal step size is derived by minimising the IMSE without regard for computational complexity, which obviously increases if we increase L. Nonetheless, we found equation (5.4) to be effective at predicting a (near) optimal step size, even when taking computational

complexity into account, if $2^L < \tau_f/4$.

To show the benefit of using the antithetic midpoint-based Poisson bridge construction we note that in the limit of $\tau_f \to \infty$ with L fixed we can, to first order, ignore any computational overhead from the Poisson bridge construction, which yields the following results (see Appendix 5.A)

$$\mathrm{IMSE}(\tau_f; \tau_s^*) = \frac{1}{\sqrt{2^L}} \cdot \frac{\gamma}{3} \cdot \tau_f^{3/2} + \mathcal{O}(\tau_f), \tag{5.5a}$$

$$\mathrm{VRF}(\tau_f) = \sqrt{2^L} \cdot \sqrt{\frac{3}{4\gamma}} \cdot \sqrt{\tau_f} + \mathcal{O}(1), \tag{5.5b}$$

where the VRF is defined relative to the standard, independent, MC sampling of a Poisson process. This means that, especially for large time intervals of interest $[0, \tau_f)$, the combination of antithetic sampling and a Poisson bridge construction can provide large improvements over the conventional sampling approaches in Section 5.1.

Though equation (5.5) indicates that increasing the number of levels, L, subject to the previously mentioned restrictions, improves the relative efficiency of the antithetic Poisson bridge method, we note that if we want to take L such that $2^L = \mathcal{O}(\tau_f)$ it is often preferable to employ Strategy 2, which combines both computational complexity and statistical error.

Strategy 2: optimal number of levels for exactly one step

The previous strategy provides optimal parameters if we are constrained by the number of levels, L, that we can use in the midpoint-based Poisson bridge. However, as already briefly alluded to, increasing L can improve the efficiency and so we now present the optimal strategy if both the number of levels, L, and the step size, τ_s, can be chosen freely. In this scenario we have to take into account both the (normalised) IMSE and the computational complexity, and combining both contributions

the optimal parameter combination is given by

$$\tau_s^* = \tau_f, \tag{5.6a}$$

$$L^* = \left\lfloor 0.94 \frac{\log 0.9\tau_f}{\log 2} \right\rfloor. \tag{5.6b}$$

Note that employing this strategy we take exactly one step of the midpoint-based Poisson bridge, but use many levels so that $\tau_s^*/2^L = \mathcal{O}(1)$, i.e. we construct a Poisson skeleton on an $\mathcal{O}(1)$ spaced time grid. Using this optimal strategy we find that

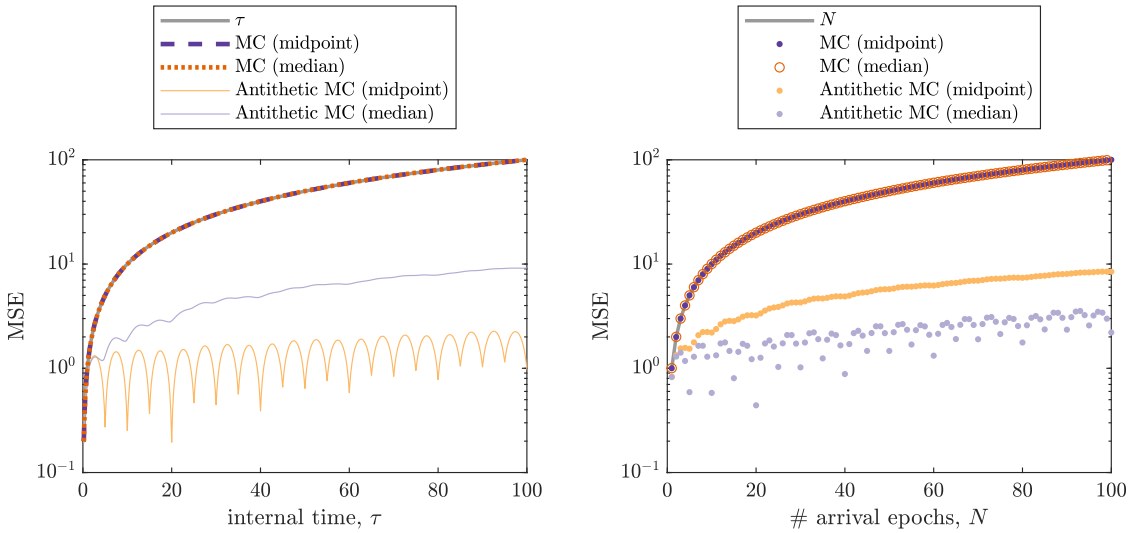
$$\text{IMSE}(\tau_f; \tau_s^*) \approx \frac{\tau_f}{4}\left(1 + \frac{\log \tau_f}{3}\right), \tag{5.7}$$

which means that we effectively created a method that samples a Poisson processes with an IMSE that grows linearly in the length of the Poisson process, rather than quadratically as is custom for the standard methods. In other words, the average MSE only grows logarithmically in the length of the Poisson process and is $\mathcal{O}(1)$ for any reasonable value of τ_f one is likely to encounter. Unless stated otherwise we will therefore use this strategy to construct the optimal antithetic midpoint-based Poisson bridge.

5.4.2 Comparison of different Poisson bridges

Though the analysis of the MSE as a function of internal time, τ, in Appendix 5.A does not hold for the median-based Poisson bridge we can still compare the two methods via simulation. To provide a fair comparison between the midpoint-based and median-based Poisson bridge we consider both the MSE of the value of a Poisson process, $Y(\tau)$, and the MSE of the N-th arrival epoch, τ_N, of a Poisson process in Figure 5.6.

Though both methods have, in some sense, equivalent parameters we see that, depending on the quantity of interest, it is clear to see which method performs the



(a) MSE of the value of a Poisson process, $Y(\tau)$, as a function of the internal time, τ.

(b) MSE of the N-th arrival epoch, τ_N, of a Poisson process as a function of the number of arrival epochs, N.

Figure 5.6: Comparison of statistical error between the median-based and midpoint-based Poisson bridge constructions, either using standard MC or via antithetic MC sampling. Both methods use $L = 2$ levels and the median-based Poisson bridge construction has step size $N_s = 20$, whereas the midpoint-based Poisson bridge construction uses step size $\tau_s = 20$. Data computed by averaging over 2^{20} samples.

best. As expected, when we are interested in having an accurate estimate of the value of a Poisson process for a given internal time, τ, the midpoint-based Poisson bridge method should be preferred. It can be seen in Figure 5.6(a) that this method indeed does introduce negative correlations at specific, regular, time points and therefore reduces the overall MSE across the whole time interval of interest[4]. The median-based Poisson bridge, however, produces two antithetic sample paths which, in terms of the internal time, do not have negative correlation introduced at synchronised time points and thus the MSE as a function of the internal time, τ, is significantly larger than that of the midpoint-based Poisson bridge. For comparison we note that the (normalised) IMSE over $[0, 100)$ is (approximately) given by 5000, 142 and 567 for

[4]The oscillatory behaviour of the MSE is due to the fact that regular MC sampling is used conditioned on a Poisson skeleton, which has lower variance when we use a variance reduction technique such as antithetic sampling. The MSE therefore increases in between the constrained points of the Poisson skeleton.

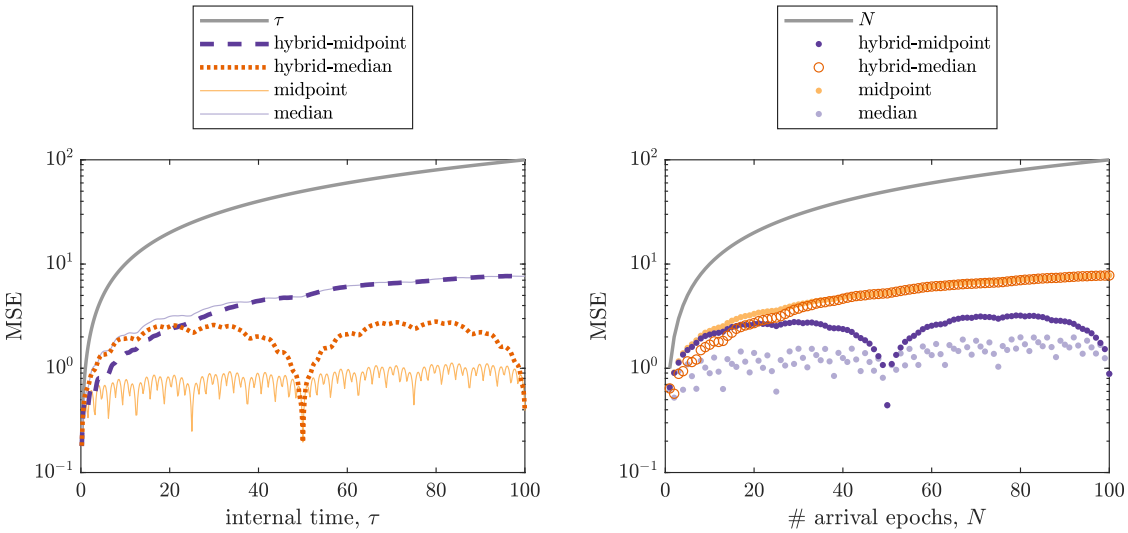
the standard MC construction, antithetic midpoint-based Poisson bridge construction and the antithetic median-based Poisson bridge construction, respectively.

If we focus on estimation of the arrival epochs in Figure 5.6(b) we see that the reverse holds, and the median-based Poisson bridge is to be preferred. Importantly, we note that both methods outperform the standard MC sampling methods in all scenarios. For reference, the IMSE (which is now a sum over N rather than an integral over τ) for the number of arrival epochs between zero and 100, is given by 5050, 549 and 215 for the standard MC construction, antithetic midpoint-based Poisson bridge construction and the antithetic median-based Poisson bridge construction, respectively.

Antithetic hybrid Poisson bridges

As the median-based Poisson bridge construction seems to suffer from a lack of temporal synchronisation we can instead use fixed time steps of size τ_s, in a similar manner to the midpoint-based Poisson bridge, whilst at the same time still use the median-based subdivision and subsampling procedure. This method, which we denote as the hybrid-median Poisson bridge, was the original format of the median-based Poisson bridge in [62]. Similarly, we can use fixed steps in the number of arrivals, N_s, in a manner akin to the median-based Poisson bridge, but subsequently use the midpoint-based subdivision and subsampling procedure, yielding the hybrid-midpoint Poisson bridge construction. The hope is that by combining both methods we can see improved performance. However, we see in Figure 5.7 that this is not the case. In fact, the lack of synchronisation in both hybrid methods remains, and increases the MSE, especially when L increases.

We therefore recommend the use of the (standard) midpoint-based Poisson bridge in the most common scenario in which we are interested in having an accurate estimate of $Y(\tau)$ for a given time, or time interval. If for a specific application one is interested in having an accurate estimate of the arrival epochs, given a specified number of arrivals, then the median-based Poisson bridge is the better method. However,



(a) MSE of the Poisson process value, $Y(\tau)$, as a function of the internal time, τ.

(b) MSE of the N-th arrival epoch, τ_N, of a Poisson process as a function of the number of arrival epochs, N.

Figure 5.7: Comparison of statistical error between the median-based and midpoint-based Poisson bridge constructions and hybrid variants, all using antithetic MC sampling. All methods use $L = 5$ levels and the median-based and hybrid-midpoint Poisson bridge constructions have step size $N_s = 50$, whereas the midpoint-based and hybrid-median Poisson bridge constructions use step size $\tau_s = 50$. Data computed by averaging over 2^{20} samples.

because in the context of chemical reaction network simulations it is more important to accurately capture the value of a Poisson process over a time interval we will from now on solely consider the midpoint-based Poisson bridge.

5.4.3 Chemical reaction network examples

In this section we consider the use of Poisson processes constructed via antithetic Poisson bridge methods in the context of chemical reactions. More specifically, we use the MNRM [1], see Algorithm 2.2, in combination with antithetic Poisson processes and we denote this combined method as the antithetic MNRM from now on. Though this concept appeared recently in [141] we note that that work is mainly a proof-of-concept and provides little guidance as to how one should implement the antithetic MNRM. Here we address this issue and provide some ideas on how we can relate

the variance reduction on the level of the input Poisson processes to the observed performance benefit when used in chemical reaction network simulations.

As we have done so far we consider both the MSE at fixed (physical system) times t and the IMSE over an interval $[0, T)$ of a mean copy number estimator to quantify the statistical error of the antithetic MNRM and its efficiency relative to the standard MNRM. To calculate the IMSE in practice we sample the MSE at 200 equispaced time points in $[0, T)$ and use the trapezoidal rule to compute the IMSE. Extensions to other functionals of the sample paths, e.g. species distributions or exit time statistics, are left for future work.

Poisson bridges for chemical reaction networks

The first hurdle to overcome when we want to effectively use the (antithetic) Poisson bridge construction in combination with the MNRM is the discrepancy between the physical system time, t, and the internal times, τ_k, of the Poisson processes $Y_k(\tau_k)$, where we recall $k = 1, \ldots, K$ for a system with K reaction channels. Via the RTCR in equation (2.7) we see how the two quantities relate to each other, but it is *a priori* impossible to determine $\tau_{k,f}$, i.e. the final internal times of the Poisson processes for each reaction channel, given that the physical time $t \in [0, T)$, unless the reaction is zero-th order. In fact, for any reaction channel not obeying zero-th order mass action kinetics the final internal time for the relevant Poisson process is itself a random variable. This poses a problem because we have seen in the previous sections that the choice of optimal parameters for the Poisson bridge methods is intricately related to the time domain $[0, \tau_f)$ over which we want to accurately generate the Poisson processes. Moreover, we note that various reaction channels can have very different final internal times for a single given physical final time T, this is for example very clear in systems with fast and slow reaction channels.

To circumvent the issue of not knowing $\tau_{k,f}$ *a priori* Maginnis et al. [141] use a fixed time step, τ_s, and number of levels, L, valid for all reaction channels. They use

these parameters repeatedly in the midpoint-based Poisson bridge method until the simulation has reached final physical time T. To establish a sensible choice of step size, τ_s, and number of levels, L, they run multiple MC simulations for a range of parameters to empirically identify a good combination of τ_s and L. In light of the previous two issues, namely that an efficient step size and number of levels depend strongly on $\tau_{k,f}$, and the possibility of multiple scales, in terms of the final internal times $\tau_{k,f}$, for the different reaction channels, this method is not only impractical, as there is no cheap way to determine such a fixed step size and number of levels, but it is also sub-optimal in terms of efficiency.

The first improvement we suggest is to allow L_k and $\tau_{k,s}$ to be different for each reaction channel. Effectively this means that we treat the Poisson processes relating to each reaction channel independently and simply ask how we can generate each individual process optimally given that we want to know the value of the Poisson process $Y_k(\tau_k)$ for $\tau_k \in [0, \tau_{k,f})$. Note that the choice of parameters, L_k and $\tau_{k,s}$, is then done via either of the two strategies in the previous section, which guarantees that we generate two antithetic Poisson processes that have minimal average MSE across the time interval of interest. Though this does not necessarily guarantee the best possible overall strategy, as we will see in the examples that follow, it is a practical choice that is easy to implement and in a worst case scenario has at most twice the computational complexity of the standard MNRM.

This choice, however, does leave us with the issue of determining $\tau_{k,f}$ for each reaction channel. As mentioned earlier, for most types of reaction kinetics we now have a random-horizon problem, where the final internal time, $\tau_{k,f}$, is a random variable, with an *a priori* intractable distribution. The obvious and practical solution to this problem is found in [63], which recommends to take the 0.95-quantile of the empirical distribution for the $\tau_{k,f}$ from a small set of pilot runs, say in the range of 10^2 to 10^3 samples. In the next example we show, however, that in the context of chemical reaction networks this is not necessarily the best strategy from an efficiency

point of view. As a second improvement we therefore suggest to use the median of the empirical distribution of the pilot runs, as we found that using the median, or the 0.5-quantile, was more efficient in terms of computational complexity and almost always equally efficient in terms of the VRF achieved. As a side note, to implement a version of the antithetic median-based Poisson bridge we recommend a similar procedure, but now using the pilot runs to generate the empirical distribution for $N_{k,f}$, i.e. the number of times reaction channel k fires over the physical time domain of interest $t \in [0, T)$. An alternative strategy, which we do not explore here further, is the use of cheaper, approximate model frameworks, such as the RRE or the CLE, to perform the pilot runs to estimate $N_{k,f}$ via an extended reaction system as defined, for example, in Example 2.4.

We can now summarise our actual implementation of the antithetic MNRM in practice. First we run a small set of pilot simulations using the standard MNRM in which we keep track of the internal times of each of the Poisson processes, $Y_k(\tau_k)$. From the empirical distribution of the final internal times, $\hat{\tau}_{k,f}$, we use the median to select a suitable step size, $\tau_{k,s}$, and number of levels, L_k, via equation (5.6). We then use antithetic sampling for each reaction channel individually over the respective intervals $[0, \hat{\tau}_{k,f})$ and supply these Poisson processes as input to the (antithetic) MNRM. Note that using this method means there will be sample paths of the MNRM which need to access the value of $Y_k(\tau_k)$ for $\tau_k > \hat{\tau}_{k,f}$. In this case we simply use the standard MC procedure from Section 5.1 to sample the arrival epochs for $\tau_k > \hat{\tau}_{k,f}$ on-the-fly. Note that this procedure still results in unbiased Poisson processes and therefore an overall unbiased estimator of the chemical species sample paths. Though this completion of the Poisson processes using standard MC sampling will increase the MSE of both the Poisson processes and the chemical species sample paths, we have chosen $\hat{\tau}_{k,f}$ in such a way that these corrections not only occur in just some of the sample paths, but also only for a few reactions. We thus found this "topping up" of the Poisson processes to have little influence on the overall performance and

efficiency in practice.

Linear monomolecular system

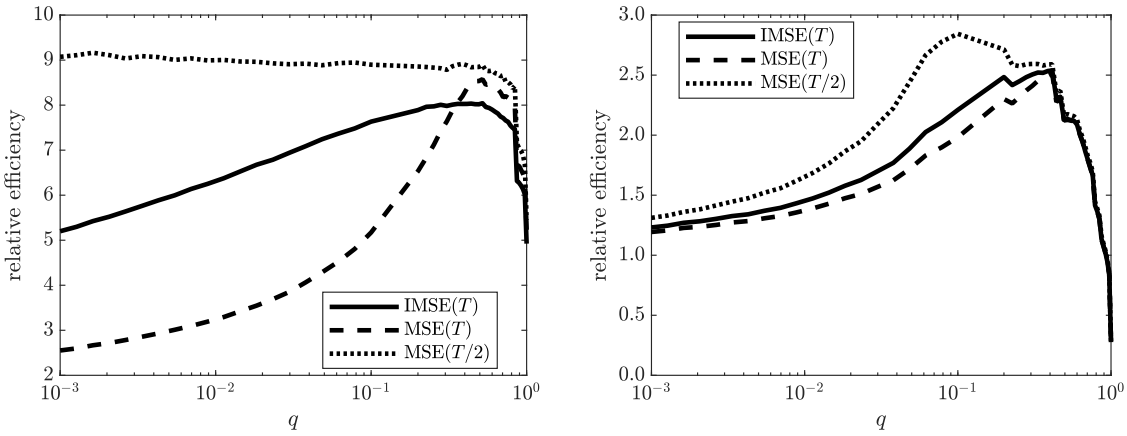
To start we consider the simple monomolecular reaction system we first encountered in Section 2.4 which comprises a single species, S_1, undergoing three different reactions

$$\emptyset \xrightarrow{c_0} S_1, \qquad S_1 \xrightarrow{c_1} 2S_1, \qquad S_1 \xrightarrow{c_2} \emptyset. \tag{5.8}$$

Note that though the three reaction channels are represented by independent Poisson processes they interact in a non-additive way via the RTCR, see equation (2.7), when used to construct sample paths for S_1. Due to this interaction it is *a priori* not clear whether the use of antithetic Poisson processes will in fact provide a benefit, and if so, how much gain in efficiency we will observe.

To investigate the efficacy of the antithetic MNRM we start with the problem of selecting the length, or final internal time, $\tau_{k,f}$, of the unit-rate Poisson processes that we need to generate when used in a chemical reaction system. As discussed earlier the only available suggestion in the literature for such random-horizon Poisson processes is to perform a pilot run and compute the 0.95-quantile from the observed internal times of the Poisson processes. We note, however, that in doing so we might be simulating Poisson processes of lengths larger than strictly necessary to observe a performance benefit. In particular, we note that the complexity of the standard MNRM is equal to the total of the mean internal final times of the Poisson processes, which can be much less than the sum of the 0.95-quantiles. We therefore generalise the suggestion in [63, Chapter 3] and consider q-quantiles for $q \in [0, 1]$ from the pilot run to select the internal times, $\tau_{k,f}$, and the results are shown in Figure 5.8.

Unsurprisingly we see that if we take q small the efficiency improvements are small. This can be understood by observing that in those cases the antithetic Poisson processes generated over $[0, \hat{\tau}_{k,f})$ are not long enough to cover the whole physical time



(a) $c_0 = 4, c_1 = 0.08$ and $c_2 = 0.08$.

(b) $c_0 = 0.4, c_1 = 0.7$ and $c_2 = 0.6$.

Figure 5.8: Effect of changing the final internal time, $\tau_{k,f}$, used to construct the midpoint-based Poisson bridge for each of the Poisson processes, $Y_k(\tau)$, representing a reaction channel. We consider taking the largest q-quantile of the empirical distribution of $\tau_{k,f}$ from pilot runs and look at the efficiency of the antithetic MNRM relative to the standard MNRM when we vary q. We use Strategy 2 from Section 5.4.1 to select the number of levels, L, and time step, τ_s, independently for each reaction channel Poisson process. Model studied is the linear system (5.8) in a volume $V = 1$ with initial condition $\mathbf{X}_0 = 10$ and final time $T = 20$.

domain $[0, T)$ and therefore a significant proportion of the time domain of interest will be simulated using standard MC, which increases the variance. Interestingly we see in Figure 5.8(a) that at time $T/2$ the MSE seems insensitive to the value of q. This follows from the fact that for any q we select the correct $\tau_{1,f}$ for the first reaction channel in the system (5.8) because it is a zero-th order reaction, i.e. the length of this Poisson process is always equal to $\tau_{1,f} = c_0 VT$. Furthermore, we note that for small time with the parameters in Figure 5.8(a) the dynamics of the system are dominated by this first reaction channel, and therefore are virtually unaffected by the choice of q. However, for later times, e.g. time $T = 20$ in Figure 5.8(a), or different model parameters, e.g. Figure 5.8(b), we see that the effect of the other two (first-order) reaction channels is more prominent. Perhaps more surprising is the sharp decrease in efficiency when q becomes close to unity, for both sets of model parameters. This is due to the fact that any extra gain from generating the antithetic Poisson processes for larger internal times is outweighed by the extra computational complexity that

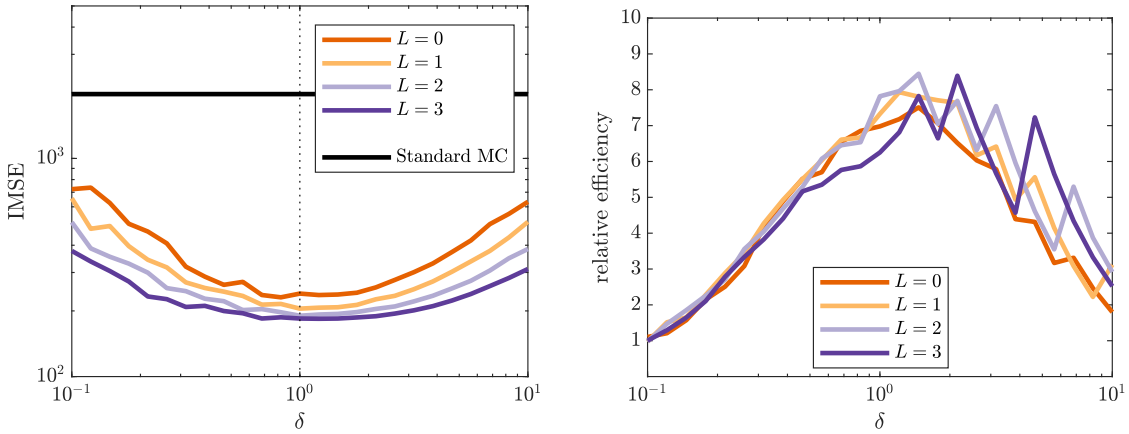
is incurred in doing so. Similar studies of the effect of q on other example systems show comparable results and we therefore conclude that taking $q = 0.95$ is, in fact, a bad idea from a complexity point of view. Instead we propose to use $q = 0.5$, i.e. the median of the pilot runs, as a feasible, stable, and at the same time easy to implement, strategy to balance variance reduction and computational complexity.

Now that we have established a good procedure for choosing the internal times, $\tau_{k,f}$, for the Poisson processes representing each of the reaction channels we will verify the validity of the optimal parameter choices proposed in Section 5.4.1. First we fix the number of levels to a constant across all the reaction channels, i.e. $L_k \equiv L$, and consider the use of Strategy 1, equation (5.4), to select the optimal step size. The resulting IMSE over the interval $[0, T)$ is shown in Figure 5.9(a) and we see that indeed the IMSE is minimised when $\tau_{k,s}$ is given by equation (5.4). Furthermore, when we consider the efficiency of antithetic MNRM relative to the standard MNRM we observe that the antithetic method is roughly eight times more efficient when we use Strategy 1 to select the step size.

Next we consider the second strategy from Section 5.4.1, which takes the step size $\tau_{k,s} = \tau_{k,f}$ for each reaction channel, but optimises the number of levels using equation (5.6). In Figure 5.10 we show that, whilst L^* using equation (5.6) does not yield the absolute best relative efficiency, it is very close to the true optimal.

This discrepancy arises due to the fact that L^* is chosen so as to minimise the variation of individual reaction channels but, due to the interaction of the reaction channels via the random time-change in the RTCR, this is not the sole contribution to the IMSE of the overall chemical reaction sample paths. We can quantify the effects of these interactions by considering the analysis of variance (ANOVA)-decomposition of the problem [201], given in alternative form by

$$\mathrm{Var}\left[X_i(t)\right] = \sum_{k=1}^{K} \mathrm{Var}\left[\mathbb{E}\left[X_i(t) \mid Y_k(\tau)\right]\right] + V_{\mathrm{HOI}}, \tag{5.9}$$

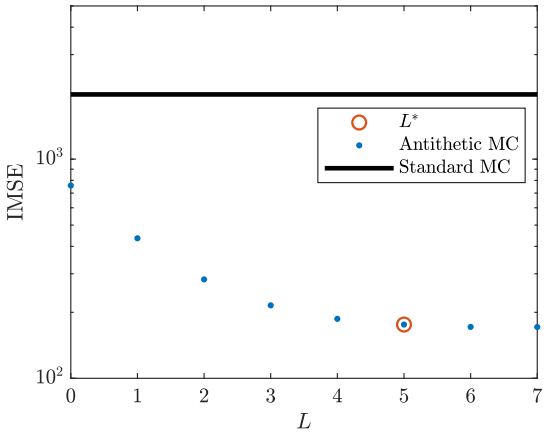


(a) IMSE at time T.

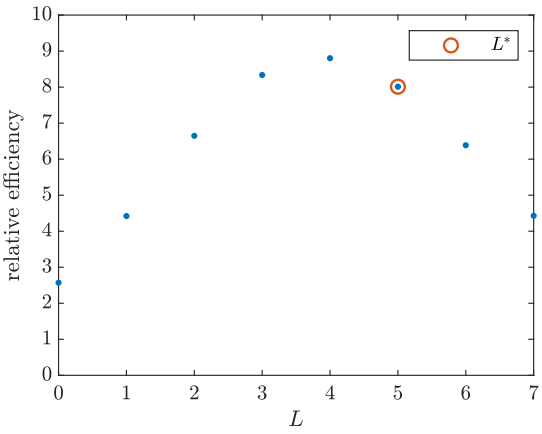
(b) Efficiency relative to the standard MC method.

Figure 5.9: Study of the efficiency of the antithetic midpoint-based Poisson bridge method when using Strategy 1 in Section 5.4.1 to select the method parameters. We let $\tau_s = \delta\tau_s^*$, where τ_s^* is given in equation (5.4), for each reaction channel, while fixing the number of midpoint levels L. Model studied is the linear system (5.8) in a volume $V = 1$ with $c_0 = 4$, $c_1 = 0.08$ and $c_2 = 0.08$, initial condition $\mathbf{X}_0 = 10$ and final time $T = 20$. Each data point corresponds to 10^6 simulations and we use a pilot run with 10^3 simulations to determine the final internal times, $\tau_{k,f}$, for the unit-rate Poisson processes via the medians of the pilot run distribution.

where V_{HOI} is the sum of all higher order interaction partial variance terms in the orthogonal ANOVA-decomposition. Some authors refer to the ANOVA-decomposition as the Sobol'-Hoeffding-decomposition of the variance and for more information we refer the reader to [162, Appendix A]. Now we note that the antithetic MNRM primarily attempts to reduce the total variance $\text{Var}\left[X_i(t)\right]$ by reducing the first order effects due to the individual reaction channels, i.e. it lowers $\text{Var}\left[\mathbb{E}\left[X_i(t) \mid Y_k(\tau)\right]\right]$. However, there is no guarantee that the interaction terms, contained in V_{HOI}, are reduced by this antithetic sampling method. We therefore also see that if we could establish a method, such as QMC sampling, that decreases the variance due to the first order effects further than the antithetic sampling method without dealing with the higher order interactions, there will be a point where the higher-order interaction terms will start to dominate $\text{Var}\left[X_i(t)\right]$ and thus the benefits of such a method relative to the antithetic sampling approach will be limited. In the example studied in



(a) IMSE at time T.



(b) Efficiency (based on IMSE at time T) relative to the standard MC method.

Figure 5.10: Study of the efficiency of the antithetic midpoint-based Poisson bridge method when using Strategy 2 in Section 5.4.1 to select the method parameters. We let $\tau_{k,s} = \tau_{k,f}$ for each reaction channel, but vary the number of midpoint levels L. Theoretical optimal L^* is given by equation (5.6). Model studied is the linear system (5.8) in a volume $V = 1$ with $c_0 = 4$, $c_1 = 0.08$ and $c_2 = 0.08$, initial condition $\mathbf{X}_0 = 10$ and final time $T = 20$. Each data point corresponds to 10^6 simulations and we use a pilot run with 10^3 simulations to determine the final internal times, $\tau_{k,f}$ for the unit-rate Poisson processes via the medians of the pilot run distribution. Note that for this system we have $\tau_{k,f} \approx 80$ for each reaction channel, and thus $L_k^* \equiv L^* = 5$.

Figure 5.10 we thus see that when $L \geq 4$ the first order variance contribution due to the individual reaction channels has been reduced to the extent that the contribution from mixing effects starts to become relatively sizeable. Further increasing the number of levels, L, then comes at a higher computational cost, whilst reducing the IMSE only a small amount. In this case we find that a slightly smaller number of levels gives a slightly higher IMSE, but at a relatively smaller computational complexity in this particular case.

Via the ANOVA-decomposition we can also immediately conclude that the antithetic MNRM will be most efficient for systems that are (nearly) additive in their contributions from each of the reaction channels. Though this property is hard to predict *a priori* for a given chemical reaction network, the next two examples will show that for more complicated chemical reaction networks the ANOVA-decomposition

can be indicative of the effectiveness of the antithetic MNRM, but that it is not able to completely predict the performance benefits. These examples are taken from [124], where the authors specifically study the ANOVA-decomposition when using mean-estimators in the context of chemical reaction networks.

Michaelis-Menten system

First we consider the Michaelis-Menten system, given by four species interacting via to the following three reactions

$$S_1 + S_2 \xrightarrow{c_1} S_3, \qquad S_3 \xrightarrow{c_2} S_1 + S_2, \qquad S_3 \xrightarrow{c_3} S_4 + S_2. \tag{5.10}$$

Following [124] we look at the evolution of the species for $t \in [0, 60]$. We start with the initial condition $\mathbf{X}_0 = (300, 120, 0, 0)^\intercal$ and also fix $c_1 = 0.0017$ and $c_3 = 0.125$ while we take c_2 equal to either 10^{-3} or $25 \cdot 10^{-3}$. The results comparing the standard and antithetic versions of the MNRM are shown in Table 5.1.

		$c_2 = 25 \cdot 10^{-3}$			$c_2 = 10^{-3}$		
		S_1	S_2	S_4	S_1	S_2	S_4
VRF	$L_k = L_k^*$	5.0	2.9	9.2	5.3	3.1	8.7
	$L_k = L_k^* - 2$	4.7	2.7	9.0	5.2	3.1	8.5
	$L_k \equiv 5$	4.9	2.8	8.9	5.2	3.1	8.5
relative efficiency	$L_k = L_k^*$	3.5	2.0	6.6	3.7	2.2	6.1
	$L_k = L_k^* - 2$	4.3	2.5	8.1	4.7	2.8	7.6
	$L_k \equiv 5$	4.3	2.5	7.9	4.5	2.6	7.3

Table 5.1: Performance metrics for the antithetic MNRM relative to the standard MNRM. We use Strategy 2 from Section 5.4.1 to tune the Poisson bridge construction and compare the use of L^* and $\max(0, L^* - 2)$ according to equation (5.6) with the best possible simple strategy that picks a single number of levels L_k, identical for each reaction channel. We consider the Michaelis-Menten system (5.10) with initial condition and parameters as described in the text. The results are calculated using 10^6 samples. Under these conditions we have $(L_1^*, L_2^*, L_3^*) = (7, 5, 7)$ and $(L_1^*, L_2^*, L_3^*) = (7, 1, 7)$ for $c_2 = 25 \cdot 10^{-3}$ and $c_2 = 10^{-3}$, respectively.

We see that, though the parameters given by Strategy 2, equation (5.6), are optimal in the sense that they provide the largest VRF, in terms of efficiency it can be better to consider fewer levels, L, in the midpoint-based Poisson bridge. In particular we see that the best possible strategy is taking $L_k = L_k^* - 2$, i.e. still optimise the parameters for each reaction channel individually, but use two fewer levels than Strategy 2 prescribes.

To link the performance of the antithetic MNRM to the ANOVA-decomposition we consider the difference between the variance of the standard and antithetic versions of the MNRM in Figure 5.11. If we compare this with [124, Figure 12] we see that the variance reduction achieved by the antithetic MNRM accurately matches with the sum of the first-order variance contributions in the ANOVA-decomposition. We therefore reiterate our earlier conclusion that the antithetic MNRM is effective at integrating out the first-order contributions from reaction channels to the variance, but not necessarily any higher-order interactions. The ANOVA-decomposition can also explain the difference in performance when looking at the different species in the Michaelis-Menten system as [124, Figure 12] shows that the variance of S_4 is nearly completely additive in the three reaction channels, whereas there is a significant variance contribution due to interactions for S_2.

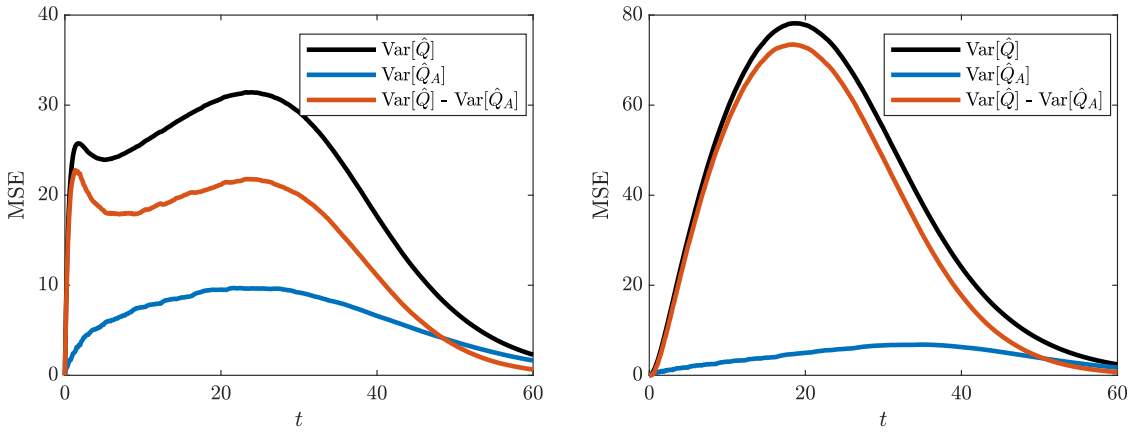
To elucidate an even more impressive performance of the antithetic MNRM we note that under certain conditions we can use the quasi-steady state approximation (QSSA) to find a reduced model with only a single reaction channel

$$S_1 \xrightarrow{a_1(\mathbf{X})} S_4, \tag{5.11}$$

where the reaction propensity is now given by the Michaelis-Menten propensity function

$$a_1(\mathbf{X}) = \frac{c_3 X_2 X_1}{X_1 + \frac{c_2+c_3}{c_1}}. \tag{5.12}$$

Note that in this model S_2 and S_3 are assumed to remain constant. Due to the

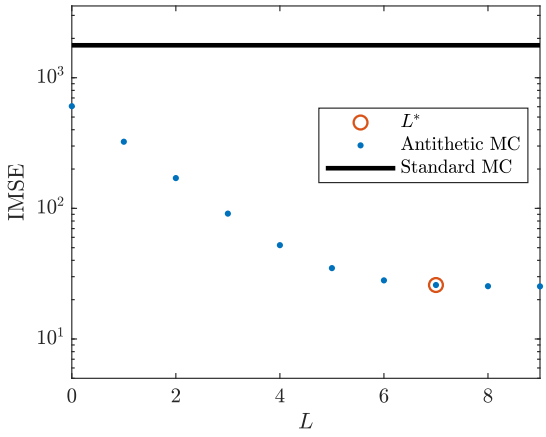


(a) S_2 (b) S_4

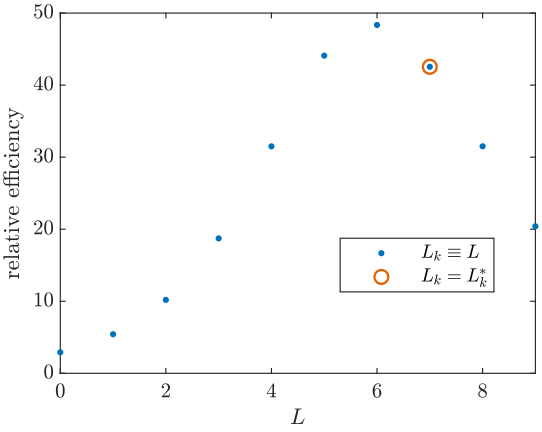
Figure 5.11: MSE for the Michaelis-Menten system, (5.10), with $c_2 = 10^{-3}$ and other parameters and initial conditions as described in the text. We compare the results from using the standard MNRM, $\hat{Q}$, against the equivalent results when we use the antithetic MNRM, $\hat{Q}_A$. The results were calculated using 10^6 samples and the number of levels for the Poisson bridge for each channel is given by $L_k = \max(0, L_k^* - 2)$ according to equation (5.6).

reduction of the chemical reaction network we note that no higher order interaction terms between different reaction channels exist and thus we expect that the antithetic MNRM should be very efficient if we want to estimate the average copy number of S_1 or S_4 in (5.11). This is verified in Figure 5.12, where we see that we can achieve an almost 50-fold improvement in the efficiency when using the antithetic MNRM.

Based on the relative efficiency results of a unit-rate Poisson process (see Appendix 5.A) we would expect to see roughly a 75-fold improvement for $L^* = 8$, but we see in Figure 5.12 that this is not achieved in practice. Because for this example there is only a single reaction channel we cannot attribute this discrepancy to any higher order interaction terms. Instead we note that the result for the unit-rate Poisson processes is derived under a deterministic and fixed final internal time τ_f. For a chemical reaction network, however, we have a random final internal time for the Poisson process and this is the reason that we lose part of the predicted efficiency gains. In particular, we see that L^* given by equation (5.6), again, is an over-prediction of the true optimal number of levels, L.



(a) IMSE at time T.



(b) Efficiency (based on IMSE at time T) relative to the standard MC method.

Figure 5.12: Study of the antithetic midpoint-based Poisson bridge method using Strategy 2 in Section 5.4.1 to select the method parameters. Theoretical optimal L^* is given by equation (5.6). Model studied is the QSSA version of the Michaelis-Menten system, (5.11), with parameters and initial conditions described in the text and final time $T = 60$. Each data point corresponds to 10^6 simulations and we use a pilot run with 10^3 simulations to determine the final internal time $\tau_{1,f}$ for the unit-rate Poisson process via the median of the pilot run distribution. Note that for this system we have $\tau_{1,f} \approx 210$ and thus $L^* = 7$.

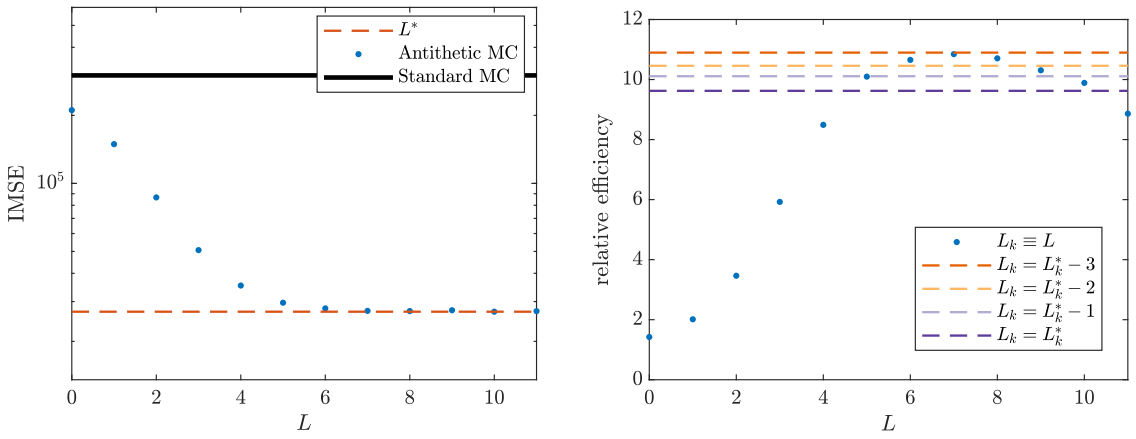
Schlögl system

Finally we consider the bistable Schlögl system, which we already encountered in Section 4.2.3

$$2S_1 + S_2 \xrightarrow{c_1} 3S_1, \qquad 3S_1 \xrightarrow{c_2} 2S_1 + S_2, \qquad S_3 \xrightarrow{c_3} S_1, \qquad S_1 \xrightarrow{c_4} S_3. \tag{5.13}$$

We assume that S_2 and S_3 are present in abundance, and therefore are modelled to have constant population over time, and take $\mathbf{X}_0 = (250, 10^5, 2 \cdot 10^5)^\intercal$ as the initial condition. With parameters $c_1 = 3 \cdot 10^{-7}$, $c_2 = 10^{-4}$, $c_3 = 10^{-3}$ and $c_4 = 3.5$ this system then gives rise to bistable behaviour. We see in Figure 5.13 that if we consider the system over $t \in [0, 8]$ we find that under optimal conditions the antithetic MNRM is more than 10 times as efficient relative to the standard MNRM. Note that this is surprising based on the ANOVA-decomposition of this problem. The decomposition,

depicted in [124, Figure 7], attributes only roughly half of the variance to first-order effects and would thus predict merely a two-fold efficiency improvement from using the antithetic MNRM.



(a) IMSE at time T.

(b) Efficiency (based on IMSE at time T) relative to the standard MC method.

Figure 5.13: Study of the antithetic midpoint-based Poisson bridge method using Strategy 2 in Section 5.4.1 to select the method parameters. Theoretical optimal L^* is given by equation (5.6). Model studied is the Schlögl system, (5.13), with parameters and initial conditions described in the text and final time $T = 8$. Each data point corresponds to 10^5 simulations and we use a pilot run with 10^2 simulations to determine the final internal time $\tau_{k,f}$ for each of the unit-rate Poisson process via the medians of the pilot run distribution. Note that for this system we found using equation (5.6) that $(L_1^*, L_2^*, L_3^*, L_4^*) = (11, 9, 9, 11)$.

As a possible explanation for this larger-than-expected performance benefit we consider what the sample paths of S_1 look like when using the antithetic MNRM in Figure 5.14. Interestingly we see that the antithetic MNRM via the negatively correlated sample paths is very effective in sampling from both modes of the bistable distribution. This means that, on average, the mean of two antithetic sample paths will be much closer to the true mean than when using two independent sample paths.

This final example therefore again shows that the antithetic MNRM can provide large efficiency improvements and, in particular, that under certain conditions such as bistability the improvement over the standard MNRM can be larger than indicated by solely looking at the first order variance components of the ANOVA-decomposition.

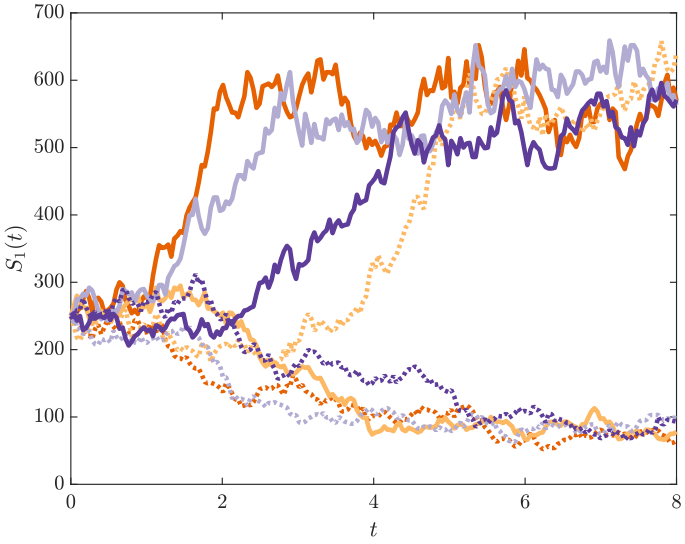


Figure 5.14: Sample paths from the antithetic MNRM for the Schlögl system, (5.13), with parameters and initial conditions as described in the text. Antithetic sample path pairs share the same color, but differ in line style (solid and dotted).

5.5 Discussion

In this chapter we investigated the generalisation of the Brownian bridge construction to the Poisson bridge, its natural counterpart for Poisson processes. As an important contribution we showed in Section 5.4.1 and Appendix 5.A that it is possible to *a priori* derive the optimal parameters for the midpoint-based variant of the Poisson bridge so as to minimise, over a time interval of interest, the average MSE of the mean of two coupled antithetic Poisson processes constructed using the Poisson bridge method.

Using these new results we found that we can effectively use the Poisson bridge construction in combination with the MNRM. Importantly, we proposed an antithetic MNRM that requires minimal parameter tweaking from the end-user to achieve (near) optimal efficiency improvements, which were shown to be an order of magnitude for a range of examples. In determining the optimal parameter values for the Poisson bridge construction we observed a slight gap between what is the optimal method to construct an isolated unit-rate Poisson process and a unit-rate Poisson process used in a chemical reaction network, which we contribute to the following two factors.

Firstly, by considering the ANOVA-decomposition of chemical reaction problems, we see that in a system with multiple reaction channels the interaction between the reaction channels add to the total variance, but the antithetic sampling method is not tailored to reduce this contribution as it generates the Poisson processes for each reaction channel independently. The reduction of the variance due to higher-order interactions, possibly via QMC sampling of the joint Poisson processes, is therefore an important next step towards further improving the MNRM.

Secondly, we note that when optimising the Poisson-bridge construction we assumed that the length, τ_f, of the Poisson processes that we need to generate is fixed and deterministic. In the context of chemical reaction networks, however, the length of the Poisson processes is often a random variable. In particular, we saw in the QSSA reduction of the Michaelis-Menten system that this was the sole cause for a discrepancy between the predicted and observed optimal parameters. In most other systems, however, we postulate that the effect this random length of the Poisson process has on the limiting variance of the sample paths for the chemical species is negligible compared to the higher order interaction terms between different reaction channels.

Despite the fact that the examples used in this chapter to illustrate the efficacy of the antithetic MNRM are all time-homogeneous reaction networks we note that the proposed method is equally applicable to any system for which the standard MNRM is a suitable SSA. For example, the MNRM is particularly suited for time-inhomogeneous reaction networks or models which incorporate delays and Poisson bridge methods can be used seamlessly to improve the performance of the MNRM in those circumstances as well.

We note that for performance critical systems one might want to profile the cost of the Poisson bridge in more detail and in particular the sampling of binomial, Poisson, beta and gamma random variables via inverse transform sampling. Though using the fast methods discussed in Sections 5.2 and 5.3 means that there is little overhead on

top of the standard uniform random variable sampling, we predict that small run-time improvements could be made when taking this system-dependent information explicitly into account. In such a scenario slightly fewer levels, or larger step sizes, than indicated by the standard theory presented in this chapter might be preferred, though any recommendation will depend on the system hardware and specific software implementation.

As a possible extension based on the results in this chapter we note that the variance reduction methods for Poisson skeletons can also be used to improve the τ-leap method, i.e. we can generate Poisson increments (see step 6 of Algorithm 2.3) conditional on Poisson skeletons sampled using a variance reduction method. However, the optimisation of the Poisson bridge method parameters provided in this chapter does not directly apply due to the discrete time stepping in the τ-leap method.

Additionally, we mainly focussed on the midpoint-based Poisson bridge construction, because it seems the most natural method for problems in the context of chemical reaction networks. We did therefore not focus on the optimisation of the median-based Poisson bridge, but by adapting the results in Appendix 5.A to reflect the variation in the arrival epochs conditional on the Poisson process value we are confident that optimal parameters for the median-based Poisson bridge can be derived in a similar manner. In particular, we note that [63] suggests that, when combined with QMC methods, the median-based construction should be preferred due to the fact that beta and gamma random variables are continuous and therefore more amenable to the use of low-discrepancy points than the discrete binomial and Poisson random variables in the midpoint-based construction. On a more general note, it would be interesting to see if limitations in the effectiveness of QMC methods in the context of chemical reaction networks that we previously observed in Chapter 4 also hold when using the Poisson bridge construction in combination with the MNRM. Furthermore, the discussion and examples in this chapter focussed on estimation of the mean summary statistic and the extent to which the antithetic MNRM works for different summary

statistics, e.g. distributions, is thus also an interesting question. We leave the exploration of these ideas for future work.

Finally, we reiterate that we have created a (general) framework for developing efficient midpoint-based Poisson bridge constructions for unit-rate Poisson processes. We used this to optimally combine the Poisson bridge construction with antithetic sampling and showed that this combination can also be used to significantly improve the efficiency of the MNRM. However, the use of the antithetic Poisson bridge construction is not just limited to chemical reaction network simulations. We note that, amongst others, jump-diffusion models and continuous-time branching process models often model the jump and branching times to be given by a Poisson process. If one is to simulate such models, the use of the antithetic Poisson bridge construction will help to reduce the overall estimator variance.

Chapter appendix

5.A Midpoint-based Poisson bridge parameter optimisation details

In [141] the authors derive explicit expressions for the statistical error and complexity of the antithetic endpoint method, but these expressions do not lend themselves to tractable analysis. In this appendix we therefore present an approximation scheme that overcomes this hurdle and thus allows us to derive optimal parameters for the midpoint-based Poisson bridge construction. The main focus of this appendix is on the use of antithetic sampling, but the results derived here are general and can also be amended to different sampling strategies, as we show at the end of the appendix.

Endpoint method optimisation

Approximation of the Maginnis-Γ function

We start by considering the Maginnis-Γ function, as defined in [141, Equation (6)], and given by

$$\Gamma(\tau) = \tau^2 - \int_0^1 F_\tau^{-1}(u) F_\tau^{-1}(1-u)\,\mathrm{d}u, \tag{5.14}$$

where F_τ^{-1} is the formal inverse of the CDF of the Poisson distribution with parameter τ. Its main use is in describing the covariance between two antithetically sampled Poisson random variables, X_1 and X_2, with the same parameter τ, namely $\mathrm{Cov}\,[X_1, X_2] = -\Gamma(\tau)$.

It is possible to bound the Maginnis-Γ function by $\min(\tau, \tau^2)$ as shown in [141], but no analytical formula exists, necessitating the numerical computation of $\Gamma(\tau)$ via standard quadrature methods. This procedure, however, is problematic for large τ as the integral argument quickly oscillates, in particular near the boundaries of the interval $[0, 1]$. As noted in [141] for $\tau < \log 2$ we can use the exact result $\Gamma(\tau) = \tau^2$

and for $\tau \gg 1$ it appears that $\Gamma(\tau)$ tends to $\tau - \gamma$ for a constant γ, given by

$$\gamma = \lim_{\tau \to \infty} \tau - \Gamma(\tau) = \frac{7}{36} \approx 0.194. \tag{5.15}$$

To derive this result we use a normal asymptotic approximation to the Poisson distribution, see for example [72, Section 2], but with a continuity correction, i.e.

$$\begin{aligned} F_\tau^{-1}(u) = -\frac{1}{2} + \tau + \sqrt{\tau} + \left(\frac{1}{3} + \frac{w^2}{6}\right) + \tau^{-1/2}\left(-\frac{w}{36} - \frac{w^3}{72}\right) \\ + \tau^{-1}\left(-\frac{8}{405} + \frac{7w^2}{810} + \frac{w^4}{270}\right) + \mathcal{O}(\tau^{-3/2}), \end{aligned} \tag{5.16}$$

where $w = \Phi^{-1}(u)$ and Φ is the standard normal CDF. Note that we only need $F_\tau^{-1}(u)F_\tau^{-1}(1-u)$ to be correct up to order $\mathcal{O}(1)$ to derive γ, so the expansion (5.16) is in fact accurate enough to yield the exact answer. Using equation (5.16) we find that

$$\int_0^1 F_\tau^{-1}(u)F_\tau^{-1}(1-u)\,\mathrm{d}u = \tau^2 - \tau + \frac{7}{36} + \mathcal{O}(\tau^{-1}), \tag{5.17}$$

and thus as $\tau \to \infty$ we have equation (5.15).

This observation in the limit of large τ might come as a surprise if we consider the normal approximation to the Poisson distribution, which is generally considered to improve as the parameter τ increases, and effectiveness of antithetic sampling when used to generate normal random variables. Small deviations away from the normal approximation, even for large Poisson parameter, τ, mean that the variance of an antithetic pair of Poisson random variables does not decay to zero as the parameter τ grows large.

Combining this knowledge of the asymptotic behaviour of the Maginnis-Γ function with its behaviour for small τ we propose a simple approximation to the complicated

and oscillating $\Gamma(\tau)$ function via

$$\tilde{\Gamma}(\tau) = \begin{cases} \tau^2, & \tau \leq \dfrac{3+\sqrt{2}}{6}, \\ \tau - \gamma, & \tau > \dfrac{3+\sqrt{2}}{6}. \end{cases} \tag{5.18}$$

We note that $\tilde{\Gamma}$ is smooth everywhere apart from $\tau = (3+\sqrt{2})/6$, where $\tilde{\Gamma}$ is merely continuous. Though this approximation might seem crude at first sight, its validity can be clearly seen in Figure 5.15. In particular, the larger the parameter τ, the better the approximation and for all practical purposes we will see that using $\tilde{\Gamma}$, even for relatively small τ, gives very satisfying results.

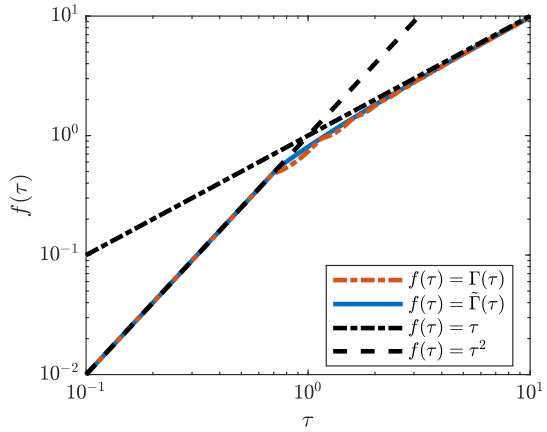
Approximation and minimisation of the IMSE

As proposed in [141] a reasonable statistical error metric to compare different methods that generate unit-rate Poisson processes is the IMSE of a mean estimator. In particular, after we generate $Y^{(1)}(t), \ldots, Y^{(N)}(t)$, possibly correlated, Poisson processes we consider the IMSE normalised to the number of sample paths given by

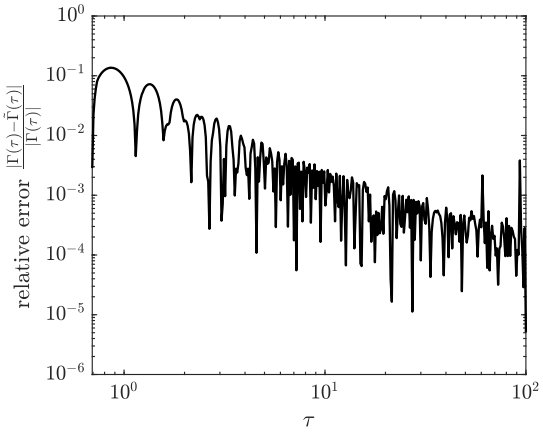
$$N \cdot \mathrm{IMSE}(\tau_f) = N \int_0^{\tau_f} \mathrm{Var}\left[\frac{1}{N}\sum_{n=1}^{N} Y^{(n)}(t)\right] \mathrm{d}t. \tag{5.19}$$

Note that for antithetic sampling it is sufficient to consider $N = 2$, i.e. we introduce a correlation between a pair of samples, and when we use standard MC sampling we can use $N = 1$. Note that in the latter case we simply have $\mathrm{IMSE}(\tau_f) = \tau_f^2/2$, i.e. the accumulated statistical error grows quadratically in the length of the Poisson process when we use the standard MC method to generate unit-rate Poisson processes. From now on we will assume that the normalised version of the statistical error is used whenever we refer to the IMSE unless explicitly stated otherwise.

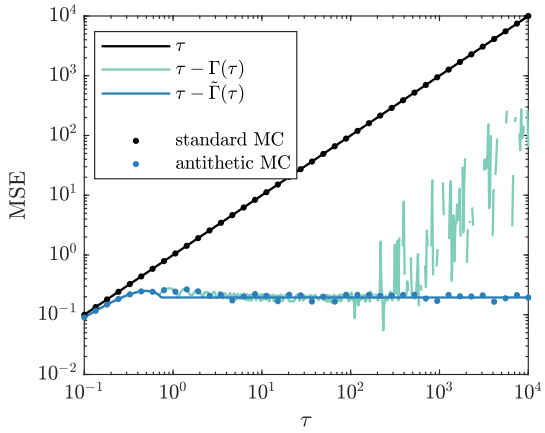
The exact IMSE at a final time τ_f when using the endpoint antithetic method with step size τ_s is given by [141, Lemma 3]. However, whilst that formula yields



(a) Comparison between the Maginnis-Γ function and the approximation $\tilde{\Gamma}$ given by equation (5.18), cf. [141, Figure 4].



(b) Relative error between the Maginnis-Γ function, computed via MATLAB's `integral` method (adapative global quadrature), and the approximation $\tilde{\Gamma}$ given by equation (5.18).



(c) Scaled MSE of the average of two Poisson random variables with parameter τ via either standard MC or antithetic sampling (dots). For comparison we also show the analytic expressions (solid lines). Note the numerical instability in the calculation of the Maginnis-Γ function via MATLAB's `integral` for large τ.

Figure 5.15: Illustration of the approximation of the Maginnis-Γ function by $\tilde{\Gamma}$ in equation (5.18).

an analytic formula, it is non-smooth and it relies on the exact computation of the Maginnis-Γ function, which, as discussed previously, is particularly challenging for large step size, τ_s. In [141] the authors acknowledge that their exact expression is difficult to analyse and therefore provide asymptotic approximations, assuming either

$\tau_s \ll \tau_f$ or $\tau_s > \tau_f$. These approximations, however, are of limited value as the step size that minimises the IMSE clearly does not belong to either asymptotic regime.

To derive a more uniformly valid and tractable approximation we therefore first approximate the IMSE using our $\tilde{\Gamma}$ approximation (5.18) to the Maginnis-Γ function. In addition we note that the IMSE is composed of a contribution of the integral over $t \in [0, \lfloor \tau_f/\tau_s \rfloor \tau_s)$ and a remainder contribution due to the integral over $t \in [\lfloor \tau_f/\tau_s \rfloor \tau_s, \tau_f)$. The latter contribution becomes negligible when τ_f grows large and we therefore ignore it. Combining these two assumptions with the approximation $\lfloor \tau_f/\tau_s \rfloor \approx \tau_f/\tau_s$ we arrive at a (mainly) smooth and simplified approximation to the IMSE

$$\widetilde{\mathrm{IMSE}}(\tau_f; \tau_s) = \begin{cases} \dfrac{\tau_f^2}{2}\left(\dfrac{\tau_s - \tilde{\Gamma}(\tau_s)}{\tau_s}\right) - \dfrac{\tau_f}{6}\left(\left(\tau_s - \tilde{\Gamma}(\tau_s)\right) - \tau_s\right), & \tau_s \leq \tau_f, \\ \dfrac{\tau_f^2}{2} - \dfrac{\tilde{\Gamma}(\tau_s)}{3\tau_s^2}\tau_f^3, & \tau_s > \tau_f. \end{cases} \tag{5.20}$$

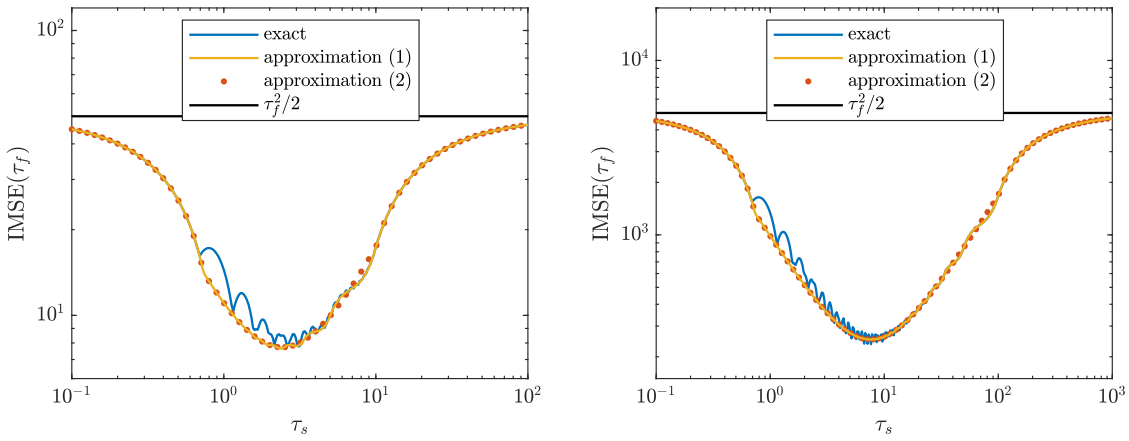
Both approximations to the IMSE, i.e. using equation (5.20) or via [141, Lemma 3] with equation (5.18), are uniformly valid and can be seen to accurately describe the true IMSE in Figure 5.16. In particular, it can be seen that, as expected, the approximation quality improves when τ_f increases and both approximations are capable of accurately capturing the step size that (approximately) minimises the IMSE.

We can now use our approximation to the IMSE to derive the optimal step size, τ_s^*, that minimises the IMSE. We note that in general $\tau_s^* < \tau_f$ which means that we can easily differentiate equation (5.20) with respect to τ_s and set the derivative to zero, which results in

$$\tau_s^* = \sqrt{3\gamma} \cdot \sqrt{\tau_f} \approx 0.76\sqrt{\tau_f}, \tag{5.21a}$$

$$\mathrm{IMSE}(\tau_f; \tau_s^*) = \sqrt{\frac{\gamma}{3}} \cdot \tau_f^{3/2} + \mathcal{O}(\tau_f) \approx 0.25\tau_f^{3/2} + \mathcal{O}(\tau_f). \tag{5.21b}$$

Note that this means that, compared to the standard method of generating unit-rate



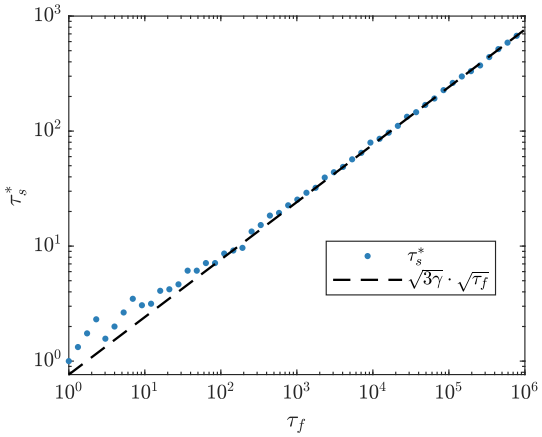
(a) $\tau_f = 10$. (b) $\tau_f = 100$.

Figure 5.16: Approximation of the IMSE at final time τ_f when using the antithetic endpoint method with different step sizes, τ_s. Exact IMSE is calculated via [141, Lemma 3], using a numerical quadrature to compute the Maginnis-Γ function. Approximation (1) uses [141, Lemma 3], but replaces the Maginnis-Γ function by the approximation in equation (5.18). Approximation (2) is computed via equation (5.20). The black reference line shows the IMSE when using standard MC to generate a unit-rate Poisson process.

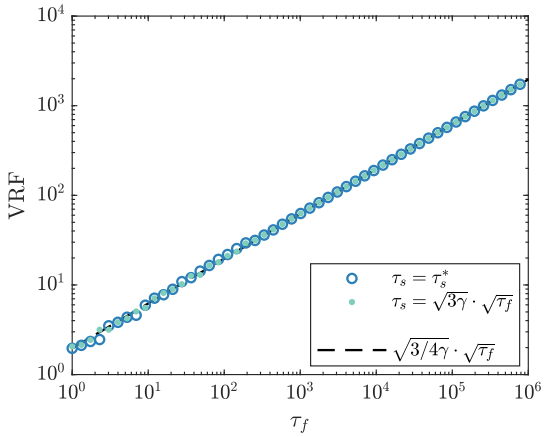
Poisson processes which has IMSE equal to $\tau_f^2/2$, the endpoint antithetic method has a VRF that is asymptotically given by

$$\mathrm{VRF}(\tau_f) = \sqrt{\frac{3}{4\gamma}} \cdot \sqrt{\tau_f} + \mathcal{O}(1) \approx 2.0\sqrt{\tau_f} + \mathcal{O}(1). \tag{5.22}$$

We verify the analysis in this section in Figure 5.17, where we compare the asymptotic optimal step size, τ_s^*, from equation (5.21) and resulting VRF, equation (5.22), with the results using a step size that is found numerically to minimise the true IMSE, explicitly given in [141, Lemma 3]. Though we see in Figure 5.17 that for small final time, τ_f, the step size, τ_s^*, that minimises the true IMSE can be a factor two larger than the approximate optimal step size in equation (5.21), we see that the resulting VRF of both step sizes is almost indistinguishable, and we therefore conclude that equation (5.21) is a very effective, yet practical, choice for the step size.



(a) Optimal step size, τ_s^*, as a function of τ_f for the antithetic endpoint method.



(b) VRF as a function of τ_f for the antithetic endpoint method.

Figure 5.17: Comparison between the step size, τ_s^*, found by numerically optimising the true IMSE for the antithetic endpoint method ([141, Lemma 3]) and its asymptotic approximation in equation (5.21) (a), and the true resulting VRF for both step sizes (b). The VRF is relative to the standard MC Poisson process construction.

Midpoint method optimisation

The results so far give an optimal step size, τ_s^*, if we restrict ourselves to the antithetic endpoint method. However, as we will show now, it is often advantageous to further subdivide our leaps of size τ_s using L midpoint divisions, thus effectively creating a Poisson skeleton with spacing $\tau_s/2^L$. To determine the optimal L^* and τ_s^* we will require some theoretical results, extending the theoretical foundations for the antithetic endpoint method from [141].

Proposition 1. *Suppose we have a collection* $\{Y^{(1)}(t), \ldots, Y^{(N)}(t)\}$ *of* N *unit-rate Poisson processes, sampled independently using standard MC conditional on an unbiased Poisson skeleton (not necessarily generated independently by standard MC) at times* $t_1, \ldots, t_R$ *such that* $0 = t_0 < t_1 < t_2 < \cdots < t_R$. *The scaled MSE of the*

corresponding mean estimator is then piecewise quadratic and given by

$$\begin{aligned}\mathrm{MSE}(t) &= N \cdot \mathrm{Var}\left[\frac{1}{N}\sum_{n=1}^{N} Y^{(n)}(t)\right] \\ &= (t_{r+1}-t_r)\left(\frac{t-t_r}{t_{r+1}-t_r}\right)\left(\frac{t_{r+1}-t}{t_{r+1}-t_r}\right) \\ &+ \mathrm{MSE}(t_r) + \left(\frac{t-t_r}{t_{r+1}-t_r}\right)^2 \Big(\mathrm{MSE}(t_{r+1}) - \mathrm{MSE}(t_r)\Big) \\ &+ 2\left(\frac{t-t_r}{t_{r+1}-t_r}\right)\left(\frac{t_{r+1}-t}{t_{r+1}-t_r}\right)\left(\frac{1}{N}\sum_{n,m}\mathrm{Cov}\left[Y^{(n)}(t_r), Y^{(m)}(t_{r+1}) - Y^{(m)}(t_r)\right]\right),\end{aligned} \tag{5.23}$$

for $t \in [t_r, t_{r+1}]$.

Proof. Without loss of generality we take $t \in [t_r, t_{r+1}]$ for some $1 \leq r < R$. Next we use the fact that if we condition the Poisson processes on the start of the interval, i.e. on $Y^{(1)}(t_r), \ldots, Y^{(N)}(t_r)$, and the end of the interval, i.e. on $Y^{(1)}(t_{r+1}), \ldots, Y^{(N)}(t_{r+1})$, the increments of the Poisson processes are binomially distributed for $t \in [t_r, t_{r+1}]$, i.e. $Y^{(n)}(t) - Y^{(n)}(t_r) \sim \mathcal{B}(N_r^{(n)}, p)$, where $p = (t-t_r)/(t_{r+1}-t_r)$ and $N_r^{(n)} = Y^{(n)}(t_{r+1}) - Y^{(n)}(t_r)$.

Then note that in general we can decompose the MSE for $t \in [t_r, t_{r+1}]$ as

$$\mathrm{MSE}(t) = N \cdot \mathrm{Var}\left[\frac{1}{N}\sum_{n=1}^{N} Y^{(n)}(t)\right] = \frac{1}{N}\sum_{n,m}\mathrm{Cov}\left[Y^{(n)}(t), Y^{(m)}(t)\right],$$

and in particular this holds for $t = t_r$ and $t = t_{r+1}$ as well. Next we use the law of total covariance to express each covariance term in the double sum over all n, m as

$$\begin{aligned}\mathrm{Cov}\left[Y^{(n)}(t), Y^{(m)}(t)\right] &= \mathbb{E}\left[\mathrm{Cov}\left[Y^{(n)}(t), Y^{(m)}(t) \mid \mathcal{F}_r\right]\right] \\ &\quad + \mathrm{Cov}\left[\mathbb{E}\left[Y^{(n)}(t) \mid \mathcal{F}_r\right], \mathbb{E}\left[Y^{(m)}(t) \mid \mathcal{F}_r\right]\right],\end{aligned}$$

where $\mathcal{F}_r = \sigma\{Y^{(1)}(t_r), \ldots, Y^{(N)}(t_r), Y^{(1)}(t_{r+1}), \ldots, Y^{(N)}(t_{r+1})\}$ is the sigma algebra generated by the random variables comprising the Poisson skeleton at times t_r and

t_{r+1}. Using the independence of the Poisson processes conditional on the skeleton we see immediately that

$$\mathrm{Cov}\left[Y^{(n)}(t), Y^{(m)}(t) \mid \mathcal{F}_r\right] = \begin{cases} 0, & n \neq m, \\ N_r^{(n)} p(1-p), & n = m. \end{cases}$$

In addition, since the Poisson skeleton is unbiased by assumption, we know that $\mathbb{E}\left[N_r^{(n)}\right] = t_{r+1} - t_r$ and thus

$$\mathbb{E}\left[\mathrm{Cov}\left[Y^{(n)}(t), Y^{(m)}(t) \mid \mathcal{F}_r\right]\right] = \begin{cases} 0, & n \neq m, \\ (t_{r+1} - t_r)\left(\frac{t-t_r}{t_{r+1}-t_r}\right)\left(\frac{t_{r+1}-t}{t_{r+1}-t_r}\right), & n = m. \end{cases}$$

Using the fact that the Poisson process conditional on its endpoints is given by the value of the Poisson skeleton at t_r plus a binomial contribution we find

$$\mathbb{E}\left[Y^{(n)}(t) \mid \mathcal{F}_r\right] = pY^{(n)}(t_{r+1}) + (1-p)Y^{(n)}(t_r).$$

This, in turn, implies that the covariance of the conditional expectation can be factored as

$$\begin{aligned} \mathrm{Cov}\left[\mathbb{E}\left[Y^{(n)}(t) \mid \mathcal{F}_r\right], \mathbb{E}\left[Y^{(m)}(t) \mid \mathcal{F}_r\right]\right] = {} & (1-p)^2 \mathrm{Cov}\left[Y^{(n)}(t_r), Y^{(m)}(t_r)\right] \\ & + p^2 \mathrm{Cov}\left[Y^{(n)}(t_{r+1}), Y^{(m)}(t_{r+1})\right] \\ & + p\ \mathrm{Cov}\left[Y^{(n)}(t_r), Y^{(m)}(t_{r+1})\right] \\ & + p\ \mathrm{Cov}\left[Y^{(n)}(t_{r+1}), Y^{(m)}(t_r)\right]. \end{aligned}$$

Finally, summing over all n, m of both contributions to the total covariance then yields equation (5.23). □

Interestingly the proof above also highlights the sources of the different terms in

the MSE of the Poisson process, equation (5.23). The first contribution is intrinsic noise in the Poisson process when conditioned on a Poisson skeleton. Importantly, this part is independent of the specific construction used to generate the Poisson skeleton. The use of, for example, an (antithetic) midpoint-based method to construct the Poisson skeleton therefore does not influence this part of the MSE. The second contribution to the MSE is due to the variation in the Poisson skeleton, and this is where the specific construction method of the Poisson skeleton becomes important. By using a variance reduction technique in the midpoint-based method to construct the Poisson skeleton we thus reduce this second contribution to the MSE.

We also note that, in the particular case where the increments of the Poisson skeleton are generated independent of previous increments, the term in the last line of equation (5.23) vanishes. In the case of using standard, independent, MC sampling for the increments of the Poisson skeleton, we have $\mathrm{MSE}(t_r) = t_r$ and thus $\mathrm{MSE}(t) = t$ for all t, as we would expect. If, however, we use the antithetic endpoint method with step size τ_s to generate the Poisson skeleton at times $t_r = r\tau_s$ we find that $\mathrm{MSE}(r\tau_s) = r(\tau_s - \Gamma(\tau_s)) \leq r\tau_s$ and we therefore recover [141, Theorem 2]. Note that this also proves that the antithetic endpoint method indeed does yield a variance reduction for all t, regardless of the step size.

Approximation and minimisation of the IMSE

Using Proposition 1 it immediately follows that the IMSE can be found by simply integrating the piecewise quadratic defined by equation (5.23), which yields Corollary 1.

Corollary 1. *Suppose we have a collection* $\{Y^{(1)}(t), \ldots, Y^{(N)}(t)\}$ *of* N *unit-rate Poisson processes, sampled independently using standard MC conditional on an unbiased Poisson skeleton (not necessarily generated independently by standard MC) at times* $t_1, \ldots, t_R$ *such that* $0 = t_0 < t_1 < t_2 < \cdots < t_R$. *The scaled IMSE over* $[0, t_R]$ *of the*

corresponding mean estimator is given by

$$\begin{aligned}\mathrm{IMSE}(t_R) &= \int_0^{t_R} \mathrm{MSE}(t)\,\mathrm{d}t \\ &= \frac{1}{6}\sum_{r=0}^{R-1}(t_{r+1}-t_r)^2 \\ &\quad + \frac{1}{3}\sum_{r=0}^{R-1}(t_{r+1}-t_r)\Big(\mathrm{MSE}(t_{r+1}) + 2\mathrm{MSE}(t_r)\Big) \\ &\quad + \frac{1}{3}\sum_{r=0}^{R-1}(t_{r+1}-t_r)\left(\frac{1}{N}\sum_{n,m}\mathrm{Cov}\left[Y^{(n)}(t_r), Y^{(m)}(t_{r+1}) - Y^{(m)}(t_r)\right]\right).\end{aligned} \tag{5.24}$$

Note that, in particular, we recover [141, Lemma 3] when we assume that the Poisson skeleton has been generated by the antithetic endpoint method and $\tau_f = R\tau_s$ so that $t_r = r\tau_s$. The IMSE for the standard MC construction of a Poisson process, $t_R^2/2$, is recovered using, as before, $\mathrm{MSE}(t_r) = t_r$.

We will analyse the midpoint-based Poisson bridge now in two steps. First we consider the IMSE of the midpoint-based Poisson bridge over a single step τ_s in Corollary 2 below.

Corollary 2. *Suppose we have a collection $\{Y^{(1)}(t), \ldots, Y^{(N)}(t)\}$ of N unit-rate Poisson processes, sampled independently using standard MC conditional on an unbiased Poisson skeleton generated by a single step of the midpoint-based method, with full step size τ_s and L levels. The scaled IMSE over $[0, \tau_s]$ of the corresponding mean estimator is given by*

$$\begin{aligned}\mathrm{IMSE}(\tau_s) &= \int_0^{\tau_s} \mathrm{MSE}(t)\,\mathrm{d}t \\ &= \underbrace{\frac{\tau_s^2}{6\cdot 2^L}}_{\substack{\text{contribution from intrinsic Poisson noise conditional on an unbiased Poisson skeleton}}} \\ &\quad + \underbrace{\frac{\tau_s}{3}\cdot\bar{\psi}_L(\tau_s)}_{\substack{\text{contribution from Poisson skeleton noise in a single midpoint step of size } \tau_s}},\end{aligned} \tag{5.25}$$

where we have defined the auxiliary functions

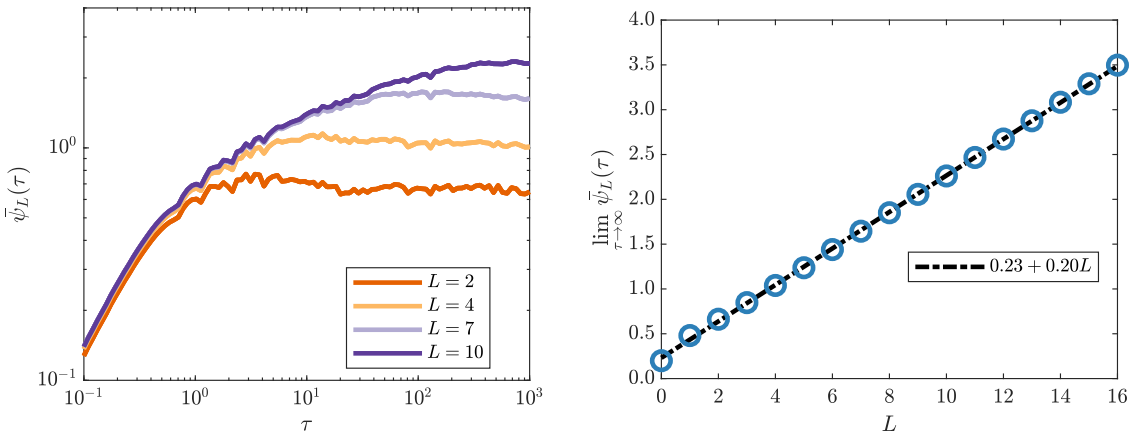
$$\begin{aligned}\bar{\psi}_L(\tau) &= \left(\frac{1}{2^L}\sum_{l=0}^{2^L-1}\psi_l\left(\frac{\tau}{2^L}\right)\right),\\ \psi_l(\tau) &= \mathrm{MSE}((l+1)\tau) + 2\mathrm{MSE}(l\tau)\\ &\quad + \left(\frac{1}{N}\sum_{n,m}\mathrm{Cov}\left[Y^{(n)}(l\tau), Y^{(m)}((l+1)\tau) - Y^{(m)}(l\tau)\right]\right).\end{aligned} \tag{5.26}$$

The proof of Corollary 2 simply uses Corollary 1 with the observation that for a Poisson skeleton generated by the midpoint-based method with step size τ_s and L levels we have $t_r = r\tau_s/2^L$.

Note that the use of multiple midpoint levels, L, introduces an extra function $\bar{\psi}_L(\tau_s)$, which captures the noise induced by the Poisson skeleton averaged over the time points in it. In particular we note that $\bar{\psi}_0(\tau_s) = \mathrm{MSE}(\tau_s) = \tau_s - \Gamma(\tau_s)$ when using antithetic sampling and thus $\bar{\psi}_0(\tau_s)$ tends to the constant γ when $\tau_s \to \infty$. Although the explicit expression for $\bar{\psi}_L(\tau_s)$ is in general intractable, Figure 5.18 strongly suggests that when using the antithetic midpoint-based method it tends to a constant, which linearly depends on the number of levels L, as $\tau_s \to \infty$.

Furthermore we note that Figure 5.18(a) suggests that if we keep τ fixed and increase L it appears that $\bar{\psi}_L$ is bounded, i.e. there is a limit for the Poisson skeleton noise contribution when τ is fixed. This is confirmed in Figure 5.19(a), where we see that the $\bar{\psi}_L$ function does not increase further if we increase L beyond roughly $\log 4\tau / \log 2$. In other words, if we subdivide the midpoint step of size τ_s into sub steps of size $\tau_s/2^L < 1/4$ then increasing the number of levels L even further has no effect on the noise in the Poisson skeleton.

Finally we consider the small step size limit for the $\bar{\psi}_L$ function. We ignore the covariance terms in equation (5.26) and approximate the MSE by the antithetic endpoint result for small step sizes, i.e. $\mathrm{MSE}(\tau) = \tau - \tau^2$. This allows us to explicitly find $\bar{\psi}_L$ using equation (5.26) in terms of τ and L. We then find (empirically) that by com-



(a) $\bar{\psi}_L(\tau)$ as a function of τ.

(b) Limiting behaviour of $\bar{\psi}_L(\tau)$ for large τ.

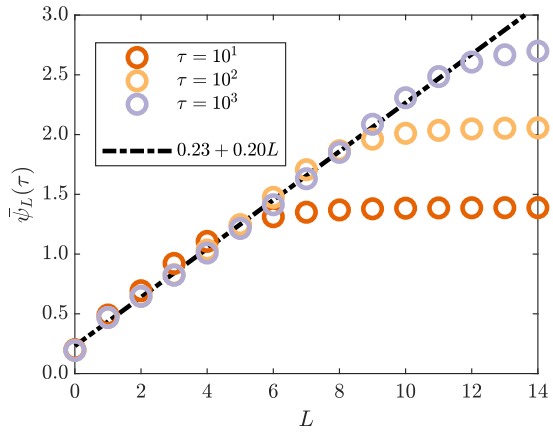
Figure 5.18: Behaviour of the $\bar{\psi}_L$ function for finite step size τ (a) and in the limit of $\tau \to \infty$ (b). To estimate the value of $\bar{\psi}_L(\tau)$ in (a) we used $128 \cdot 10^4$ samples per time point. To estimate the limiting value of $\bar{\psi}_L$ (blue circles in (b)) we average $\bar{\psi}_L(\tau)$ using $32 \cdot 10^3$ samples for each of 20 logarithmically spaced time points in $\tau \in [10^6, 10^8]$.

bining the behaviour in the two asymptotic regimes of step size we can approximate $\bar{\psi}_L(\tau)$ using

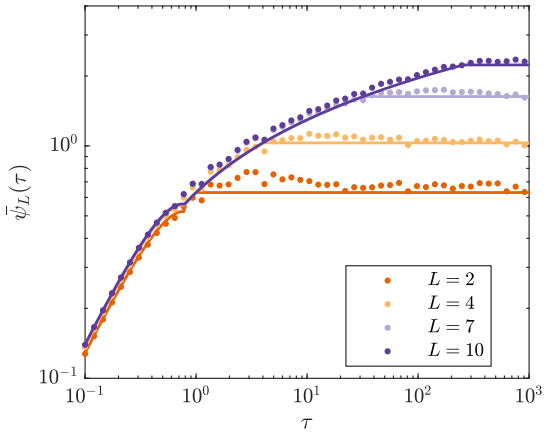
$$\bar{\psi}_L(\tau) \approx \begin{cases} \tau\left(\frac{3}{2} - \frac{1}{2^{L+1}}\right) - \tau^2\left(1 - \frac{1}{2^{1+L}} + \frac{1}{2^{1+2L}}\right), & \tau < 0.78, \\ 0.23 + 0.20 \cdot \min\left(\frac{\log 4\tau}{\log 2}, L\right), & 0.78 < \tau, \end{cases} \tag{5.27}$$

where we chose the switching point between the two regimes, $\tau \approx 0.78$, such that the approximation is continuous in the limit of $L \to \infty$. Note, however, that for any finite L the approximation in equation (5.27) is discontinuous at the switching point.

The performance of the approximation can be judged in Figure 5.19, and, as we saw before in the case of the Maginnis-Γ function, the approximation is of good quality for all τ. However, unlike equation (5.18), this approximation is phenomenological in nature and we see no direct way to prove its validity analytically in the way we did for the approximation to the Maginnis-Γ function earlier. It is purely intended as a tool to practically optimise and predict the performance of the midpoint-based method.



(a) $\bar{\psi}_L(\tau)$ as a function of L for fixed τ.



(b) Comparison between $\bar{\psi}_L(\tau)$ and the approximation given by equation (5.27).

Figure 5.19: Behaviour of the $\bar{\psi}_L$ function for fixed step size τ when varying L (a) and a comparison of $\bar{\psi}_L$ (dots) with the approximation (solid lines) in equation (5.27) (b). To estimate the value of $\bar{\psi}_L(\tau)$ in both plots we used $128 \cdot 10^4$ samples per time point.

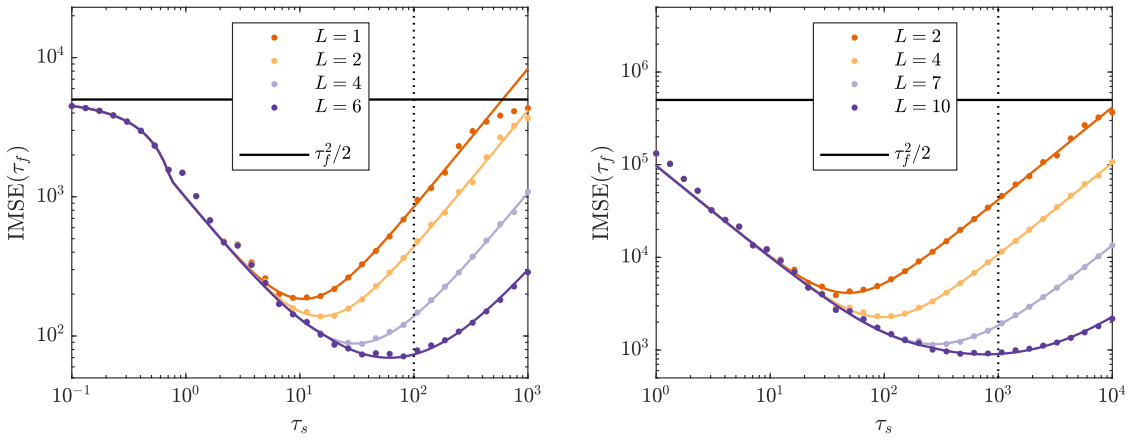
Now that we have an understanding of the different components of the IMSE from a single step of the midpoint-based Poisson bridge method we generalise this approach in Corollary 3 to include multiple steps of the same method.

Corollary 3. *Suppose we have a collection* $\{Y^{(1)}(t), \ldots, Y^{(N)}(t)\}$ *of* N *unit-rate Poisson processes, sampled independently using standard MC conditional on an unbiased Poisson skeleton generated by the midpoint-based method, with full step size* τ_s *and* L *levels. The scaled IMSE over* $[0, \tau_f]$*, where* $\tau_f = R\tau_s$ *with* $R \in \mathbb{N}$*, i.e. a multiple of* τ_s*, of the corresponding mean estimator is given by*

$$\begin{aligned}
\mathrm{IMSE}(\tau_f) &= \int_0^{\tau_f} \mathrm{MSE}(t)\,\mathrm{d}t \\
&= \underbrace{\frac{\tau_f \tau_s}{6 \cdot 2^L}}_{\text{contribution from intrinsic Poisson noise conditional on an unbiased Poisson skeleton}} \\
&\quad + \underbrace{\frac{\tau_f}{3} \cdot \bar{\psi}_L(\tau_s)}_{\text{contribution from Poisson skeleton noise in a single midpoint step of size } \tau_s} \\
&\quad + \underbrace{\frac{\tau_f^2}{2} \cdot \frac{\mathrm{MSE}(\tau_s)}{\tau_s} - \frac{\tau_f}{2}\mathrm{MSE}(\tau_s)}_{\text{contribution from accumulation of noise in the Poisson skeleton after each full step of size } \tau_s},
\end{aligned} \tag{5.28}$$

where the auxiliary function $\bar{\psi}_L(\tau)$ is defined in equation (5.26).

To prove Corollary 3 we simply use Corollary 2 and the fact that the full midpoint steps are independent of each other. Note that the general case where $R\tau_s < \tau_f < (R+1)\tau_s$ yields an IMSE given by equation (5.28) plus a small correction which is $\mathcal{O}(\tau_s)$. The combination of equations (5.27) and (5.28) turns out to be a very good approximation to the true IMSE for a wide range of step sizes, τ_s, as can be seen in Figure 5.20. Note that the only point where the approximation breaks down is in the regime for which $\tau_s \gg \tau_f$, which is a practically irrelevant regime in terms of performance benefit over the standard MC construction method.



(a) $\tau_f = 100$. (b) $\tau_f = 1000$.

Figure 5.20: Approximation of the IMSE at final time τ_f when using the antithetic midpoint-based method for a varying number of midpoint levels, L, as a function of different step sizes, τ_s. Exact IMSE is estimated using $128 \cdot 10^4$ samples per data point. Approximation (solid coloured lines) uses equations (5.27) and (5.28). The solid black reference line shows the IMSE when using standard MC to generate a unit-rate Poisson process. The dotted black reference line indicates τ_f.

Equations (5.27) and (5.28) allow us to determine the optimal step size τ_s^* which minimises the IMSE in the case of a general antithetic midpoint-based Poisson bridge where we fix the number of levels to L. As noted before for the endpoint (antithetic) method, we see from equation (5.28) that a good choice for the step size, τ_s, in the midpoint-based Poisson bridge construction will have to be a balanced choice; a small step size, τ_s, decreases the first term in equation (5.28), but increases the last term

in equation (5.28). Using the observation that for large enough τ_s both $\mathrm{MSE}(\tau_s)$ and $\bar{\psi}_L(\tau_s)$ tend to constants we find, under the assumption that $\tau_s < \tau_f$ and for large enough τ_f, that the optimal step size and corresponding IMSE are given by

$$\tau_s^* = \sqrt{2^L} \cdot \sqrt{3\gamma} \cdot \sqrt{\tau_f}, \tag{5.29a}$$

$$\mathrm{IMSE}(\tau_f; \tau_s^*) = \frac{1}{\sqrt{2^L}} \cdot \sqrt{\frac{\gamma}{3}} \cdot \tau_f^{3/2} + \mathcal{O}(\tau_f). \tag{5.29b}$$

Equation (5.29) seems to suggest that increasing L indefinitely will keep decreasing the IMSE. However, because the above conditions were derived under the assumption that $\tau_s < \tau_f$, we see that $L \leq \lfloor \log \tau_f / \log 2 \rfloor$ must hold for equation (5.29) to be valid. The VRF from using the antithetic midpoint-based method relative to the standard MC construction is finally given by

$$\mathrm{VRF}(\tau_f) = \sqrt{2^L} \cdot \sqrt{\frac{3}{4\gamma}} \cdot \sqrt{\tau_f} + \mathcal{O}(1). \tag{5.30}$$

Complexity consideration

Thus far we have ignored the computational complexity of the midpoint-based method which from an efficiency point of view is technically incorrect as smaller steps and more levels L will incur a larger overall complexity overhead. The computational complexity of the midpoint-method using L levels is given in [141, Section 2.4] and can be written as

$$\mathcal{C}_{\mathrm{midpoint}}(\tau_f; \tau_s) = 2^L \cdot \max\left(\left\lceil \frac{\tau_f}{\tau_s} \right\rceil, 1\right) + \max\left(\left\lceil \frac{\tau_f}{\tau_s} \right\rceil \tau_s, \tau_f\right). \tag{5.31}$$

Again this expression is slightly complicated by the fact that it is non-smooth and we therefore note that a reasonable smooth approximation is given by

$$\tilde{\mathcal{C}}_{\mathrm{midpoint}}(\tau_f; \tau_s) = 2^L \cdot \max\left(\frac{\tau_f}{\tau_s}, 1\right) + \tau_f. \tag{5.32}$$

Note that when we fix L and increase τ_f the complexity of the midpoint-based method is $\mathcal{C}_{\text{midpoint}}(\tau_f; \tau_s^*) = \tau_f + \mathcal{O}(\tau_f^{1/2})$, where τ_s^* is given in equation (5.29). Thus, to first order, the complexity of the midpoint-based construction of a Poisson process is equal to the complexity of the standard MC construction in the limit of large τ_f if we fix the number of midpoint levels, L.

We can now instead also choose to directly minimise the IMSE normalised to the complexity, as is customary when considering the efficiency of a method (see Section 2.3.2). If we consider $\mathcal{C}_{\text{midpoint}}(\tau_f; \tau_s) \cdot \text{IMSE}(\tau_f; \tau_s)$, or an approximation of this quantity, for example via equations (5.20) and (5.32), we find that the optimal step size, τ_s^*, is marginally larger than $\sqrt{2^L 3\gamma\tau_f}$. By taking slightly larger step sizes we make a trade-off between an increase in IMSE and a reduction in complexity. In practice, however, the analytical formula for τ_s^* in the complexity-normalised case is much more complicated and the difference with equation (5.29) is very small. For ease of implementation we therefore simply recommend to use equation (5.29) to determine the appropriate step size τ_s when one wishes to use the midpoint-based method with a fixed number of levels, L.

The discussion up to now has focussed on the scenario in which we fix the number of midpoint levels, L, and ask what the optimal step size τ_s should be. The reverse question, however, is equally interesting, i.e. given that we fix the step size, τ_s, what should be number of midpoint levels, L, so that we get the most efficient method? Though we will not consider this angle in as much detail, we note that for general L the complexity of the midpoint-based method is minimised when we take $\tau_s = \tau_f$, i.e. in the case where we consider a single full step. In fact, though we will not prove this, the only alternative, from a complexity point of view, to the strategy which picks the optimal step size for a fixed τ_s, is a strategy which picks the optimal number of levels, L, when we fix $\tau_s = \tau_f$. Intuitively this strategy first minimises the complexity by selecting $\tau_s = \tau_f$ and then subsequently the IMSE by choosing the correct number of levels, L. Note that taking $\tau_s = \tau_f$ simplifies the IMSE significantly and we get the

exact result

$$\mathrm{IMSE}(\tau_f;\tau_f) = \frac{\tau_f^2}{6\cdot 2^L} + \frac{\tau_f}{3}\cdot\bar{\psi}_L(\tau_f), \tag{5.33}$$

where we recall that we can approximate $\bar{\psi}_L(\tau_f)$ accurately using equation (5.27). If we now solely consider the IMSE we would select a number of levels that is excessively large. Therefore we combine the IMSE in equation (5.33) with the exact complexity, $\mathcal{C} = \tau_f + 2^L$, and directly optimise their product. The intuitive solution is to take $\tau_f \approx 2^L$ so that all the terms in both the IMSE and complexity are $\mathcal{O}(\tau_f)$. It turns out that this strategy, i.e.

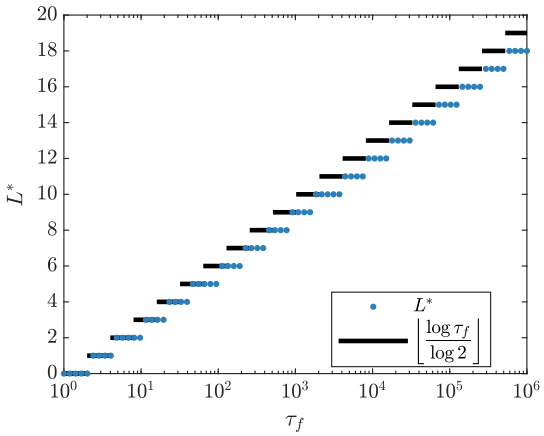
$$L^* = \left\lfloor \frac{\log \tau_f}{\log 2} \right\rfloor, \tag{5.34}$$

is remarkably close to the true optimal solution that is found by numerically minimising the complexity-normalised IMSE, which can be verified in Figure 5.21(a). Note that the complexity of this approach is $\mathcal{C} \approx 2\tau_f$, i.e. twice that of the standard MC Poisson process construction. The IMSE, however, is significantly reduced and using, our approximation (5.27), can be estimated to be

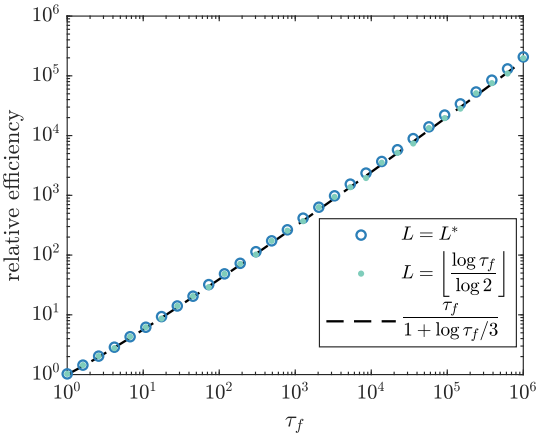
$$\tilde{\mathrm{IMSE}}(\tau_f;\tau_f) \approx \frac{\tau_f}{4}\left(1 + \frac{\log \tau_f}{3}\right). \tag{5.35}$$

This means that if we compute the efficiency of this strategy relative to the standard MC method we find it is $\mathcal{O}(\tau_f \log^{-1} \tau_f)$, thus achieving increasingly large improvements when the length of the Poisson process, τ_f, increases. We verify in Figure 5.21(b) that in practice this is very close to the actual limit when finding the optimal L^* by numerically optimising the complexity normalised IMSE. Effectively, for a given computational budget, the resulting antithetic midpoint-based method has an IMSE and MSE of $\mathcal{O}(\tau_f)$ and $\mathcal{O}(1)$, respectively, whereas the standard MC method achieves, for the same statistical error measures, only $\mathcal{O}(\tau_f^2)$ and $\mathcal{O}(\tau_f)$, respectively.

For completeness we note that the optimal solution L^* found by numerically optimising the complexity-normalised IMSE grows slower than predicted by equa-



(a) Optimal number of midpoint levels, L^*, for the antithetic midpoint-based Poisson bridge method when step size $\tau_s = \tau_f$.



(b) Efficiency of the antithetic midpoint-based Poisson bridge method with $\tau_s = \tau_f$ and optimal number of midpoint levels, L, relative to the standard MC Poisson process construction.

Figure 5.21: Comparison between choosing the number of midpoint levels, L^*, either by numerically optimising the complexity normalised IMSE for the antithetic midpoint-based Poisson bridge method with step size $\tau_s = \tau_f$ or by using the approximation given in equation (5.34).

tion (5.34). By empirically fitting a larger class of functions than equation (5.34), of the form $\lfloor a + b \log \tau_f \rfloor$ for free parameters a and b, we find that L^* is more accurately described, at least on the interval $\tau_f \in [10^0, 10^6]$, by

$$L^* = \left\lfloor 0.94 \cdot \frac{\log 0.9\tau_f}{\log 2} \right\rfloor, \tag{5.36}$$

i.e. the growth of L^* with τ_f is only slightly slower than predicted by equation (5.34). Note that this approximation to the optimal L^* differs by not more than unity for $\tau_f \in [10^0, 10^{10}]$, which should be sufficient for all practical purposes, and for large $\tau_f > 10^4$ forms a lower bound for L^*. If we use this method to select the (approximate) optimal number of levels the previous conclusions about the efficiency of the resulting antithetic midpoint method still hold.

Going beyond antithetic sampling

Though the previous sections mainly focussed on the antithetic midpoint-based Poisson bridge construction it is possible to use these results to study more general versions of the midpoint-based Poisson bridge method. In particular, we note that the key result to derive optimal parameters for the Poisson bridge construction is Corollary 3, which describes the IMSE of the full Poisson process purely in terms of the statistical error in the Poisson skeleton constructed via the midpoint-based method. Importantly, we reiterate that this result is independent of the specific sampling technique used in the construction of the Poisson bridge. We can therefore study the performance of many other variance reduction techniques, such as Latin hypercube sampling, stratified sampling or QMC sampling in the same framework. The only quantity needed for this procedure is $\bar{\psi}_L(\tau)$, given in equation (5.26), and in an ideal scenario we can (approximately) calculate it analytically, as was done when we considered the antithetic endpoint method. The more likely scenario, however, is that $\bar{\psi}_L(\tau)$ is intractable for many sampling techniques, but as shown in the previous section, a reasonable phenomenological approximation is all that is needed in practice to get excellent results.

5.B Algorithm details

In this section we provide explicit descriptions of the algorithms used in the midpoint-based and median-based Poisson bridge constructions. This is largely based on [63, Chapter 15] with minor corrections and additions to allow for more memory efficient implementations.

Efficient sampling of uniform order statistics

We first describe in Algorithm 5.1 an efficient and stable implementation of the exponential spacings method to sequentially generate the order statistics of uniform

random variables on a finite interval based on [63, Section 15.6]. This subroutine is used in both the median-based and midpoint-based Poisson bridge constructions.

Algorithm 5.1 UOS(a, b, N).
This generates N ordered uniform random variates on (a, b).

Input: Interval (a, b)
Input: Number, N, of uniform random variates on (a, b)
1: $S \leftarrow 0$
2: **for** $n = 1, \ldots, N$ **do**
3: Generate $v_n \sim \mathcal{U}(0, 1)$
4: $S \leftarrow S + \log v_n/(N - n + 1)$
5: $P \leftarrow \exp(S)$
6: $P \leftarrow 1 - P$ ▷ n-th smallest uniform random variable, out of N, on $(0, 1)$.
7: $u_n \leftarrow a + (b - a)P$ ▷ Scale to interval (a, b).
8: **end for**
9: **return u**

Midpoint-based Poisson bridge

The full midpoint-based Poisson bridge construction is described in Algorithm 5.2, which uses two subroutines; Algorithm 5.4 to construct a Poisson skeleton and Algorithm 5.5 to fill in such a Poisson skeleton. Note that to use variance reduction techniques in combination with the Poisson bridge method we simply make sure that the uniform random variables, $\mathbf{u}$, that are used to construct the Poisson skeleton are sampled via a variance reduction technique, such as antithetic sampling.

Median-based Poisson bridge

The full median-based Poisson bridge construction is described in Algorithm 5.3, which uses of two subroutines; Algorithm 5.6 to construct a Poisson skeleton and Algorithm 5.7 to fill in such a Poisson skeleton. Note that to use variance reduction techniques in combination with the Poisson bridge method we simply make sure that the uniform random variables, $\mathbf{u}$, that are used to construct the Poisson skeleton are sampled via a variance reduction technique, such as antithetic sampling.

Algorithm 5.2 $\mathrm{PB}_{\mathrm{midpoint}}(\tau_f, \tau_s, L, \mathbf{u})$.
This generates a single Poisson process $Y(\tau)$ when $\tau \in [0, \lceil \tau_f/\tau_s \rceil \tau_s)$ via the midpoint-based Poisson bridge construction with L levels and step size τ_s.

Input: Final time τ_f
Input: Step size τ_s
Input: Number of levels L
Input: $\mathbf{u} \in [0,1)^d$ uniform random variable, where $d = \lceil \tau_f/\tau_s \rceil 2^L$

$\tau \leftarrow 0$
$\mathbf{T} \leftarrow \emptyset$ ▷ Initialise the arrival epochs of $Y(\tau)$.
$k \leftarrow 0$
while $\tau < \tau_f$ **do**
  $\mathbf{t}, \mathbf{M} \leftarrow \mathrm{stepPB}_{\mathrm{midpoint}}(\tau_s, L, \mathbf{u}_{1+k2^L:(k+1)2^L})$ ▷ Sample a Poisson skeleton on $[0, \tau_s)$.
  $\boldsymbol{\mathcal{T}} \leftarrow \tau + \mathrm{fillPB}_{\mathrm{midpoint}}(\mathbf{M}, \mathbf{t})$ ▷ Fill in Poisson skeleton and scale to $[\tau, \tau + \tau_s)$.
  $\mathbf{T} \leftarrow \mathbf{T} \cup \boldsymbol{\mathcal{T}}$ ▷ Append to previously sampled arrival epochs.
  $\tau \leftarrow \tau + \tau_s$
  $k \leftarrow k + 1$
end while
return T ▷ Arrival epochs for Poisson process $Y(\tau)$ when $\tau \in [0, \lceil \tau_f/\tau_s \rceil \tau_s)$.

Algorithm 5.3 $\mathrm{PB}_{\mathrm{median}}(N_f, N_s, L, \mathbf{u})$.
This generates a single Poisson process $Y(\tau)$ when $0 \leq Y(\tau) \leq \lceil N_f/N_s \rceil N_s$ via the median-based Poisson bridge construction with L levels and step size N_s.

Input: Final time N_f
Input: Step size N_s
Input: Number of levels L
Input: $\mathbf{u} \in [0,1)^d$ uniform random variable, where $d = \lceil N_f/N_s \rceil 2^L$

$N \leftarrow 0$
$\tau \leftarrow 0$
$\mathbf{T} \leftarrow \emptyset$ ▷ Initialise the arrival epochs of $Y(\tau)$.
$k \leftarrow 0$
while $N < N_f$ **do**
  $\mathbf{t}, \mathbf{j} \leftarrow \mathrm{stepPB}_{\mathrm{median}}(N_s, L, \mathbf{u}_{1+k2^L:(k+1)2^L})$ ▷ Sample a Poisson skeleton.
  $\boldsymbol{\mathcal{T}} \leftarrow \tau + \mathrm{fillPB}_{\mathrm{median}}(\mathbf{j}, \mathbf{t})$ ▷ Fill in Poisson skeleton and scale.
  $\mathbf{T} \leftarrow \mathbf{T} \cup \boldsymbol{\mathcal{T}}$ ▷ Append to previously sampled arrival epochs.
  $N \leftarrow N + N_s$
  $\tau \leftarrow \tau + \mathbf{t}_{\mathrm{end}}$
  $k \leftarrow k + 1$
end while
return T ▷ Arrival epochs for Poisson process $Y(\tau)$ when $0 \leq Y(\tau) \leq \lceil N_f/N_s \rceil N_s$.

Algorithm 5.4 $\text{stepPB}_{\text{midpoint}}(\tau_s, L, \mathbf{u})$.
This generates a single Poisson skeleton for the Poisson process $Y(\tau)$ when $\tau \in [0, \tau_s)$ via the midpoint-based Poisson bridge construction with L levels.

Input: Step size τ_s
Input: Number of levels L
Input: $\mathbf{u} \in [0,1)^{2^L}$ uniform random variable

$\mathbf{M} \leftarrow \mathbf{0}$ ▷ Initialise the increments of the Poisson skeleton with a zero vector of size 2^L.
Generate $N \sim \mathcal{P}(\tau_s)$ using inverse transform sampling on u_1.
$k \leftarrow 2$
$M_1 \leftarrow N$ ▷ Poisson skeleton at level $l = 0$.
for $l = 0, \ldots, L-1$ **do**
    for $n = 2^l, 2^l - 1, \ldots, 1$ **do** ▷ Construct increments of Poisson skeleton at level l.
        Generate $B \sim \mathcal{B}(M_n, 0.5)$ using inverse transform sampling on u_k.
        $k \leftarrow k + 1$
        $h \leftarrow 2n - 1$
        $M_{h+1} \leftarrow M_n - B$
        $M_h \leftarrow B$
    end for
end for
for $i = 1, \ldots, 2^L$ **do**
    $t_i \leftarrow i\tau_s / 2^L$
end for
return $\mathbf{t}, \mathbf{M}$ ▷ Poisson skeleton such that $\sum_{j=1}^{i} M_j = Y(t_i^-)$.

Algorithm 5.5 $\text{fillPB}_{\text{midpoint}}(\mathbf{M}, \mathbf{t})$.
This fills in a single Poisson skeleton generated by the midpoint-based Poisson bridge construction.

Input: Poisson skeleton increments $\mathbf{M}$
Input: Poisson skeleton times $\mathbf{t}$

$N \leftarrow \sum_i M_i$ ▷ Total number of arrival epochs.
$\mathbf{T} \leftarrow \mathbf{0}$ ▷ Initialise arrival epochs with a zero vector of size N.
$q \leftarrow 0$
$a \leftarrow 0$
for $n = 1, \ldots, N$ **do**
    $p \leftarrow q + 1$
    $q \leftarrow q + M_n$
    $b \leftarrow t_n$
    $\mathbf{T}_{p:q} \leftarrow \text{UOS}(a, b, M_n)$ ▷ Sample M_n arrival epochs, indexed p to q, on (a, b).
    $a \leftarrow b$
end for
return $\mathbf{T}$

Algorithm 5.6 $\text{stepPB}_{\text{median}}(N_s, L, \mathbf{u})$.
This generates a single Poisson skeleton for the Poisson process $Y(\tau)$ when $0 \leq Y(\tau) \leq N_s$ via the median-based Poisson bridge construction with L levels.

Input: Step size N_s
Input: Number of levels L
Input: $\mathbf{u} \in [0,1)^{2^L}$ uniform random variable

$\mathbf{t} \leftarrow \mathbf{0}$ ▷ Initialise the Poisson skeleton with a zero vector of size $2^L + 1$.
$\mathbf{j} \leftarrow \mathbf{0}$ ▷ Index vector for arrival epochs in median-based construction.
Generate $\tau \sim \text{Gamma}(N_s, 1)$ using inverse transform sampling on u_1.
$k \leftarrow 2$
$t_{2^L+1} \leftarrow \tau$ ▷ Poisson skeleton at level $l = 0$.
for $l = 0, \ldots, L-1$ **do**
    $j_{2^l+1} \leftarrow N_s$
    for $n = 2^l, 2^l - 1, \ldots, 1$ **do** ▷ Construct Poisson skeleton at level l.
        $c \leftarrow j_n$
        $d \leftarrow j_{n+1}$
        $m \leftarrow \lceil (c+d)/2 \rceil$ ▷ Find median index between c and d.
        if $d \neq m$ **then** ▷ Find median arrival epoch.
            Generate $B \sim \text{Beta}(m-c, d-m)$ using inverse transform sampling on u_k.
            $A \leftarrow t_{d+1} - t_{c+1}$
            $C \leftarrow t_{c+1}$
            $t_{m+1} \leftarrow AB + C$
            $k \leftarrow k + 1$
        end if
        $h \leftarrow 2n$
        $j_h \leftarrow m$
        $j_{h+1} \leftarrow d$
    end for
end for
return $\mathbf{t}, \mathbf{j}$ ▷ Poisson skeleton such that t_i is exactly the j_i-th arrival epoch.

Algorithm 5.7 $\text{fillPB}_{\text{median}}(\mathbf{j}, \mathbf{t})$.
This fills in a single Poisson skeleton generated by the median-based Poisson bridge construction.

Input: Poisson skeleton values $\mathbf{j}$, assumes $j_1 = 0$
Input: Poisson skeleton times $\mathbf{t}$, assumes $t_1 = 0$

$N \leftarrow \mathbf{j}_{\text{end}}$ ▷ Total number of arrival epochs.
$\mathbf{T} \leftarrow \mathbf{0}$ ▷ Initialise arrival epochs with a zero vector of size $N + 1$.
for $m = 1, \ldots, \text{length}(\mathbf{j}) - 1$ **do**
    $p \leftarrow j_m + 1$
    $q \leftarrow j_{m+1}$
    $T_{q+1} \leftarrow t_{m+1}$
    $\mathbf{T}_{(p+1):q} \leftarrow \text{UOS}(t_m, t_{m+1}, q-p)$ ▷ Sample $q-p$ arrival epochs, indexed p to $q-1$, on (t_m, t_{m+1}).
end for
return $\mathbf{T}_{2:\text{end}}$ ▷ Remove the entry for the trivial 0-th arrival epoch.

Chapter 6

Discussion

The common theme throughout this thesis has been the efficiency improvement of simulation approaches for chemical reaction networks via incorporation of variance reduction techniques into SSAs. Our approach has focused on adapting existing variance reduction techniques from the MC sampling literature to the simulation of CTMCs and we have shown that this can lead to sizeable efficiency improvements. Importantly, these new methods are largely complementary and orthogonal to many of the algorithmic improvements that have already been proposed in the literature, thus allowing one to combine improvements to even further increase the efficiency of SSAs. In this final chapter we first provide an overview of the results achieved in this thesis in Section 6.1 before we lay down some open questions and challenges, arising from these results, in Section 6.2.

6.1 Review

Chapter 2

In this thesis we studied the synthesis of SSAs for chemical reaction networks and variance reduction techniques. In Chapter 2 we therefore first reviewed the two key components to this thesis, stochastic model and simulation approaches for chemical

reaction networks, and variance reduction methodology for MC methods, respectively. We discussed both exact SSAs, which were further studied in Chapters 3 and 5, and approximate SSAs, which played an important role in Chapter 4. Importantly, we placed extra emphasis on the problem of estimating distributions, for example in Section 2.3.1, because it is a common problem of interest in the context of chemical reaction networks and, perhaps, not as widely discussed as necessary in the literature. The concept of efficiency was also introduced as a trade-off between computational complexity and statistical error and we showed via simple examples throughout the chapter how algorithmic and statistical improvements can be made so as to improve the efficiency of stochastic simulations of chemical reaction networks, thereby setting the scene for the rest of this thesis.

Chapter 3

In Chapter 3 we focussed on the uniformisation technique for CTMCs. We introduced the UDM, which is an algorithmic improvement of Gillespie's DM, and showed how, in terms of complexity, this new method is on par with standard SSAs for chemical reaction networks. This allowed us to consider variance reduction techniques that can make use of the extra structure in a uniformised chemical reaction network. Firstly, via stratification of the total number of reactions firing in the uniformised system we could get modest improvements when estimating (raw) moments of chemical species using the UDM without increasing the computational complexity. Secondly, we used the uniformised system to construct the wUDM, a weighted variant of the UDM, which has the important property that it acts as a low-pass filter when used to estimate distributions of chemical species. For systems in which the species of interest evolves on the fast time-scale of the system the wUDM can be much more efficient than conventional approaches.

Chapter 4

Motivated by the impressive efficiency improvements researchers and practitioners have observed when using QMC methodology to simulate SDEs and SPDEs we first studied in Chapter 4 the combination of RQMC methods with the closest analogue to SSAs used for SDEs, namely the τ-leap method. Though RQMC variants improved upon standard SSAs we found that the efficiency improvements were smaller than perhaps anticipated. By comparing the difference between the results for the τ-leap method with those from numerically solving the CLE approximation we could pinpoint the reasons why standard QMC methodology is less well suited to direct application in SSAs for chemical reaction networks, namely the inherent discreteness and the large effective dimension of the problems relevant to this thesis.

A possible way to overcome these limitations is the use of array-RQMC, a recently developed variant of RQMC. In Section 4.3 we showed, empirically, how the combination of array-RQMC methodology with a variety of SSAs is a more suitable approach when simulating chemical reaction networks compared to standard RQMC methodology. In particular, for two different approximate SSAs we discussed how the type of summary statistic and the number of sample paths influence the observed performance benefit. Finally, we combined the array-RQMC method with the wUDM from Chapter 3 and showed that this combination can also be more efficient than a standard MC implementation of the wUDM for certain problems. We also showed that for this specific combination previously reported results in the literature misspecified the efficiency improvements.

Chapter 5

Finally, in Chapter 5 we considered the discrete analogue of the successful Brownian bridge construction for Wiener processes in the form of Poisson bridge methods, which can be used to construct unit-rate Poisson processes. Though such methods have been

around for roughly 20 years, recent interest in them sparked us to provide a theoretical analysis of a particularly promising variant of the Poisson bridge, the midpoint-based Poisson bridge. Using our results we were able to show how the Poisson bridge can be constructed optimally, in the sense of having the smallest average statistical error of the Poisson process over a fixed time interval of interest, when we use antithetic sampling to generate two negatively correlated Poisson process sample paths.

We then used the results on the optimal configuration of the midpoint-based Poisson bridge to create an antithetic variant of the MNRM, which can yield efficiency improvements compared to regular SSAs of up to orders of magnitude when we want to estimate the mean number of species in a chemical reaction network, importantly with minimal SSA configuration tweaking.

6.2 Future work and challenges

In this thesis we have explored the use of a number of variance reduction techniques in the context of MC simulations of chemical reactions. Though we have been able to derive and study new, more efficient, simulation approaches, many open questions and challenges related to the scope of this work remain. We now discuss some future ideas and projects split into three themes, namely: i) the testing and understanding of variance reduction techniques, both current and new; ii) the development of new variance reduction techniques; and iii) the application of variance reduction techniques.

6.2.1 Testing and understanding variance reduction techniques

Firstly, in order to exploit variance reduction techniques to their fullest potential we need to be sure of their correctness and understand their merits and weaknesses so that we can guide practical users of MC simulations, or developers of stochastic

simulation software, as to how and when to use a given approach. Here, we highlight two worthwhile avenues towards these goals.

Benchmark suite for SSAs

A crucial step in the development of any computational method is the verification and testing of the method's correctness and accuracy. It is therefore common practice to consider test problems for which we know exact answers, or can approximate answers to high-precision, and compare them to the output of simulations. For the numerical computation of both partial and ordinary differential equations one can, for example, use the method of manufactured solutions [188] or rely on a benchmark suite, e.g. as done for the DifferentialEquations.jl solver suite in the Julia language [176]. For SSAs this procedure is, however, complicated by the stochastic nature of the output. Nevertheless, given enough sample paths, a statistical comparison of the SSA output can be made to an exact solution of the CME if one is available, or alternatively to a reference output of an SSA using many samples. An important step towards verification of SSAs for chemical reaction networks is the Systems Biology Markup Language (SBML) discrete stochastic models test suite (DSMTS) [60], which contains mean and standard deviation solutions for three simple chemical reaction models in different scenarios. Though useful as a first step, we believe that a larger and better benchmark suite is required moving forward.

Firstly, the enrichment of the benchmark suite with summary statistics beyond the first two central moments would better reflect the breadth of applications for SSAs. For example, (marginal) distributions and exit-time distributions are widely used in practice to study chemical systems, but are not included in any benchmark. Note that for the problems currently incorporated in the DSMTS benchmark it is possible to solve the CME for the full species distribution, either analytically [104] (and Appendix 2.C) or numerically, e.g. using [200].

In addition we note that the behaviour of the systems in the DSMTS is relatively

trivial, a necessity to derive exact answers. Due to the lack of a method of manufactured solutions for stochastic models this means that little consistent benchmarking is done on systems beyond simple (linear) models that exhibit more exotic behaviour such as multiple scales, oscillations, rare events, delays or influence from external noise. However, note that such models, and their investigation via SSAs, are in fact commonplace in the literature. In particular, most new SSA development papers include at least one non-trivial reaction network, often tailored to the specifics of the new method presented. However, these papers generally fail to discuss weaknesses of the new SSA relative to other approaches. As such it is often not clear under what circumstances one should prefer one method over another. A central collection of models, in each case including parameters, initial conditions and time-course data, combined with a description of the class of behaviour for which they are suitable, would provide a level playing field to compare different (implementations of) SSAs. Ideally such a benchmark would also include the results of an (exact) SSA using many sample paths if no analytic results are known.

Effective dimension, dimension reduction and smoothing

A different challenge lies in gaining a deeper understanding of the performance of variance reduction methods in the context of chemical reaction networks. Many variance reduction techniques are introduced on problems with more smoothness or problems with a smaller state space than is common in the context of this thesis, and therefore it is often far from clear if and how effective a technique, taken from a different context, will be in the generation of sample paths for chemical reaction systems. In particular, a deeper understanding of the problem structure in the context of chemical reaction models could lead to better QMC methods.

For example, the concept of effective dimension is devised to quantify the relative importance of sources of statistical error in an SSA [162, Chapter 17]. Techniques to reduce said dimension are widely studied in the context of QMC methods for fi-

nancial applications, but the implications of problem dimension for the models of interest in biology have received little attention. In [124, 159] the authors study the statistical error decomposition in reaction networks via individual reaction channels and their interactions. Such knowledge could be used in conjunction with, for example, the Poisson bridge method in Chapter 5. Extending such a decomposition to approximate SSAs, such as the τ-leap method, could provide further explanation of the observations in Chapter 4.

Another factor influencing the efficacy of QMC methods discussed in this thesis is the discreteness of the state space in chemical reaction network models. In [63, Chapters 2, 7 and 8] it was suggested that smoothing techniques, such as conditional MC or interpolation for discrete random variables, can boost the effectiveness of QMC methods. In practice such techniques have already proven to be effective in MLMC methods for digital options, see e.g. [72] and references therein. We have, however, not seen adaptation of smoothing methods for models in the context of this thesis and there might therefore be scope to improve upon the QMC methods discussed in Chapter 4. For example, the conditional MC technique, as recently discussed in [3], either using an exact or approximate SSA, combined with a QMC method could prove to be an interesting alternative approach to QMC methods in this thesis.

6.2.2 Developing new variance reduction techniques

Secondly, we believe that, despite growing computational resources, a demand for more efficient simulation methods will remain. The development of new variance reduction techniques, either through combining known variance reduction techniques or by developing bespoke adaptations of techniques from a different context, will make accurate exploration of larger and more complex reaction systems with MC methods possible and we therefore mention two such ideas for new variance reduction techniques.

MFMC methods for chemical reactions

Over the past few decades a range of model reduction techniques have been derived, such as those based on a separation of scales [35, 44, 82, 93, 96, 108, 178], species-reaction coupling arguments [61], a marginal process framework [29], or on approximations like the CLE and the RRE from Section 2.1. The output of sample paths from such reduced models, though generally faster to generate, will normally be biased and therefore not always be informative (enough) of the dynamics of the full model. If, however, we can generate coupled sample paths from reduced models and the full model we can exploit MFMC techniques [172] to construct a reduced variance estimator. Such an approach first appeared in [5] and by using just a single reduced model it was shown that the resulting MFMC method was over an order of magnitude more efficient. Though these MFMC approaches share characteristics with hybrid-methods, such as [54, 97], we reiterate that MFMC methods yield unbiased estimators, whereas hybrid-methods are biased by construction and generally aim to merely control the bias arising from model reductions in a trade-off with efficiency.

MLQMC methods for chemical reactions

In the context of SDEs and stochastic PDEs the combination of MLMC and QMC methods has already been realised, e.g. [45, 73, 115]. Despite the lack of a general multilevel quasi-Monte Carlo (MLQMC) theory these methods in practice outperformed both standard MLMC and (R)QMC methods. In the context of CTMCs this feat has not been achieved yet, partially due to the fact that QMC methods have received little attention to date. We see two possible avenues towards a MLQMC method for the simulation of sample paths of chemical reaction networks, based on extending the currently available MLMC methods using the τ-leap or R-leap method.

- If the main cost of the MLMC simulation takes place at the coarse levels the priority in improving complexity and run-time lies in lowering the statistical

error of the (coarse) τ-leap/R-leap steps. This can be achieved by using either standard RQMC methods or the array-RQMC method and should be a relatively straightforward application of previous work. Note, however, that if we use the array-RQMC method the state space is now doubled due to presence of two (coupled) sample paths on a single level and the sorting step therefore is non-trivial, even if the underlying model is one-dimensional.

- If, however, sizeable cost in the standard MLMC simulation comes from the exact final level (Gillespie's DM), then one could try to use the combination of uniformisation and array-RQMC (Chapter 3 and [95]). Note that it is not clear how to couple this uniformised approach to a τ-leap approach and therefore the original MLMC method [5] cannot be used. It might, however, be possible to couple an array-RQMC exact level with an R-leap level, which would allow the MLMC method with R-leap [135] to be run with low-discrepancy points across all levels. Alternatively one could use results from Chapter 5, i.e. the use of QMC methods to generate the Poisson processes driving the sample paths on the exact final level. Note that this approach can also be used in conjunction with the original coupling method for MLMC in the context of chemical reaction systems [5].

6.2.3 Applying variance reduction techniques

Practical use of variance reduction techniques

Results in this thesis and the literature show that also in the context of chemical reaction network simulation variance reduction techniques can provide significant improvements over standard approaches. However, to date, uptake of these techniques by practitioners has been lagging behind. A major contributing factor to this discrepancy between theory and practice is the lack of implementations of variance reduction techniques in standard software suites, such as StochSS [53] and COPASI [101]. As

computational resources are inherently limited, moving forward it should therefore be the aim to marry efficient standard SSAs provided in such software suites with complementary variance reduction techniques in order to maximise the efficiency of exploration of (stochastic) chemical reaction networks.

A first practical concern, however, when implementing variance reduction techniques could be the parallelisation of SSAs when they are combined with a variance reduction technique. We note that apart from array-RQMC all variance reduction methods encountered in the context of chemical reaction networks are still embarrassingly parallel. The array-RQMC method is not strictly embarrassingly parallel due to the sorting step, which requires communication between synchronised trajectories. In addition, some implementations of variance reduction techniques can result in a larger memory footprint, e.g. a standard implementation of the Poisson bridge in Chapter 5 first generates the Poisson processes needed for a simulation with the MNRM and thus have to store these in memory, rather than generating them on-the-fly. Such considerations have to be explored further in the context of simulation software development.

A further concern for software developers could be the conception that model complexity must be low or a certain structure needs to be present in models for variance reduction techniques to work effectively. Efforts to speed up SSAs have often been motivated by the need to simulate increasingly 'larger and more complex' chemical reaction networks, which might impede the effectiveness of variance reduction techniques. Notably, spatially extended models modelled in the framework of CTMCs can certainly lead to systems with many reactions and reactants and the use of variance reduction for such systems has received little attention. Nevertheless, there is a special structure present in spatial models, represented by the distinct nature of diffusion reactions compared to regular chemical reactions. The development of variance reduction methods tailored to this specific structure is therefore something that could be investigated in future work. We note, however, that many well-mixed chemical

reaction network models used in practice are relatively small in size, see for example Figure 6.1. We can attribute this to the fact that models are often build to be as simple as possible so as to remain interpretable, an application of Occam's razor. With this in mind, the examples in this thesis have been relatively low-dimensional (up to 8 species and 10 reactions) in order to provide a thorough investigation of the workings and limitations of variance reduction techniques in the context of chemical reaction networks. The outcome of this thesis and related work in the literature shows that the use of effective variance reduction techniques for many practical models is thus a genuine possibility.

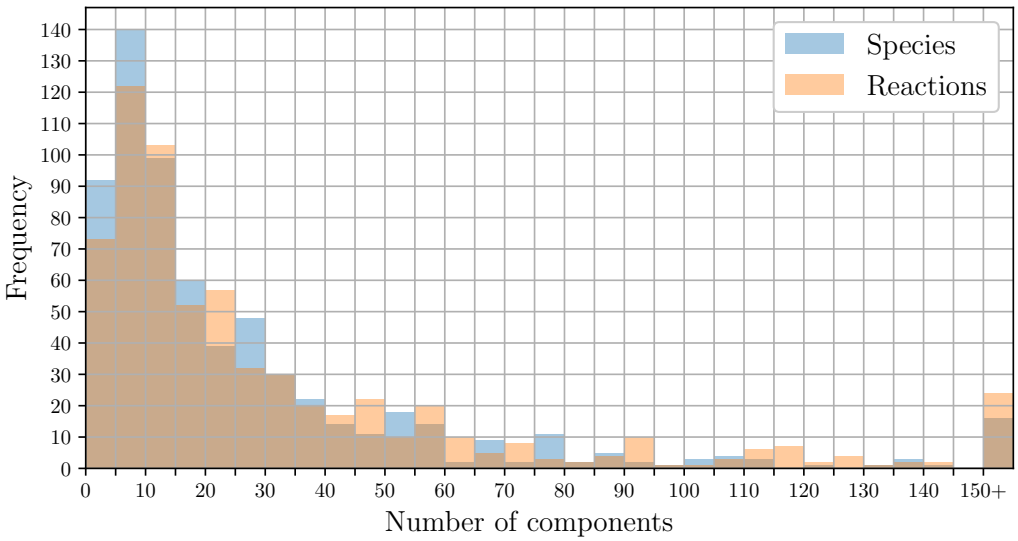


Figure 6.1: Size, in terms of the number of species and reactions, of the 640 curated models in the 31st release (June 2017) of the BioModels database [125]. Note that 75% of the models have less than 32 reactants and 40 reactions.

Inference

Finally, we look ahead and consider the use of variance reduction techniques in one of the most active areas of research in the field of chemical reaction networks, namely (Bayesian) inference. Problems in this context concern the identification of model structure or parameters from observed data and, as such, are known as inverse problems, as opposed to forward problems in which we are tasked with generating data or sample paths given a model structure and parameters. The variance reduction

methods in this thesis are defined purely in the context of forward problems, but we will discuss here how they can be employed to also improve inference methods.

For simplicity we focus here solely on the inverse problem of parameter inference and thus assume we have a model with unspecified parameters θ that we believe gave rise to some observed data $\mathbf{Y}_{\mathcal{D}}$, where we do not restrict ourselves as to the type of data. We can then use Bayes' theorem to find the probability distribution, $p(\theta \mid \mathbf{Y}_{\mathcal{D}})$, for the parameters, θ, given the data, $\mathbf{Y}_{\mathcal{D}}$, via

$$\underbrace{p(\theta \mid \mathbf{Y}_{\mathcal{D}})}_{\text{posterior}} \propto \underbrace{p(\theta)}_{\text{prior}} \cdot \underbrace{p(\mathbf{Y}_{\mathcal{D}} \mid \theta)}_{\text{likelihood}}, \tag{6.1}$$

where the beliefs about θ before observing the data are encoded in the prior. Methods to establish the posterior directly based on equation (6.1) generally require special structure in the model and data. If such an approach is not applicable we can often find methods that generate samples θ_i from the posterior distribution, such as Markov chain Monte Carlo (MCMC). In the context of chemical reaction networks with data consisting of (discrete) observations of (some of) the species we can sample from the posterior distribution using the MCMC methods in [66, 179, 180, 222]. Note, however, that these methods, which are exact in the sense that they target the true posterior distribution, $p(\theta \mid \mathbf{Y}_{\mathcal{D}})$, are generally only computationally feasible for models with small state spaces, because they rely on the computation of the likelihood, $p(\mathbf{Y}_{\mathcal{D}} \mid \theta)$. Such approaches are infeasible for many chemical reaction systems of interest, where the likelihood function becomes too expensive to evaluate, and therefore most recent research has been focused on approximate inference techniques. The first important class of such methods performs exact inference on approximate models with tractable likelihoods, such as the RRE, the CLE or its linear noise approximation, and we refer the reader to [193, Section 6.4] and references therein for more information. The second class of methods, likelihood-free inference methods, instead works with approximate proxies for the likelihood of the original model, generally computed via

the use of forward simulations, and here we highlight three such approaches that we think can be improved upon by incorporating variance reduction techniques.

Pseudo-marginal MCMC

The crucial observation from [12] is that for an MCMC method to correctly target the posterior distribution it is sufficient to work with unbiased estimates of the likelihood functions, $\hat{p}(\mathbf{Y}_{\mathcal{D}} \mid \theta)$, rather than the actual likelihood, $p(\mathbf{Y}_{\mathcal{D}} \mid \theta)$, leading to a pseudo-marginal approach. This implies that, rather than considering the likelihood of the data under all possible sample paths, we can run a finite number, N, of forward simulations to generate a MC estimate of the likelihood based on these N sample paths. In the context of chemical reactions such estimates can be constructed using blind forward simulations, but such an approach is bound to be ineffective if the observation noise in (part of) the data is small as noted, for example, in [87]. To improve the efficiency of pseudo-marginal methods we can use conditional path sampling methods, e.g. using the recent method provided in [86]. We note that such conditional sampling methods often rely on forward simulations with (sequential) importance sampling to bring simulations closer to the observed data and thus can already be viewed as a type of variance reduction technique. This, however, does not prevent the use of further additional variance reduction techniques, as for example shown by the recent sequential quasi-Monte Carlo (SQMC) method [67], which combines QMC with a variant of sequential importance sampling.

In particular, we propose that exploration of the SQMC method in combination with a bridged τ-leap method, inspired by the conditional sampling method by [86], could be interesting based on the promising results in Section 4.3.1 and the relationship between the SQMC and array-RQMC methods. Though this approach would be approximate in nature, it could be competitive when used in a pseudo-marginal MCMC sampler due to an improved effective sample size, see, for example, [67, Section 5.3]. Alternatively, and slightly more involved, we could combine the array-RQMC

version of the wUDM as discussed in Section 4.3.3 with the conditional sampling method proposed in [86] to construct an 'exact approximate' inference method.

Approximate Bayesian computation

In order to estimate the true likelihood, pseudo-marginal MCMC methods still rely on it being feasible to compute the path-wise likelihood of the data. For large chemical reaction models or high-dimensional data even calculating the likelihood for single sample paths can become computationally very demanding. A viable alternative to previously discussed pseudo-marginal methods in this case is the use of approximate Bayesian computation (ABC). The basic ingredient of the ABC method is that it replaces the true likelihood by instead considering a non-parametric approximation of the likelihood which is much cheaper to compute. To construct this proxy we define a distance metric, $\rho(\mathbf{Y}_{\mathcal{D}}, \mathbf{Y}_{\mathcal{S}})$, which compares the data, $\mathbf{Y}_{\mathcal{D}}$, with simulated data from the model, $\mathbf{Y}_{\mathcal{S}}$, and approximate the likelihood via

$$p_{\text{ABC}}(\mathbf{Y}_{\mathcal{D}}|\theta) = p(\rho(\mathbf{Y}_{\mathcal{D}}, \mathbf{Y}_{\mathcal{S}}) \leq \varepsilon \mid \theta) \approx p(\mathbf{Y}_{\mathcal{D}} \mid \theta), \tag{6.2}$$

where ε is a distance parameter. Note that, due to the approximation of the likelihood, the ABC method does not target the true posterior distribution. In practice the ABC likelihood is also not analytically tractable and instead is computed using forward simulations. This makes the ABC method effectively a pseudo-marginal method, but one that targets an approximate posterior distribution. We refer the reader to [217] and references therein for a comprehensive review on ABC methods with a focus on the context of chemical reaction networks.

Variance reduction via coupling arguments (Section 2.4.2) in the context of ABC methods has received a lot of attention recently, see for example [174, 217]. However, despite what one might think based on results for many pseudo-marginal methods [52, 197], there is an intrinsic limit to the usefulness of variance reduction techniques in

the forward simulations of ABC methods. For many of the standard ABC methods, such as ABC rejection sampling, ABC importance sampling and ABC MCMC, it was recently proven in [26] that if the parameter proposals are generated via standard MC methodology then it is most efficient to use only a single sample path to estimate the ABC likelihood. Effectively this is due to the fact that the statistical error in these ABC approaches stems from two sources, the sampling of parameter proposals and the estimation of the ABC likelihood, respectively. Using more samples to estimate the ABC likelihood will reduce part of the statistical error, but at the same time generating more samples incurs a proportional cost. The statistical error due to the parameter sampling, however, remains, regardless of the number of samples used to estimate the ABC likelihood. The overall efficiency of the standard MC version of ABC methods can therefore not be improved, and in fact is often lowered, by generating more samples to estimate the ABC likelihood. Since many variance reduction techniques for forward simulation methods rely on inducing a correlation between at least two sample paths this effectively rules out such approaches for most ABC methods using standard MC parameter proposals.

Recently, however, it was shown in [30] that the use of QMC methods to generate parameter proposals removes this intrinsic limit for some important ABC methods, such as ABC sequential Monte Carlo (SMC), ABC rejection sampling and ABC importance sampling. As a result it becomes worthwhile to use multiple forward simulations per parameter proposal to estimate the likelihood and we can leverage this by introducing a correlation between sample paths that reduces the variance of ABC likelihood estimates, e.g. using antithetic sampling or QMC methods. We therefore believe that moving forward in the context of inference on chemical reaction network models the use of variance reduction methods to reduce both sources of statistical error in ABC methods, i.e. due to parameter proposals as in [30], and forward simulations using methods such as described in this thesis, could become the standard.

Similar in spirit to the result in [30] it was shown in [167] that the combination of

certain MCMC approaches with specific QMC methods can also significantly reduce the statistical error due to parameter sampling. If this result extends to pseudo-marginal MCMC approaches then a similar conclusion regarding variance reduction methods for forward simulations could also hold for ABC MCMC.

Bayesian synthetic likelihood

Lastly, we mention a third likelihood-free method that recently gained attention, namely Bayesian synthetic likelihood (BSL) [175]. Similar in spirit to the ABC method, the BSL method replaces the true model likelihood by an approximate proxy, but, contrary to the ABC method, it assumes a parametric form for this approximate likelihood. In particular, the BSL method assumes that the summary statistics follow a multivariate normal distribution, implying that the (synthetic) likelihood under this simplification is also Gaussian. For most models this synthetic likelihood, however, is not known and in practice the BSL method is therefore often used in a pseudo-marginal approach, i.e. the synthetic likelihood is constructed using (estimated) summary statistics from N forward simulations where, contrary to the standard ABC methods, $N > 1$ generally. Due to its reliance on forward simulations to estimate likelihoods many of the possible improvements using variance reduction techniques of the ABC method that we discussed in the previous section could equally well apply for the BSL method.

6.3 Conclusions

Looking back on the work contained in this thesis and by standing on the shoulders of giants, we see that no one-size-fits-all approach exists for stochastic simulations of chemical reaction networks. However, at the same time we realise that we can certainly do much better than following the conventional approaches. The results in this thesis contribute not only to the development of improved simulation procedures, but also to a deeper understanding of the challenges and limitations one is faced with in the field of chemical reaction network simulation.

If our small minds, for some convenience, divide this glass of wine, this universe, into parts – physics, biology, geology, astronomy, psychology, and so on – remember that nature does not know it!
So let us put it all back together, not forgetting ultimately what it is for.
Let it give us one more final pleasure: drink it and forget it all!

Richard Feynman (1918-1988)

References

1. Anderson, D. F. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *The Journal of Chemical Physics* **127,** 214107 (2007).

2. Anderson, D. F. Incorporating postleap checks in tau-leaping. *The Journal of Chemical Physics* **128,** 054103 (2008).

3. Anderson, D. F. & Ehlert, K. W. Conditional Monte Carlo for reaction networks. arXiv: 1906.05353 (2019).

4. Anderson, D. F., Ganguly, A. & Kurtz, T. G. Error analysis of tau-leap simulation methods. *The Annals of Applied Probability* **21,** 2226–2262 (2011).

5. Anderson, D. F. & Higham, D. J. Multilevel Monte Carlo for continuous time Markov chains, with applications in biochemical kinetics. *Multiscale Modeling & Simulation* **10,** 146–179 (2012).

6. Anderson, D. F., Higham, D. J., Leite, S. C. & Williams, R. J. On constrained Langevin equations and (bio)chemical reaction networks. *Multiscale Modeling & Simulation* **17,** 1–30 (2019).

7. Anderson, D. F., Higham, D. J. & Sun, Y. Complexity of multilevel Monte Carlo tau-leaping. *SIAM Journal on Numerical Analysis* **52,** 3106–3127 (2014).

8. Anderson, D. F., Higham, D. J. & Sun, Y. Computational complexity analysis for Monte Carlo approximations of classically scaled population processes. *Multiscale Modeling & Simulation* **16,** 1206–1226 (2018).

9. Anderson, D. F. & Koyama, M. Weak error analysis of numerical methods for stochastic models of population processes. *Multiscale Modeling & Simulation* **10,** 1493–1524 (2012).

10. Anderson, D. F. & Kurtz, T. G. *Continuous time Markov chain models for chemical reaction networks* in *Design and Analysis of Biomolecular Circuits* 3–42 (Springer, New York, NY, 2011).

11. Anderson, D. F. & Yuan, C. Low variance couplings for stochastic models of intracellular processes with time-dependent rate functions. *Bulletin of Mathematical Biology* (2018).

12. Andrieu, C. & Roberts, G. O. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics* **37,** 697–725 (2009).

13. Auger, A., Chatelain, P. & Koumoutsakos, P. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *The Journal of Chemical Physics* **125,** 084103 (2006).

14. Avramidis, A. N. & Wilson, J. R. A splitting scheme for control variates. *Operations Research Letters* **14,** 187–198 (1993).

15. Backenköhler, M., Bortolussi, L. & Wolf, V. *Control variates for stochastic simulation of chemical reaction networks* in *Computational Methods in Systems Biology* (eds Bortolussi, L. & Sanguinetti, G.) (Springer, Cham, 2019), 42–59.

16. Balázsi, G., van Oudenaarden, A. & Collins, J. J. Cellular decision making and biological noise: From microbes to mammals. *Cell* **144,** 910–925 (2011).

17. Barrio, M., Burrage, K. & Burrage, P. Stochastic linear multistep methods for the simulation of chemical kinetics. *The Journal of Chemical Physics* **142,** 064101 (2015).

18. Barrio, M., Burrage, K., Leier, A. & Tian, T. Oscillatory regulation of Hes1: Discrete stochastic delay modelling and simulation. *PLoS Computational Biology* **2,** e117 (2006).

19. Beentjes, C. H. L. & Baker, R. E. Quasi-Monte Carlo methods applied to tau-leaping in stochastic biological systems. *Bulletin of Mathematical Biology* **81,** 2931–2959 (2019).

20. Beentjes, C. H. L. & Baker, R. E. Uniformization techniques for stochastic simulation of chemical reaction networks. *The Journal of Chemical Physics* **150,** 154107 (2019).

21. Ben Hammouda, C., Ben Rached, N. & Tempone, R. Importance sampling for a robust and efficient multilevel Monte Carlo estimator for stochastic reaction networks. arXiv: 1911.06286 (2019).

22. Ben Hammouda, C., Moraes, A. & Tempone, R. Multilevel hybrid split-step implicit tau-leap. *Numerical Algorithms* **74,** 527–560 (2017).

23. Berblinger, M., Schlier, C. & Weiss, T. Monte Carlo integration with quasi-random numbers: experience with discontinuous integrands. *Computer Physics Communications* **99,** 151–162 (1997).

24. Biancalani, T., Dyson, L. & McKane, A. J. Noise-induced bistable states and their mean switching time in foraging colonies. *Physical Review Letters* **112,** 038101 (2014).

25. Blake, W. J., Kærn, M., Cantor, C. R. & Collins, J. J. Noise in eukaryotic gene expression. *Nature* **422,** 633–637 (2003).

26. Bornn, L., Pillai, N. S., Smith, A. & Woodard, D. The use of a single pseudo-sample in approximate Bayesian computation. *Statistics and Computing* **27,** 583–590 (2017).

27. Bowsher, C. G. & Swain, P. S. Identifying sources of variation and the flow of information in biochemical networks. *Proceedings of the National Academy of Sciences of the United States of America* **109,** E1320–8 (2012).

28. Bronstein, L. & Koeppl, H. A variational approach to moment-closure approximations for the kinetics of biomolecular reaction networks. *The Journal of Chemical Physics* **148,** 014105 (2018).

29. Bronstein, L. & Koeppl, H. Marginal process framework: A model reduction tool for Markov jump processes. *Physical Review E* **97,** 062147 (2018).

30. Buchholz, A. & Chopin, N. Improving approximate Bayesian computation via quasi-Monte Carlo. *Journal of Computational and Graphical Statistics* **28,** 205–219 (2019).

31. Burrage, K. & Tian, T. *Poisson Runge-Kutta methods for chemical reaction systems* in *Advances in Scientific Computing and Applications* (eds Lu, Y., Sun, W. & Tang, T.) (Science Press, 2004), 82–96.

32. Caflisch, R. E. Monte Carlo and quasi-Monte Carlo methods. *Acta Numerica* **7,** 1 (1998).

33. Caflisch, R. E., Morokoff, W. & Owen, A. B. Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension. *The Journal of Computational Finance* **1,** 27–46 (1997).

34. Cai, X. Exact stochastic simulation of coupled chemical reactions with delays. *The Journal of Chemical Physics* **126,** 124108 (2007).

35. Cao, Y., Gillespie, D. T. & Petzold, L. R. Avoiding negative populations in explicit Poisson tau-leaping. *The Journal of Chemical Physics* **123,** 054104 (2005).

36. Cao, Y., Gillespie, D. T. & Petzold, L. R. Efficient step size selection for the tau-leaping simulation method. *The Journal of Chemical Physics* **124,** 044109 (2006).

37. Cao, Y., Li, H. & Petzold, L. R. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *The Journal of Chemical Physics* **121,** 4059–4067 (2004).

38. Cao, Y. & Petzold, L. R. Accuracy limitations and the measurement of errors in the stochastic simulation of chemically reacting systems. *Journal of Computational Physics* **212,** 6–24 (2006).

39. Cao, Y., Petzold, L. R., Rathinam, M. & Gillespie, D. T. The numerical stability of leaping methods for stochastic simulation of chemically reacting systems. *The Journal of Chemical Physics* **121,** 12169 (2004).

40. Carruth, J., Tygert, M. & Ward, R. A comparison of the discrete Kolmogorov-Smirnov statistic and the Euclidean distance. arXiv: 1206.6367 (2012).

41. Ceccato, A. & Frezzato, D. Remarks on the chemical Fokker-Planck and Langevin equations: Nonphysical currents at equilibrium. *The Journal of Chemical Physics* **148,** 064114 (2018).

42. Chatterjee, A., Vlachos, D. G. & Katsoulakis, M. A. Binomial distribution based τ-leap accelerated stochastic simulation. *The Journal of Chemical Physics* **122,** 024112 (2005).

43. Cochran, W. G. *Sampling Techniques* 3rd ed. (John Wiley & Sons, New York, 1977).

44. Cotter, S. L. Constrained approximation of effective generators for multiscale stochastic reaction networks and application to conditioned path sampling. *Journal of Computational Physics* **323,** 265–282 (2016).

45. Croci, M., Giles, M. B. & Farrell, P. E. Multilevel quasi Monte Carlo methods for elliptic PDEs with random field coefficients via fast white noise sampling. arXiv: 1911.12099 (2019).

46. Daigle, B. J., Roh, M. K., Gillespie, D. T. & Petzold, L. R. Automated estimation of rare event probabilities in biochemical systems. *The Journal of Chemical Physics* **134,** 044110 (2011).

47. Dana, S. & Raha, S. Physically consistent simulation of mesoscale chemical kinetics: The non-negative FIS-α method. *Journal of Computational Physics* **230,** 8813–8834 (2011).

48. Devroye, L. *Non-Uniform Random Variate Generation* 1st ed. (Springer-Verlag, New York, NY, 1986).

49. Dick, J., Kuo, F. Y. & Sloan, I. H. High-dimensional integration: The quasi-Monte Carlo way. *Acta Numerica* **22,** 133–288 (2013).

50. van Dijk, N. M. Approximate uniformization for continuous-time Markov chains with an application to performability analysis. *Stochastic Processes and their Applications* **40,** 339–357 (1992).

51. van Dijk, N. M., van Brummelen, S. P. J. & Boucherie, R. J. J. Uniformization: Basics, extensions and applications. *Performance Evaluation* **118,** 8–32 (2018).

52. Doucet, A., Pitt, M. K., Deligiannidis, G. & Kohn, R. Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. *Biometrika* **102,** 295–313 (2015).

53. Drawert, B. *et al.* Stochastic simulation service: Bridging the gap between the computational expert and the biologist. *PLoS Computational Biology* **12,** e1005220 (2016).

54. Duncan, A., Liao, S., Vejchodský, T., Erban, R. & Grima, R. Noise-induced multistability in chemical systems: Discrete versus continuum modeling. *Physical Review E* **91,** 042111 (2015).

55. E, W., Liu, D. & Vanden-Eijnden, E. Nested stochastic simulation algorithm for chemical kinetic systems with disparate rates. *The Journal of Chemical Physics* **123,** 194107 (2005).

56. E, W., Liu, D. & Vanden-Eijnden, E. Nested stochastic simulation algorithms for chemical kinetic systems with multiple time scales. *Journal of Computational Physics* **221,** 158–180 (2007).

57. Elowitz, M. B., Levine, A. J., Siggia, E. D. & Swain, P. S. Stochastic gene expression in a single cell. *Science* **297,** 1183–1186 (2002).

58. Emsermann, M. & Simon, B. Improving simulation efficiency with quasi control variates. *Stochastic Models* **18,** 425–448 (2002).

59. Erban, R., Chapman, S. J. & Maini, P. K. A practical guide to stochastic simulations of reaction-diffusion processes. arXiv: 0704.1908 (2007).

60. Evans, T. W., Gillespie, C. S. & Wilkinson, D. J. The SBML discrete stochastic models test suite. *Bioinformatics* **24,** 285–286 (2008).

61. Feng, C. & Hillston, J. Accelerating simulation of population continuous time Markov chains via automatic model reduction. *Performance Evaluation* **120,** 20–35 (2018).

62. Fox, B. L. *Generating Poisson processes by quasi-Monte Carlo* tech. rep. (1996).

63. Fox, B. L. *Strategies for Quasi-Monte Carlo* 1st ed. (Springer US, 1999).

64. Fox, B. L. & Glynn, P. W. Discrete-time conversion for simulating finite-horizon Markov processes. *SIAM Journal on Applied Mathematics* **50,** 1457–1473 (1990).

65. Gardiner, C. *Stochastic Methods: A Handbook for the Natural and Social Sciences* 4th ed. (Springer-Verlag, Berlin, Heidelberg, 2009).

66. Georgoulas, A., Hillston, J. & Sanguinetti, G. Unbiased Bayesian inference for population Markov jump processes via random truncations. *Statistics and Computing* **27,** 991–1002 (2017).

67. Gerber, M. & Chopin, N. Sequential quasi Monte Carlo. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **77,** 509–579 (2015).

68. Gibson, M. A. & Bruck, J. Efficient exact stochastic simulation of chemical systems with many species and many channels. *The Journal of Physical Chemistry A* **104,** 1876–1889 (2000).

69. Gil, A., Segura, J. & Temme, N. M. Efficient and accurate algorithms for the computation and inversion of the incomplete gamma function ratios. *SIAM Journal on Scientific Computing* **34,** A2965–A2981 (2012).

70. Giles, M. B. Algorithm 955: Approximation of the inverse Poisson cumulative distribution function. *ACM Transactions on Mathematical Software* **42,** 1–22 (2016).

71. Giles, M. B. Multi-level Monte Carlo path simulation. *Operations Research* **56,** 607–617 (2008).

72. Giles, M. B. Multilevel Monte Carlo methods. *Acta Numerica* **24,** 259–328 (2015).

73. Giles, M. B. & Waterhouse, B. J. Multilevel quasi-Monte Carlo path simulation. *Advanced Financial Modelling, Radon Series on Computational and Applied Mathematics* **8,** 165–181 (2009).

74. Gillespie, C. S. & Golightly, A. Guided proposals for efficient weighted stochastic simulation. *The Journal of Chemical Physics* **150,** 224103 (2019).

75. Gillespie, D. T. A rigorous derivation of the chemical master equation. *Physica A: Statistical Mechanics and its Applications* **188,** 404–425 (1992).

76. Gillespie, D. T. Approximate accelerated stochastic simulation of chemically reacting systems. *The Journal of Chemical Physics* **115,** 1716–1733 (2001).

77. Gillespie, D. T. Deterministic limit of stochastic chemical kinetics. *The Journal of Physical Chemistry B* **113,** 1640–1644 (2009).

78. Gillespie, D. T. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry* **81,** 2340–2361 (1977).

79. Gillespie, D. T. The chemical Langevin equation. *The Journal of Chemical Physics* **113,** 297 (2000).

80. Gillespie, D. T., Hellander, A. & Petzold, L. R. Perspective: Stochastic algorithms for chemical kinetics. *The Journal of Chemical Physics* **138,** 170901 (2013).

81. Gillespie, D. T. & Petzold, L. R. Improved leap-size selection for accelerated stochastic simulation. *The Journal of Chemical Physics* **119,** 8229–8234 (2003).

82. Gillespie, D. T., Roh, M. & Petzold, L. R. Refining the weighted stochastic simulation algorithm. *The Journal of Chemical Physics* **130,** 174103 (2009).

83. Glasserman, P. & Gong, W.-B. *Derivative estimates from discontinuous realizations: smoothing techniques* in *Proceedings of the 21st conference on Winter simulation - WSC '89* (ACM Press, New York, New York, USA, 1989), 381–389.

84. Glasserman, P. *Monte Carlo Methods in Financial Engineering* (Springer-Verlag, 2003).

85. Glasserman, P. & Merener, N. Convergence of a discretization scheme for jump-diffusion processes with state–dependent intensities. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **460,** 111–127 (2004).

86. Golightly, A. & Sherlock, C. Efficient sampling of conditioned Markov jump processes. *Statistics and Computing* **29,** 1149–1163 (2019).

87. Golightly, A. & Wilkinson, D. J. Bayesian inference for Markov jump processes with informative observations. *Statistical Applications in Genetics and Molecular Biology* **14,** 169–88 (2015).

88. Grassmann, W. Transient solutions in markovian queueing systems. *Computers & Operations Research* **4,** 47–53 (1977).

89. Grima, R. Linear-noise approximation and the chemical master equation agree up to second-order moments for a class of chemical systems. *Physical Review E* **92,** 042124 (2015).

90. Grima, R., Thomas, P. & Straube, A. V. How accurate are the nonlinear chemical Fokker-Planck and chemical Langevin equations? *The Journal of Chemical Physics* **135,** 084103 (2011).

91. Gupta, A., Mikelson, J. & Khammash, M. A finite state projection algorithm for the stationary solution of the chemical master equation. *The Journal of Chemical Physics* **147,** 154101 (2017).

92. Hammersley, J. M. & Morton, K. W. A new Monte Carlo technique: antithetic variates. *Mathematical Proceedings of the Cambridge Philosophical Society* **52,** 449–475 (1956).

93. Haseltine, E. L. & Rawlings, J. B. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *The Journal of Chemical Physics* **117,** 6959–6969 (2002).

94. He, Z. & Wang, X. On the convergence rate of randomized quasi-Monte Carlo for discontinuous functions. *SIAM Journal on Numerical Analysis* **53,** 2488–2503 (2015).

95. Hellander, A. Efficient computation of transient solutions of the chemical master equation based on uniformization and quasi-Monte Carlo. *The Journal of Chemical Physics* **128,** 154109 (2008).

96. Hellander, A. & Lötstedt, P. Hybrid method for the chemical master equation. *Journal of Computational Physics* **227,** 100–122 (2007).

97. Hepp, B., Gupta, A. & Khammash, M. Adaptive hybrid simulations for multiscale stochastic reaction networks. *The Journal of Chemical Physics* **142,** 034118 (2015).

98. Higham, D. J. Modeling and simulating chemical reactions. *SIAM Review* **50,** 347–368 (2008).

99. Hilfinger, A. & Paulsson, J. Separating intrinsic from extrinsic fluctuations in dynamic biological systems. *Proceedings of the National Academy of Sciences of the United States of America* **108,** 12167–72 (2011).

100. Hong Li, H. & Petzold, L. Efficient parallelization of the stochastic simulation algorithm for chemically reacting systems on the graphics processing unit. *The International Journal of High Performance Computing Applications* **24,** 107–116 (2010).

101. Hoops, S. *et al.* COPASI–a COmplex PAthway SImulator. *Bioinformatics* **22,** 3067–3074 (2006).

102. Hou, Z. & Xin, H. Internal noise stochastic resonance in a circadian clock system. *The Journal of Chemical Physics* **119,** 11508 (2003).

103. Hu, Y., Li, T. & Min, B. A weak second order tau-leaping method for chemical kinetic systems. *The Journal of Chemical Physics* **135,** 024113 (2011).

104. Jahnke, T. & Huisinga, W. Solving the chemical master equation for monomolecular reaction systems analytically. *Journal of Mathematical Biology* **54,** 1–26 (2007).

105. Jensen, A. Markoff chains as an aid in the study of Markoff processes. *Scandinavian Actuarial Journal* **1953,** 87–91 (1953).

106. Kahn, H. & Marshall, A. W. Methods of reducing sample size in Monte Carlo computations. *Journal of the Operations Research Society of America* **1,** 263–278 (1953).

107. van Kampen, N. *Stochastic Processes in Physics and Chemistry* 3rd ed. (Elsevier, Amsterdam, The Netherlands, 2007).

108. Kang, H.-W. & Kurtz, T. G. Separation of time-scales and model reduction for stochastic reaction networks. *The Annals of Applied Probability* **23,** 529–583 (2013).

109. Kholodenko, B. N. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *European Journal of Biochemistry* **267,** 1583–1588 (2000).

110. Klingbeil, G., Erban, R., Giles, M. B. & Maini, P. K. Fat versus thin threading approach on GPUs: Application to stochastic simulation of chemical reactions. *IEEE Transactions on Parallel and Distributed Systems* **23,** 280–287 (2012).

111. Kloeden, P. E. & Platen, E. *Numerical Solution of Stochastic Differential Equations* (Springer-Verlag, Berlin, Heidelberg, 1992).

112. Kokonendji, C. C. & Kiessé, T. S. Discrete associated kernels method and extensions. *Statistical Methodology* **8,** 497–516 (2011).

113. Komarov, I. & D'Souza, R. M. Accelerating the Gillespie exact stochastic simulation algorithm using hybrid parallel execution on graphics processing units. *PloS ONE* **7,** e46693 (2012).

114. Komarov, I., D'Souza, R. M. & Tapia, J.-J. Accelerating the Gillespie τ-leaping method using graphics processing units. *PLoS ONE* **7,** e37370 (2012).

115. Kuo, F. Y., Schwab, C. & Sloan, I. H. Multi-level quasi-Monte Carlo finite element methods for a class of elliptic PDEs with random coefficients. *Foundations of Computational Mathematics* **15,** 411–449 (2015).

116. Kurtz, T. G. *Approximation of Population Processes* (Society for Industrial and Applied Mathematics, 1981).

117. Kurtz, T. G. *Limit theorems and diffusion approximations for density dependent Markov chains* in *Stochastic Systems: Modeling, Identification and Optimization, I* 67–78 (Springer, Berlin, Heidelberg, 1976).

118. Kuwahara, H. & Mura, I. An efficient and exact stochastic simulation method to analyze rare events in biochemical systems. *The Journal of Chemical Physics* **129,** 165101 (2008).

119. L'Ecuyer, P., Lécot, C. & Tuffin, B. A randomized quasi-Monte Carlo simulation method for Markov chains. *Operations Research* **56,** 958–975 (2008).

120. L'Ecuyer, P. & Lemieux, C. *Recent advances in randomized quasi-Monte Carlo methods* in *Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications* (eds Dror, M., L'Ecuyer, P. & Szidarovszky, F.) 419–474 (Kluwer Academic Publishers, Boston, 2002).

121. L'Ecuyer, P. & Munger, D. Algorithm 958: Lattice builder: A general software tool for constructing rank-1 lattice rules. *ACM Transactions on Mathematical Software* **42,** 1–30 (2016).

122. L'Ecuyer, P., Munger, D., Lécot, C. & Tuffin, B. Sorting methods and convergence rates for Array-RQMC: Some empirical comparisons. *Mathematics and Computers in Simulation* **143,** 191–201 (2018).

123. L'Ecuyer, P. & Simard, R. Inverting the symmetrical beta distribution. *ACM Transactions on Mathematical Software* **32,** 509–520 (2006).

124. Le Maître, O. P., Knio, O. M. & Moraes, A. Variance decomposition in stochastic simulators. *The Journal of Chemical Physics* **142,** 244115 (2015).

125. Le Novere, N. *et al.* BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic Acids Research* **34,** D689–D691 (2006).

126. Lécot, C. & Tuffin, B. *Quasi-Monte Carlo methods for estimating transient measures of discrete time Markov chains* in *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing 2002* (ed Niederreiter, H.) (Springer, Berlin, Heidelberg, 2004), 329–343.

127. Lehmann, E. L. Some concepts of dependence. *The Annals of Mathematical Statistics* **37,** 1137–1153 (1966).

128. Leite, S. C. & Williams, R. J. A constrained Langevin approximation for chemical reaction networks. *The Annals of Applied Probability* **29,** 1541–1608 (2019).

129. Lemieux, C. *Monte Carlo and Quasi-Monte Carlo Sampling* 1st ed. (Springer-Verlag, New York, NY, 2009).

130. Leobacher, G. Fast orthogonal transforms and generation of Brownian paths. *Journal of Complexity* **28,** 278–302 (2012).

131. Lester, C., Yates, C. A. & Baker, R. E. Efficient parameter sensitivity computation for spatially extended reaction networks. *The Journal of Chemical Physics* **146,** 044106 (2017).

132. Lester, C., Yates, C. A., Giles, M. B. & Baker, R. E. An adaptive multi-level simulation algorithm for stochastic biological systems. *The Journal of Chemical Physics* **142,** 024113 (2015).

133. Lester, C., Baker, R. E., Giles, M. B. & Yates, C. A. Extending the multi-level method for the simulation of stochastic biological systems. *Bulletin of Mathematical Biology* **78,** 1640–1677 (2016).

134. Lester, C., Baker, R. E. & Yates, C. A. Efficiently simulating discrete-state models with binary decision trees. arXiv: 2001.07247 (2020).

135. Lester, C., Yates, C. A. & Baker, R. E. Robustly simulating biochemical reaction kinetics using multi-level Monte Carlo approaches. *Journal of Computational Physics* **375,** 1401–1423 (2018).

136. Levien, E. & Bressloff, P. C. Coupling sample paths to the thermodynamic limit in Monte Carlo estimators with applications to gene expression. *Journal of Computational Physics* **346,** 1–13 (2017).

137. Levin, D. & Peres, Y. *Markov Chains and Mixing Times* 2nd ed. (American Mathematical Society, 2017).

138. Li, H. & Petzold, L. *Logarithmic direct method for discrete stochastic simulation of chemically reacting systems* tech. rep. (University of California Santa Barbara, 2006), 1–11.

139. Lipková, J., Arampatzis, G., Chatelain, P., Menze, B. & Koumoutsakos, P. S-Leaping: An adaptive, accelerated stochastic simulation algorithm, bridging τ-leaping and R-leaping. *Bulletin of Mathematical Biology* **81,** 3074–3096 (2019).

140. MacNamara, S., Burrage, K. & Sidje, R. B. Multiscale modeling of chemical kinetics via the master equation. *Multiscale Modeling & Simulation* **6,** 1146–1168 (2008).

141. Maginnis, P. A., West, M. & Dullerud, G. E. Exact variance-reduced simulation of lattice continuous-time Markov chains with applications in reaction networks. *Bulletin of Mathematical Biology* **81,** 3159–3184 (2019).

142. Maginnis, P. A., West, M. & Dullerud, G. E. Variance-reduced simulation of lattice discrete-time Markov chains with applications in reaction networks. *Journal of Computational Physics* **322,** 400–414 (2016).

143. Maginnis, P. A., West, M. & Dullerud, G. E. *Anticorrelated discrete-time stochastic simulation* in *52nd IEEE Conference on Decision and Control* (IEEE, 2013), 618–623.

144. Matoušek, J. On the L2-discrepancy for anchored boxes. *Journal of Complexity* **14,** 527–556 (1998).

145. Mauch, S. & Stalzer, M. Efficient formulations for exact stochastic simulation of chemical systems. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **8,** 27–35 (2011).

146. McAdams, H. H. & Arkin, A. P. Stochastic mechanisms in gene expression. *Proceedings of the National Academy of Sciences* **94,** 814–819 (1997).

147. McAdams, H. H. & Arkin, A. P. It's a noisy business! Genetic regulation at the nanomolar scale. *Trends in Genetics* **15,** 65–69 (1999).

148. McCollum, J. M., Peterson, G. D., Cox, C. D., Simpson, M. L. & Samatova, N. F. The sorting direct method for stochastic simulation of biochemical systems with varying reaction execution behavior. *Computational Biology and Chemistry* **30,** 39–49 (2006).

149. McKay, M. D., Beckman, R. J. & Conover, W. J. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21,** 239 (1979).

150. McLeish, D. A general method for debiasing a Monte Carlo estimator. *Monte Carlo Methods and Applications* **17,** 301–315 (2011).

151. Mélykúti, B., Burrage, K. & Zygalakis, K. C. Fast stochastic simulation of biochemical reaction systems by alternative formulations of the chemical Langevin equation. *The Journal of Chemical Physics* **132,** 84103 (2010).

152. van Moorsel, A. P. & Sanders, W. H. Adaptive uniformization. *Communications in Statistics: Stochastic Models* **10,** 619–647 (1994).

153. Moraes, A., Tempone, R. & Vilanova, P. A multilevel adaptive reaction-splitting simulation method for stochastic reaction networks. *SIAM Journal on Scientific Computing* **38,** A2091–A2117 (2016).

154. Moraes, A., Tempone, R. & Vilanova, P. Multilevel hybrid Chernoff tau-leap. *BIT Numerical Mathematics* **56,** 189–239 (2016).

155. Morokoff, W. J. & Caflisch, R. E. Quasi-Monte Carlo integration. *Journal of Computational Physics* **122,** 218–230 (1995).

156. Moskowitz, B. & Caflisch, R. E. Smoothness and dimension reduction in quasi-Monte Carlo methods. *Mathematical and Computer Modelling* **23,** 37–54 (1996).

157. Munsky, B. & Khammash, M. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of Chemical Physics* **124,** 044104 (2006).

158. Nakatsukasa, Y. Approximate and integrate: Variance reduction in Monte Carlo integration via function approximation. arXiv: 1806.05492 (2018).

159. Navarro Jimenez, M., Le Maître, O. P. & Knio, O. M. Global sensitivity analysis in stochastic simulators of uncertain reaction networks. *The Journal of Chemical Physics* **145,** 244106 (2016).

160. Nobile, M. S., Cazzaniga, P., Besozzi, D., Pescini, D. & Mauri, G. cuTauLeaping: A GPU-powered tau-leaping stochastic simulator for massive parallel analyses of biological systems. *PLoS ONE* **9,** e91963 (2014).

161. Owen, A. B. Latin supercube sampling for very high-dimensional simulations. *ACM Transactions on Modeling and Computer Simulation* **8,** 71–102 (1998).

162. Owen, A. B. *Monte Carlo Theory, Methods and Examples* (2013).

163. Owen, A. B. *Randomly Permuted (t,m,s)-Nets and (t, s)-Sequences* in *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing 1994* (eds Niederreiter, H. & Shiue, P. J.-S.) (Springer New York, New York, NY, 1995), 299–317.

164. Owen, A. B. Scrambled net variance for integrals of smooth functions. *The Annals of Statistics* **25,** 1541–1562 (1997).

165. Owen, A. B. Scrambling Sobol' and Niederreiter-Xing Points. *Journal of Complexity* **14,** 466–489 (1998).

166. Owen, A. B. Variance with alternative scramblings of digital nets. *ACM Transactions on Modeling and Computer Simulation* **13,** 363–378 (2003).

167. Owen, A. B. & Tribble, S. D. A quasi-Monte Carlo Metropolis algorithm. *Proceedings of the National Academy of Sciences of the United States of America* **102,** 8844–9 (2005).

168. Ozbudak, E. M., Thattai, M., Kurtser, I., Grossman, A. D. & van Oudenaarden, A. Regulation of noise in the expression of a single gene. *Nature Genetics* **31,** 69–73 (2002).

169. Pasupathy, R., Schmeiser, B. W., Taaffe, M. R. & Wang, J. Control-variate estimation using estimated control means. *IIE Transactions* **44,** 381–385 (2012).

170. Paulsson, J., Berg, O. G. & Ehrenberg, M. Stochastic focusing: fluctuation-enhanced sensitivity of intracellular regulation. *Proceedings of the National Academy of Sciences of the United States of America* **97,** 7148–53 (2000).

171. Peherstorfer, B., Gunzburger, M. & Willcox, K. Convergence analysis of multifidelity Monte Carlo estimation. *Numerische Mathematik* **139,** 683–707 (2018).

172. Peherstorfer, B., Willcox, K. & Gunzburger, M. Optimal model management for multifidelity Monte Carlo estimation. *SIAM Journal on Scientific Computing* **38,** A3163–A3194 (2016).

173. Peherstorfer, B., Willcox, K. & Gunzburger, M. Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Review* **60,** 550–591 (2018).

174. Prescott, T. P. & Baker, R. E. Multifidelity approximate Bayesian computation. *SIAM/ASA Journal on Uncertainty Quantification* **8,** 114–138 (2020).

175. Price, L. F., Drovandi, C. C., Lee, A. & Nott, D. J. Bayesian synthetic likelihood. *Journal of Computational and Graphical Statistics* **27,** 1–11 (2018).

176. Rackauckas, C. & Nie, Q. DifferentialEquations.jl – A performant and feature-rich ecosystem for solving differential equations in Julia. *Journal of Open Research Software* **5** (2017).

177. Raj, A. & van Oudenaarden, A. Nature, nurture, or chance: Stochastic gene expression and its consequences. *Cell* **135,** 216–226 (2008).

178. Rao, C. V. & Arkin, A. P. Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the Gillespie algorithm. *The Journal of Chemical Physics* **118,** 4999–5010 (2003).

179. Rao, V. & Teh, Y. W. Fast MCMC sampling for Markov jump processes and extensions. *The Journal of Machine Learning Research* **14,** 3295–3320 (2013).

180. Rao, V. & Teh, Y. W. *MCMC for continuous-time discrete-state systems* in *Advances in Neural Information Processing Systems* (2012), 701–709.

181. Rathinam, M. Convergence of moments of tau leaping schemes for unbounded Markov processes on integer lattices. *SIAM Journal on Numerical Analysis* **54,** 415–439 (2016).

182. Rathinam, M., Petzold, L. R., Cao, Y. & Gillespie, D. T. Consistency and stability of tau-leaping schemes for chemical reaction systems. *Multiscale Modeling & Simulation* **4,** 867–895 (2005).

183. Rathinam, M., Petzold, L. R., Cao, Y. & Gillespie, D. T. Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method. *The Journal of Chemical Physics* **119,** 12784–12794 (2003).

184. Reis, M., Kromer, J. A. & Klipp, E. General solution of the chemical master equation and modality of marginal distributions for hierarchic first-order reaction networks. *Journal of Mathematical Biology* **77,** 377–419 (2018).

185. Rhee, C.-H. & Glynn, P. W. *A new approach to unbiased estimation for SDE's* in *Proceedings of the 2012 Winter Simulation Conference* (eds Laroque, C., Himmelspach, J., Pasupathy, R., Rose, O. & Uhrmacher, A. M.) (2012).

186. Rhee, C.-H. & Glynn, P. W. Unbiased estimation with square root convergence for SDE models. *Operations Research* **63,** 1026–1043 (2015).

187. Richtmyer, R. D. *The evaluation of definite integrals and a quasi-Monte Carlo method based on the properties of algebraic numbers* tech. rep. (Divison of Technical Information Extension, 1951), LA–1342.

188. Roache, P. J. Code verification by the method of manufactured solutions. *Journal of Fluids Engineering* **124,** 4 (2002).

189. Roh, M. K. Data-driven method for efficient characterization of rare event probabilities in biochemical systems. *Bulletin of Mathematical Biology* **81,** 3097–3120 (2019).

190. Roh, M. K., Daigle, B. J., Gillespie, D. T. & Petzold, L. R. State-dependent doubly weighted stochastic simulation algorithm for automatic characterization of stochastic biochemical rare events. *The Journal of Chemical Physics* **135,** 234108 (2011).

191. Roh, M. K., Gillespie, D. T. & Petzold, L. R. State-dependent biasing method for importance sampling in the weighted stochastic simulation algorithm. *The Journal of Chemical Physics* **133,** 174106 (2010).

192. Sandmann, W. Discrete-time stochastic modeling and simulation of biochemical networks. *Computational Biology and Chemistry* **32,** 292–297 (2008).

193. Schnoerr, D., Sanguinetti, G. & Grima, R. Approximation and inference methods for stochastic biochemical kinetics - A tutorial review. *Journal of Physics A: Mathematical and Theoretical* **50,** 093001 (2017).

194. Schnoerr, D., Sanguinetti, G. & Grima, R. Comparison of different moment-closure approximations for stochastic chemical kinetics. *The Journal of Chemical Physics* **143,** 185101 (2015).

195. Schnoerr, D., Sanguinetti, G. & Grima, R. The complex chemical Langevin equation. *The Journal of Chemical Physics* **141,** 024103 (2014).

196. Shahrezaei, V. & Swain, P. S. Analytical distributions for stochastic gene expression. *Proceedings of the National Academy of Sciences of the United States of America* **105,** 17256–61 (2008).

197. Sherlock, C., Thiery, A. H., Roberts, G. O. & Rosenthal, J. S. On the efficiency of pseudo-marginal random walk Metropolis algorithms. *The Annals of Statistics* **43,** 238–275 (2015).

198. Slepoy, A., Thompson, A. P. & Plimpton, S. J. A constant-time kinetic Monte Carlo algorithm for simulation of large biochemical reaction networks. *The Journal of Chemical Physics* **128,** 205101 (2008).

199. Smith, S. & Grima, R. Spatial stochastic intracellular kinetics: A review of modelling approaches. *Bulletin of Mathematical Biology,* 1–50 (2018).

200. Smith, S. & Shahrezaei, V. General transient solution of the one-step master equation in one dimension. *Physical Review E* **91,** 062119 (2015).

201. Sobol', I. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation* **55,** 271–280 (2001).

202. Sobol', I. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics* **7,** 86–112 (1967).

203. Srivastava, R., Anderson, D. F. & Rawlings, J. B. Comparison of finite difference based methods to obtain sensitivities of stochastic chemical kinetic models. *The Journal of Chemical Physics* **138,** 074110 (2013).

204. Swain, P. S., Elowitz, M. B. & Siggia, E. D. Intrinsic and extrinsic contributions to stochasticity in gene expression. *Proceedings of the National Academy of Sciences of the United States of America* **99,** 12795–800 (2002).

205. Thanh, V. H. Efficient anticorrelated variance reduction for stochastic simulation of biochemical reactions. *IET systems biology* **13,** 16–23 (2019).

206. Thanh, V. H., Marchetti, L., Reali, F. & Priami, C. Incorporating extrinsic noise into the stochastic simulation of biochemical reactions: A comparison of approaches. *The Journal of Chemical Physics* **148,** 064111 (2018).

207. Thanh, V. H., Priami, C. & Zunino, R. Accelerating rejection-based simulation of biochemical reactions with bounded acceptance probability. *The Journal of Chemical Physics* **144,** 224108 (2016).

208. Thanh, V. H., Priami, C. & Zunino, R. Efficient rejection-based simulation of biochemical reactions with stochastic noise and delays. *The Journal of Chemical Physics* **141,** 134116 (2014).

209. Thanh, V. H., Zunino, R. & Priami, C. Efficient stochastic simulation of biochemical reactions with noise and delays. *The Journal of Chemical Physics* **146,** 084107 (2017).

210. Thanh, V. H., Zunino, R. & Priami, C. On the rejection-based algorithm for simulation and analysis of large-scale reaction networks. *The Journal of Chemical Physics* **142,** 244106 (2015).

211. Tian, T. & Burrage, K. Binomial leap methods for simulating stochastic chemical kinetics. *The Journal of Chemical Physics* **121,** 10356–10364 (2004).

212. Vastola, J. J. Solving the chemical master equation for monomolecular reaction systems analytically: a Doi-Peliti path integral view. arXiv: 1911.00978 (2019).

213. Viana, F. A. C. *Things you wanted to know about the Latin hypercube design and were afraid to ask* in *10th World Congress on Structural and Multidisciplinary Optimization* (2013).

214. Voliotis, M., Thomas, P., Grima, R. & Bowsher, C. G. Stochastic simulation of biomolecular networks in dynamic environments. *PLoS Computational Biology* **12,** e1004923 (2016).

215. Wakimoto, K. On unbiased estimation of the population variance based on the stratified random sample (II). *Annals of the Institute of Statistical Mathematics* **22,** 429–433 (1970).

216. Wallace, E. W. J., Petzold, L. R., Gillespie, D. T. & Sanft, K. R. Linear noise approximation is valid over limited times for any chemical system that is sufficiently large. *IET Systems Biology* **6,** 102–115 (2012).

217. Warne, D. J., Baker, R. E. & Simpson, M. J. Simulation and inference algorithms for stochastic biochemical reaction networks: from basic concepts to state-of-the-art. *Journal of The Royal Society Interface* **16,** 20180943 (2019).

218. Wilkie, J. & Wong, Y. M. Positivity preserving chemical Langevin equations. *Chemical Physics* **353,** 132–138 (2008).

219. Wilkinson, D. J. Stochastic modelling for quantitative description of heterogeneous biological systems. *Nature Reviews Genetics* **10,** 122–133 (2009).

220. Wilson, D. & Baker, R. E. Multi-level methods and approximating distribution functions. *AIP Advances* **6,** 075020 (2016).

221. Yates, C. A. & Klingbeil, G. Recycling random numbers in the stochastic simulation algorithm. *The Journal of Chemical Physics* **138,** 094103 (2013).

222. Zhang, B. & Rao, V. Efficient parameter sampling for Markov jump processes. *Journal of Computational and Graphical Statistics,* 1–18 (2020).

223. Zhang, J., Watson, L. T. & Cao, Y. A modified uniformization method for the solution of the chemical master equation. *Computers & Mathematics with Applications* **59,** 573–584 (2010).