An Efficient Tau-Leaping Simulation Method for Stochastic Biochemical Kinetics

by

Serguei Rousskikh

Bachelor of Science, Ryerson University, 2015

A thesis

presented to Ryerson University

in partial fulfilment

of the requirements for the degree of

Master of Science

in the program of

Applied Mathematics

Toronto, Ontario, Canada, 2018

Declaration

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

Abstract

An Efficient Tau-Leaping Simulation Method for Stochastic Biochemical Kinetics

Serguei Rousskikh

Master of Science in Applied Mathematics

Ryerson University

2018

Stochastic modeling and simulation of biochemical systems are topics of high interest in Computational Biology. Stochastic mathematical models are critical in accurately capturing the variability observed experimentally in cellular processes, in particular when some species have low molecular numbers. Many, realistic biochemical networks exhibit stiffness, due to the presence of multiple time-scales. For such networks explicit simulation methods are computationally quite intensive. In this thesis, we introduce an improved implicit tau-leaping strategy for the simulation of stochastic biochemical kinetic models. Numerical tests on various biochemical systems of interest in applications show the efficiency of our method.

Acknowledgements

I would like first of all to express my appreciation to everyone who has supported me in my studies and beyond, and without whom this thesis would not have been possible. I would also like to take the time to posthumously acknowledge the late Dr. Daniel Gillespie, the pioneer and architect of this research topic.

I would like to begin by expressing my outmost gratitude to my supervisor Dr. Silvana Ilie who has spent much time, energy and resources in order to give me the best possible opportunity to succeed in this project. I thank her for the support given to me throughout this endeavor as well as the remarkable poise and patience she demonstrated during very trying times and during periods when I clearly was not at my best.

Subsequently, I would like to thank all of my classmates, friends and all with whom I have had the pleasure to work closely with over the past two years. I have had a wonderful time, met some extraordinary people and have had the chance to learn a lot from them in this time.

Finally, I would like to thank all of my family, here with me in Canada and back home in Russia. In particular I would like to thank my mother and father who have sacrificed so much to give me the best opportunities in life. The patience, guidance and support that they give me on a daily basis plays an immeasurable role in all aspects of my life. A very special mention goes to my little sister, who has witnessed the ups and downs of this process firsthand, and has always been there to provide a shoulder especially during the downs. I cannot end this without expressing my gratitude to my grandparents, who despite being thousands of miles away are aware of every little intricate detail surrounding my master's degree.

Dedication

To all that I have mentioned I dedicate this thesis.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction to Biochemical Systems

1.1 Motivation

Why is this topic so important? First and foremost, biochemical systems are at the heart of modern medicine and biomedical research. The medical and biomedical industry is already one of the highest grossing and socially vital economic sectors in the world. As the global population continues to age, particularly in North America and Europe, the reliance upon this industry will only continue to grow. Biochemical research already provides us with the basis for revolutionary medical procedures, treatments and medication. Yet, with all of the progress that has been made in recent history, the field still has boundless potential. In this thesis we strive to build on the massive amount of work already done with respect to biochemical simulation, and develop a more efficient way of achieving results.

Aside from the broad economic and medical implications of computational biology, the motivational factor at the heart of this research is our interest in improving stochastic simulation of mathematical models of biochemical systems. In particular, our goal is to identify and correct inefficiencies in stochastic methods. The focus of our study will be the tau-leaping method developed by Gillespie [13] for simulating stochastic discrete models of biochemical kinetics.

In the study of biochemical networks we encounter a wide spectrum of systems and a range of numerical strategies to approximate the solution of their mathematical models. Biochemical systems can be categorized into spectrum ranging from small systems with few species and small population sizes to large systems with many species and large populations sizes. On the small system side of the scale, an appropriate solution algorithm would be the stochastic simulation algorithm (SSA) [9], while on the opposite end one would often utilize ordinary differential equation (ODE) solvers for the reaction rate equations (RRE). These methods have been extensively studied and are currently heavily relied upon in the industry. This however, does not mean that these techniques and others do not come without their drawbacks. Thus, we turn our attention to the tau-leaping scheme. At its core, the tau-leaping strategy is an improvement on the SSA. The SSA is built upon the premise that we carry out each reaction consecutively, whereas the tau-leaping mechanism allows us to jump several reactions ahead under certain conditions. The tau-leaping method addresses two shortcomings of the SSA in particular,

1) the SSA is extremely slow for a large number of reactant species and/or large molecular amounts of certain species [13]; 2) the tau-leaping method, especially the implicit variation, is far better suited to approximate stiff problems. Referencing what was stated a few sentences before, no method comes without its faults, and the tau-leaping methods are no different. Within the realm of this thesis we strive to identify the inefficiencies associated with existing tau-leaping methods, on our way to establishing a modified tau-leaping algorithm capable of being an accurate and efficient alternative for a wider range of biochemical systems.

Finally, as with any other research undertaking we are motivated by a desire to advance the field of computational biology and build on the work done by scientists before us (McAdams & Arkin [21]; McAdams & Arkin [22]; Arkin et al. [23]; Elowitz et al. [6]; Fedoroff & Fontana [7]). In the next few paragraphs, we will briefly explore the work of other computational biologists and the influence that their work has had on this particular research area as well as the field in general.

Our first motivational piece comes from Harley H. McAdams and Adam Arkin and their work concerning modeling of genetic activity. In 1997, they published a paper titled "Stochastic mechanisms in gene expression" where they proposed that the pattern of protein concentration, essential in controlling the promoter, which in turn is responsible for gene expression, can be modeled using stochastic processes at varying time intervals [21]. In 1998, McAdams and Arkin with the help of John Ross explored the effect of fluctuations (noise) in rates of gene expression and con-

sidered molecular level stochastic modeling as a noise-modeling mechanism [22]. McAdams and Arkin followed this up with a paper in 1999, "It's a noisy business". They again explored the effects of noise on reaction rates and subsequently gene expression, but this time on a nanomolar level [23].

In 2002, Michael B. Elowitz along with his team expanded on the research done by McAdams and Arkin in a paper titled "Stochastic gene expression in a single cell". The endeavour focused on noise that arises from stochasticity. The team analysed both intrinsic (stochasticity inherent in the biochemical process of gene expression) and extrinsic (fluctuations in other cellular components) noise [6]. Their results established a quantitative base for modeling noise in gene expression and revealed how low intracellular copy numbers of molecules can fundamentally limit the precision of gene regulation. A similar study was conducted by Fedoroff and Fontana titled "Small Numbers of Big Molecules" in 2002 [7].

1.2 Introduction

In the previous section we shed light upon the influence of biomedicine and biochemical systems on healthcare, the economy and laid out our motivations for choosing stochastic modeling of biochemical kinetics as a research topic. In this section we will present the key concepts at high level, in order to facilitate a big picture understanding of the subject for the reader.

Consider a biochemical system, which contains a system of N different types of

molecules, known as chemical species, which are involved in M types of chemical reactions. We can think of this as a simple mathematical equation where different numbers through various mathematical operations give some result, for instance $x + y = z + u$. If we have several equations, then we have a system of equations, which we can then solve using a variety of methods available at our disposal. Similarly, we have chemical equations, where reactants (left side of the equation) go through a chemical reaction to generate some products (right side of the equation). For instance, chemical x reacts with chemical y to produce chemicals z and u (this is expressed as $x+y \rightarrow z+u$). If we have several reactions, then we have a chemical system. Our goal is to model the evolution of the population of each of the chemical species. The most accurate approach to modeling the effects of such systems on molecular populations is to track the position and velocity of each individual molecule while allowing it to evolve under the appropriate laws of physics [14]. Subsequently, we monitor the reactions as they take place and make the necessary observations. This approach is known as the *"molecular dynamics"* approach. Naturally, we can deduce that even with a moderate species population this approach becomes far too time-consuming when simulated on larger time intervals, relevant in applications. However, we can often simplify the problem by ignoring the spatial information and tracking the population size of each type of molecule as a function of time. We can achieve this by making three critical assumptions. The first being, the *well-stirred* assumption, which states that molecules are uniformly spread throughout the spatial domain [14]. In order to disregard the spatial aspect, we must also assume thermal equilibrium and constant volume throughout the reac-

tion system [16]. With these assumptions, the behaviour of the biochemical system may be described using the Chemical Master equation (CME)[10].

As was aforementioned rather than taking a brute force approach known as the molecular dynamics approach, we are interested in calculating the number of molecules for each molecular species at a given time; this is known as the system state. The Chemical Master equation (CME) is a large system of ordinary differential equations (ODE′s), with one equation for every state [10]. The CME, as we stated before, is a valid model under three key assumptions: the system is well-stirred, it is in thermal equilibrium and the volume remains constant throughout the reaction. The issue with the CME is that once the system or the population size becomes large, the system of ODE′s cannot be solved analytically and are computationally very challenging to simulate directly.

This problem remained unsolved, until the breakthrough work of Dr. Daniel Gillespie. Dr. Gillespie is a renowned American physicist, with a Ph.D. from Johns Hopkins University. His research took him to some of the leading technical universities in the United States, such as the California Institute of Technology (Caltech) and the University of California Berkeley. In 1976, his research culminated in the establishment of the stochastic simulation algorithm [9]. The SSA computes only a single realization of the state vector rather than the entire probability distribution [16]. This method allowed scientists the opportunity to simulate chemical reactions stochastically. However, the SSA is not without faults of its own. While an exact

solution of the CME, it comes at the price of high computational cost, particularly when dealing with large species populations. It is this high cost, which leads us to the tau-leaping methods.

Theoretically, one way that we can accelerate the simulation, in contrast to the SSA, is by having more than one reaction fire during a time-step [14]. Mathematically, we can achieve this by making the time-step (τ) larger. This strategy is known as the tau-leaping method. It was also developed by Dr. Gillespie as an answer to the computational issues associated with the SSA. To avoid compromising the accuracy of the method, there is a restriction that we must placed upon the length of the step τ. Known as the leap condition, this restriction states that one must ensure τ is small enough such that the propensity functions will not change significantly. Naturally, there are other safeguards within the algorithm that guarantee the accuracy of the method, such as error percentages. In this thesis, we will reference different types of tau-leaping algorithms. In particular, the explicit [14], implicit [27] and adaptive tau-leaping methods,[2, 3]. Each of these strategies is especially effective under specific conditions and, as was stated previously, our main objective is to develop a modified algorithm. The modified algorithm will incorporate elements of the former and correct inefficiencies associated with each technique. Given that tau-leaping will serve as the keystone element of this research undertaking, this scheme will be discussed in much greater depth in the subsequent chapters.

Tau-leaping has been referred to in the past as a *"bridge process"* [14]. This is

indeed correct. Tau-leaping serves as the bridge between the CME and the Chemical Langevin equation (CLE) [13] a stochastic continuous model. In the previous section we have established that over the new time-step $[t, t+\tau]$ we have more than one firing of a reaction. Additionally, we also make the assumption that the mean of the Poisson random variables in the tau-leaping method is large. Once again we will delve deeper into this in consequent chapters; for now, however, probability theory dictates that a Poisson random variable can be accurately approximated by a normal random variable with the same mean and variance. This assumption serves as the backbone for the CLE. While our previous processes were discrete and stochastic in nature, the normal random variable approximation has turned the system state into a continuous and stochastic process.

While tau-leaping is the bridge to the CLE, it ultimately acts as the bridge between the CME and the reaction rate equations (RRE). We have seen that under certain assumptions, through probabilistic approximation, a discrete and stochastic process was reduced to a stochastic and continuous one. Now, if we were to disregard the stochastic term of the CLE we reduce the model to the RRE, a continuous and deterministic model. However, it is imperative to note that this reduction is valid under a key assumption, the thermodynamic limit [14]. In Chapter 3 we will describe the thermodynamic limit mathematically and how it facilitates the reduction to the reaction rate equations.

1.3 Review of Stochastic Simulation Techniques

This final section of the introductory Chapter will serve as a review or a quick tutorial on the different types of stochastic simulation techniques at our disposal. Some will be familiar, for instance the stochastic simulation algorithm (SSA), while some may be new to the reader. We will break this section down into two categories, exact methods and approximate methods.

1.3.1 Exact Methods

Before we begin, it is important to emphasize that when it comes to simulations there is no such thing as an "*exact*" algorithm. The first exact method that we mention is Gillespie's stochastic simulation algorithm (SSA). We briefly discussed this method in the previous section and will discuss it at much greater length in Chapter 3. For now, all we will add, is that this method epitomizes accuracy and if we were given unlimited computational capability this would be the method of choice.

Next we move to a method that can be thought of as the predecessor to the SSA, the first reaction method, also developed by Gillespie [14]. The algorithm computes the time τ_i at which a reaction could be occurring, barring any reaction firing. Subsequently, the index j of the first reaction is directly correlated to the index of the reaction with the shortest time to reaction. Finally, we examine the exact method developed by Michael A. Gibson and Jehoshua Bruck, called the next reaction method [8]. The next reaction method is a modification of the aforementioned

first reaction method. This method has the computational time proportional to the logarithm of the number of reactions $\log(M)$. This is accomplished by constructing a dependency graph from the set of reactions and incorporating an appropriate data structure capable of amassing the propensities a_i and possible times τ_i. This method is unique for two particular reasons, i) it uses only a single random number per simulation event, and ii) its computational time is proportional to the logarithm of the number of reactions rather than the number of reactions itself.

1.3.2 Approximate Methods

Methods of this category are designed to approximate the exact solution as well as possible, while being far more efficient than their exact counterparts. The first type of approximate schemes that we mention are the various tau-leaping methods that are at the heart of this work. Other methods are presented for the CME, CLE and RRE. We briefly delved into these topics earlier, and we dissect these methods much more meticulously later on.

In this thesis we propose a modified adaptive explicit-implicit tau-leaping strategy to simulate a large class of well-stirred biochemical systems. Our work improves the state-of-the-art adaptive tau-leaping strategy [3], by eliminating the need for symbolic computation the Jacobian required by Newton's method for the implicit tau-leaping technique. For this we use the finite-difference approximations. Our results show that the new method maintains a similar accuracy and computational cost, with minimal intervention from the user.

1.4 Outline

This thesis will be constructed in the following way. The subsequent chapter will entail a comprehensive review of the mathematical background, which will form the basis for the rest of the thesis. The mathematical review will contain a discussion on probability theory, Monte Carlo simulation techniques and stochastic processes. In Chapter 3, we will delve into the background of biochemical systems including in-depth analysis and rigorous derivations of the CME, SSA, tau-leaping, CLE and the RRE. Chapter 4, will form the core of the of this thesis, with the introduction of several tau-leaping strategies notably featuring the state-of-the art adaptive explicit-implicit tau-leaping method followed by the new user-friendly modified adaptive explicit-implicit tau-leaping scheme. In the following Chapter, we present the numerical results which will justify the theoretical component of this thesis. The final chapter, will entail rationalization the preceding work and contemplation of topics for future study.

Chapter 2

Mathematical Background

In this Chapter we cover the necessary mathematical theory for studying stochastic modelling and simulation of biochemical systems. To begin this section we review the necessary probability distributions, upon which our theories are based. Consequently, we consider the Monte Carlo method, which is essential to the numerical simulation of the stochastic models of biochemical kinetics. Finally, we discuss several properties of stochastic processes that are key to stochatic modelling of biochemical processes.

2.1 Probability Models

In this Section we explore three probability distributions, which will prove to be critical in the derivations and proofs of future concepts. In particular, we consider exponential, normal and Poisson distributions. Nonetheless, we begin with several crucial definitions, including, probability density and mass functions, cumulative distribution function and expectation (we refer the reader to [30] for more details).

Definition 2.1.1. ***Probability mass function [30].*** *For any discrete random variable X, we define the probability mass function (PMF) to be the function which gives the probability of each $x \in S_X$ (where S_X is the set of possible observed values for X). We denote this as,*

$$P(X = x) = \sum_{\{s \in S | X(s) = x\}} P(\{s\}),$$

where S represents any arbitrary sample set. This function has a continuous counterpart, known as the probability density function (PDF), which serves the same purpose, only in continuous space.

Definition 2.1.2. ***Probability density function [30].*** *If X is a continuous random variable, then there exists a function $f_X(x)$, called the probability density function, which satisfies the following conditions,*

1. *$f_X(x) \geq 0, \forall\ x$ for any real number;*

2. *$\int_{-\infty}^{\infty} f_X(x)dx = 1$;*

3. *$P(a \leq X \leq b) = \int_a^b f_X(x)dx$ for any $a \leq b$.*

A related concept, is the cumulative distribution function (CDF). The CDF applies to both discrete and continuous distributions.

Definition 2.1.3. ***Cumulative distribution function [30].*** *The cumulative*

distribution function for a discrete random variable is signified by,

$$F_X(x) = P(X \leq x) = \sum_{\{y \in S | y \leq x\}} P(X = y).$$

The continuous analogue has the following structure,

$$\begin{aligned} F_X(x) &= P(X \leq x) \\ &= P(-\infty \leq X \leq x) \\ &= \int_{-\infty}^{x} f_X(z)dz. \end{aligned}$$

The *expectation* is the average of the random variable we are interested in, it should not be confused with the sample mean. Again, we combine the discrete and continuous interpretations of expectation into one definition.

Definition 2.1.4. ***Expectation [30].*** *The expectation of a discrete random variable X, designated by E(X) is denoted by,*

$$E(X) = \sum_{\{x \in S_X\}} xP(X = x).$$

The expectation of the continuous analogue is as follows,

$$E(X) = \int_{-\infty}^{\infty} x f_X(x)dx.$$

Now, we discuss some important probability distributions. We begin with the exponential distribution. In general a random variable, X, that is exponentially

distributed is represented as [30],

$$X \sim Exp(\lambda),$$

where λ is the rate parameter. The exponential distribution has the probability density function,

$$f_X(x) = \begin{cases} \lambda e^{-\lambda x}, x \geq 0; \\ \\ 0, \text{ otherwise.} \end{cases}$$

The exponential has the cumulative distribution function,

$$F_X(x) = \begin{cases} 0, & x < 0, \\ \\ 1 - e^{-\lambda x}, & x \geq 0, \end{cases}$$

and expected value,

$$E(X) = \frac{1}{\lambda}.$$

The exponential distribution will prove to be essential in the derivation of the stochastic simulation algorithm.

The Poisson distribution is equally important, for, it will serve as the backbone for our tau-leaping methods. This discrete probability distribution, has a parameter λ, and a Poisson random variable is expressed as,

$$X \sim P_o(\lambda).$$

Since it is discrete, it has an associated PMF given by,

$$P(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}, \quad \text{for } k = 0, 1, 2, 3, \ldots.$$

The expectation value is λ, that is,

$$E(X) = \lambda.$$

The final distribution to be discussed is the normal (or Gaussian) distribution. This probability distribution will serve as the bridge from tau-leaping to the Chemical Langevin Equation. A normal random variable X with mean μ and variance σ^2, is denoted by,

$$X \sim N\Big(\mu, \sigma^2\Big).$$

The associated PDF is written as,

$$f_X(x) = \frac{1}{\sigma}\sqrt{2\pi}\exp\left\{ -\frac{1}{2}\Big(\frac{x-\mu}{\sigma}\Big)^2 \right\}$$

Through integration of

$$\int_{-\infty}^{\infty} x f_X(x) dx,$$

we obtain the expected value of the normal distribution X,

$$E(X) = \mu.$$

As it was already mentioned, the normal distribution is essential in making the

transition from tau-leaping to the CLE. This is achieved through the following proposition.

Proposition 2.1.1. ***Normal approximation of Poisson distribution [30].*** *The Poisson distribution can be approximated by the normal distribution in circumstances where the mean of the Poisson distribution is large; in general the mean has to be greater than a threshold, more precisely,*

$$X \sim P_o(\lambda) \simeq N(\lambda, \lambda), \quad for\ \lambda > 20.$$

In the context of this thesis, a formal proof is unnecessary, nonetheless we will present the reasoning behind this property. The Poisson distribution is derived from the binomial distribution. Using the Central Limit Theorem it can be proven that the Binomial Theorem is well approximated by the normal distribution if the number of successes (n) is large. Given, that the Poisson distribution is derived from the binomial distribution, it also possesses this property. The only discrepancy is the fact that the Poisson is approximated with the parameter λ as opposed to the number of successes.

With this we conclude our review of probabilistic methods and distributions. The next section will delve into the Monte Carlo method and its application to the simulation of stochastic models of biochemical systems.

2.2 Monte Carlo Method

The Monte Carlo scheme is a broad class of computational algorithms that use random distribution over a large number of iterations, taking the average of said iterations and providing the desired results. An early variant of the method was first used in Buffons needle experiment, and subsequently again in the 1930′s by Enrico Fermi when studying neutron diffusion [1]; however, its use was not acknowledged. Physicists Stanislaw Ulam and John von Neumann working at the Los Alamos Scientific Laboratory in Los Alamos, New Mexico on a project regarding radiation shielding, first coined the term Monte Carlo, which was a code name for their work [25].

As was mentioned in the previous paragraph, the Monte Carlo strategy was first applied to physics. Nevertheless, over time it has been increasingly applied to fields other than physics such as computational biology and financial analytics. In regards to systems biology, the Monte Carlo method is vital to the implementation, among others, of the stochastic simulation algorithm and various tau-leaping methods.

Since the Monte Carlo technique proves to be a useful computational tool, let us present it briefly below. Monte Carlo simulation is a strategy by which we simulate many different realizations of the stochastic process of interest. We accomplish this through random number generation from the appropriate probability distribution and subsequent application within the parameters of the problem we wish to solve. Our desired outcome is a probability distribution of the results.

The Monte Carlo method is a crucial step in the implementation of both the SSA and subsequent tau-leaping methods. In each case we wish to construct a distribution of the number of species remaining following their evolution through time in a biochemical system. Each method will adhere to a similar structure. In Sections 3.2 and 3.3 we will provide a detailed description of the algorithms, but for now we focus strictly on the Monte Carlo component. Our initial step is to define all the variables and set the parameter values for the simulation. In this case, the parameters will include the number of species and reaction channels, the initial conditions, the stoichiometric matrix and the reaction rate constants. The variables are the molecular amounts of the biochemical species, depending on the time t. After initialization, the algorithm simulates, the various trajectories species populations can take. In biochemical systems the accepted practice is to simulate ten thousand trajectories. While this number may not seem huge, especially, when by comparison, Monte Carlo simulations in financial mathematics require upwards of a million trajectories, it will more than suffice. We have to be mindful of the fact that we are dealing with tiny molecules, where for instance a 5% error is insignificant, while in financial terms a 5% error on a billion dollar deal could potentially have a devastating impact. Using the ten thousand trajectories simulated, we can construct a probability distribution based on the results.

In conclusion, the Monte Carlo method is essentially a practical manifestation of the Law of Large Numbers fused with relatively simple statistics tools. Nevertheless, it

is an indispensable part of our computational capabilities.

2.3 Introduction to Stochastic Processes

This section gives a brief introduction to stochastic processes, which are used in stochastic modeling approaches of biochemical systems. Particular attention is placed upon comprehension, derivation and application of Markov processes and Kolmogorov's equations. The interested reader is referred to [30] for more details.

2.3.1 Markov Process Introduction

Introductory sections dedicated to probability theory and computational aspects of stochastic modeling, allows us to delve deeper into stochastic theory before transitioning to biochemical systems theory. A stochastic process is a random variable, in this case, the state change vector $\mathbf{X}(t)$,

$$\mathbf{X}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_N(t) \end{bmatrix}$$

which evolves in time [16]. The random variable, or hereafter, the system state vector, can either be continuous or discrete.

Definition 2.3.1. ***Markov Process [30].*** *A Markov process is a stochastic process that possesses the property that future states do not depend on the past states,*

given the present state.

In other words, Markov processes can be thought of as history-less. It is known that Markov processes model the behaviour of biochemical kinetics remarkably well and will serve as the underlying theoretical foundation to our research.

Armed with a general understanding of what stochastic and Markov processes are, the next task is to construct a mathematical framework based on what was stated earlier. Assume that the set

$$\{\theta^{(n)} | n = 0, 1, 2, 3, \dots\}$$

is a discrete time stochastic process. It is worth repeating that the state space S, is such that $\theta^{(n)} \in S$, for all n can be continuous or discrete. A first order Markov chain, is a stochastic process where future states are only dependent on the present state,

$$P(\theta^{(n+1)} \in A | \theta^{(n)} = x, \theta^{(n-1)} = x_{n-1}, \dots, \theta^{(0)} = x_0) = P(\theta^{(n+1)} \in A | \theta^{(n)} = x), \quad (2.1)$$

where $A \subseteq S$. The first order Markov chain depends on A, x and n. However, if the process is independent of n, then,

$$P(\theta^{(n+1)} \in A | \theta^{(n)} = x) = P(x, A). \quad (2.2)$$

In this case, the Markov Chain is said to be (time) *homogeneous*, and the *transition kernel* $P(x, A)$ determines the behaviour of the chain [30].

2.3.2 Markov Process Notation

If S is discrete, the following notation is used,

$$P(x, y) = P(\theta^{(n+1)} = y | \theta^{(n)} = x). \tag{2.3}$$

Furthermore, assuming the presence of a discrete and finite state space, $S = \{x_1, \ldots, x_m\}$, probability P can be rewritten in matrix form

$$P = \begin{bmatrix} P(x_1, x_1) & P(x_1, x_2) & \ldots & P(x_1, x_m) \\ P(x_1, x_1) & P(x_2, x_2) & \ldots & P(x_2, x_m) \\ \vdots & \vdots & \ddots & \vdots \\ P(x_m, x_1) & P(x_m, x_2) & \ldots & P(x_m, x_m) \end{bmatrix}.$$

Matrix P is known as a **stochastic matrix**.

Definition 2.3.2. *Stochastic Matrix [30]. An $m \times m$ matrix P is a stochastic matrix if its entries are non-negative and the sum of all of its elements on each row equal to 1.*

Proposition 2.3.1. *1) If P_1, P_2 are $m \times m$ stochastic matrices then, the product of P_1 and P_2 is also a stochastic matrix.*

2) For all eigenvalues (λ) of a stochastic matrix P satisfy $|\lambda| \leq 1$.

3) For a stochastic matrix P, there exists at least one eigenvalue $\lambda = 1$.

Proof. Proof of Proposition 2.3.1.

Suppose we take an eigenvalue, λ, of a stochastic matrix P, there exists a vector $x \neq 0$ such that $Px = \lambda x$. Let us denote by $||\cdot||_\infty$ the matrix ∞-norm, and by $||\cdot||_\infty^V$ the ∞-norm for m-dimensional column vectors. It then follows that

$$||Px||_\infty^V = ||\lambda x||_\infty^V \Rightarrow |\lambda| \cdot ||x||_\infty^V = ||Px||_\infty^V. \tag{2.4}$$

$$||P||_\infty = \max_{x \neq 0} \frac{||Px||_\infty^V}{||x||_\infty^V} \geq \frac{||Px||_\infty^V}{||x||_\infty^V}$$

$$||Px||_\infty^V \leq ||P||_\infty \cdot ||x||_\infty^V \quad \text{for any} \quad x \neq (0, 0, \ldots, 0)^T \tag{2.5}$$

Substituting (2.4) into (2.5), we get.

$$|\lambda| \cdot ||x||_\infty^V = ||Px||_\infty^V \leq ||P||_\infty \cdot ||x||_\infty^V \quad \text{where} \quad ||x||_\infty^V \neq (0, 0, \ldots, 0)^T$$

$$\begin{aligned} |\lambda| \leq ||P||_\infty = &\max(P_{11} + P_{12} + \cdots + P_{1m}, \\ &P_{21} + P_{22} + \cdots + P_{2m}, \\ &\cdots, \\ &P_{m1} + P_{m2} + \cdots + P_{mm}) \\ = &\max(1, 1, \ldots, 1) \\ = &1 \end{aligned}$$

Thus,

$$|\lambda| \leq 1.$$

□

Now we set up the basis for the $Chapman - Kolmogorov$ equations. Let us define, for time t_n [30],

$$P(\theta^{(n)} = x_1) = \pi^{(n)}(x_1)$$

$$P(\theta^{(n)} = x_2) = \pi^{(n)}(x_2)$$

$$\vdots$$

$$P(\theta^{(n)} = x_m) = \pi^{(n)}(x_m)$$

$$\pi^{(n)} = (\pi^{(n)}(x_1), \pi^{(n)}(x_2), \ldots, \pi^{(n)}(x_m)) \text{ at time } t_n.$$

Then,

$$\begin{aligned} P(\theta^{(n+1)} = x_1) =& P(\theta^{(n)} = x_1)P(x_1, x_1) + \\ &+ P(\theta^{(n)} = x_2)P(x_2, x_1) + \\ &+ \cdots + \\ &+ P(\theta^{(n)} = x_m)P(x_m, x_1) \end{aligned}$$

$$= (\pi^{(n)}(x_1), \pi^{(n)}(x_2), \ldots, \pi^{(n)}(x_m)) \cdot \begin{bmatrix} P(x_1, x_1) \\ P(x_2, x_1) \\ \vdots \\ P(x_m, x_1)) \end{bmatrix}$$

$$\Rightarrow \pi^{(n+1)}(x_1) = P(\theta^{(n+1)}) = \pi^{(n)} \cdot \begin{bmatrix} P(x_1, x_1) \\ P(x_2, x_1) \\ \vdots \\ P(x_m, x_1)) \end{bmatrix}$$

$$\pi^{(n+1)}(x_2) = \pi^{(n)} \cdot \begin{bmatrix} P(x_1, x_2) \\ P(x_2, x_2) \\ \vdots \\ P(x_m, x_2)) \end{bmatrix}$$

$$\vdots$$

$$\pi^{(n+1)}(x_m) = \pi^{(n)} \cdot \begin{bmatrix} P(x_1, x_m) \\ P(x_2, x_m) \\ \vdots \\ P(x_m, x_m) \end{bmatrix}$$

$$\pi^{(n+1)} = \pi^{(n+1)}(x_1), \pi^{(n+1)}(x_2), \ldots, \pi^{(n+1)}(x_m) =$$

$$= \pi^{(n)} \cdot \begin{bmatrix} P(x_1, x_1) & P(x_1, x_2) & \ldots & P(x_1, x_m) \\ P(x_1, x_1) & P(x_2, x_2) & \ldots & P(x_2, x_m) \\ \vdots & \vdots & \ddots & \vdots \\ P(x_m, x_1) & P(x_m, x_2) & \ldots & P(x_m, x_m) \end{bmatrix}$$

$$\Rightarrow \pi^{(n+1)} = \pi^{(n)} \cdot P$$

Therefore, we obtain

$$\pi^{(n+1)} = \pi^{(n)} \cdot P = \pi^{(n-1)} \cdot P \cdot P = \pi^{(n-1)} \cdot P^2 = \pi^{(n-2)} \cdot P^3 = \cdots = \pi^{(0)} \cdot P^{(n+1)}$$

$$\pi^{(n)} = \pi^{(0)} P^{(n+1)} \tag{2.6}$$

Why does this result bear significance? Equation (2.6) above, states that the initial state and the stochastic matrix P determine future probability distributions. Equipped with this knowledge, it can be deduced that if one step is dependent on P, then two steps are dependent on P^2 and the n^{th}-step is determined by P^n, accordingly. Furthermore, suppose we have two different step sizes, for instance n and p, then $P^n \cdot P^p = P^{(n+p)}$ [30]. And this statement is of the utmost importance to us because it forms the basis for the $Chapman - Kolmogorov$ equations, which are vital in deriving the stochastic discrete model of well-stirred biochemical kinetics, namely the Chemical Master equation.

2.3.3 Markov Processes: Continuous Time, Finite State-Space

This thesis deals with stochastic processes that are continuous in time and have finite state-spaces. Therefore, it is necessary to get acquainted with this concept. More details on these processes may be found in [30].

Definition 2.3.3. *A stochastic process $X(t)$ is a Markov process continuous in time*

if,

$$P(X(t+dt) = x|X(s) = x(s)|s \in [0,t])$$

$$= P(X(t+dt) = x|X(t) = x(t)), \quad \forall t \in [0,\infty), x \in S \quad (2.7)$$

where S is the state space, $S = \{1, 2, \ldots, m\}$.

Identically to the discrete cases covered earlier, future behaviour of the process does not depend on the past states, if the current state is known. Consider a process which is characterized by one of the m states denoted earlier, if at time t it is in the state $x \in S$, then future behaviour will be contingent upon the transition kernel,

$$p(x, t, x', t') \equiv P(X(t+t') = x'|X(t) = x),$$

the notation $P(X(t+t') = x'|X(t) = x)$ means the conditional probability $X(t+t') = x'$ given that $X(t) = x$. If the transition kernel is independent of t, then it is considered to be homogeneous, and can be written as $p(x, x', t')$. For each t' the transition is denoted by P(t'), a $m \times m$ matrix. There are a few properties which can be attributed to the transition kernel. First of all,

$$P(0) = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{bmatrix} = I$$

where I is an $m \times m$ identity matrix. This is intuitive as no transition can take

place in the absence of time. Similarly to what was stated in the previous section, we can carry out regular multiplication operations with P because it is a transition matrix for each value of t. The latter sentence gives the following identity,

$$P(t+t') = P(t) \cdot P(t') = P(t') \cdot P(t).$$

Let us denote by, Q, the *transition rate matrix* or just the rate matrix. The rate matrix is defined to be [30]

$$Q := \frac{d}{dt} P(t') \Big|_{t'=0}. \tag{2.8}$$

Thus,

$$\begin{aligned} Q &= \lim_{\delta t \to 0} \frac{P(\delta t) - P(0)}{\delta t} \\ &= \lim_{\delta t \to 0} \frac{P(\delta t) - I}{\delta t} \end{aligned}$$

and therefore,

$$P(dt) = I + Qdt. \tag{2.9}$$

It should be stated that $P(dt)$ is a stochastic matrix (a matrix where its entries are non-negative and the sum of all of its elements on each row equal to 1). From this we can make a few deductions. First, given that the identity matrix I consists of zeros, aside from the diagonal, leads us to the realization that off-diagonal elements of Q are also non-negative. Secondly, since diagonal elements of $P(dt)$ are bounded above by 1, then Q's diagonal elements must be non-positive. Lastly, knowing that all rows of $P(dt)$ and I sum to 1, logic dictates that the rows of Q must sum to 0.

These properties must be satisfied by a rate matrix Q. Indeed,

$$\begin{aligned}
1 &= sum\ of\ elements\ in\ one\ row\ of\ P \\
&= sum\ of\ all\ elements\ in\ same\ row\ of\ I \\
&+ (sum\ of\ all\ elements\ in\ same\ row\ of\ Q)dt \\
&= 1 + (sum\ of\ all\ elements\ in\ a\ row\ of\ Q)dt \\
&\Longrightarrow (sum\ of\ all\ elements\ in\ a\ row\ of\ Q) = 0.
\end{aligned}$$

Using equation (2.9) we can calculate the stationary distribution of the Markov chain. If π is the stationary distribution of $P(dt)$ it the follows that [30],

$$\pi P(dt) = \pi.$$

Since,

$$\pi P(dt) = \pi + \pi Q dt = \pi$$

$$\Longrightarrow \pi Q dt = 0,$$

where we know $dt \neq 0$

$$\Longrightarrow \pi Q = 0.$$

It should be noted here that π is a vector.

We can write,

$$\begin{aligned}\frac{d}{dt}P(t) &= \frac{P(t+dt)-P(t)}{dt} \\ &= \lim_{dt\to 0}\frac{P(t+dt)-P(t)}{dt} \\ &= \frac{P(dt)\cdot P(t)-P(t)}{dt} \\ &= \frac{(P(dt)-I)\cdot P(t)}{dt} \\ &\underset{(2.9)}{=} \frac{Qdt\cdot P(t)}{dt}.\end{aligned}$$

From the above we obtain,

$$\frac{d}{dt}P(t) = Q\cdot P(t). \tag{2.10}$$

Then,

$$\begin{aligned}\frac{d}{dt}P(t) &= \frac{P(t)\cdot P(dt)-P(t)}{dt} \\ &= \frac{P(t)[P(dt)-I]}{dt} \\ &\underset{(2.9)}{=} \frac{P(t)\cdot Qdt}{dt}.\end{aligned}$$

Thus, we derived

$$\frac{d}{dt}P(t) = P(t)\cdot Q. \tag{2.11}$$

Equation (2.10) can be written out using the components i and j, which leads to equations (2.12) [30],

$$\boxed{\frac{d}{dt}p(i,j,t) = \sum_{k=1}^{m} q_{ik}\cdot p(k,j,t) \quad \text{for} \quad i,j = 1,2,\ldots,m.} \tag{2.12}$$

Equations (2.12) are known as *Kolmogorov's backward equations*. Although the set does look different then its predecessor, equation (2.10), upon a closer look it is easy to identify that $[p(i,j,t)]_{(i,j)} = P(t)$, $(q)_{ik} = Q$ and $[p(k,j,t)]_{(k,j)} = P(t)$. Carrying out a similar rearrangement of equation (2.11) we arrive at the following equation [30],

$$\boxed{\frac{d}{dt}p(i,j,t) = \sum_{k=1}^{m} p(i,k,t) \cdot q_{ik} \quad \text{for} \quad i,j = 1,2,\ldots,m.} \tag{2.13}$$

The equation (2.13) is the set of *Kolmogorov's forward equations*. Kolmogorov's forward equation can now be used to derive the Chemical Master equation in the next chapter.

Chapter 3

Biochemical Systems Background

The previous chapter provided the mathematical framework essential to studying stochastic models of well-stirred biochemical systems upon which this thesis is based. This Chapter commences with a thorough examination of the discrete stochastic model of biochemical kinetics, the Chemical Master equation (CME) including definitions of all assumptions made, foundational theory and the derivation. The ensuing section will explore the motivation and underlying concepts behind the stochastic simulation algorithm (SSA). This will be followed by a detailed analysis of the tau-leaping method, which will be central in our use of the adaptive tau-leaping method and the modification of the latter. Next, we will demonstrate that the tau-leaping method can be reduced to the Chemical Langevin equation (CLE) and subsequently to the reaction rate equations (RRE) under certain assumptions. The culmination of this Chapter will be an outline of potential future work and practical applications of this research.

3.1 Chemical Master Equation

Definition 3.1.1. ***Chemical Master Equation [14].*** *The CME is the system of equations that determines the probability of the system state to be in each possible state of the well-stirred biochemical network, at the current time, provided that the initial state is known.*

The most refined model of biochemical systems is that of molecular dynamics. That is, the position and velocity of each molecule are obtained at each time t. However, this molecular dynamics approach bears enormous computational costs and is highly impractical. The foundations upon which the CME is built is probabilistic. Under certain simplifying assumptions, rather than keeping track of the positions and velocities for every single molecule, the objective is to find the molecular population number of each species depending on time. Consider a system where there are N different types of molecules, or *chemical species*, denoted by $\{S_1, \ldots, S_N\}$. These molecules are involved in M types of chemical reactions denoted by $\{R_1, \ldots, R_M\}$. Implementation is contingent upon ignoring positions and velocities of individual molecules, however, this simplification can only occur if the system is assumed to be *"well-stirred"*.

Definition 3.1.2. ***Well-stirred.*** *A well-stirred system is one where molecules of each type are uniformly spread throughout the spatial domain.*

The "$well - stirred$" assumption is fundamental in deriving the CME because, most molecular collisions are non-reactive (elastic) [14]. Two consequences arise. First, molecules, as stated in the definition, are spread uniformly throughout the

spatial domain; secondly, velocities of molecules become thermally randomized to the Maxwell-Boltzman distribution. Ergo, non-reactive collisions are negated and the focus shifts to completed reactions, which significantly reduces computational time. In addition to the well-stirred assumption, two more assumptions have to be made. One, the system has to be in thermal equilibrium and two, the volume of the spatial domain is constant.

At this point the introduction of a biochemical system would be beneficial to the reader, as it can be used to demonstrate concepts currently being discussed. Consider the following biochemical system [20], known as the decay-dimerization model. We let $X_i(t)$ represent the number of molecules of species S_i at some time t.

Definition 3.1.3. ***State-change vector.*** *The change in the vector of the species' molecular populations induced by a single occurrence of a particular reaction is known as the state-change vector of that reaction.*

The system state vector at time t is denoted by,

$$\mathbf{X}(t) = \begin{bmatrix} X_1(t) \\ X_2(t) \\ \vdots \\ X_N(t) \end{bmatrix}.$$

At time $t = 0$ the initial state vector is given, $\mathbf{X}(t_0) = x_0$. Change in the state vector synonymously, the population of the species, is the result of a chemical reaction. Thus, if the system is in state $\mathbf{X}(t)$ at time t and one reaction R_j happens, then

the system state becomes $\mathbf{X}(t) + \nu_j$, where ν_j is the state change vector of reaction R_j. For example, for the decay-dimerization model [20] we have,

$$S_1 \xrightarrow{C_1} 0 \tag{3.1}$$

$$S_1 + S_1 \xrightarrow{C_2} S_2 \tag{3.2}$$

$$S_2 \xrightarrow{C_3} S_1 + S_1 \tag{3.3}$$

$$S_2 \xrightarrow{C_4} S_3. \tag{3.4}$$

Prior to advancing, we must make note of a few things. First of all, only the molecules that are reactants, that is molecules that appear on the left side of the reactions will be considered. For instance, in decay-dimerization it is apparent that S_3 does not react, as such, S_3 can be disregarded. Next, it should be said that all state-change vectors for each reaction channel will combine to form the "stoichiometric matrix". Drawing from our model, the state change vectors associated with the first reaction channel being,

$$\nu_1 = \begin{bmatrix} -1 \\ \\ 0 \end{bmatrix}.$$

For complete clarity, let us consider Reaction (3.1). In Reaction (3.1) one molecule of species S_1 is lost and nothing is gained in return, thus ν_1 is written as above.

Similarly, in Reaction (3.2) one molecule of species S_2 is exhausted and two molecules

of species S_2 are formed. The union of all the associated state change vectors will form the stoichiometric matrix,

$$\nu = \begin{bmatrix} -1 & -2 & 2 & 0 \\ 0 & 1 & -1 & -1 \end{bmatrix}.$$

Let us regress. The CME as has been noted at various times, is a stochastic model. It would not be egregious to purport that a reaction can only occur if certain molecules are to collide. Using probability theory, we know that $P(A \cap B)$ for two independent variables is P(A) $\cdot$ P(B). Similar logic applies here. Therefore, the probability that the next reaction takes place in the interval $[t, t+dt)$, where dt is an infinitesimal time step, is proportional to some combination of S_i's. The constant of proportionality is called a reaction rate parameter. Naturally arises a question, why is a constant necessary? As was discussed earlier, not all collisions lead to reactions, therefore this constant is designed to account for unreactive collisions. This very general equation reduces to three cases, first and second order, and dimerization. This probability is also known as, and from henceforth will be referred to as, the *propensity function.*

Definition 3.1.4. ***Propensity Function.*** *The function $a_j(x)$ whose product with dt gives the probability that a reaction R_j will occur in $[t, t+dt)$ for an infinitesimal time dt, given that $\mathbf{X}(t) = x$.*

First and second order propensities are fairly intuitive, however the dimerization propensity leaves room for questions. The answer is also in fact straightforward; it

represents the number of ways we can choose an unordered pair of objects from a total of X_m molecules using combinatorics.

The expression of these propensities are justified by kinetic theory principles [16].

First order. $S_m \xrightarrow{C_j}$ products of reaction, $\Longrightarrow a_j(X(t)) = c_j X_m(t)$.

Second order. $S_m + S_n \xrightarrow{C_j}$ products of reaction, where $m \neq n, \Longrightarrow a_j(X(t)) = c_j X_m(t) X_n(t)$.

Dimerization. $S_m + S_m \xrightarrow{C_j}$ products of reaction,$\Longrightarrow a_j(X(t)) = \frac{1}{2} c_j X_m(t)$ $(X_m(t) - 1)$.

Utilizing the results from the preceding section we can now derive the Chemical Master equation from the Kolmogorov forward equations [30].

Theorem 3.1.1. *Kolmogorov forward equations for a biochemical system may be written as ,*

$$\frac{d}{dt} p(x_0, t_0, x, t) = \sum_{i=1}^{M} \Big[a_i(x - \nu_i, C_i) p(x_0, t_0, x - \nu_i, t) - a_i(x, C_i) p(x_0, t_0, x, t) \Big], \quad (3.5)$$

for any $t_0 \in \mathbb{R}, x_0, x \in S,$ *where* S *is the state − space.*

Equations (3.5) are known as the Chemical Master equation, a discrete stochastic model of well-stirred biochemical kinetics.

Proof. Using Kolmogorov's forward equations (2.13), we obtain,

$$\frac{d}{dt}p(x_0,t_0,x,t) = \sum_{x'\in S} q_{x',x}p(x_0,t_0,x',t) \tag{3.6}$$

$$= \left[\sum_{x'\in S,x'\neq x} q_{x',x}p(x_0,t_0,x',t)\right] + q_{x,x}p(x_0,t_0,x,t) \tag{3.7}$$

Since

$$P = I + Qdt,$$

we derive the following property of the entries of the matrix Q:

$$q_{x,x} + \sum_{x'\in S,x'\neq x} q_{x,x'} = 0$$

and thus

$$q_{x,x} = -\sum_{x'\in S,x'\neq x} q_{x,x'} \tag{3.8}$$

Substituting (3.8) into (3.7), we get

$$\frac{d}{dt}p(x_0,t_0,x,t) = \sum_{x'\in S,x'\neq x} q_{x',x}p(x_0,t_0,x',t) - \sum_{x'\in S,x'\neq x} q_{x,x'}p(x_0,t_0,x,t)$$

$$= \sum_{x'\in S,x'\neq x}\left[q_{x',x}p(x_0,t_0,x',t) - q_{x,x'}p(x_0,t_0,x,t)\right].$$

Given that $x' \neq x$ and $q_{x',x} \neq 0$ it then follows that it is only possible that $x' = x-\nu_j$. This in turn means that $q_{x-\nu_j,x} \neq 0$. Following a similar train of thought, we can state that given $x' \neq x$ and $q_{x,x'} \neq 0$ then $x' = x + \nu_j$, and thus $q_{x+\nu_j,x} \neq 0$.

Using what we just stated, it follows,

$$\frac{d}{dt}p(x_0, t_0, x, t) = \sum_{j=1}^{M} \Big[q_{x-\nu_j,x} p(x_0, t_0, x - \nu_j, t) - q_{x,x+\nu_j} p(x_0, t_0, x, t) \Big]. \tag{3.9}$$

Also, we note from the definition of a propensity function that,

$$q_{x-\nu_j,x} = a_j\big(x - \nu_j, C_j\big)$$

$$q_{x,x+\nu_j} = a_j(x, C_j)$$

Then, after substituting in (3.9), we derive:

$$\frac{d}{dt}p(x_0, t_0, x, t) = \sum_{j=1}^{M} \Big[a_j(x - \nu_j, C_j) p(x_0, t_0, x - \nu_j, t) - a_j(x, C_j) p(x_0, t_0, x, t) \Big] \tag{3.10}$$

□

Thus we have arrived at the Chemical Master equation. Parts of this section will be revisited in the rest of the chapter, in particular when we derive the Reaction Rate Equation (RRE). In what follows we use the notation $P(x, t|x_0, t_0)$ to represent the probability that $\mathbf{X}(t) = x$ if $\mathbf{X}(t_0) = x_0$. With this notation the CME becomes,

$$\frac{d}{dt}P(x,t|x_0,t_0) = \sum_{j=1}^{M}\left(P(x-\nu_j,t|x_0,t_0)a_j(x-\nu_j) - P(x,t|x_0,t_0)a_j(x)\right). \quad (3.11)$$

The CME (3.10) is clearly faster to solve numerically than the molecular dynamics approach and is accurate. However, the solution of the CME is computationally very challenging to approximate directly, hence it is still very slow. The CME is a system of ordinary differential equations with one ordinary differential equation (ODE) for every single state thus it has a large dimension in general. Solving the CME directly came to be replaced by the SSA, which will be discussed at length in the next section.

3.2 Stochastic Simulation Algorithm

As we discussed in the introduction and the preceding section, the set of CME equations becomes impossible to solve analytically when the system is sufficiently large. The stochastic simulation algorithm (SSA) also known as Gillespie's algorithm was first introduced in 1945 by Joseph L. Doob [4]. It however was not until 1976, and Daniel Gillespie presented it, that the method became a biokinetic mainstay [9].

The SSA uses the Monte Carlo method to generate trajectories with a distribution in exact agreement with the solution of the CME. Enhanced computational capability of the SSA is a direct result of the explicit simulation mechanism. Additionally,

we have numerously mentioned that the SSA is an upgrade on the direct solution of the CME, but never examined why. In order to facilitate our explanation we refer to definition 2.1.1 in Chapter 2, which declares, *"for a discrete random variable X, the probability mass function (PMF) is the function that outputs the probability of each event $x \in S_x$ (where S_x is the sample space)"* [30]. Indeed, the SSA takes a sample from the probability mass function of the solution set of CME rather than the whole PMF.

Below, we provide the theoretical justification of the SSA. Let us first introduce the quantity $P_0(\tau|x,t)$ [16] which denotes the probability that no reaction occurs in the next time interval $[t, t+\tau]$, given the state vector $\mathbf{X}(t) = x$. Next, consider the time interval $[t, t+\tau+d\tau)$, where $d\tau$ is an infinitesimal time step, where at most one reaction can occur. The probability that no reaction occurs over $[t, t+\tau+d\tau)$ is denoted by "event C". Similarly, "event A" will signify the probability that no reaction happens during $[t, t+\tau)$ and "*event B*" as the probability that no reaction happens over $[t+\tau, t+\tau+d\tau)$. Then the probability that event C occurs is equivalent to the probability of events A **and** B taking place.

$$P(C) = P(A) \cdot P(B). \tag{3.12}$$

Since the events in the interval $[t, t+\tau)$ and those in $[t+\tau, t+\tau+d\tau)$ are independent the "**and**" can be expressed using multiplication, and then the key to advancing the derivation is, restating P(B) differently. P(B) can be thought of as *1 - the probability of each reaction occurring in the interval*, $[t+\tau, t+\tau+d\tau)$. Using this

reconfiguration, equation (3.12) leads to,

$$P(C) = P(A) \cdot (1 - P(B^{\mathsf{c}})). \tag{3.13}$$

Then using the definition of a propensity (recall that the propensity function coupled with an infinitesimal time step is equivalent to the probability of a reaction occuring in the next time step), equation (3.13), can be expressed as,

$$P(C) = P(A) \cdot \left(1 - \sum_{k=1}^{M} a_k(x) d\tau \right). \tag{3.14}$$

Then, using our initial statements and notation, this can be rewritten as,

$$P_0(\tau + d\tau | x, t) = P_0(\tau | x, t) \left(1 - \sum_{k=1}^{M} a_k(x) d\tau \right). \tag{3.15}$$

Rearranging the above we arrive at,

$$\frac{P_0(\tau + d\tau | x, t) - P_0(\tau | x, t)}{d\tau} = -a_{sum}(x) P_0(\tau | x, t),$$

where $a_{sum}(x) := \sum_{k=1}^{M} a_k(x)$. Equivalently, this can be restated as,

$$\frac{d}{d\tau} P(\tau | x, t) = -a_{sum}(x) P_0(\tau | x, t). \tag{3.16}$$

Approaching the limit as $d\tau \to 0$, leads to a linear scalar ODE, which by definition has the initial condition $P_0(0|x,t) = 1$ [16]. We remark that, $P_0(0|x,t) = 1$, signifies that the probability that no reactions take place in no time is always 1. Solving the

ODE (3.16) with this initial condition, the following result is obtained,

$$P_0(\tau|x,t) = e^{-a_{sum}(x)\tau}. \tag{3.17}$$

Now we introduce the quantity $p(\tau, j|x,t)$ which denotes the probability that the next reaction will be i) the j^{th} reaction and ii) will occur in the time interval $[t+\tau, t+\tau+d\tau)$. As before event i) is signified by D and ii) by E. Events D and E are again independent of each other by virtue of similar logic expressed a few paragraphs earlier, thus,

$$P(D \cap E) = P(D) \cdot P(E). \tag{3.18}$$

As earlier we assume $d\tau$ to be an infinitesimal time step where no more than one reaction can take place. Using established definitions of P_0 and propensity, we arrive at equation,

$$p(\tau, j|x,t)d\tau = P_0(\tau|x,t)a_j(x)d\tau. \tag{3.19}$$

Recall the earlier result in the form of equation (3.17). Substituting equation (3.17) into (3.19) and cancelling the $d\tau$ terms, we obtain the following,

$$p(\tau, j|x,t) = e^{-a_{sum}(x)\tau}a_j(x). \tag{3.20}$$

Finally, equation (3.20) can be rearranged ensuring that the a's are gathered and outside the exponential function, simplifying our algorithm,

$$p(\tau, j|x, t) = \frac{a_j(x)}{a_{sum}(x)} a_{sum}(x) e^{-a_{sum}(x)\tau}. \tag{3.21}$$

Before analysing the significance of what was just derived, let us examine in greater detail the mathematics behind this equation. This will require two propositions. We begin with the time to next reaction.

Proposition 3.2.1. ***Simulation of time to next reaction [30].*** *Recall the definitions presented in Section 2.1, it is apparent that τ is exponentially distributed with parameter $a_{sum}(x)$. Then,*

$$\tau = \frac{1}{a_{sum}(x)} \ln\left(\frac{1}{\xi_1}\right)$$

where ξ_1 is a uniformly distributed random variable ($\xi_1 \sim U(0,1)$) necessary for simulation.

Proof. We have to solve $e^{-a_{sum}(x)} = \xi_1$ with ξ_1 is uniformly distributed in [0,1]. Consequently,

$$e^{-a_{sum}}(x) = \xi_1$$

$$-a_{sum}(x) = \ln(\xi_1)$$

$$\tau = -\frac{1}{a_{sum}(x)} \ln(\xi_1)$$

$$\tau = \frac{1}{a_{sum}(x)} \ln\left(\frac{1}{\xi_1}\right) \tag{3.22}$$

□

Equation (3.22) is the time to next reaction simulated by the SSA, with $a_{sum}(x)e^{-a_{sum}(x)\tau}$ serving as the PDF.

Proof that the index of the next reaction is indeed $\frac{a_j(x)}{a_{sum}(x)}$, requires Proposition 3.2.2.

Proposition 3.2.2. ***Index of time to next reaction [30].*** *The reaction channels R_j are exponentially distributed, with the parameters $a_j(x)$, that is,*

$$\tau_j \sim Exp(a_j(x)),$$

where $j = 1, 2, \ldots, n$, are independent random variables. Then,

$$\tau_0 \equiv \min_{i=1,2,\ldots,M} \tau_j \sim Exp(a_{sum}(x)).$$

Given that the SSA is a critical component of this thesis, consequently, this is an equally important proposition; thus, we will take the time to provide the proofs.

Proof. For a exponential random variable $X \sim Exp(\lambda)$, $P(X > x) = e^{-\lambda x}$.In our

case, $X = \tau_j, \lambda = a_j(x)$

$$\begin{aligned} P(X_0 > \tau) &= P\Big(\min_j\{X_j\} > \tau\Big) \\ &= P(|X_1 > \tau| \cap |X_2 > \tau| \cap \cdots \cap |X_M > \tau|) \\ &= \prod_{j=1}^{M} P(X_j > \tau) \\ &= \prod_{j=1}^{M} e^{-a_j(x)\tau} \\ &= e^{-x\sum_{j=1}^{M} a_j(x)\tau} \\ &= e^{-a_{sum}(x)\tau}. \end{aligned}$$

□

Thus, $\tau_0 \sim Exp(a_{sum}(x))$.

Lemma 3.2.3. **[30]** *If we suppose $X \sim Exp(\lambda)$ and $Y \sim Exp(\mu)$ are independent random variables, then,*

$$P(X < Y) = \frac{\lambda}{\lambda + \mu}.$$

Proof. We can derive that,

$$\begin{aligned} P(X < Y) &= \int_0^\infty P(X < Y)|Y = y)f(y)dy \\ &= \int_0^\infty P(X < Y)f(y)dy \\ &= \int_0^\infty (1 - e^{-\lambda y})\mu e^{-\mu y}dy \\ &= \frac{\lambda}{\lambda + \mu}, \end{aligned}$$

since $f(y) = \mu e^{-\mu y}$. □

Proposition 3.2.4. **[30]** *If $\tau_j \sim Exp(a_j(x)), i = 1, 2, \ldots, M$ are independent random variables, let k be the index of the smallest of the τ_j. Then k is a discrete random variable with the PMF,*

$$\pi_j = \frac{a_j(x)}{a_{sum}(x)},\ j = 1, 2, \ldots, M.$$

Proof. Let us consider,

$$\begin{aligned}\tau_j &= P(\tau_k < \min_{j \neq K}\{\tau_j\}) \\ &= P(\tau_k < Y).\end{aligned}$$

Then, according to Lemma 3.2.3,

$$\begin{aligned}\tau_j &= \frac{a_k(x)}{a_k(x) + a_{-k}(x)} \\ &= \frac{a_k(x)}{a_{sum}(x)}.\end{aligned}$$

□

Note: $Y = \min_{j \neq k}\{\tau_j\}, such\ that,\ Y \sim Exp(a_{-k}(x)), where\ a_{-k}(x) = \sum_{j \neq k}$ $a_j(x)$. Why is this result meaningful? Primarily, it is because our two variables are gathered in one equation (3.21), a joint density function. This in turn, maintains independence between the two variables, while optimizing the computational cost. Recall that, j represents the next reaction index and τ defines the time to next reaction. Each of the independent variables can be simulated using a uniform sample

on the (0,1) interval, with time to next reaction being simulated using the exponential distribution. With all of that being said, we present Gillespie's algorithm also known as the SSA,

1. Initialize the simulation $\mathbf{X}(t_0) = x_0$ at $t = 0$ and set the parameters M, N, c_j's and ν.
2. Evaluate $\{a_k(\boldsymbol{X}(t))\}_{k=1}^{M}$ and $a_{sum}((\boldsymbol{X}(t))) := \sum_{j=1}^{M} a_k(\boldsymbol{X}(t))$.
3. Select two independent uniform (0,1) random numbers ξ_1 and ξ_2.
4. Evaluate j, the smallest integer satisfying $\sum_{k=1}^{j} a_k(\boldsymbol{X}(t)) > \xi_1 a_{sum}((\boldsymbol{X}(t)))$.
5. Compute $\tau = \ln(1/\xi_2)/a_{sum}((\boldsymbol{X}(t)))$.
6. Update $\boldsymbol{X}(t+\tau) = \boldsymbol{X}(t) + \nu_j$ and t to $t+\tau$.
7. Go to step 1.

Nearing the end of the topic, conclusions can be drawn. The SSA is a vast improvement over solving the CME directly, in terms of computational cost. Also, the SSA is an exact Monte Carlo method for the CME, generating a possible sequence of reaction events. In the following section, our attention is turned to tau-leaping, a topic at the heart of this thesis. An obvious shortcoming of the SSA is that the algorithm advances through time one reaction at a time, therefore, it can be very slow when some very fast reactions happen in the system. In contrast, tau-leaping provides a platform where the system can fire several reactions during one time-step.

3.3 Tau-Leaping

Gillespie [13] proposed a strategy to accelerate the SSA in which each time step τ advances the system through possibly many reaction events.

A common theme within this thesis will be the inefficiencies of methods associated with high computational costs. The SSA, while an exact method, is nonetheless computationally expensive when some reactions are fast. We mentioned in the previous Section that the SSA progresses one reaction at a time. This in turn implies that at each iteration random number generation has to be utilized, the state vector updated and so on. The idea behind tau-leaping is allowing many reactions of each type to fire over one time step and then to update the state vector. However, the key to the tau-leaping method is maintaining accuracy comparable to the SSA, while improving execution speed. This is accomplished by mandating the leap condition, which states that the propensity cannot change its value significantly as a result of the larger time step.

Definition 3.3.1. *Leap Condition [14]. A time step τ satisfies the leap condition if τ is sufficiently small such that the propensity $a_j(x(s))$does not undergo any observable change for any $1 \leq j \leq M$ and any $t \leq s \leq t + \tau$.*

Mathematically speaking, this can be written as,

$$a_j(x(t+\tau)) \simeq a_j(x(t)), \quad \text{for any } 1 \leq j \leq M.$$

The leap condition requires propensities $a_j(x(t))$ to remain almost constant during

the step, while the number of reactions that will fire is calculated using a counting process. The probability of the j-th reaction firing over the time step τ is $a_j(x(t))\tau$, by the multiplication rule for independent variables. Subsequently, we need to determine how many of these events occur over $[t, t+\tau)$. This can be well approximated using a Poisson distribution, with mean and variance, $a_j(x(t))\tau$ in $P_j(a_j(x), \tau)$. Putting everything together garners the general tau-leaping equation [13],

$$\mathbf{X}(t+\tau) = x + \sum_{j=1}^{M} \nu_j P_j(a_j(x), \tau). \tag{3.23}$$

where the random variables $\{P_j(a_j(x), \tau)\}_{j=1}^{M}$ are independent Poisson random variables and $\mathbf{X}(t) = x$.

An exact representation of the stochastic process $\mathbf{X}(t)$ was given by Kurtz [18]. If $\mathbf{X}(t) = x$, then

$$\mathbf{X}(t+\tau) = \mathbf{X}(t) + \sum_{j=1}^{M} \nu_j P_j\left(\int_t^{t+\tau} a_j(\mathbf{X}(s))ds \right). \tag{3.24}$$

Using the leap condition we can make the following assumption,

$$a_j(\mathbf{X}(s)) \simeq a_j(\mathbf{X}(t)), \quad \text{for all } t \leq s \leq t+\tau.$$

Then,

$$\int_t^{t+\tau} a_j(\mathbf{X}(s))ds \simeq \int_t^{t+\tau} a_j(\mathbf{X}(t))dt = a_j(\mathbf{X}(t))\tau \tag{3.25}$$

Using (3.24) and (3.25) we get,

$$\mathbf{X}(t+\tau) \simeq x + \sum_{j=1}^{M} \nu_j P_j(a_j(x), \tau).$$

Thus the tau-leaping method (3.23) is an approximate Monte Carlo strategy for solving the CME. In the introduction tau-leaping was characterized as the "*bridge equation*" to the Chemical Langevin Equation (CLE) from the CME. Let us investigate this claim further. We begin by making the assumption that τ is small enough to satisfy the leap condition, but also large enough to ensure that the number of firings for each reaction channel R_j is much larger than 1 (i.e. $a_j(x(t))\tau \gg 1$, for $1 \leq j \leq M$). Now we invoke Proposition 2.1.1 from Chapter 2, which states that a Poisson random variable with a large mean and variance, can be well approximated by a normal random variable with the same mean and variance [30].

3.4 Chemical Langevin Equation

Section 3.3 featured comprehensive coverage of the tau-leaping method; tau-leaping is considered to be a bridge to the Chemical Langevin equation (CLE). Recall from the previous section that we made the assumption that τ is chosen such that (i) the leap condition is satisfied and (ii) the average number of firings for each reaction channel R_j is $a_j(x(t)) \cdot \tau \gg 1$ for any $1 \leq j \leq M$.

If $P_j(a_j(x), \tau)$ in equation (3.23) is replaced by $a_j(X(t))\tau + \sqrt{a_j(Y(t))}Z_j$, where Z_j

are independent normal variables with mean 0 and variance 1, then we get,

$$\mathbf{X}(t+\tau) = \mathbf{X}(t) + \tau \sum_{j=1}^{M} \nu_j a_j(\mathbf{X}(t)) + \sqrt{\tau} \sum_{j=1}^{M} \nu_j \sqrt{a_j(\mathbf{X}(t))} Z_j. \tag{3.26}$$

The algorithm for simulating the above is,

1. Select independent samples $\{Z_j\}_{j=1}^{M}$ from the normal distribution with mean 0 and variance 1.

2. Substitute samples from first step into equation (3.26), to obtain $\mathbf{X}(t+\tau)$ and update time t to $t+\tau$.

3. Return to step 1.

This algorithm is to be repeated for as many simulations as needed (the standard number of Monte Carlo trajectories used for stochastic simulation of biochemical systems is 10,000). Also noteworthy is the fact that equation (3.26) is the Euler-Maruyama solution to equation (3.27).

$$d\mathbf{X}(t) = \sum_{j=1}^{M} \nu_j a_j \mathbf{X}(t) dt + \sum_{j=1}^{M} \nu_j \sqrt{a_j \mathbf{X}(t)} dW_j(t), \tag{3.27}$$

where, $W_j(t)$ are independent scalar Brownian motions. Equation (3.27), is, in fact a system of stochastic differential equations which is called the *Chemical Langevin equation.*

In the paragraph above we briefly mentioned a key concept, Brownian motion, let us explore it further. Brownian motion is a physical phenomenon pioneered by Robert Brown [26] and later developed by Albert Einstein and Jean Perrin, for which Perrin would eventually be awarded a Nobel Prize in Physics in 1926 [19]. From the physics perspective, Brownian motion is the random motion of a particle surrounded by fluid. The motion is the result of continuous and random pounding of the particle by the surrounding atoms. Einstein was able to derive an equation for the average displacement of the particle, however this is not necessary in the context of this thesis. Eventually Brownian motion was adopted into the world of biochemical simulation, since the simulations tend to move in random trajectories, mimicking the movement of the particle in fluid described above.

The conclusory paragraph is a good time to make two remarks. First of all, the CLE is dependent on two assumptions, one, the time step has to be small enough not to cause a significant variation in the propensities, yet large enough to satisfy the approximation of the Poisson distribution by the normal distribution. Secondly, the CLE is a "*bridge process*" itself, as we shall see in the next section.

3.5 Reaction Rate Equation

The reaction rate equations constitutes a model of well-stirred biochemical systems. We have traced simulation of biochemical systems from the molecular dynamics approach, to the CME, then via the tau-leaping method we arrived at the CLE and

now it is time to dissect the RRE. As was aforementioned, the RRE is simply the deterministic part of the CLE [16]. We said that this simplification can be achieved through the thermodynamic limit. Well in that case, the question begs itself, what is the thermodynamic limit?

Definition 3.5.1. *Thermodynamic Limit [14]. The thermodynamic limit is defined as the limit in which the species populations X_i, and the system volume Ω all approach infinity, but in such a way that the species concentrations X_i/Ω stay constant.*

As this limit approaches infinity, the propensities grow proportionally to the size of the system. This occurs for both types of propensities, unimolecular and bimolecular. The latter is a result of the inversely proportional relationship between the reaction constants and the system volume. Therefore, as the propensities grow, so do both sides of equation (3.27). However, the term on the right $\left(\sum_{j=1}^{M} \nu_j a_j \mathbf{Y}(t)dt\right)$ will grow much faster than the square root term on the left $\left(\sum_{j=1}^{M} \nu_j \sqrt{a_j \mathbf{Y}(t)} dW_j(t)\right)$. Naturally, as the limit approaches infinity, the term on the left becomes negligible, thus reducing (3.27) to the reaction rate equations (RRE).

Similarly to what we presented in Section 3.1, the rate constants, c_j, can be catogorized into the three identical scenarios we described for the CME; for first and second order reactions and dimerization.

Propensity functions for the first and second order reactions and the dimeriza-

tion are:

First order. $S_m \xrightarrow{C_j}$ products of reaction, then $a_j(X(t)) = c_j X_m(t)$.

Second order. $S_m + S_n \xrightarrow{C_j}$ products of reaction, where $m \neq n$, then $a_j(X(t)) = c_j X_m(t) X_n(t)$.

Dimerization. $S_m + S_m \xrightarrow{C_j}$ products of reaction, then $a_j(X(t)) = \frac{1}{2} c_j X_m(t)^2$.

Therefore, to achieve the transformation from the CLE to the RRE, we induce the necessary assumptions mentioned in the previous paragraph followed by application of Definition 3.5.1.

All that remains is derivation of the reaction rate equations. We start by recalling the general model of the Chemical Master equation,

$$\frac{d}{dt} P(x, t|x_0, t_0) = \sum_{j=1}^{M} \left(P(x - \nu_j, t|x_0, t_0) a_j(x - \nu_j) - P(x, t|x_0, t_0) a_j(x) \right).$$

Proof. In order to derive the RRE the expectation of both sides of the CME has to

be taken [30].

$$\begin{aligned}
\frac{\partial}{\partial t}E(X_t) &= \frac{\partial}{\partial t}\sum_{x\in S} x \cdot p(x,t) \\
&= \sum_{x\in S} x \cdot \frac{\partial}{\partial t}p(x,t) \\
&= \sum_{x\in S} x \cdot \left[\sum_{j=1}^{M} a_j(x-\nu_j, c_j)\cdot p(x-\nu_j,t) - a_j(x,c_j)p(x,t)\right] \\
&= \sum_{j=1}^{M}\sum_{x\in S}\left[x \cdot a_j(x-\nu_j,c_j)p(x-\nu_j,t) - x\cdot a_j(x,c_j)p(x,t)\right]
\end{aligned}$$

where $x = (x-\nu_j)+\nu_j$

$$\begin{aligned}
&= \sum_{j=1}^{M}\left[\sum_{x\in S}(x-\nu_j)a_j(x-\nu_j,c_j)p(x-\nu_j,t)\right. \\
&\left.+\sum_{x\in S}\nu_j a_j(x-\nu_j,c_j)p(x-\nu_j,c_j) - \sum_{x\in S} x\cdot a_j(x,c_j)p(x,t)\right]
\end{aligned}$$

taking $y = (x - \nu_j)$

$$= \sum_{j=1}^{M} \Bigg[\sum_{y \in S} y \cdot a_j(y, c_j) p(y, t)$$
$$+ \sum_{y \in S} \nu_j a_j(y, c_j) p(y, c_j) - \sum_{x \in S} x \cdot a_j(x, c_j) p(x, t) \Bigg]$$

the *y and x terms will cancel*

$$= \sum_{j=1}^{M} \nu_j \Bigg[\sum_{y \in S} a_j(y, c_j) p(y, c_j) \Bigg].$$

Consequently,

$$\frac{\partial}{\partial t} E(X_t) = \sum_{j=1}^{M} \nu_j \sum_{y \in S} a_j(y, c_j) p(y, t)$$

where

$$\sum_{y \in S} a_j(y, c_j) p(y, t) = E(a_j(c_j)).$$

We obtained [30],

$$\frac{\partial}{\partial t} E(X_t) = \sum_{j=1}^{M} \nu_j E(a_j(c_i)) \tag{3.28}$$

The above equation is derived when expectation is taken in the CME. However, the reaction rate equations are

$$\frac{\partial}{\partial t} E(X_t) = \sum_{j=1}^{M} a_j(E(x_t), c_j)) \tag{3.29}$$

or, if we denote $y(t) = E(X_t)$, then, □

$$\frac{d\mathbf{y}(t)}{dt} = \sum_{j=1}^{M} \nu_j a_j(\mathbf{y}(t)).$$

At this point, it is critical to mention that equation (3.28), as it is written, may be different than equation (3.29). The two equations coincide for systems with at most order one reactions. For second order reactions they may differ, nevertheless empirical results attest that the RRE maintains its purpose. The reason behind this discrepancy is that, in general,

$$E(c_j X_i X_k) = c_j E(X_i X_k) \neq c_j E(X_i) E(X_k).$$

as X_i and X_k may not be independent.

As was briefly mentioned in the introduction, the reaction rate equations was the conventional model of biochemical systems until stochastic model of the CME proved to be more accurate. An additional drawback to the RRE, as demonstrated earlier, is the fact that there are inconsistencies between the theoretical base and empirical results. Nevertheless, for simulating systems with very large numbers of chemical species, the RRE remains the gold standard to this day.

3.6 Potential Applications

In the introduction we briefly touched on the overall impact of the medical and biomedical industry on society and what role biochemical systems have to play within that. The future of systems biology will continue to be intertwined with

medicine and biomedical engineering [17]. In this section we will delve into the specific projects for which systems biology is utilized as well as other industries that can benefit from this research.

The medical field that has perhaps benefited most from the advancement of computational biology is cancer research [17]. Cancer is the biggest medical challenge of our time. As the global population continues to live longer, cancer rates are rising and according to the Canadian Cancer Statistics 2017 report by the Canadian Cancer Society, it is said that half of the Canadian population will develop it during their lifetime. Biochemical systems strive to predict the future stages of the disease, as well as the response to medication in hopes of a cure. Just in recent history we have seen that patients are living longer with their cancers and some forms of it which we considered untreatable are now being if not treated at the very least managed. Other promising applications for systems biology include treatment or possible cure for inflammatory diseases, diabetes and disorders of the nervous system [24].

With that we conclude this chapter. We started by outlining the basic assumptions and constructs, followed by an exhaustive examination that took us from the Chemical Master equation (a stochastic model, discrete in time and space) all the way to the reaction rate equations (a deterministic model continuous in time and space). Along the way we provided a step-by-step dissemination of each of the methods we used, in tandem with rigorous proofs. In the next chapter, we focus on

the development of effective tau-leaping strategies.

Chapter 4

Algorithms and Models

This Chapter will serve as the apex of this research endeavour and is based upon the following framework. Building on the content from Section 3.3, the explicit and implicit tau-leaping techniques will be discussed at length. Subsequently, we will introduce and analyze adaptive tau-leaping methods, using the explicit and implicit tau-leaping schemes. In this chapter we propose an innovative adaptive explicit-implicit tau-leaping method which generalizes the state of the art variable tau-selection strategy of Cao et al. [3]. We use a pseudo-Newton's scheme to approximate the solution of the tau-leaping method, by employing finite-difference strategies to approximate the Jacobian. This eliminates the need for user's intervention. The final act will serve as a review of methods and tools we use in the analysis of the results, followed by the results themselves.

4.1 Explicit Tau-Leaping

Recall that, if the leap condition is satisfied on $[t, t+\tau)$, then the explicit tau-leaping method is

$$\mathbf{X}(t+\tau) = x + \sum_{j=1}^{M} \nu_j P_j(a_j(x)\tau), \tag{4.1}$$

given that $\mathbf{X}(t) = x$.

The implicit tau-leaping scheme operates on a similar principle. It is known that the explicit method is well-suited to handle non-stiff problems. In the next section, we devote a paragraph to stiffness, which plays a central role in selecting the appropriate tau-leaping simulation strategy: explicit for non-stiff systems or implicit for stiff ones. For now, let us return to explicit tau-leaping. The explicit tau-leaping strategy requires that the step-size τ is chosen such that the leap condition is obeyed. Applying this condition leads to a sequence of non-uniform step-sizes on each trajectory. Such a method is said to incorporate adaptivity. A constant step-size implementation of the tau-leaping scheme is not justified theoretically and it may lead to inaccurate results. The next two paragraphs will contain the algorithm for each respective τ selection process.

We begin with the "*vanilla*", or the explicit method sans adaptivity. The first step, as with all other algorithms, is to choose the initial conditions $\mathbf{X}(t) = x_0$ at $t = 0$. For this *vanilla method*, the leap-size τ is fixed, thus it is chosen at this step. Secondly, within a time loop ranging from time $t = 0$ to the preconditioned

final time, the propensities are calculated. Subsequently, using the Poisson random number generator with the parameter $a_j(x(t)) \cdot \tau$ (still within the loop), the solution is advanced according to (4.1) and t is updated by $t + \tau$. Thus, one trajectory of the biochemical system is computed. Finally, return to step one and continue the process until an appropriate number of trajectories is available.

Our attention will now shift to the adaptive explicit tau-leaping method. As the reader might have guessed, the only difference between the algorithm above and the current method is the selection strategy of τ. First of all, automatic selection of τ is now performed over each step of the time-loop. In this method, the choice of τ is contingent upon the type of reaction that will fire next. In this context, there are two types of reactions, *critical* and *noncritical* [2]. Critical reactions are those, for which the population of reactant falls below a certain threshold; non-critical reactions are those that do not satisfy this condition. Throughout the simulation, the algorithm categorizes reactions in this manner at each time-step. Based on whether or not a reaction is critical, the algorithm has to make a decision, either proceed with the explicit tau-leaping method with an associated τ for non-critical reactions or the SSA with also an accompanying τ for critical reactions. Adaptive switching between tau-leaping and the SSA is the result of the SSA being better equipped to simulate smaller population sizes; this is evident given that the SSA progresses one reaction at a time. Finally, once the algorithm has classified the reactions and chosen the appropriate τ, steps three and four are identical to the prior paragraph.

Everything discussed previously can be algorithmized into a multi-step procedure and formally integrated into the following algorithm due to Cao et al [2].

1. Initialize $t = 0$, $X^{(0)} = x_0$; set the simulation parameters: tolerance ϵ, critical threshold n_c, the final time T and the reaction rate constants c_j.

2. At each time t categorize the reactions. We begin, by introducing L_j which represents the number of times a reaction can fire before one of the reactants is exhausted

$$L_j = \min_{i \in [1,N], \nu_{i,j} < 0} \left[\frac{x_i}{|\nu_{i,j}|} \right],$$

where $\nu_{i,j}$ represents the state-change vector. A reaction R_j is deemed to be critical if $L_j < n_c$; for our purposes the generally accepted value of $n_c = 10$ was used .

3. Armed with the knowledge of which reactions adhere to which class, we introduce J_{cr} and J_{ncr}, respectively denoting the set of critical and non-critical reaction indices. If all the reactions happen to be critical, then set $\tau = \infty$ and proceed to step 5.

4. If non-critical reactions are present, then the following multi-step is used to calculate the explicit candidate for τ.

I) First the *highest order of reaction* (or HOR(i)) for a chemical species S_i is determined. As was discussed earlier, reactions found only on the left side of the

equation will be considered. The order of a reaction is equivalent to the number of times a reactant is seen in each reaction; the highest such number is the HOR(i).

II) In this step the value of ϵ_i is computed, where,

$$\epsilon_i = \frac{\epsilon}{g_i}. \tag{4.2}$$

ϵ is the tolerance and $g_i = g_i(x_i)$ is calculated in the following way,

i) If HOR(i)=1,

$$g_i = 1. \tag{4.3}$$

ii) If HOR(i)=2, g_i=2, unless two S_i molecules are used in one reaction, in which case,

$$g_i = \left(2 + \frac{1}{x_i - 1}\right) \tag{4.4}$$

iii) If If HOR(i)=3, g_i=3, unless two S_i molecules are used in one reaction, in which case,

$$g_i = \frac{3}{2}\left(2 + \frac{1}{x_i - 1}\right). \tag{4.5}$$

If a third-order reaction requires three S_i molecules, then,

$$g_i = \left(3 + \frac{1}{x_i - 1} + \frac{2}{x_i - 2}\right). \tag{4.6}$$

III) Calculation of the explicit candidate for the next time step requires the following two quantities,

$$\alpha_i(x) := \sum_{j \in J_{ncr}} \nu_{ij} a_j(x), \tag{4.7}$$

$$\beta_i(x) := \sum_{j \in J_{ncr}} \nu_{ij}^2 a_j(x). \tag{4.8}$$

IV) The explicit candidate for the next time step is given by τ',

$$\tau' = \min\left\{\frac{\max\{\epsilon x_i/g_i, 1\}}{|\alpha_i(x)|}, \frac{\max\{\epsilon x_i/g_i, 1\}^2}{[\beta_i(x)]^2}\right\}. \tag{4.9}$$

5) Compute the sum $a_0^c(x)$ of all the propensities of critical reactions. Using this generate a second candidate τ'', with equation (4.10)

$$\tau'' = \left(\frac{1}{a_0^c(x)}\right) \cdot \ln\left(\frac{1}{r_1}\right), \tag{4.10}$$

for r_1 a sample of $\mathcal{U}(0,1)$, the uniform distribution over (0,1).

6) If $\tau' < \tau''$ then $\tau = \tau'$ and we proceed with the explicit tau-leaping (4.1), else $\tau = \tau''$ followed by simulation with the SSA of the slow reactions.

7) Exit this sequence, return to step 2 and repeat until a sufficient number of trajectories has been generated.

This marks the end of adaptive explicit tau-leaping scheme, for now. As we shall soon see, it also features prominently in adaptive tau-leaping.

4.2 Implicit Tau-Leaping

The main objective of the implicit tau-leaping method is to efficiently simulate stiff systems, which are expensive to solve numerically by the explicit tau-leaping strategy.

4.2.1 Stiffness

The primary reason for different variants of tau-leaping is stiffness. Stiffness is defined as the presence of slow and fast dynamics in the system, with the fast ones being stable [27]. Often biochemical systems arising in applications involve slow and fast reactions. After a short transient, fast reactions reach a partial equilibrium. To be more precise, in order for a system to be considered stiff, there has to be at least two orders of separation between the fast and slow reaction propensities. For problems that are stiff we use implicit methods because they are better suited than their explicit counterparts. The issue with explicit methods is that the step size has to be kept small in order to ensure stability [27]. Implicit methods on the other hand have no such restriction on the step size in order to be stable.

4.2.2 Newton's Method

In many ways the discussion regarding the implicit tau-leaping method is very similar to the previous Section. For this very reason we shall focus only on the major discrepancy between the two, the implicit part. The implicit tau-leaping equation is given by [27],

$$\mathbf{X}(t+\tau) \doteq x + \sum_{j=1}^{M} [P_j(a_j(x)\tau) - \tau a_j(x) + a_j(\boldsymbol{X}(t+\tau))\tau]\nu_j, \tag{4.11}$$

if $X(t) = x$. Note that only the deterministic part is implicit, while the stochastic component is in an explicit form. Equation (4.11) may be written as

$$F(\mathbf{X}(t+\tau), x) = 0 \tag{4.12}$$

where

$$F(\mathbf{X}(t+\tau), x) = \mathbf{X}(t+\tau) - \sum_{j=1}^{M} a_j(x(t+\tau)) * \tau_j\nu_j - \left[x + \sum_{j=1}^{M} [P_j\Big(a_j(x), \tau) - \tau a_j(x)\Big]\nu_j\right]. \tag{4.13}$$

Equation (4.12) will be solved to find $\mathbf{X}(t+\tau)$, the system state at the future time, $t+\tau$. Note that (4.12) is an implicit equation in $\mathbf{X}(t+\tau)$. To solve this implicit problem numerically, we use Newton's method.

Newton's method was first developed by its namesake Isaac Newton. The method originally proposed by Newton through the years has evolved into a version that

varies from the original. An important contributor to Newton's method was Joseph Raphson, so much so that the method is often referred to as the Newton-Raphson method. Newton's method for a generic equation $F(X) = 0$ is

$$X^{(n+1)} = X^{(n)} - \left[\frac{\partial F}{\partial X}\Big(X^{(n)}\Big)\right]^{-1} \cdot F\Big(X^{(n)}\Big), \tag{4.14}$$

where the n-th iteration $X^{(n)}$ is an N-dimensional array and F is an N-dimensional function of X. What we have is a system composed of N equations and N unknowns, where

$$X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{pmatrix}$$

and

$$F(X) = \begin{pmatrix} F_1(X) \\ F_2(X) \\ \vdots \\ F_N(X) \end{pmatrix}.$$

$X^{(0)} = \mathbf{X}(t)$ serves as the initial guess for the implicit tau-leaping method (4.11) on $[t, t+\tau)$. A challenge of this strategy is the computation of the Jacobian. This portion of the method has to be derived symbolically, which may be challenging, especially in the presence of large systems. Also, it requires the user's intervention, which is a drawback of this technique. Nevertheless, the Jacobian is given by the

following matrix,

$$J = \begin{bmatrix} \frac{\partial F_1(X)}{\partial X_1} & \frac{\partial F_1(X)}{\partial X_2} & \cdots & \frac{\partial F_1(X)}{\partial X_N} \\ \frac{\partial F_2(X)}{\partial X_1} & \frac{\partial F_2(X)}{\partial X_2} & \cdots & \frac{\partial F_2(X)}{\partial X_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_N(X)}{\partial X_1} & \frac{\partial F_N(X)}{\partial X_2} & \cdots & \frac{\partial F_N(X)}{\partial X_N} \end{bmatrix} \tag{4.15}$$

The algorithm outlined in Section 4.1 will form the skeleton of the implicit algorithm. A novelty is the addition of Newton's method. This addition is expressed in the form of one extra step. This step consists of solving the implicit equation (4.12) using Newton's method. We iterate until $||X^{(n+1)} - X^{(n)}|| \leq TOL$, for some given tolerance TOL. As before the simulation continues until one trajectory is attained, at which time we return to the first step. More formally, the algorithm will adhere to the following structure (see also [27]).

1. Specify the parameters, including the stoichiometric matrix, number of species, reaction channels and simulations, tolerances TOL and ϵ, rate constants, final time T, n_c and empty arrays capable for storing critical and non-critical reaction indices.

2. Initialize the time $t = 0$ and the state $X^{(0)} = x_0$.

3. Compute the propensity functions and consequently update after each simulation.

4. For each trajectory, at each time t categorize the reactions. We begin, by

introducing L_j which represents the number of times a reaction can fire before one of the reactants is exhausted:

$$L_j = \min_{i \in [1,N], \nu_{i,j} < 0} \left[\frac{x_i}{|\nu_{i,j}|} \right].$$

A reaction R_j is deemed to be a critical reaction if $L_j < n_c$; for our purposes the generally accepted value of $n_c = 10$ was used.

5. Armed with the knowledge of which reactions adhere to which class, we introduce J_{cr} and J_{ncr}, respectively denoting the set of critical and non-critical reaction indices. If all the reactions happen to be critical, then set $\tau = \infty$ and proceed to step 7.

6. Identify reversible reactions in the system at hand. Our objective is to create a set of indices which correspond to reversible reactions not in partial equilibrium, which will be signified by J_{ne}. Partial equilibrium is defined as the condition where correspondent, $a_+(x)$ and $a_-(x)$ are close to each other. The difference between the two must be smaller than each respective propensity. Specifically, the partial equilibrium condition is,

$$|\mathbf{a}_+(\mathbf{x}) - \mathbf{a}_-(\mathbf{x})| \leq \delta \min\{\mathbf{a}_+(\mathbf{x}), \mathbf{a}_-(\mathbf{x})\},$$

where the generally accepted value of δ is 0.05. If the system is already in equilibrium then τ can be chose to be sufficiently large.

7. If reactions that are non-critical and not in partial equilibrium are present, then the following multi-step is used to calculate the implicit candidate for τ.

I) We first determine the *highest order of reaction* (or HOR(i)) for a chemical species S_i. As was discussed earlier, only the reactant species are considered. The order of a reaction is equivalent to the number of times a reactant is seen in each reaction; the highest such number is the HOR(i).

II) In this step the value of ϵ_i is computed, where,

$$\epsilon_i = \frac{\epsilon}{g_i}. \tag{4.16}$$

ϵ is the tolerance and $g_i = g_i(x_i)$ is calculated using 4.3, 4.4, 4.5 and 4.6.

III) Calculation of the implicit candidate for the next time step requires the following two quantities,

$$\alpha_i(x) := \sum_{j \in J_{necr}} \nu_{ij} a_j(x), \tag{4.17}$$

$$[\beta_i(x)]^2 := \sum_{j \in J_{necr}} \nu_{ij}^2 a_j(x), \tag{4.18}$$

where $J_{necr} = J_{ncr} \cap J_{ne}$ is the set of non-critical reactions which are not in partial equilibrium.

IV) The implicit candidate for the next time step is τ',

$$\tau' = \min\left\{\frac{\max\{\epsilon x_i/g_i, 1\}}{|\alpha_i(x)|}, \frac{\max\{\epsilon x_i/g_i, 1\}^2}{[\beta_i(x)]^2}\right\}. \tag{4.19}$$

8. Compute the sum $a_0^c(x)$, of the propensities of all critical reactions. Using this generate a second candidate τ'' according to,

$$\tau'' = \left(\frac{1}{a_0^c(x)}\right) \cdot \ln(1/r_1), \tag{4.20}$$

for r_1 a sample of $\mathcal{U}(0,1)$, the uniform distribution over (0,1). τ'' is the step to the next slow reaction.

9. If $\tau' < \tau''$ then $\tau = \tau'$ and we proceed with the implicit tau-leaping method, else $\tau = \tau''$ followed by simulation with the SSA.

10. Applying the implicit tau-leap step by solving (4.11) with respect to $X(t+\tau)$, using Newton's method (4.14) for F given by (4.13).

11. Exit this sequence, return to step 2. Repeat until a sufficient number of trajectories has been generated.

The next section will entail a detailed discussion regarding an adaptive tau-leaping

method, a strategy which fuses the explicit, implicit schemes and the SSA.

4.3 Adaptive Explicit-Implicit Tau-Leaping Method

This Section commences with what brought the preceding one to a close, fusion of the three central schemes into what is known as an explicit-implicit adaptive tau-leaping method. A discussion regarding the effectiveness of this algorithm will be given at the end of the section.

The adaptive tau-leaping algorithm is similar to the implicit method with adaptivity, with some exceptions. Let us present the adaptive implicit-explicit tau-leaping algorithm (see also [3]).

1. Specify the parameters, including the stoichiometric matrix, number of species, reaction channels and simulations, tolerances TOL and ϵ, rate constants, final time T, n_c and empty arrays capable for storing critical and non-critical reaction indices.

2. Initialize the time $t = 0$ and the state $X^{(0)} = x_0$.

3. Compute the propensity functions and consequently update after each simulation.

4. For each trajectory, at each time t categorize the reactions. We begin, by introducing L_j which represents the number of times a reaction can fire before one

of the reactants is exhausted:

$$L_j = \min_{i \in [1,N], \nu_{i,j} < 0} \left[\frac{x_i}{|\nu_{i,j}|} \right].$$

A reaction R_j is deemed to be a critical reaction if $L_j < n_c$; for our purposes the generally accepted value of $n_c = 10$ was used.

5. Armed with the knowledge of which reactions adhere to which class, we introduce J_{cr} and J_{ncr}, respectively denoting the set of critical and non-critical reaction indices. If all the reactions happen to be critical, then set $\tau = \infty$ and proceed to step 8.

6. Identify reversible reactions in the system at hand. Our objective is to create a set of indices which correspond to reversible reactions not in partial equilibrium, which will be signified by J_{ne}. Partial equilibrium is defined as the condition where correspondent, $a_+(x)$ and $a_-(x)$ are close to each other. The difference between the two must be smaller than each respective propensity. Specifically, the partial equilibrium condition is,

$$|\mathbf{a}_+(\mathbf{x}) - \mathbf{a}_-(\mathbf{x})| \leq \delta \min\{\mathbf{a}_+(\mathbf{x}), \mathbf{a}_-(\mathbf{x})\},$$

where the generally accepted value of δ is 0.05. If the system is already in equilibrium then τ can be chose to be sufficiently large.

7. If non-critical reactions are present, then the following multi-step sequence is used to calculate the explicit candidate for τ. Note in the adaptive explicit implicit method, we now have two τ candidates, $\tau^{(ex)}$ and $\tau^{(im)}$ corresponding to the explicit and implicit candidates respectively. The way we compute the quantities changes. The explicit scheme will still correspond to the indices set J_{ncr}, while the implicit method will draw from the set J_{necr}. The latter is set that we have not yet seen, and represents the set that is non-critical and not in partial equilibrium, in the set theory notation this is formulated as $J_{necr} = J_{ncr} \cap J_{ne}$.

I) We first determine the *highest order of reaction* (or HOR(i)) for a chemical species S_i. As was discussed earlier, reactions found only on the left side of the equation will be considered. The order of a reaction is equivalent to the number of times a reactant is seen in each reaction; the highest such number is the HOR(i).

II) In this step the value of ϵ_i is computed, where,

$$\epsilon_i = \frac{\epsilon}{g_i}. \tag{4.21}$$

ϵ is the tolerance and $g_i = g_i(x_i)$ is calculated using 4.3, 4.4, 4.5 and 4.6.

III) Calculating the explicit candidate for the next time step requires the following two quantities,

$$\alpha_i(x)^{(ex)} := \sum_{j \in J_{ncr}} \nu_{ij} a_j(x), \tag{4.22}$$

$$\left[\beta_i(x)^{(ex)}\right]^2 := \sum_{j\in J_{ncr}} \nu_{ij}^2 a_j(x). \tag{4.23}$$

IV) Calculating the implicit candidate for the next time step requires the next two quantities,

$$\alpha_i(x)^{(im)} := \sum_{j\in J_{necr}} \nu_{ij} a_j(x), \tag{4.24}$$

$$\left[\beta_i(x)^{(im)}\right]^2 := \sum_{j\in J_{necr}} \nu_{ij}^2 a_j(x), \tag{4.25}$$

where $J_{necr} = J_{ncr} \cap J_{ne}$ is the set of non-critical reactions which are not in partial equilibrium.

IV) The explicit candidate for the next time step is given by $\tau^{(ex)}$,

$$\tau^{(ex)} = \min\left\{\frac{\max\{\epsilon x_i/g_i, 1\}}{|\alpha_i(x)^{(ex)}|}, \frac{\max\{\epsilon x_i/g_i, 1\}^2}{[\beta_i(x)^{(ex)}]^2}\right\}. \tag{4.26}$$

And the implicit candidate by $\tau^{(im)}$,

$$\tau^{(im)} = \min\left\{\frac{\max\{\epsilon x_i/g_i, 1\}}{|\alpha_i(x)^{(im)}|}, \frac{\max\{\epsilon x_i/g_i, 1\}^2}{[\beta_i(x)^{(im)}]^2}\right\}. \tag{4.27}$$

8. Compute the sum of $a_0^c(x)$, the propensities of all critical reactions. Using

this generate a second candidate τ_2, with equation (4.37)

$$\tau_2 = \left(\frac{1}{a_0^c(x)} \right) \cdot \ln(1/r_1), \tag{4.28}$$

where r_1 is a sample from the unit-interval uniform distribution. τ_2 represents the step to the next slow reaction.

9. If $\tau^{(im)}$ is greater than $N_{stiff}\tau^{(ex)}$, where N_{stiff} usually takes on the value 100, then the system is considered to be stiff, we let $\tau_1 = \tau^{(im)}$. Otherwise $\tau_1 = \tau^{(ex)}$.

10. If $\tau_2 > \tau_1$ then $\tau = \tau_1$ and we proceed with explicit tau-leaping (4.1) if $\tau_1 = \tau^{(ex)}$ or implicit tau-leaping (4.11) if $\tau_2 = \tau^{(im)}$. Else $\tau = \tau_2$ followed by simulation with the SSA for the slow reactions.

11. Update $\mathbf{X}(t+\tau)$, set time to $t = t + \tau$ and exit this sequence. Return to step 2 and repeat until a sufficient number of trajectories has been generated.

This adaptive strategy is considered to be the state-of-the-art tau-leaping method [3]. Aside from the fact that adaptivity along with explicit and implicit schemes is far more efficient than the SSA, there are compelling reasons that underscore the superiority of this algorithm. The first such reason is harmonization of explicit, implicit methods and the SSA. Throughout this thesis we have demonstrated time and time again that each of the aforementioned schemes is well designed for specific degrees of stiffness of the system. For the regions where the problem is non-stiff, the

algorithm uses the explicit tau-leaping scheme, while in the regions of stiffness, it switches to the implicit tau-leaping strategy. In the regions where some molecular amounts are below the threshold, the SSA is the preferred strategy, as it prevents negative population numbers. Thus, the adaptive explicit-implicit method expands the computational horizons and broadens the scope of problems that can be solved using it. This elicits a smooth transition to the second reason, automatization; for without it the marriage of the three strategies would be difficult at best. However, in the case of the implicit tau-leaping method, a Newton step is employed to solve a non-linear system of equations. Symbolic computation of the Jacobian maybe expensive or it may require the user's input. This is a drawback of Newton's method for the implicit tau-leaping step. The method we proposed in the next Section will address this issue.

Despite its computational prowess, the adaptive explicit-implicit tau-leaping method is not with out faults, however, we will leave this discussion for the subsequent Section and the conclusion.

4.4 Modified Adaptive Tau-Leaping Method

In the previous section we heaped praise upon the increased automatization observed in the adaptive explicit-implicit method but also noted that room for improvement exists. Automatization is the proverbial double-edged sword. On one hand, the checks and balances are carried out mechanically, one the other, the Jacobian has to be inputted by the user. Approximating the Jacobian using the

finite-difference method limits the need for symbolic computation.

The finite-difference strategy will be used to approximate the Jacobian in Newton's method. The results we publish in Chapter 5, will affirm the accuracy of the new user-friendly modified algorithm.

From the previous paragraph it is apparent that the difference between this amalgamated method and its adaptive predecessor lies in Newton's method. As such, all of the steps outlined in the previous section apply here as well. While other finite-difference schemes may be used to estimate first order derivatives, we apply the forward finite-difference scheme, for simplicity. Thus we estimate $\frac{\partial F_k}{\partial X_i}$ by,

$$\frac{\partial F_k}{\partial X_i}(X_1, \ldots, X_n) = \frac{F(X_1, \ldots, X_i, X_i + h, X_{i+1}, \ldots, X_n) - F_k(X_1, \ldots, X_n)}{h}$$

for any $1 \leq i \leq N$ and $1 \leq h \leq N$. Here $0 < h \ll 1$. Recall that X is an N-dimension array and F is N-dimensional function of X. The Jacobian will be approximated by matrix (4.29)

$$\frac{\partial F}{\partial X}(x) \simeq \frac{1}{h}\Bigg[F(X_1 + h, X_2, \ldots, X_n) - F(X), F(X_1, X_2 + h, \ldots, X_n) - F(X), \\ \ldots, F(X_1, X_2, \ldots, X_n + h) - F(X)\Bigg] \quad (4.29)$$

with

$$F(X) = [F_1(X), F_2(X), \ldots, F_N(X)]^T.$$

In the case of the implicit tau-leaping method $F(X)$ is given by formula (4.13) in Section 4.2.2. We note that for large biochemical systems, the computation of the exact Jacobian is challenging, while the finite-difference approximation is straightforward.

With this approximation Newton's step in the implicit scheme becomes a pseudo-Newton's method, which may, theoretically, be less accurate per iteration and therefore it may require more iterations to achieve the same accuracy. However, the numerical tests performed (see Chapter 5) show that the same accuracy is obtained with very similar computational costs. The implementation of the new method is straightforward, even for large systems.

Leaning upon the foundation built in the previous three Sections we are now ready to present the algorithm for the modified explicit-implicit tau-leaping method.

1. Specify the parameters, including the stoichiometric matrix, number of species, reaction channels and simulations, tolerances TOL and ϵ, rate constants, final time T, n_c and empty arrays capable for storing critical and non-critical reaction indices.

2. Initialize the time $t = 0$ and the state $X(0) = x_0$.

3. Compute the propensity functions and consequently update after each simulation.

4. For each trajectory, at each time t categorize the reactions. We begin, by introducing L_j which represents the number of times a reaction can fire before one of the reactants is exhausted:

$$L_j = \min_{i \in [1,N], \nu_{i,j} < 0} \left[\frac{x_i}{|\nu_{i,j}|} \right].$$

A reaction R_j is deemed to be a critical reaction if $L_j < n_c$; for our purposes the generally accepted value of $n_c = 10$ was used.

5. Armed with the knowledge of which reactions adhere to which class, we introduce J_{cr} and J_{ncr}, respectively denoting the set of critical and non-critical reaction indices. If all the reactions happen to be critical, then set $\tau = \infty$ and proceed to step 7.

6. Identify reversible reactions in the system at hand. Our objective is to create a set of indices which correspond to reversible reactions not in partial equilibrium, which will be signified by J_{ne}. Partial equilibrium is defined as the condition where correspondent, $a_+(x)$ and $a_-(x)$ are close to each other. The difference between the two must be smaller than each respective propensity. Specifically, the partial equilibrium condition is,

$$|\mathbf{a}_+(\mathbf{x}) - \mathbf{a}_-(\mathbf{x})| \leq \delta \min\{\mathbf{a}_+(\mathbf{x}), \mathbf{a}_-(\mathbf{x})\},$$

where the generally accepted value of δ is 0.05. If the system is already in equilibrium then τ can be chose to be sufficiently large.

7. If non-critical reactions are present, then the following multi-step sequence is used to calculate the explicit candidate for τ. Note in the adaptive explicit implicit method, we now have two τ candidates, $\tau^{(ex)}$ and $\tau^{(im)}$ corresponding to the explicit and implicit candidates respectively. The way we compute the quantities changes. The explicit method will still correspond to the indices set J_{ncr}, while the implicit scheme will draw from the set J_{necr}. The latter represents the set of reactions which are both non-critical and not in partial equilibrium, in the set theory notation this is formulated as $J_{necr} = J_{ncr} \cap J_{ne}$.

I) We first determine the *highest order of reaction* (or HOR(i)) for a chemical species S_i. As was discussed earlier, reactions found only on the left side of the equation will be considered. The order of a reaction is equivalent to the number of times a reactant is seen in each reaction; the highest such number is the HOR(i).

II) In this step the value of ϵ_i is computed, where,

$$\epsilon_i = \frac{\epsilon}{g_i}. \tag{4.30}$$

ϵ is the tolerance and $g_i = g_i(x_i)$ is computed using 4.3, 4.4, 4.5 and 4.6.

III) Computing the explicit candidate for the next time step requires the following

two quantities,

$$\alpha_i(x)^{(ex)} := \sum_{j \in J_{ncr}} \nu_{ij} a_j(x), \tag{4.31}$$

$$\left[\beta_i(x)^{(ex)}\right]^2 := \sum_{j \in J_{ncr}} \nu_{ij}^2 a_j(x). \tag{4.32}$$

IV) Computing the implicit candidate for the next time step requires the next two quantities,

$$\alpha_i(x)^{(im)} := \sum_{j \in J_{necr}} \nu_{ij} a_j(x), \tag{4.33}$$

$$\left[\beta_i(x)^{(im)}\right]^2 := \sum_{j \in J_{necr}} \nu_{ij}^2 a_j(x), \tag{4.34}$$

where $J_{necr} = J_{ncr} \cap J_{ne}$ is the set of non-critical reactions which are not in partial equilibrium.

IV) The explicit candidate for the next time step is given by $\tau^{(ex)}$,

$$\tau^{(ex)} = \min\left\{\frac{\max\{\epsilon x_i/g_i, 1\}}{|\alpha_i(x)^{(ex)}|}, \frac{\max\{\epsilon x_i/g_i, 1\}^2}{[\beta_i(x)^{(ex)}]^2}\right\}. \tag{4.35}$$

And the implicit candidate by $\tau^{(im)}$,

$$\tau^{(im)} = \min\left\{\frac{\max\{\epsilon x_i/g_i, 1\}}{|\alpha_i(x)^{(im)}|}, \frac{\max\{\epsilon x_i/g_i, 1\}^2}{[\beta_i(x)^{(im)}]^2}\right\}. \tag{4.36}$$

8. Compute the sum of $a_0^c(x)$, the propensities of all critical reactions. Using this generate a second candidate τ_2, with equation (4.37)

$$\tau_2 = \left(\frac{1}{a_0^c(x)} \right) \cdot \ln(1/r_1), \tag{4.37}$$

where r_1 is a sample from the unit-interval uniform distribution. τ_2 represents the step to the next slow reaction.

9. If $\tau^{(im)}$ is greater than $N_{stiff}\tau^{(ex)}$, where N_{stiff} usually takes on the value 100, then the system is considered to be stiff, we let $\tau_1 = \tau^{(im)}$. Otherwise $\tau_1 = \tau^{(ex)}$.

10. If $\tau_2 > \tau_1$ then $\tau = \tau_1$ and we proceed with explicit tau-leaping (4.1) if $\tau_1 = \tau^{(ex)}$ or implicit tau-leaping (4.11) if $\tau_2 = \tau^{(im)}$. The implicit system (4.11) with F given by (4.13) is solved by the pseudo-Newton method using the approximate Jacobian (4.29). Else $\tau = \tau_2$ followed by simulation with the SSA for the slow reactions.

11. Update $X(t + \tau)$, set time to $t = t + \tau$ and exit this sequence. Return to step 2 and repeat until a sufficient number of trajectories has been generated.

Before presenting the numerical results, let us review the content of this Section. We began with the explicit tau-leaping method and introduced the concept of adaptivity. We described the automatic selection of the time-step based on quasi-steady states when stiffness is present; we also presented the implicit tau-leaping method, which is well-suited to approximate stiff systems, due to the absence of step-size

restriction. In the same subsection, we also introduced Newton's method, a numerical solution to the implicit component of equation (4.11). Using the algorithms developed in Sections 4.1 and 4.2 the state-of-the-art adaptive explicit-implicit tau-leaping method was presented. It is considered to be state-of-the-art because it combines the explicit and implicit methods with the SSA to form an algorithm designed to approximate a wide spectrum of systems.

Finally, our contribution is the modification of the adaptive explicit-implicit tau-leaping scheme is the creation of a new algorithm that is far more user-friendly in the absence of symbolic computation that is equally accurate and efficient in comparison to the original adaptive strategy. The new strategy is designed for larger systems, but not large enough where simulation with the CLE or RRE would preferable, and systems featuring complex propensity functions (for instance, propensity functions that are not polynomials).

Chapter 5

Numerical Results

Numerical results presented in this thesis will be underpinned by three models, stiff, decay-dimerization and cycle systems. We illustrate the advantages of our new adaptive explicit-implicit tau-leaping strategy for the Chemical Master equation over the state-of-the-art tau-selection scheme by Cao et al. [3] and the exact Stochastic Simulation Algorithm developed by Gillespie.

5.1 Stiff Model

The first model we consider is a stiff model [28].

$$S_1 + S_2 \xrightarrow{C_1} S_3$$

$$S_3 \xrightarrow{C_2} S_1 + S_2$$

$$S_1 + S_3 \xrightarrow{C_3} S_2$$

$$S_2 \xrightarrow{C_4} S_1 + S_3$$

$$S_3 + S_2 \xrightarrow{C_4} S_1$$

$$S_2 \xrightarrow{C_4} S_3 + S_2$$

The simulation interval is [0,0.01], the stoichiometric matrix is,

$$\nu = \begin{bmatrix} -1 & 1 & -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 \end{bmatrix},$$

with rate constants,

$$c = \begin{pmatrix} 25 \\ 10^4 \\ 10^{-3} \\ 10^{-1} \\ 10^{-2} \\ 2 \end{pmatrix},$$

and propensities,

$$a = \begin{pmatrix} c_1 x_1 x_2 \\ c_2 x_3 \\ c_3 x_1 x_3 \\ c_4 x_2 \\ c_5 x_2 x_3 \\ c_6 x_1 \end{pmatrix}.$$

For this model the initial conditions are,

$$X(0) = \begin{pmatrix} 1000 \\ 1000 \\ 10 \end{pmatrix},$$

with a tolerance (TOL) of 0.0275 and $h = 0.1$.

After simulating 10,000 trajectories of the exact SSA, the explicit-implicit variable step-size tau-leaping scheme and the modified explicit-implicit variable step-size tau-leaping strategy, we present our findings below. Figure 5.1 shows the histograms at $t = 0.01$ of species X_1 generated with the above three methods. Figures 5.2 and 5.3 present the histograms for species X_2 and X_3 respectively. The accuracy of the modified adaptive tau-lap method matches very well that of the standard adaptive tau-leaping scheme, both matching well the accuracy of the exact SSA. Moreover,

the speed-up of the modified adaptive tau-leaping scheme is defined as,

$$speed - up(\%) = \frac{CPU(SSA)}{CPU(mod\ \tau - leap)} \cdot 100.$$

For this model

$$speed - up(\%) = 1496.82$$

for the modified adaptive explicit-implicit tau-leaping scheme.

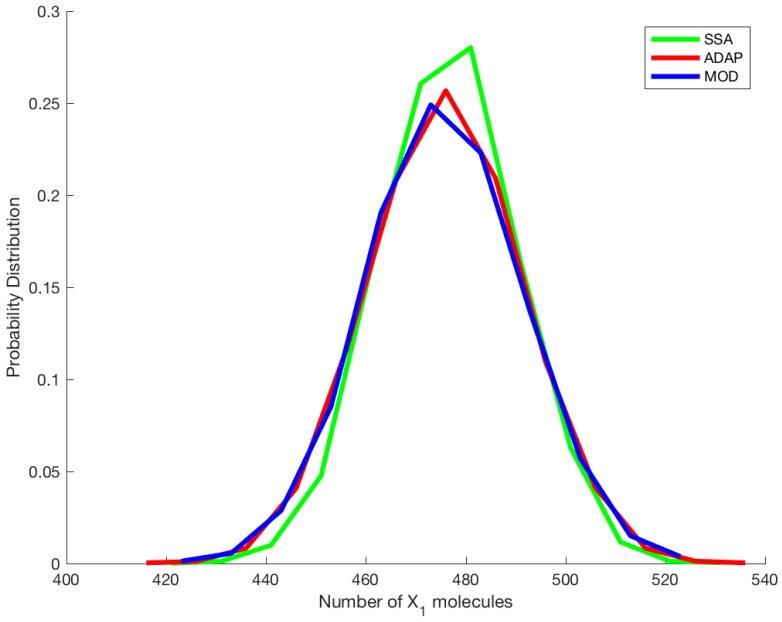

Figure 5.1: Stiff Model: Histogram of the X_1 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at t=0.01

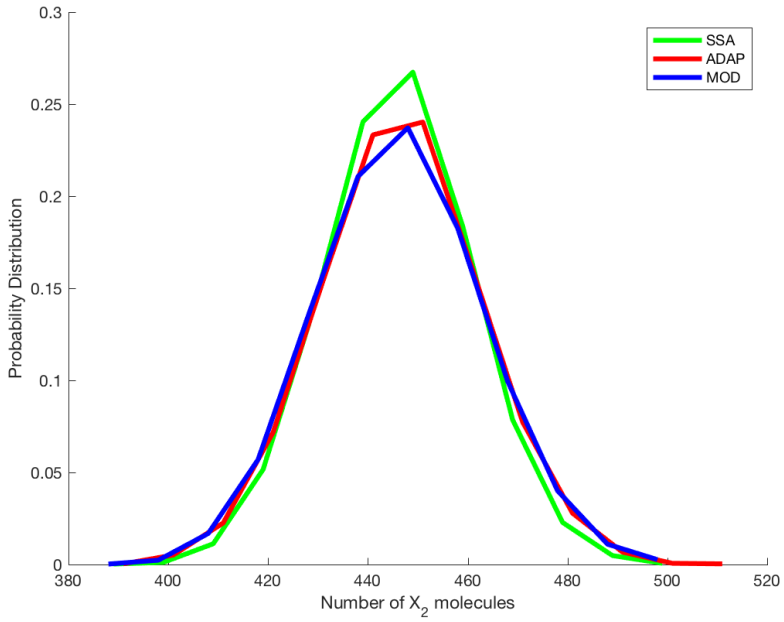

Figure 5.2: Stiff Model: Histogram of the X_2 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 0.01$

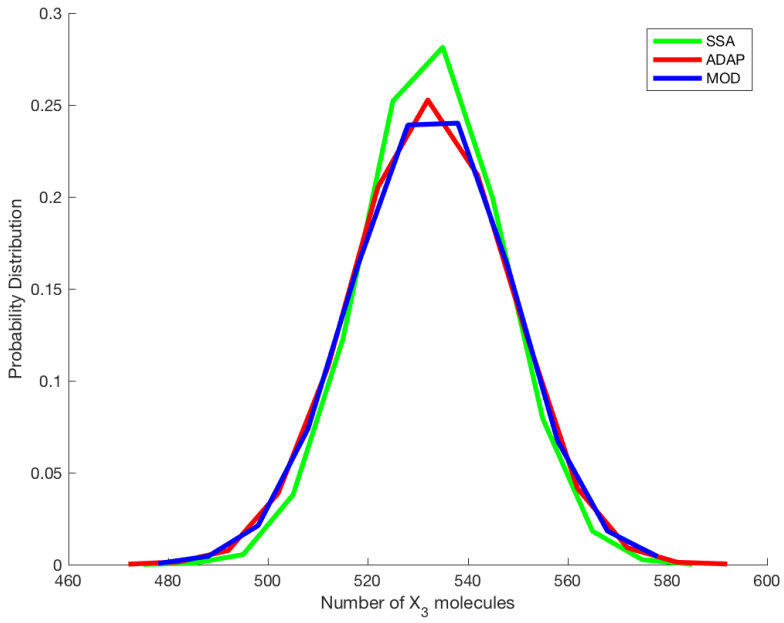

Figure 5.3: Stiff Model: Histogram of the X_3 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 0.01$

5.2 Decay-Dimerization Model

We have partially familiarized ourselves with this model through our demonstration of fundamental concepts in Section 3.1. Nonetheless, we restate it while filling in missing information such as the rate constants and propensities. It should be noted this model is inherently stiff. The model is subjected to the following reaction channels [20],

$$S_1 \xrightarrow{C_1} 0$$

$$S_1 + S_1 \xrightarrow{C_3} S_2$$

$$S_2 \xrightarrow{C_2} S_1 + S_1$$

$$S_2 \xrightarrow{C_4} S_3$$

The model operates on the time interval [0,3], with the stoichiometric matrix,

$$\nu = \begin{bmatrix} -1 & -2 & 2 & 0 \\ 0 & 1 & -1 & -1 \end{bmatrix},$$

rate constants,

$$c = \begin{pmatrix} 1 \\ 0.1 \\ 25 \\ 0.04 \end{pmatrix},$$

and propensities,

$$a = \begin{pmatrix} c_1 x_1 \\ c_2 \frac{x_1(x_1-1)}{2} \\ c_3 x_2 \\ c_4 x_2 \end{pmatrix}.$$

The initial conditions are,

$$X(0) = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$$

with a tolerance (TOL) of 0.04 and $h = 0.1$.

We ran simulations on 10,000 trajectories for the SSA, the adaptive explicit-implicit method and the modified adaptive explicit-implicit strategy. The histograms obtained with the above techniques at $t = 3$ are presented in Figure 5.4 for species X_1 and in Figure 5.5 for species X_2. We remark the good agreement among these methods, showing the accuracy of the adaptive explicit-implicit algorithms. The modified variable step-size tau-leaping is as accurate as the state-of-the-art adaptive tau-leap strategy. From Table 5.2, we see that the speed-up of the modified tau-leaping scheme over the SSA is

$$speed - up(\%) = 463.86.$$

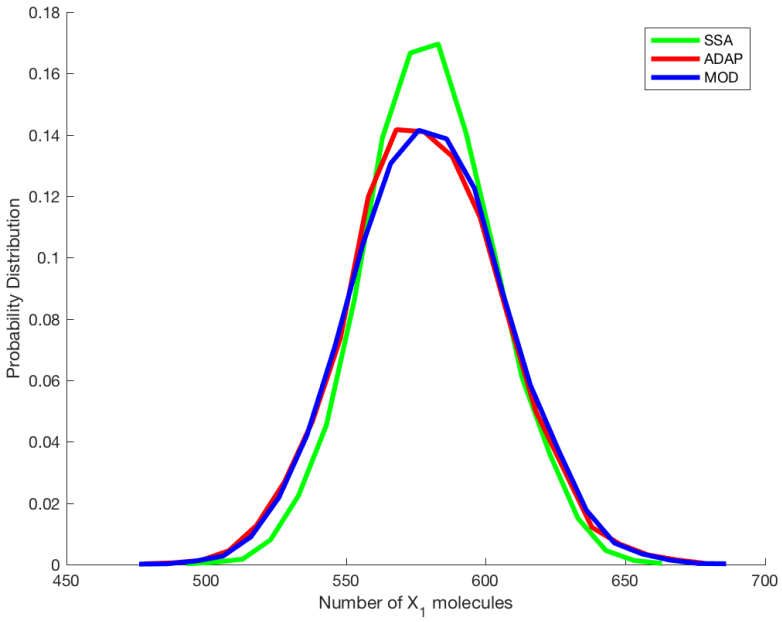

Figure 5.4: Decay-Dimerization model: Histograms of X_1 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 3$

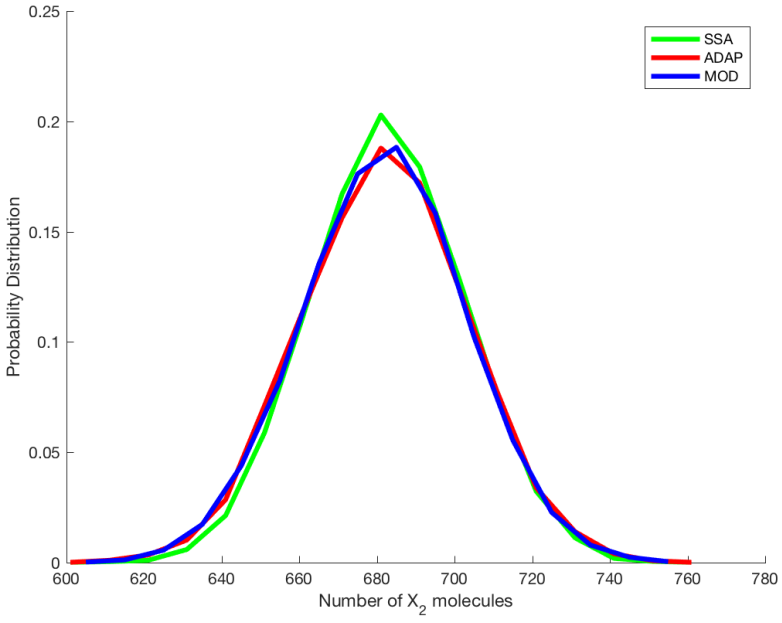

Figure 5.5: Decay-Dimerization model: Histograms of X_3 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 3$

5.3 Modified Cycle Model

The last model we ponder upon is the cycle model [29].

$$S_1 \xrightarrow{C_1} S_2$$

$$S_2 \xrightarrow{C_2} S_3$$

$$S_3 \xrightarrow{C_3} S_1$$

$$S_1 + S_4 \xrightarrow{C_4} S_5$$

$$S_5 \xrightarrow{C_5} S_1 + S_4$$

The model operates on the time interval [0,0.05], with the stoichiometric matrix,

$$\nu = \begin{bmatrix} -1 & 0 & 1 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}.$$

With rate constants,

$$c = \begin{pmatrix} 1.50 \cdot 10^3 \\ 5.00 \cdot 10^3 \\ 1.00 \cdot 10^3 \\ 1.66 \cdot 10^{-4} \\ 8.00 \cdot 10^{-2} \end{pmatrix},$$

and propensities,

$$a = \begin{pmatrix} c_1 x_1 \\ c_2 x_2 \\ c_3 x_3 \\ c_4 x_1 x_4 \\ c_5 x_5 \end{pmatrix}.$$

The initial conditions are,

$$X(0) = \begin{pmatrix} 1000 \\ 800 \\ 400 \\ 40 \\ 50 \end{pmatrix}.$$

with a tolerance (TOL) of 0.05 and $h = 0.1$.

For this model, we performed 10,000 simulations with each of the following algorithms: SSA, the state-of-the-art adaptive explicit-implicit algorithm and the modified explicit-implicit technique. The histograms for at $t = 0.05$ of the three simulation methods for the species X_1 are shown in Figure 5.6, Figure 5.7 for

species X_2 and Figure 5.8 for species X_3. These results demonstrate that the modified variable step-size tau-leaping performs as well as the state-of-the-art adaptive tau-leaping scheme, both consistent with the results obtained using the SSA. This demonstrates that the leaping techniques are accurate. The CPU times of the three methods are given in Table 5.1. We note that the speed-up of the modified adaptive tau-leaping scheme over the SSA is,

$$speed - up(\%) = 395.53.$$

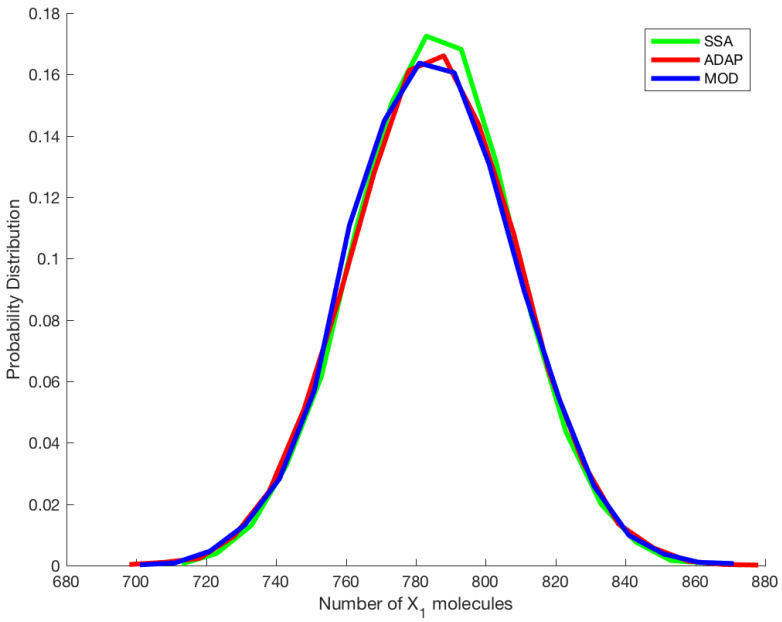

Figure 5.6: Modified Cycle model: Histograms of X_1 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 0.05$.

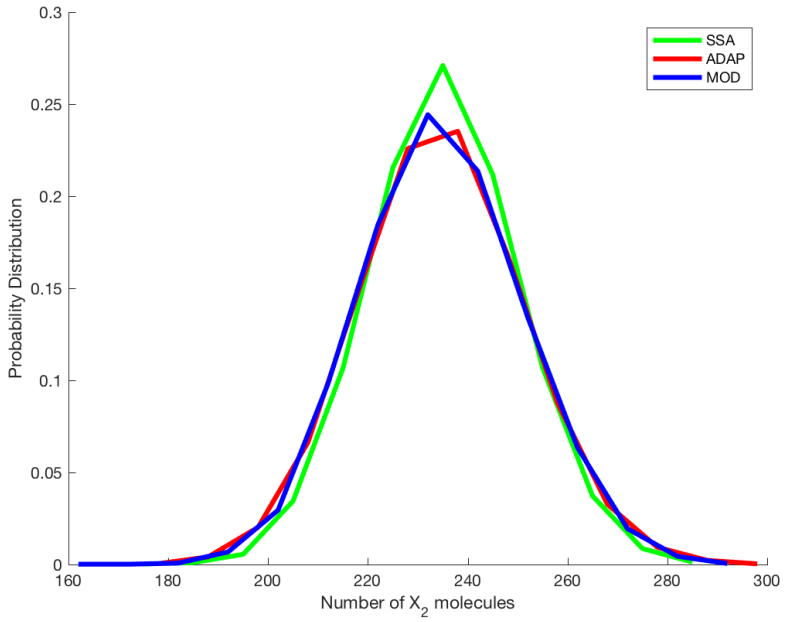

Figure 5.7: Modified Cycle model: Histograms of X_2 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 0.05$.

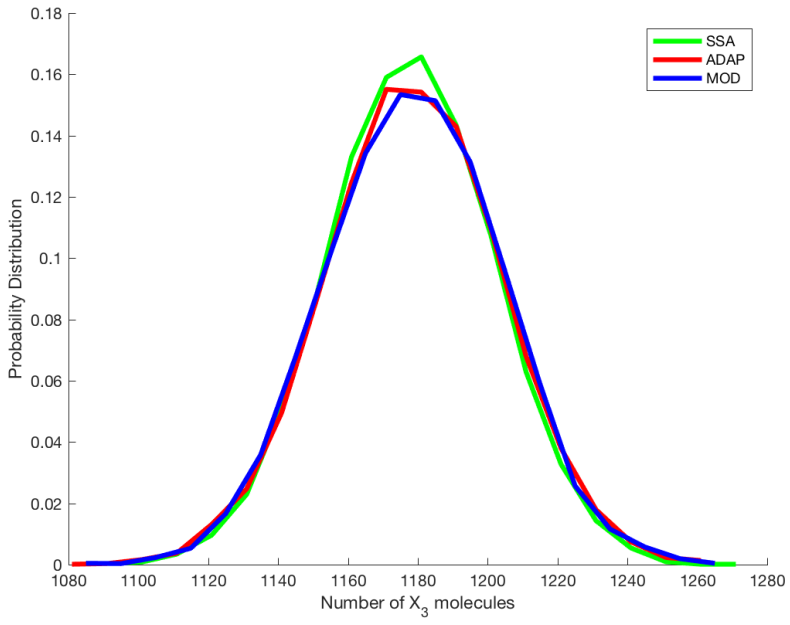

Figure 5.8: Modified Cycle model: Histograms of X_3 species (SSA vs. Adaptive Tau-Leaping vs. Modified Adaptive Tau-Leaping) at $t = 0.05$.

5.4 Table of Results

	SSA (s)	Adaptive (s)	Modified Adaptive (s)
Stiff	4926.64	375.41	329.14
Decay-Dimerization	8398.42	1856.73	1810.57
Cycle	8696.21	2260.13	2198.61

Table 5.1: Computational times of the SSA, Adaptive and Modified Tau-Leaping Methods.

	Adaptive vs. SSA (%)	Modified Adaptive vs. SSA (%)
Stiff	1312.33	1496.82
Decay-Dimerization	452.50	463.86
Cycle	382.70	395.53

Table 5.2: Improvement in computational speed of the adaptive tau-leaping method vs. SSA.

Chapter 6

Conclusion and Further Research Topics

This thesis studied effective simulation methods for stochastic models of well-stirred biochemical systems, with a focus on the explicit and implicit tau-leaping strategies for the Chemical Master equation. Since many biochemical systems in applications have stiff mathematical models, it is essential to develop effective computational tools to study them. One critical tool for overcoming stiffness is adaptive time-stepping. Our first conclusion is confirmation of time-wise computational improvement of the adaptive explicit-implicit tau-leaping scheme in contrast to the exact stochastic simulation algorithm for the Chemical Master equation. The numerical results substantiated that the state-of-the art adaptive tau-leaping method was faster than the SSA up to an order of magnitude of 14. This strategy uses Newton's method for the implicit tau-leaping steps. Arguably the most important result garnered is the performance of the modified adaptive explicit-implicit tau-leaping

algorithm in terms of accuracy and efficiency. The proposed modified scheme employs a pseudo-Newton's method. The numerical results and computational cost of the modified method were closely aligned with the state-of-the-art adaptive scheme. This is vital given that the modified adaptive explicit-implicit tau-leaping technique is more user-friendly following the replacement of the symbolic computation component, a challenge to potential users, with the finite-difference approximation of the Jacobian. The method is ideal for larger systems not well-suited for simulation using the CLE or RRE and those with complex propensity functions, particular propensities not in polynomial form. Finally, we have demonstrated using smaller and larger models that the methods are well-suited to handle a large class of problems.

Reflecting on this work three topics of interest immediately come to mind. First is the issue of negative species populations. This scenario presented itself on more than one occasion. This phenomenon becomes especially problematic in systems with species with low population numbers, which remain close to zero on some time-interval. Species populations falling below zero contaminate the results; intuitively it is evident that negative populations do not exist. A second topic worth exploring is a machine learning problem. A parameter that did not receive much attention and was left largely unchanged, was the tolerance. Using machine learning techniques one could condition the algorithm to optimize the relationship between efficiency and accuracy, by adjusting the tolerance according to system predispositions. Finally, there remains a need for further automatization to minimize the

amount of human input. Additionally, this would allow the algorithm to handle more complex systems, perhaps even problems beyond biochemical kinetics.

Bibliography

[1] H.L. Anderson,1986. Metropolis, Monte Carlo, and the MANIAC. *Los Alamos Science* **14**, 96-107.

[2] Y. Cao, D.T. Gillespie, L.R. Petzold, 2006. Efficient step size selection for the tau-leaping simulation method. *J. Phys. Chem.* **124**, 044109-1-11.

[3] Y. Cao, D.T. Gillespie, L.R. Petzold, 2007. Adaptive explicit-implicit tau-leaping method with automatic tau selection. *J. Phys. Chem.* **126**, 224101.

[4] J.L. Doob, 1953. Stochastic Processes. *Wiley,New York.*

[5] R. Eckhardt,1987. Stan Ulam, John von Neumann, and the Monte Carlo method. *Los Alamos Science* **15**, 131-137.

[6] M.B. Elowitz, A. Levine, E. Siggia, & P. Swain (2002). Stochastic Gene Expression in a Single Cell. *Science* **297(5584)**, 1183–1186.

[7] N. V. Fedoroff, W. Fontana, 2002. Small Numbers of Big Molecules. *Science* **297**, 1129–1131.

[8] M.A. Gibson, J. Bruck, 2000. Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels. *J. Phys. Chem. A* **104**, 18761889

[9] D.T. Gillespie, 1976. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics* **22**, 403434.

[10] D.T. Gillespie, 1992. A rigorous derivation of the chemical master equation. *Physica A* **188**, 402–425.

[11] D.T. Gillespie, 1992. Markov processes, an introduction for physical scientists. *Academic Press, New York.*

[12] D.T. Gillespie, 2000. The chemical Langevin equations. *J. Phys. Chem.* **113**, 297–306.

[13] D.T. Gillespie, 2001. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.* **115**, 1716–1733.

[14] D.T. Gillespie, 2007. Stochastic Simulation of Chemical Kinetics. *Annu. Rev. Phys. Chem.* **58**, 35–55.

[15] D.T. Gillespie, L.R. Petzold, 2003. Improved leap-size selection for accelerated stochastic simulation. *J. Chem. Phys.* **119**, 8229–8234.

[16] D.J. Higham, 2008. Modeling and Simulating Chemical Reactions. *SIAM Review.* **50**, 347–368.

[17] T. Ideker, T. Galitski, L. Hood, 2001. NEW APPROACH TO DECODING LIFE: Systems Biology. *Annu. Rev. Genom. Hum. Genet.* **2**, 343–372.

[18] T. G. Kurtz, 1972. The relationship between stochastic and deterministic models for chemical reactions. *J. Chem. Phys.* **57(7)**, 29762978.

[19] R. Kyle, 1979. Jean Baptiste Perrin. *JAMA: The Journal of the American Medical Association*, **242 (8)**, 744.

[20] H. Li, L.R. Petzold, 2005. Stochastic Simulation of Biochemical Systems on the Graphics Processing Unit. *Bioinformatics* **00**, 1-5.

[21] H. H. McAdams, A. Arkin, 1997. Stochastic mechanisms in gene expression. *Proc. Natl. Acad. Sci. U.S.A.* **94(3)**, 814–819.

[22] H. H. McAdams, A. Arkin, J. Ross, 1997. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected Escherichia coli cells. *Genetics* **149(4)**, 1633–1648.

[23] H. H. McAdams, A. Arkin, 1999. It's a noisy business!. *Trends in Genetics* **15(2)**, 65–69.

[24] Q. Meng, V-P. Mkinen, H. Luk, X. Yang, 2013. Systems Biology Approaches and Applications in Obesity, Diabetes, and Cardiovascular Diseases. *Current Cardiovascular Risk Reports*. **7(1)**, 73-83.

[25] N. Metropolis, 1987. The beginning of the Monte Carlo method, *Los Alamos Science* **15**, 125-130.

[26] R. Feynman, 1964. The Brownian Movement. *The Feynman Lectures of Physics*, **1**,s 41-1.

[27] M. Rathinam, Y. Cao, D.T. Gillespie, L.R. Petzold, 2003. Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method. *J. Phys. Chem.* **119**, 12784–12794.

[28] Y. Sotiropoulos, Y.N. Kaznessis, 2008. An adaptive time step scheme for a system of stochastic differential equations with multiple multiplicative noise: Chemical Langevin equation, a proof of concept. *J. Chem. Phys.* **128**, 014103.

[29] Y. Sotiropoulos, M. N. Contou-Carrere, P. Daoutidis, Y.N. Kaznessis, 2009. Model Reduction of Multiscale Chemical Langevin Equations: A Numerical Case Study. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **6**, 470.

[30] D.J. Wilkinson, 2006. Stochastic Modeling for Systems Biology. *Chapman & Hall/CRC Mathematical and Computational Biology Series*, 45–89, 91–92, 109–133, 157–160.