Purdue University
Purdue e-Pubs

Open Access Dissertations | Theses and Dissertations

5-2018

Transfer of Antibiotic Resistance in Enterococcus faecalis. Modeling and Computational Study

Vu Tran
Purdue University

Follow this and additional works at: https://docs.lib.purdue.edu/open_access_dissertations

Recommended Citation

Tran, Vu, "Transfer of Antibiotic Resistance in Enterococcus faecalis. Modeling and Computational Study" (2018). *Open Access Dissertations*. 1888.
https://docs.lib.purdue.edu/open_access_dissertations/1888

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

TRANSFER OF ANTIBIOTIC RESISTANCE IN ENTEROCOCCUS FAECALIS. MODELING AND COMPUTATIONAL STUDY

by

Vu Tran

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Davidson School of Chemical Engineering

West Lafayette, Indiana

May 2018

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Doraiswami Ramkrishna
Department of Chemical Engineering
Dr. Chongli Yuan
Department of Chemical Engineering
Dr. Julie C. Liu
Department of Chemical Engineering
Dr. David M. Umulis
Department of Agricultural & Biological Engineering

Approved by:
Dr. Sangtae Kim
Head of the Graduate Program

“Life is like photography. You need the negatives to develop.”

ACKNOWLEDGMENTS

I would like first to thank my advisor, Professor Doraiswami Ramkrishna for providing me the opportunity to pursue my Ph.D. under his guidance. Even though I entered the program with not much knowledge on the subject, Professor Ramkrishna has been patient with me and supported me throughout the process. He has fostered and equipped me with valuable knowledge and skills that I would have not gained otherwise. During this professional endeavor, there have been numerous times that I have felt unmotivated and lost in research obstacles, but he has always been there to support me and has provided guidance to overcome those problems and to succeed. Along with his support, he has also granted me intellectual freedom to discuss and explore possibilities in order to obtain the solutions for problems that we have encountered. He has constantly taught me to be a good presenter, communicator, and collaborator. All of these have benefited me and helped me to become who I am today. I feel ready and am looking forward to growing and learning more from challenges that I will encounter in the next the next phase of my career.

I would also like to thank many good friends and the collaborator at the University of Minnesota. Without all the support from groupmates including Frank, Conor, Akancha and Parul, my journey would have been so much harder. Not only have I had great opportunities to discuss our problems as modelers, we shared good times together at coffee breaks, lunches and skating activities. I would also thank Dr. Hu and Arpan for the ongoing support and discussion about the project. This collaboration helped me realize the power of working in a team. People think very differently and each individual in a unit can therefore learn and grow together. I would also especially thank Oscar, Agnes, Rohit, David and Ramon who have supported me more as good friends throughout the most difficult and

stressful times here. From outside of the department, I would like to thank Jeff and Sean for ongoing support. We have shared good/bad times together.

Without my parents, I would not have been able to pursue and complete my Ph.D. They have been always there loving and supporting me. They have done everything they can to help me stay focused and manage stress. Without them, this work would not have been possible. I also want to thank my brother and sister for inspiring me to be a model for them to follow. I love you all and promise to advance further in my future endeavors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Author: Tran, Vu, T. Ph.D
Institution: Purdue University
Degree Received: May 2018
Title: Transfer of Antibiotic Resistance in *Enterococcus Faecalis*. Modeling and Computational Study.
Major Professor: Doraiswami Ramkrishna

Bacteria of the genus *Enterococcus*, commonly found in the intestinal tract, are the main cause of antibiotic-resistant infections that are acquired in hospitals[1], [2]. Donor cells that contain plasmid pCF10 have the ability to resist to antibiotics and are capable of transferring this plasmid to recipient cells. This transfer occurs via a rapid horizontal inducible conjugation regulated by peptide-mediated cell-cell signaling molecules (quorum sensing), known as cCF10 and iCF10. This quorum sensing system functions by producing low levels of an inducing substance that accumulates in the environments until a threshold is reached, at which point there is a change in cellular behavior. Cells of this type can either exist in the free floating form or in biofilms, which are composed of cells attached on biotic and abiotic surfaces. Complexity of the biofilm structure hinders and affects the exposures of cells to antibiotics and hence reduces treatment efficacy. Successful models of this mechanism can lead to useful techniques/methods in controlling or interfering with the plasmid transfer.

Several efforts to model this phenomenon have been initiated and developed by our group in recent years. Recently, the collaborative experimental group in University of Minnesota has discovered new mechanisms that are associated with the system. This discover invalidates previous assumptions and hence requires modifications on both reactions and modeling assumptions. Moreover, various variables in the system have shown stiff

behaviors that are much more challenging to work with. Explicit SDE, used in previous system, can be no longer capable of obtaining accurate solutions. For these reasons, this thesis presents new updated strategies to capture the drug resistance transfer in both Planktonic and biofilm environments. Since the two systems are inherently different in structure and physics, usage of varied modeling formulations for each environment is inevitable. Deterministic models are very simple and can be used to acquire a rough prediction of Planktonic environment. However, their simplicity also limits their capability of capturing large complex systems such as biofilms and other highly heterogeneous systems. Unfortunately, stochastic models can also carry a huge burden on CPU time. Therefore, another part of this thesis is dedicated to illustrate techniques, which can be used to reduce stochastic simulation time without losing accuracy. Successfully solving these two major problems together can potentially serve as a tool to gain knowledge about the system and eventually develop methods to treat/control this phenomenon.

1. INTRODUCTION

The investigation in this thesis focuses on two main categories: (1) drug-resistance transfer in *Enterrrococus faecalis* and (2) advance computational methodologies and algorithms for stochastic systems. The first topic can be then broken down into two sub sections where mechanisms and cell behaviors are investigated and studied in two different environments: planktonic and biofilms, using stochastic simulations. In the second topic, accelerated computational algorithms are discussed. Success in developing and implementing these algorithms plays a critical role in obtaining solutions and gaining understanding of chemical/biological stochastic networks.

The thesis therefore can be broken down as follows. Chapter 2 will provide general background on the biological system and stochastic methods (Master equations and stochastic differential equations). Chapter 3 focuses on updated assumptions suggested from new experimental results. With these new assumptions, the models are reestablished and formulated. In this chapter, dynamic simulations for the deterministic single cell as well as population levels are shown. Because of this modification, the system now exhibits stiff behaviors, which cannot be solved effectively by explicit SDE, utilized in the previous work. Chapter 4 provides some useful insights on the dynamics of drug transfer in the biofilm case, from the simulation point of view. Chapter 3 also includes a section to show equivalent results obtained by solving SDE and master equations. My Phd work has its focus on instituting various ways of accelerating simulation techniques to obtain the solutions of the master equation. Development of stochastic algorithms using Chebyshev's inequality is described in chapter 5. Chapter 6 focuses on a "parallel" strategy for capturing behaviors of species in chemical/biological systems. Both chapters 5 and 6

provide remedies for ongoing CPU burden problems associated with stochastic simulations. Chapter 7 advances problem solving with the goal of obtaining the averaged solution of the master equations by combing an existing method of Grima and a proposed method from our group. The validity of this method has also been verified directly on the example of drug resistance transfer in biofilms. Chapter 8 in the thesis is a recent work and is dedicated to developing/implementing an accelerated strategy to compute the average behaviors without going through a time-consuming process of generating numerous sample paths for the differential stochastic equations. This method is tested/verified on this actual biological system and can potentially serve as a tool to suggest future experimental design space. The final chapter will focus on similarity and differentiation between the two scenarios, from which some potential solutions for controlling this mechanism are recommended. This chapter will also include proposals for future work.

1.1 Drug resistance transfer in *Enterrococus faecalis*

Conjugation is one of the most common ways from which bacteria acquire resistance, contributing to the emergence of multi-drug resistant "superbugs."[3] *Enterococcus faecalis*, known as one of the most frequent causes of hospital-acquired infection, utilizes the mechanism of conjugation to transfer antibiotic resistance from plasmid-harboring antibiotic-resistant donors to plasmid-free antibiotic-sensitive recipients. In the *Enterococcus faecalis* family, the plasmid carrying the tetracycline resistance is known as pCF10, shown in Figure 1.1

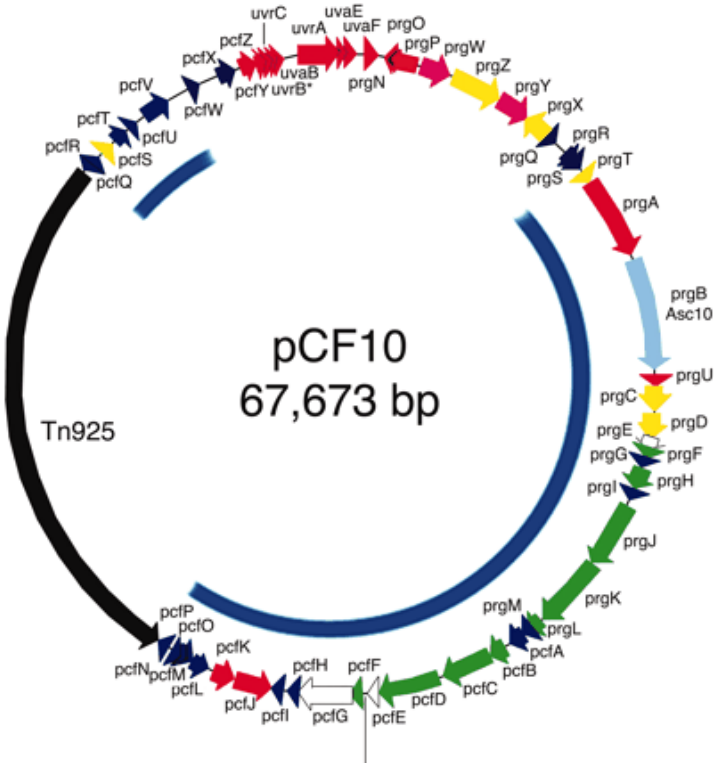

Figure 1.1: Schematic of the plasmid pCF10

Two types of signaling molecules which regulate conjugative transfer of the plasmid pCF10 are inhibitor iCF10 and inducer cCF10. cCF10 molecules, responsible for inducing conjugation, are generated by recipient cells. Donor cells, on the other hand, produce iCF10, whose role is to repress conjugation. Depending on the ratio of cCF10/iCF10, cells can proceed to conjugation, a direct cell-cell contact which allows the transfer resistance determinant to be transferred between donors and recipients. Two main genes that regulate this circuit are *prgX* and *prgQ*, coding for a repressor protein and coding for poly-transcript (inhibitor peptide and conjugation machinery), respectively. The promoters for the two genes are located against each other with an overlapping region of more than 230 bp. The convergent transcription from *prgX* and *prgQ* leads to RNA polymerase collision in the

overlapping region, varying the dynamic range of *prgQ* transcript level between induced and repressed states. In addition, there is additional evidence for the control level of mRNA transcribed from two promoters P_Q, *prgQ* promoter, and P_X, *prgX* promoter, which are located at the two opposite ends. Because the two promoters approach from two opposite locations during the transcription, the two transcriptions corresponding to those promoters are convergent for about 230 bp and complementary in this region [4]. P_X generates both *prgX* mRNA and anti-Q molecules. In the absence of pheromone induction cCF10, the transcription from P_Q produces Qs, a 380 nt RNA which in turn generates iCF10 inhibitor. When the donor cells import pheromone into the cells, cCF10 starts binding to PrgX, disrupting the structure of PrgX-iCF10 tetramers and resulting in the generation of a longer transcript $Q_{l,}$ about 530 nt[5]. Q_L stimulates the expression of *prgB* encoding a surface aggregation protein and making conjugation to occur. While the basal levels of I produced from Qs transcripts help prevent self-induction I the event of no recipients, an increase of I after induction serves to rapidly shut down the response. Multiple positive/negative feedback loops in the *pCF10* system and corresponding response of *prgQ* expression in donor populations of donor cells suggested the system could function as a bistable switch. Several processes are shown in Figure 1.2:

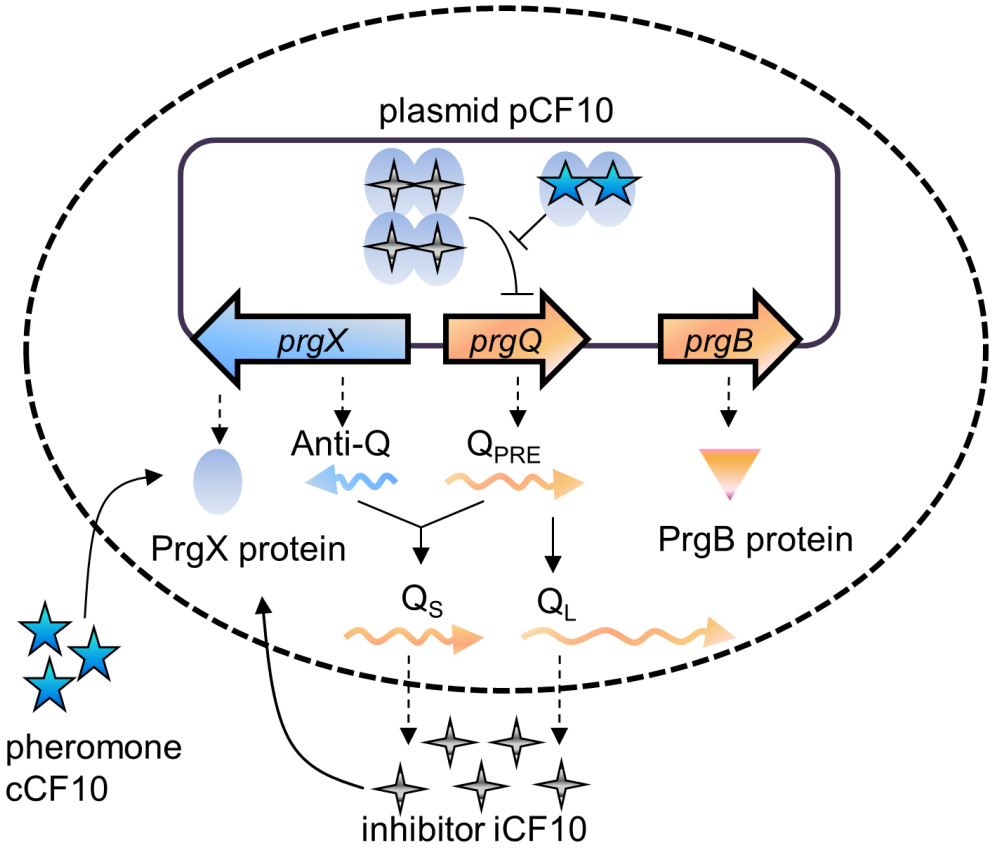

Figure 1.2: The gene regulation of pCF10

Evolution is able to shed light on the observed phenomenon described above**.** Due to any adverse changes in environment, cells will accommodate and/or adjust their own structure to enhance their ability to survive. This explains why recipient cells send out the induction pheromone and seek out donor cells. At the same time, a natural response of a cell is to protect itself from any strange invasion or exchange materials between their "bodies" and the environment, explaining the existence of anti-sensing molecule iCF10, whose function is to shut down the conjugation. The second reason for the existence of the inhibitors comes from the cell perspective, is to minimize energy/resource expenditure. Conjugation and

plasmid transfer require a great deal of energy expenditure and so therefore, cells do not necessarily involve in this process unless there is a threat to their survival

Cells can exist either in a Planktonic environment where materials are well mixed or in biofilms. Biofilms, produced by bacteria, is a matrix composed of different biopolymers: polysaccharides, polypeptides, lipids and nucleic acids. From the treatment standpoint, bacterial cells in the biofilm state have shown an extreme elevated resistance to many clinically relevant antimicrobials agents [6]. The hypotheses for elevated phenomenon are due to either a more dense inhomogeneous structure of biofilms and/or a functional change in gene expression as cells try to modify themselves when switching from planktonic to biofilm environments, but the actual cause is as yet unclear [7]. In the presence of both donors and recipient cells, each produces different types of peptide-mediated cell-cell signaling (either inducers or inhibitors) to communicate and to regulate the conjugation process. The two types of signaling molecules which regulate the conjugative transfer of this plasmid are inducer **cCF10** and inhibitor **iCF10**. Depending on the domination of either **cCF10** or **iCF10**, donors will have different responses: promoting the conjugation when the recipient cells are abundant and restraining the process when the recipient concentration is low. Understanding the balancing between different signaling mechanisms in this system may shed light on new ways of controlling the spread of antibiotic resistance.

1.2 Accelerated modeling algorithms for the drug resistance transfer and other biological/ chemical systems

Mathematical modeling has played an important role in analyzing biological/chemical systems. However, modeling capabilities and its merit depend strongly on their approaches

and assumptions. Each model can in fact provide some rough depiction of nature but how detailed or accurate one wants a model to be is dependent on the choice of the model. A modeling classification is available in [8], [9]. A more elaborate classification is of made by Ramkrishna [10]. One main criterion that many scientists and others have utilized in the past few decades is the question of using deterministic vs. stochastic models. The deterministic approach, based on the phenomenological law of mass action, involves solving ordinary differential equations (ODEs). Solutions of this approach provide averaged behaviors of the system but of course cannot capture any associated noise. This type of model has shown some success in describing homogenous systems in theoretical biology, examples of which include description of metabolism [11], signaling[12] or gene regulations within cells[13], and the analysis of population dynamics[14]. However, biological systems are subject to stochastic effect from molecular to macroscopic levels. Randomness is an important factor in signaling and regulations, where the number of particles is usually low and gene expression is intrinsically noisy[15].

Two main approaches to obtaining solutions for stochastic formulations are

- Formulating master equations from which Langevin equations and/or Fokker-Planck equations can then be derived. This method accomplished the solutions by solving continuous stochastic differential equations.
- Discretizing systems in which quantities of species exists in number, captured in Chemical master equation (CME). The strategy can be accomplished using Monte Carlo simulations based either on an exact stochastic algorithm or an approximate methodology to speed up the process using τ -leap and its modified versions.

The first method was employed in our previous papers [16], [17] and has shown success in demonstrating the system interactions, but encountered problems related to stiffness of the

differential equations of the modified new set of equations and assumptions. In our new dynamics responses, some quantities change too rapidly as compared to others, causing the system dynamics to exhibit very stiff behavior. Standard explicit methods, such as the Maruyama method used in the previous paper [16] and others, usually face huge burdens from step-size restriction. Several papers have proposed strategies of using implicit stochastic differential equations [18], [19] as a remedy for this problem. Unfortunately, in many cases, this method can result in users solving either large or badly conditioned linear systems**.** Moreover, this method is also very specific with respect to which systems to which the algorithm is applied. To overcome this obstacle, usage of sample path wise simulations was introduced and formulated as an attempt to approximate the real solutions [18], [20]. This approach has shown several successes and has proved its merits in a much broader set of scenarios.

For biochemical networks where discrete particles are associated with small quantities, the chemical master equations (CME) is frequently applied [21]. Unfortunately, analytical solutions of CME are usually intractable. The Gillespie's algorithm [22] and Shah's algorithm [23] offer a way to obtain the exact simulations of trajectories for the master equations yet require a large computational cost for systems that is composed of multiple components undergoing many reactions. In recent years, different groups have developed methods to approximate the exact solutions. One of those strategies is called tau-leap, a leaping technique through which the method captures multiple events, rather than a single event in the exact method. This method, developed as an effort to speed up the CPU time that can be large in the case of exact stochastic simulation for systems with large numbers of species and reactions, is an attempt to compromise accuracy for speedier solutions. The

underlying idea was to leap down the history by some calculated τ that comprises many reaction events. The number of reaction occurrences is approximated using either Poisson [24], [25] or binomial random variables [26]–[28]. Logically, the larger leap that the system can jump, the shorter the simulation time to complete a simulation can be acquired. On the other hand, too big of a leap can introduce more errors to the approximation. In order to preserve the accuracy, a leap condition [24], [29], [30], which implies that the state change in any leap should be small enough that no propensity functions experience a significant change, was proposed. The validity of these algorithms strongly depends on the "leap condition" under which the propensity functions change insignificantly. Ramkrishna's group later proposed a modified version of the leap condition using Chebyshev's inequality [31], to reduce the number of generated "bad" sample paths and hence reduce CPU time for obtaining a more accurate solution. Due to the nature of stochastic simulation that each sample path is independent and varies greatly from others, the group also developed another algorithm which can be applied to both exact and tau-leap methods to reduce burden on CPU time. By simultaneously starting all sample paths[32], the method allows some of the trajectories to finish before and hence avoid waiting time between sample paths generated by traditional sequential stochastic simulation. With all the foregoing developments, good approximate solutions with a reasonable simulated CPU time have been facilitated. However, as the numbers of reactions and species increase, simulation time is scaled up quickly, the need for further improvements continues to be felt. This similar problem also applies to systems where diffusion happens at a comparable rate with reactions. The CME can capture the fluctuations of small quantities in chemical/biological systems that are assumed to be well

mixed. Due to this assumption, relative position of reacting molecules has no effect on the propensity functions. Reaction diffusion master equations (RDME) provides an additional capability of capturing diffusion effect where structure and cell interaction are non-uniform throughout the domain. The RDME incorporates the local effect in the mesoscopic limit and has been proven validity under some specific conditions [33]–[35]. Similarly, if systems are composed of large numbers of reactions and species with the effect diffusion, obtaining solutions for these systems in an acceptable simulation time becomes very difficult.

A refreshing approach toward resolving the problems mentioned above was introduced in a paper by Grima [36], which raises the possibility of directly obtaining average(mesoscopic) behaviors. The underlying methodology lies with quasi-linearization leading to derivation of dynamical equations, which represent the potential source of average behavior [37]. The work is first introduced to obtain directly the average solutions without simulating sample paths for well-mixed systems. An extension of this algorithm with a linear operator approach for diffusion provides for a full capture of average results without generating sample paths in systems where both reactions and diffusion take place [38]. A similar approach for calculating average behaviors without generating trajectories, which utilizes Taylor series expansion to approximate the mean and corresponding standard deviation at each time step from a previous time point, is presented elsewhere in this thesis.

2. BACKGROUND

2.1 Chebyshev's inequality

For any random variable X with finite expectation EX and variance VX satisfies the following inequality (Feller, 1968) for an arbitrary $\varepsilon > 0$.

$$\Pr\{|X - EX| \geq \varepsilon\} \leq \frac{VX}{\varepsilon^2} \tag{1}$$

The version used in the paper[31] which is equivalent with above inequality can be stated

$$\Pr\{|X - EX| < \varepsilon\} > 1 - \frac{VX}{\varepsilon^2} \tag{2}$$

Clearly this inequality stipulates the minimum of the probability with which the random variable X deviates from EX by less than ε

2.2 Basic tau-leap method

Given a well-stirred chemically reacting system, the state vector $X(t) = (X_1(t), ..., X_N(t))$ is the number of molecules of species S_i in the system at time t. N species can undergo reactions in M different channels $\{R_1, ..., R_M\}$. The dynamics of reaction channels R_j are defined by a propensity function a_j with stoichiometry vector $\nu_j = (\nu_{1j}, ..., \nu_{Nj})$. $a_j(x)\tau$ is the probability that one R_j reaction will occur during the next infinitesimal time interval dt. Here is the leap condition :

$$|a_j(x + \Lambda(\tau; x)) - a_j(x)| \leq \varepsilon a_o(x), \quad \forall j = 1, ..., M \tag{3}$$

Given a leap τ that satisfies the leap condition, the number of fires for j-th reaction can be approximated using Poisson random variable: $K_j(\tau; x) \approx P_j(a_j(x), \tau)$

$$x(t+\tau)-x(t)\equiv\Lambda(\tau;x)=\sum_{j=1}^{M}K_j(\tau;x)\nu_j \tag{4}$$

Taylor approximation:

$$\Delta a_j(\tau;x)\overset{\Delta}{=}a_j(x+\Lambda(\tau;x))-a_j(x)\approx\Lambda(\tau;x).\nabla a_j(x)=\sum_{i=1}^{N}\Lambda_i(\tau;x)\frac{\partial a_j(x)}{\partial x_j} \tag{5}$$

Combining Eqs. (4) and (5),

$$\Delta a_j(\tau;x)\approx\sum_{j'=1}^{M}f_{jj'}(x)P_{j'}(a_{j'}(x),\tau)$$

The above indicates that $\Delta a_j(\tau;x)$ is a linear combination of independent Poison random variables with mean and variance given by:

$$<\Delta a_j(\tau;x)>\approx\sum_{j'=1}^{M}f_{jj'}(x)<P_{j'}(a_{j'}(x),\tau> \tag{6}$$

$$\mathrm{var}\{\Delta a_j(\tau;x)\}\approx\sum_{j'=1}^{M}f_{jj'}^{2}(x)\,\mathrm{var}\{P_{j'}(a_{j'}(x),\tau\} \tag{7}$$

Since $<P(a,\tau)>=\mathrm{var}\{P(a,\tau\}=a\tau$, this leads to

$$<\Delta a_j(\tau;x)>\approx\sum_{j'=1}^{M}f_{jj'}(x)(a_{j'}(x)\tau)\equiv\mu_j(x)\tau \tag{8}$$

$$\mathrm{var}\{\Delta a_j(\tau;x)\}\approx\sum_{j'=1}^{M}f_{jj'}^{2}(x)(a_{j'}(x)\tau)\equiv\sigma_j^2(x)\tau \tag{9}$$

The leap can be calculated, according to equation (10)

$$\tau=\underset{j\in[1,M]}{\mathrm{Min}}\{\frac{\varepsilon a_0(x)}{|\mu j(x)|},\frac{\varepsilon^2 a_0^2(x)}{\sigma_j^2(x)}\} \tag{10}$$

2.3 Basic Milstein's method

For the Ito's equation:

$$dX=a(t,X)dt+\sigma(t,X)dw,\qquad X(t_0)=x,\quad t_0\le t\le t_0+T \tag{11}$$

where $w(t)$ is a standard Wiener process [39]. The Milstein's approximate is as follows:

$$\overline{X}(t_0) = X(t_0), \tag{12}$$

$$\overline{X}(t_0 + (k+1)h) = \overline{X}(t_0 + kh) + \overline{\sigma} w_{k+1} + (\overline{a} - \frac{1}{2}\overline{\sigma}\frac{\overline{\partial\sigma}}{\partial x})h + \frac{1}{2}\overline{\sigma}\frac{\overline{\partial\sigma}}{\partial x} w_{k+1}^2 \tag{13}$$

where $h = T/m;\ k = 0,1,...,m-1;\ w_1,...,w_m$ are independent normal $N(0,h)$ variables.

It is also shown that $\mathrm{M}(X(t_0+T) - \overline{X}(t_0+T))^2 = O(h^2)$

2.4 Relationships between tau-leap methods and Langevin's equation

Let assume there exists a leap $\tau > 0$ such that the leap condition is satisfied. Given that condition, $a_j(x)$ is constant in this interval, and the probability that reaction jth occurs in $[t, t+\tau]$ is $a_j(x)\tau$. Therefore, the number of fires for reaction j-th during that interval follows a Possion distribution with parameter $a_j(x)\tau$. Leap methods approximate:

$$x(t+\pi) = x(t) + \sum_{j=1}^{M} \nu_j P_j(a_j(x), \pi) \tag{14}$$

If π is such that $a_j(x)\tau \geq 1$ for all reactions, Poisson random variable with mean and variance $a_j(x)\tau$ can be approximated by a normal random variable with the same mean and variance[40]

$$P_j(a_j(x), \tau) \approx a_j(x)\tau + \sqrt{a_j(x)\tau} N_j(0,1) \tag{15}$$

Substituting back we then have:

$$x(t+\tau) = x(t) + \sum_{j=1}^{M} \nu_j a_j(x)\tau + \sum_{j=1}^{M} \nu_j \sqrt{a_j(x)\tau} N_j(0,1) \tag{16}$$

Equation (21) is also known as Euler-Maruyama numerical approximation for the Langevin equation.

3. UPDATED ASSUMPTIONS LEADING TO NEW MODELING FRAMEWORK

According to the new experimental results, rapid equilibrium assumptions for all reversible reactions used in the previous models are no longer valid. Instead of separating variables and utilizing quasi steady state, reactions in the present model need to be solved all simultaneously. Binding reactions between *prgX* and cCF10 or iCF10 are no longer irreversible and once they bind together, they will stay intact until degradation. Moreover, incorporating new assumptions into our stochastic differential equations results in stiff behaviors that can be ineffectively and inaccurately solved by the explicit Murayama-Newton method utilized in previous papers. Moreover, since species in our interested system exist in discrete quantities, solving the Master equation yields a more accurate solution. Some efforts on formulating and developing new strategic approach with results as well as future approaches will be discussed in this work.

3.1 Update assumptions and corresponding results from single average deterministic modeling

Donor cells in Planktonic case have a plasmid number distribution between 1 and 9. Steady state curves of Q_L, responsible for activation of conjugation, at different numbers of plasmids have shown similar trend. The simulations are done in order to capture Q_L response at different fixed values of extracellular concentration of C. Increasing the concentration of the pheromone led to the elevation of Q_L, shown in Figure 3.1A, which supports our understanding about the pheromone's role in this regulation. Figure 3.2B

includes steady state curves for Q_L level corresponding to varied plasmid numbers. As expected, a higher plasmid number results in a higher concentration of Q_L for a same concentration of extracellular C.

Steady state curve for Q_L and Q_S at various [cCF10] for plasmid number =5

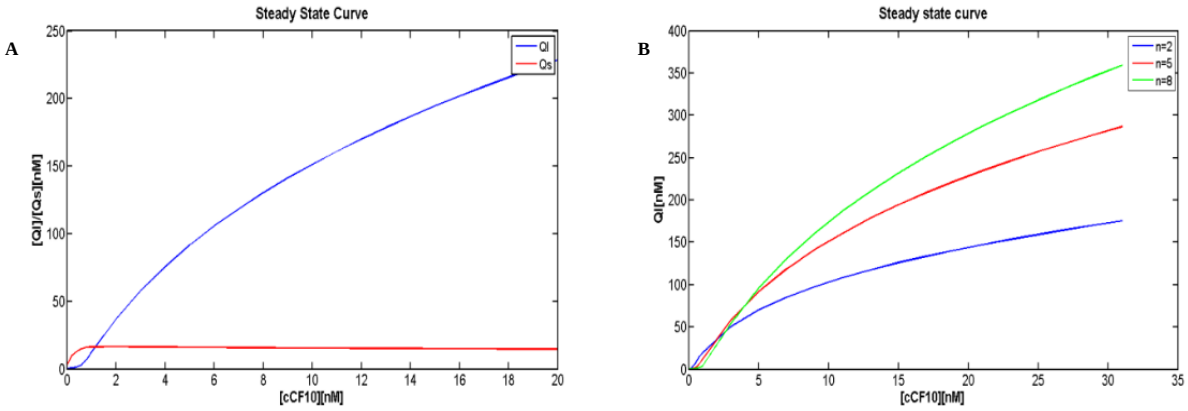

Figure 3.1: Steady state curve. (A): Q_L and Q_s at various concentration of C. (B): Q_L responses to various [C] for different plasmid number

Bistability, shown in the previous work, are not present in this model with the new update assumptions. This can be a result of two new assumptions, made based on experiments, which were applied to the updated formulation. First, the binding reaction between iCF10 or cCF10 and *PrgX* used to be written down as a reversible reaction. For that reason, as concentration of either signaling molecule elevates, they can compete and replace the other at the binding sites. This phenomenon is somewhat similar to what is termed a "cooperative relationship" which is usually shown in bi-stability curves that are composed of two main regions: one with lower concentration of stimulant and the other with higher concentration of it. Recent experiments have demonstrated that the binding reaction rate constant is at least 10^6 times faster than that of the reversed reaction. Thus, we can assume once either signaling molecules iCF10 or cCF10 bind to *PrgX*, they will remain there until degradation.

As a result, cooperative activity is lost in this scenario and the two reactions occur more in an independent sense. The corresponding change is reflected as a gradual change, rather than the big jump that is observed in the previous case. Second, the binding reaction between iCF10 or cCF10 and *PrgX* was modeled as fourth order in iCF10 or cCF10. It is similar to the scenario in enzyme reactions. Cooperative activity and bistability usually emerge if the power associated with the substrates (in this case they are either cCF10 or iCF10) is larger than one, fitting the previous formulation. Our updated development assumed the orders of these reactions are first order in signaling molecules and should behave more similar to the ordinary cases.

Dynamic models for the single averaged cells have displayed agreement with the functions of each component in the system. A set of all anticipated reactions occurring in this process is provided in the Appendix section. Dynamic behavior was modeled as a set of nonlinear ordinary differential equations. Solving the system to acquire steady state and dynamic curves requires an input of initial conditions. In order to make sure the initial conditions have physical meaning, a set of values, except the value for extracellular cCF10 chosen to be zero, were selected which produced the steady state value for all variables. This set of values corresponded to the off-state steady state values for the system. This set with an additional update on concentration of extracellular cCF10 to the value of interest was then utilized as initial condition for our dynamic simulation. Codes for the dynamic model, developed in Matlab language, utilized ODE15s and ODE23s as tools to produce plots for various variables.

The cells started with off-state conditions (no cCF10 in the environment) then evolved as cCF10 being added to the environment and fixed at 5nM. Diffusion leads to an increase of

intracellular concentration of cCF10. As a result, concentration of Q_L was increased. To counteract this effect, donors also produced more iCF10 to generate more Qs. This also explained an increase of Qs but much slower also being observed. These two productions influenced one another, showing clearly in the Q_L curve. Addition of extracellular cCF10 leaded to an elevated Q_L at first followed by a small reduction of Q_L (due to the counteracted production of Qs) and stabilized after some time when both reached equilibrium. Dynamic behaviors of all quantities are shown below:

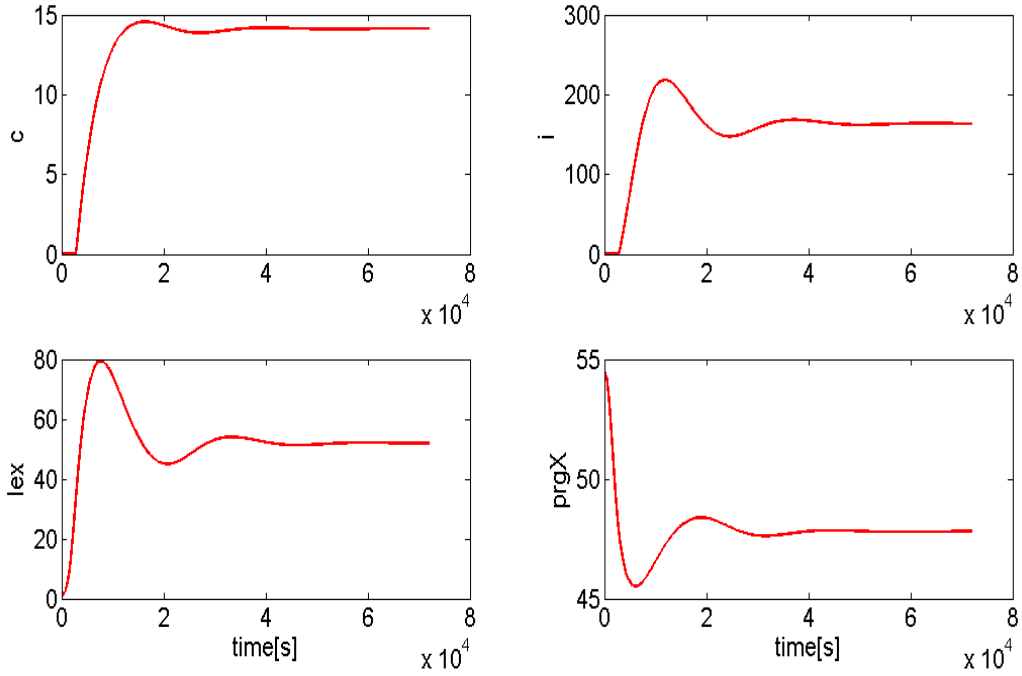

Figure 3.2: Dynamic behaviors at C=5nM

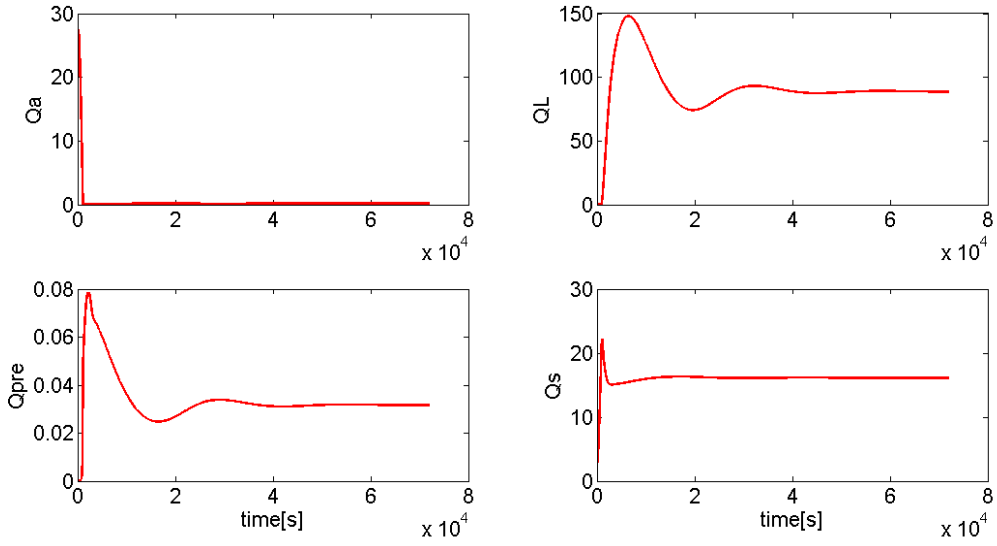

Figure 3.3: Dynamic behaviors at C=5nM (cont.)

Experiments have provided us with a set of some kinetic parameters. Some others were transferred from the previous models, and the rest require parameter fittings. Tables 3.1 and 3.2 in the Supplementary Materials section include the set of all reactions occurring in this system as well as a chosen set of parameters which were used to generate dynamics

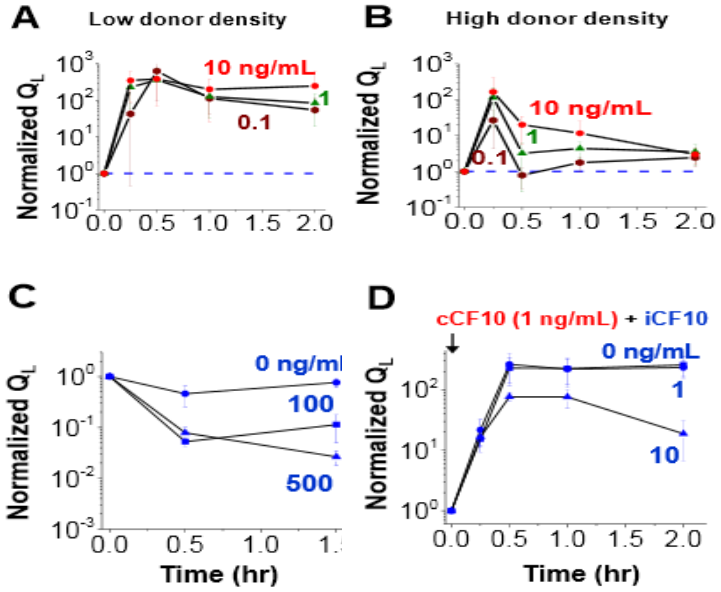

Figure 3.4: Q_L responses at various conditions

plots. Table 3.2 shows ranges of some investigated parameters for parameter fitting purposes. The set of final values was selected in a way so that it can best fit and explains the physical phenomenon. They produced similar trends in most cases, but those selected satisfy both physical meaning and accepted ranges of actual biological systems. The experimental group has collected various results testing the function of each type of signaling molecules. Figure 3.4 below describe dynamic responses of Q_L at various conditions:

Figure 3.4 includes and arranges 4 subplots in the following manners. First column indicates the condition of a low donor density, whereas the second column shows the Q_L responses for the case of a high donor density exists. Plots on the top row describe the behaviors when a fixed concentration of inducer was kept in the environment. The two plots on the bottom row show response when the environment is controlled to have only the inhibitor. As expected, when cCF10 is present in the environment Q_L level increases shown in A and B, but decreases a lot quicker in the case of more donors are around in the surrounding (B). In the contrast, when only the inhibitor I exists in the environment, Q_L level remain near zero, indicating an inactivate state for the conjugation.

3.2 Dynamic models for a population of both donors and recipients using deterministic approach

All previous models developed to dates only attempt to capture the activation of donor cells prior to actual conjugation. For that reasons, the level of Q_L was utilized as the quantitative measure of this entire process of plasmid transfer. This also explains why the model could

be developed as an averaged single cell model. Actual conjugation requires presence of both donors and recipients in the environment as well as interactions between the same species (donor-donor, recipient-recipient) or different species (donor-recipient). Since it was shown from previous models that deterministic models work fairly well for Planktonic case, our first effort is to simply, capture all of these interactions and processes, including conjugation, without consideration of stochastic effects.

Simulations were performed at the population level for transconjugants, new donor cells resulted from a successful conjugation, which is determined from the population balance equation integrated over a discrete time interval assuming that intracellular states do not vary during the small time interval. A two parameter (λ, α) distribution for contact time (with a minimum value for successful conjugation) was used although the computations in Figure below used $\alpha = 0$. This formulation also took into account a few new concepts:

- For a conjugation to occur, a donor cell needs to be activated and make contact with a recipient. Therefore, the contacting frequency depends on both recipients and donors population.
- $\tau_{\min} = 10$ mins is defined as a minimum contacting time for a conjugation to be considered successful. If two cells depart before that time, the plasmid has not successfully transferred; therefore, the recipient is still a recipient. This time generator follows exponential distribution.
- If the transfer is successful, transconjugants will not be able to perform plasmid transfer for t_d=15mins.

  Using subscripts on the number density to denote copy number and prime to represent cells not ready to conjugate we obtain

$$N_o(t+h) \approx N_o(t)\{1-hk_{con}Ql(t)\eta f'(t)\} \quad , \quad N_o(0) = N_0 \tag{17}$$

$$N_1(\mathbf{x}(t+h), t+h) \approx N_1(\mathbf{x}, t) + hk_{con}\delta_{\mathbf{x},\mathbf{0}} q_L(t) N_o(t) \eta f'(t), \tag{18}$$
$$N_1(\mathbf{x}, 0) = N_1 \delta_{\mathbf{x},\mathbf{0}}$$

$$N_k'(\mathbf{x}_k(t+h), t+h) \approx N_k'(\mathbf{x}_k, t) + hk_{con}\delta_{\mathbf{x},\mathbf{0}} q_L(t) N_o(t) \eta f'(t-t_d), \tag{19}$$
$$N_k'(\mathbf{x}_k, 0) = 0, \quad k = 1, 2, \ldots, 9$$

$$\mathbf{x}(t+h) = \dot{\mathbf{X}}(\mathbf{x}(t))h + \mathbf{x}(t), \quad \mathbf{x}_k(t+h) = \dot{\mathbf{X}}_k(\mathbf{x}_k(t))h + \mathbf{x}_k(t) \tag{20}$$

where N_o denotes recipients, N_k denotes donors with copy number k and $\dot{\mathrm{X}}(\mathrm{x})$ is the signaling kinetic for single cells, $f'(t) \equiv \sum_k N_k'(\mathbf{x}_k, t) / N_{total}$ and

$$\eta \equiv \Gamma\left(\lambda\tau_{\min}(1+\alpha)^{-1}, (1+\alpha)^{-1}\right) / \Gamma\left(0, (1+\alpha)^{-1}\right)$$

Figure 3.5 was simulated with initial conditions of 1000 recipients and 4500 donors that were composed of 500 cells for each type of donor whose plasmid numbers varied between 1 and 9.

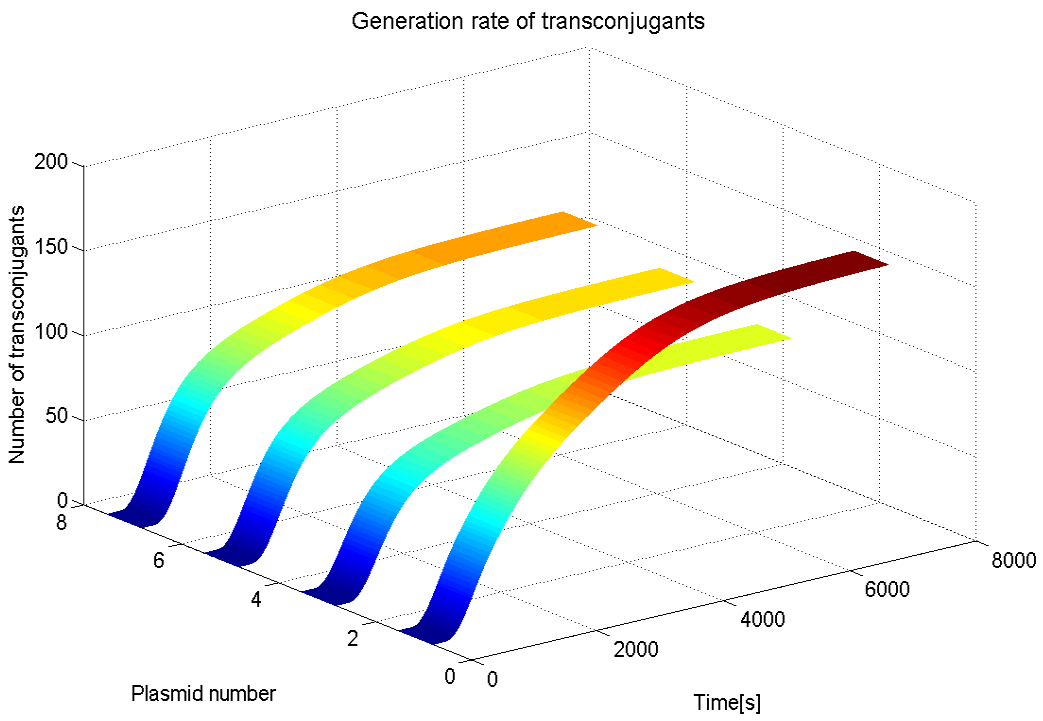

Figure 3.5: Generation rate of transconjugants

Off-state conditions are applied as initial conditions for the dynamic simulation. Even though the simulation is performed for plasmid numbers varied between 1 and 9, only results corresponding to plasmid numbers of 1,3 ,5 and 7 are shown above to avoid dense representation. The plot demonstrated generation rate of transconjugants as results of plasmid transfer from donors cells with plasmid number shown in y-axis. As expected, donor cells with a higher number of plasmids have a higher rate of plasmid transfer to recipients. This trend can be observed clearly from the figure, except for the case of 1 plasmid number. The generation rate of transconjugants due to donor cells with one plasmid is higher than the others because all new transconjugants will only have one plasmid and will contribute their new role as plasmid donors 15 minutes after their successful conjugation.

3.3 Chemical master equations vs. Langevin's equations:

Our new goal is to try to use stochastic models to investigate more rigorously the behavior of both donors and recipients in Planktonic and especially biofilms. Our previous system behaved with stability and little sensitivity, and so stochastic differential equations (Langevin equations) were developed and solved to get an approximate solution for our biological system. Current reactions with new assumptions have introduced new modifications which result in a much more challenging problem to solve. These difficulties come from an increase in total reactions, invalidation of some previous assumptions which can help simplify the formulation, and stiffness associated with cell responses to any environmental change. Solution methods for SDE are very specific and are typically rigorous and complicated. Our corresponding approach is to solve a more accurate version of the master equations using exact method [29], [41] or tau-leap methods [24], [27], [28], [30]. One important step prior to application of this methodology to our system is to ensure that the two methods produce similar results. Usage of Schlogl's system, well-known for its bistability, for simulation has proved their ability as an accurate substitution (Figure 3.6 and 3.7)

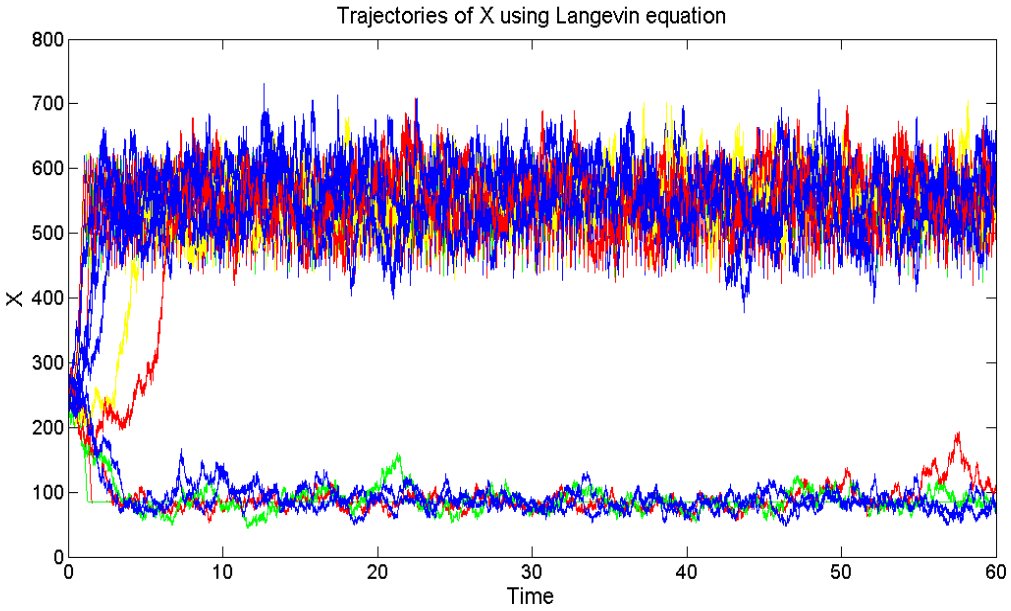

Figure 3.6: Stochastic simulation using Langevin equation

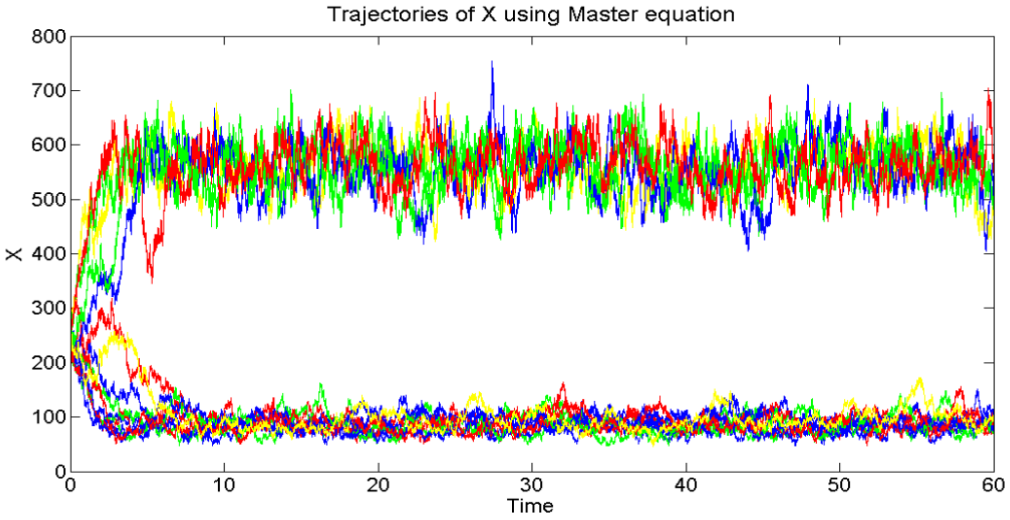

Figure 3.7: Stochastic simulation using exact method

Due to the nature of stochastic simulation, a great number of trajectories, also known as sample paths, needs to be generated and averaged to provide a good picture of the actual behaviors of the system. Even though in both Figure 3.6 and Figure 3.7, only twenty

trajectories are displayed, they already illustrate quantitatively their ability to replicate results for one another. A more rigorous comparison of the number distribution of X at t=60s was shown in Figure 3.8 below, where 10,000 sample paths were generated. Two curves overlapped throughout the entire distribution

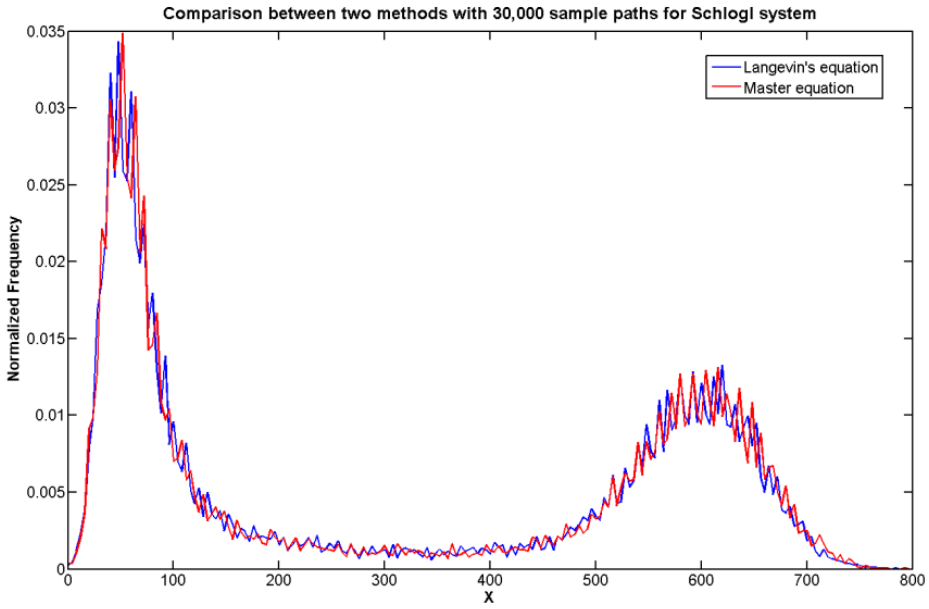

Figure 3.8: Distribution of X generated by two methods

This example is used to show that the equivalence of two methods with high number of trajectories can be guaranteed. This testing confirms no conflicts would arise between two approaches that our group has used in the previous work and the current work.

3.4 Supplementary:

Table 3-1: Reactions and kinetic parameters

	Reactions	Kinetic parameters	Reaction type
1	$X_4 + 4i \rightarrow X_4 i_4$	8.016x10^7 ($M^{-1}s^{-1}$)	1st order on i, 1st order on x_4
2	$X_4 + 4c \rightarrow X_4 c_4$	1.377x10^8 ($M^{-1}s^{-1}$)	1st order on c, 1st order on x_4
3	$B + X_4 \rightleftarrows BX_4$	F: **10^6** ($M^{-1}s^{-1}$) R: **10^{-1}**(s^{-1})	(parameters base on the lacI)
4	$B + X_4 i_4 \rightleftarrows BX_4 i_4$	F: **10^8** ($M^{-1}s^{-1}$) R: **10^{-3}**(s^{-1})	
5	$B + X_4 c_4 \rightleftarrows BX_4 c_4$	F: **10^9** ($M^{-1}s^{-1}$) R: **10^{-2}**(s^{-1})	
6	$\xrightarrow{Pq-Induced} Q_{pre} \quad (Induced : B)$	0.1 (s^{-1})	1st order on B
7	$\xrightarrow{Pq-Induced} Q_{pre} \quad (Uninduced : BX_4 c_4)$	0.1(s^{-1})	1st order on $BX_4 c_4$
8	$\xrightarrow{Pq-Uninduced} Q_{pre} \quad (Uninduced : BX_4)$	0.000723 (s^{-1})	1st order on BX_4
9	$\xrightarrow{Pq-Uninduced} Q_{pre} \quad (Uninduced : BX_4 i_4)$	0.000723 (s^{-1})	1st order on $BX_4 i_4$
10	$\xrightarrow{Px-Induced} Q_a \quad (Induced : B)$	0.00121(s^{-1})	1st order on B
11	$\xrightarrow{Px-Uninduced} Q_a \quad (Uninduced : BX_4 c_4)$	0.00121(s^{-1})	1st order on $BX_4 c_4$
12	$\xrightarrow{Px-Uninduced} Q_a \quad (Uninduced : \mathrm{B}X_4)$	0.00823(s^{-1})	1st order on BX_4
13	$\xrightarrow{Px-Uninduced} Q_a \quad (Uninduced : BX_4 i_4)$	0.00823(s^{-1})	1st order on $BX_4 i_4$
14	$Q_{pre} \longrightarrow Q_L *$	1(s^{-1})	(the time for transcript the length from Px to pass IRS1, roughly 1s)
15	$Q_{pre} + Q_a \longrightarrow Q_s$	4.43x10^9($M^{-1}s^{-1}$)	From previous Kq = 4.43(nM^{-1}); 4.43x(6 x10^8)/0.6, the latter is one reaction per second PS. 1nM ≈ 0.6 particle per cell for cell volume

16	$\xrightarrow{Q_s+Q_L} I_{ex}$	**0.5(s^{-1}) x delta**	**delta is volume conversion factor, x 10^{-3} for 10^9 cells/ml**
17	$I_{ex} \xrightarrow{uptake} i$	0.001(s^{-1})	
18	$C_{ex} \xrightarrow{uptake} c$	**0.001(s^{-1})**	
19	$\xrightarrow{Px-Induced} prgX \quad (Induced : B)$	0.000121(s^{-1})	1st order on B
20	$\xrightarrow{Px-Induced} prgX \quad (Uninduced : BX_4c_4)$	0.000121(s^{-1})	1st order on BX_4c_4
21	$\xrightarrow{Px-Uninduced} prgX \quad (Uninduced : BX_4)$	0.001021(s^{-1})	1st order on BX_4
22	$\xrightarrow{Px-Induced} prgX \quad (Uninduced : BX_4i_4)$	0.001021(s^{-1})	1st order on BX_4i_4
23	$\xrightarrow{prgX} X_2$	0.005(s^{-1})	
24	$2X_2 \rightleftarrows X_4$	F: **1×10^5** ($M^{-1}s^{-1}$) R: **0.01**(s^{-1})	

Table 3-2: Degradation rates for different species

Species	Degradation Rate (1/s)
X_4	1×10^{-5}
X_4i_4	1×10^{-5}
i	1×10^{-5}
X_4c_4	1×10^{-5}
c	1×10^{-5}
B	-
BX_4	-
BX_4i_4	-
BX_4c_4	-
Q_{pre}	-
Q_a	1×10^{-3}
Q_L	1×10^{-4}
Q_s	2×10^{-3}
I_{ex}	1×10^{-5}
C_{ex}	1×10^{-5}
$prgX$	2×10^{-4}

X_2	1×10^{-5}

Table 3-3: Range of parameters

	Reactions	Chosen values	Ranges
1	$X_4 + 4i \rightarrow X_4i_4$	$8.016\text{x}10^7$ ($M^{-1}s^{-1}$)	
2	$X_4 + 4c \rightarrow X_4c_4$	$1.377\text{x}10^8$ ($M^{-1}s^{-1}$)	
3	$B + X_4 \rightleftarrows BX_4$	F: **10^6** ($M^{-1}s^{-1}$) R: **10^{-1}**(s^{-1})	F: **10^6 – 10^8** ($M^{-1}s^{-1}$) R: **10^{-2}-10^{-1}** (s^{-1})
4	$B + X_4i_4 \rightleftarrows BX_4i_4$	F: **10^8** ($M^{-1}s^{-1}$) R: **10^{-3}**(s^{-1})	F: **10^7 – 10^8** ($M^{-1}s^{-1}$) R: **10^{-2}**(s^{-1})
5	$B + X_4c_4 \rightleftarrows BX_4c_4$	F: **10^9** ($M^{-1}s^{-1}$) R: **10^{-2}**(s^{-1})	F: **10^8 – 10^9** ($M^{-1}s^{-1}$) R: **10^{-2}**(s^{-1})
6	$\xrightarrow{Pq-Induced} Q_{pre}$ $(Induced : B)$	0.1 (s^{-1})	**0.05-0.15(s^{-1})**
7	$\xrightarrow{Pq-Induced} Q_{pre}$ $(Uninduced : BX_4c_4)$	0.1(s^{-1})	
8	$\xrightarrow{Pq-Uninduced} Q_{pre}$ $(Uninduced : BX_4)$	0.000723 (s^{-1})	**0.000723-0.0072(s^{-1})**
9	$\xrightarrow{Pq-Uninduced} Q_{pre}$ $(Uninduced : BX_4i_4)$	0.000723 (s^{-1})	
10	$\xrightarrow{Px-Induced} Q_a$ $(Induced : B)$	0.00121(s^{-1})	**0.00121/5-0.00121*5(s^{-1})**
11	$\xrightarrow{Px-Uninduced} Q_a$ $(Uninduced : BX_4c_4)$	0.00121(s^{-1})	**0.00121/5-0.00121*5(s^{-1})**
12	$\xrightarrow{Px-Uninduced} Q_a$ $(Uninduced : \mathrm{B}X_4)$	0.00823(s^{-1})	**0.00823/5-0.00823*5(s^{-1})**
13	$\xrightarrow{Px-Uninduced} Q_a$ $(Uninduced : BX_4i_4)$	0.00823(s^{-1})	**0.00823/5-0.00823*5(s^{-1})**
14	$Q_{pre} \longrightarrow Q_L$ *	1(s^{-1})	
15	$Q_{pre} + Q_a \longrightarrow Q_s$	$4.43\text{x}10^9$($M^{-1}s^{-1}$)	
16	$\xrightarrow{Q_s+Q_L} I_{ex}$	**0.5(s^{-1}) x delta**	**0.01-0.5**
17	$I_{ex} \xrightarrow{uptake} i$	0.001(s^{-1})	**0.0001-0.001**
18	$C_{ex} \xrightarrow{uptake} c$	**0.001(s^{-1})**	
19	$\xrightarrow{Px-Induced} prgX$ $(Induced : B)$	0.000121(s^{-1})	
20	$\xrightarrow{Px-Induced} prgX$ $(Uninduced : BX_4c_4)$	0.000121(s^{-1})	
21	$\xrightarrow{Px-Uninduced} prgX$ $(Uninduced : BX_4)$	0.001021(s^{-1})	
22	$\xrightarrow{Px-Induced} prgX$ $(Uninduced : BX_4i_4)$	0.001021(s^{-1})	

23	$\xrightarrow{prgX} X_2$	0.005(s^{-1})	
24	$2X_2 \rightleftarrows X_4$	F: **1×10^5** ($M^{-1}s^{-1}$) R: **0.01**(s^{-1})	**F:10^5 –10^8 ($M^{-1}s^{-1}$)** **R: 0.01-0.1(s^{-1})**

4. DYNAMICS OF TRANSFER OF DRUG RESISTANCE BETWEEN BACTERIAL SPECIES IN AN IDEALIZED BIOFILM

4.1 Introduction:

Enterococci have become the major contributors to nosocomial infections and are known to adapt to and resist antibiotics [42]. Moreover, with their special mobile genetic elements, rapid inducible horizontal conjugation allows them to transfer the plasmid pCF10(the resistance determinant) from donor cells to recipient cells[43]–[46].Cells communicate with one another via signaling molecules: inducer cCF10 and iCF10 produced by recipients and donors, respectively. Depend on the donor densities the conjugation operon is induced either at a high level or a low level. In the event that the donor population is sufficiently high, a calibrated response of reducing conjugation operon serves to increase the fitness of the population by saving energy expenditure [47], [48]. Past studies have characterized the genetic response at the single cell level as well as the population level where the well-mixed condition for environment is assumed, but there are no studies specifically aiming toward the direction of biofilm scenarios.

Research on modeling biofilms has been attracting attentions in the past few decades. However, the high level of complexity associated with biofilms has limited most past work to somewhat specific issues for specific applications. For example in the case of biofilm located in reactors, the layer model is commonly used [49]–[51]. Cellular automata [52]–[54] and/or particle models [55], [56] have been utilized to capture various biofilm characteristics such as biofilm thickness, density, and surface shapes. However, in the absence of mechanistic understanding, those models often lead users to modify

assumptions with some arbitrariness. We introduction the reaction-diffusion master equations (RDME as an expanded version of the Chemical master equation (CME) to capture behaviors in biofilms where both reactions and diffusion take place.

We use RDME as a base model to simulate biofilm in which both donors and recipients are present. The system features more than 40 reactions with diffusion occuring between neighboring areas. Scenarios with various initial conditions, numbers of donors and spatial arrangements of donor cells have been investigated. It emerges from that biofilms actually are more stable to consist of both donors and recipients, and spatial arrangement of donor cells can affect strongly affect the final distribution of each type of cell in the biofilm.

4.2 Diffusion reaction master equation and its application toward the biofilm case

The chemical master equation captures the fluctuations of small quantities existing in chemical/biological systems that are assumed to be well mixed. Due to this assumption, relative position of reacting molecules has no effect on the propensity (probability) of them reacting with one another. Hence, the application of CME holds only under well-mixed conditions [16] (i.e. diffusion occurs at a much higher rate than reactions in the system) Moreover, the RDME provides an additional capability of capturing diffusion effect in biological systems where macromolecule crowding and cell interactions are non-uniform throughout the entire spatial domain. The RDME has greater capturing capability since it incorporates the local effect in the mesoscopic limit. Validity of RDME has been proven under some specific conditions [33]–[35]. Solution of RDME was then proved to converge to that of CME if the system has (1) diffusion happens at a much higher rate

and (2) “convergent propensity functions.” [57], [58].In our specific biofilm, donor and recipient cells are fixed at each vortex in the 2-D domain, reactions take place within the donor cell and materials (extracellular C and I) are allowed to hop between neighboring vortexes. All reactions are modeled as elementary reactions and therefore, RDME can be utilized with the guarantee of its solution converging to that of CME.

4.2.1 The CME

Let us consider a well-mixed compartment of volume Ω in which N chemical species, $X_1, X_2, ..., X_n$ undergo R chemical reactions with the following form:

$$s_{1j}X_1 + ... + s_{Nj}X_N \rightarrow r_{1j}X_1 + ... + r_{Nj}X_N \quad , \ j^{\text{th}} \text{ reaction} \tag{21}$$

here r_{ij} and s_{ij} are the stoichiometric coefficients. The master equation for this stochastic system can be written as follows:

$$\frac{d}{dt}P(\vec{x},t) = \sum_{j=1}^{R}(\prod_{i=1}^{N}E_i^{s_{ij}-r_{ij}} - 1)\hat{a}_j(\vec{x},\Omega)P(\vec{x},t), \tag{22}$$

where $\vec{x}$ is the vector of molecule numbers of species $X_1, ..., X_n$, and $P(\vec{x},t)$ is the probability of the system in the state $\vec{x}$ at time t. $\hat{a}_j(\vec{x},\Omega)$ is the propensity function corresponding to j^{th} reaction. $E_i^{\Delta x}$ is the operator to replace x_i with $x_i + \Delta x$

4.2.2 The RDME

Let us consider the same chemical system above but the domain with volume Ω is divided into M voxels with a volume of Ω / M each. The j^{th} reaction taking place in a specific voxel can be written as:

$$s_{1j}X_1^k + ... + s_{Nj}X_N^k \rightarrow r_{1j}X_1^k + ... + r_{Nj}X_N^k \tag{23}$$

where X_i^k refers to species X_i in the voxel . Molecules are also allowed to exchange between neighboring voxels, modeling as follows:

$$X_i^k \underset{D_i'}{\overset{D_i}{\rightleftarrows}} X_i^{k'}, \qquad k' \in Ne(k) \tag{24}$$

where k' is located next to k (i.e. $Ne(k)$ is the set of all neighboring voxels of voxel k), and D_i is the diffusion rate of species X_i (which is spatially independent for any given species)

The stochastic dynamics of this system is described by the RDME [35], [58]

$$\frac{d}{dt}P(\vec{x}^1,...,\vec{x}^M,t) = \sum_{k=1}^{M}\sum_{j=1}^{R}(\prod_{i=1}^{N}E_{i,k}^{s_{ij}-r_{ij}}-1)\hat{a}_j(\vec{x}^k,\frac{\Omega}{M})P(\vec{x}^1,...,\vec{x}^M,t) + \\ + \sum_{k=1}^{M}\sum_{k'\in Ne(k)}\sum_{i=1}^{N}(E_{i,k}^{1}E_{i,k'}^{-1}-1)D_i x_i^k P(\vec{x}^1,...,\vec{x}^M,t) \tag{25}$$

where n_i^k is the molecule number of species X_i in voxel k, and $\vec{x}^k = (x_1^k,...,x_N^k)$. $P(\vec{x}^1,...,\vec{x}^M,t)$ is the probability of the system at state $(\vec{x}^1,...,\vec{x}^M)$ at time t. The first and second term on the right hand side of the equation above contribute to changes in the probability caused by reaction and diffusion respectively.

4.3 Results and Discussion

A unique feature of cell communication in the enterococcal pheromone system is the competition of two antagonistic peptide-signaling molecules: inducer peptide C and the inhibiting peptide I. Each peptide besides serving as an inducer or an inhibitor can also functions as quorum sensing which reflects density of each type within the populations. To understand the system with this level of complexity requires results gathered from both experiments and modeling simulations. Various experiments and studies of the donors' dynamic responses have been done but mostly in the planktonic environments where the extracellular environment is being controlled and monitored. This paper provides some

insight from a modeling perspective on donor dynamic responses as well as plasmid transfer in an environment where both donors and recipients are present. We investigate the plasmid transfer dynamics in various initial extracellular conditions as well as spatial configuration for donors in the population. The biofilm's domain is divided into 121 squared units. Donors and recipients are placed at those vertexes with a condition that each vertex can only have one cell. Each cell is modeled as a circle in the space and we assume the local concentration (the concentration measured in each square box around the circle) to be homogeneous. We allow particles to jump between neighboring units, and reactions only occur inside the donor cells. Recipient cells also constantly produce inducer C, whereas donor cells generate inhibitor I to the environment. Both extracellular C and I are allowed to diffuse around in the biofilm and enter donor cells to interact with the plasmid and cause downstream reactions which either induce or inhibit the conjugation process. To mimic the actual realistic case in vivo, we also allow the particles to transfer through the exterior boundary of the biofilm. Moreover, donor cells can only perform conjugation when Q_L level is higher than 10. Once the conjugation starts, if the recipient and the donor stay together for more than 15 minutes, the recipient is assumed to become a transconjugant(i.e. a new donor cell); otherwise, it will stay as a recipient. We consider below different initial scenarios for our simulations

4.3.1 Single donor cell located at the center and surrounded by recipients

In this configuration, the donor is fixed at the center of the biofilm surrounded by all recipients. The dynamic results for all cells in the biofilm are collected as the average values from 1000 independent sample paths, each of which is a full simulation for the biofilm from t=1 to t=20 hrs. For convenience, we number the cells from left to right and

bottom to top; therefore, the initial donor cell is cell 62. Since we only model reactions happening in the plasmid of the donor cells, behavior of a recipient cell can only be evaluated once it turns into a donor cell. Fig. 4.1 shows the distributions of conversion time for recipients 71, 85 and 109 being converted into donor cells.

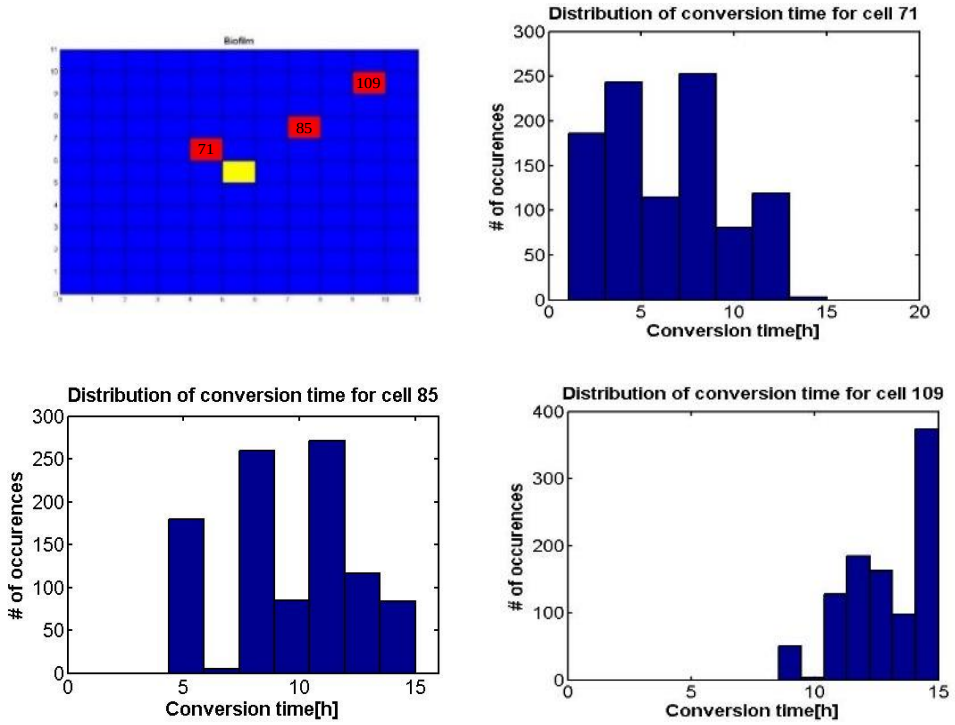

Figure 4.1: Distribution of conversion time of different recipients

Even though the three cells were selected randomly, distances between them and the center cell are different. As one may expect, recipients that are located farther away from the center donor cell are converted later than the ones that are near the center cell. We also evaluate the Q_L levels of the center donor cell versus the cell 71 known as a recipient initially and converted to a donor cell later. Figure 4.2 below indicates the average dynamic change of Q_L level of the two cells:

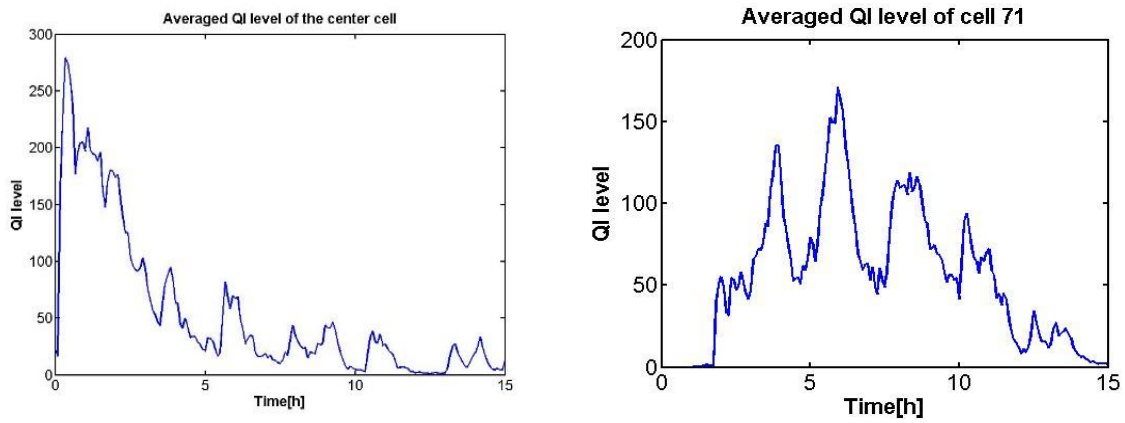

Figure 4.2: Comparison in Q_L level between two cells (L) Center cell and (R) Cell 71

The two graphs in Figure 4.2 share a common trend of Q_L in that it increases at first to a peak value and thereafter declining progressively until a near-zero value. The explanation of lies in all donor cells being initially uninduced and remaining so until encountering the inducer C, originating from the recipient cells, following which Q_L increases quickly.As the donors consume the local inducer C they also produce inhibitor to the environment, which reenter the donor to diminish the Q_L level and shutting down induction in the process. Moreover, the concentration of local C becomes low, acting as a sink. Areas where concentration of C is high serve as sources and cause C diffuse toward the low concentration area. Due to that effect, Q_L fluctuates back up a little bit, and that explains why we observe the fluctuation in Q_L level in both cases. Eventually as more recipients are

being converted into donors, production of C becomes less, more I is being generated, and hence Q_L keeps diminishing until reaching zero. The fluctuation observed in Q_L dynamics represent effects coming from both reaction and diffusion. The only difference between the two graphs comes from the fact that cell 71, unlike cell 62, is a recipient at first. It was not being converted into a donor cell until around 2 hours. In addition, after it becomes a donor, the overall amount of inducer C in the biofilm is a lot lower than it was initially because there are fewer recipients and more donors at this point. For that reason, the peak value of Q_L is only around 170 whereas it is around 270 for the center donor cell. At the end of 15 hours, all the recipient cells are being converted into donor cells. We do realize that as more recipients become donors, less C will be produced while more I will be generated. With this explanation, we also would expect to see the rate at which the plasmid being transferred from the donors to recipients goes down but from this case Q_L level just does not go down quickly enough to shut down the conjugation before it happens. Various generation rates of C and I have been examined (not shown in this paper), simulations conclude similar results. Rate of spreading change accordingly to the given conditions but all recipient cells are being converted into donor cells at the end of the simulation. Also, for the actual application of controlling/interfering this plasmid transfer as well as controlling this phenomenon, we also posit to ourselves the question if we would be able to control this spreading by alternating extracellular conditions as well as spatial cell configuration.

4.3.2 Multiple donor cells with two different spatial cell configurations:

To investigate how the donor density and its spatial variation would affect the spreading event, we simulated the biofilms in two different scenarios where different spatial

arrangements for 9 donor cells are tested. In these configurations, donors are located either together in the center part of the biofilm or distributed throughout the biofilm. The first configuration and Q_L level of a recipient close to the boundary are shown in Figure 4.3 below:

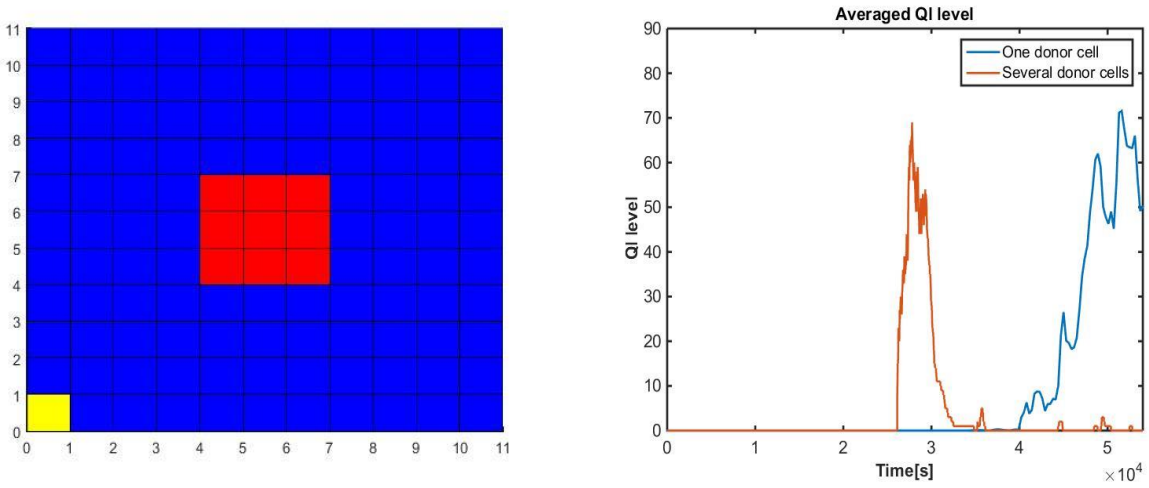

Figure 4.3: Dynamic Q_L of the "initial" recipient cell 1

In the figure on the left, red cells indicate donor cells whereas recipient cells are in blue at time t=0. The yellow is also a recipient but chosen in special to trace its Q_L level with time. In the figure to the right, the red curve shows the behavior of yellow cells in the previous configuration where only one donor cell is placed at the center of the biofilm. The red curve also describes Q_L of the yellow cell but in the configuration of the biofilm, which is shown in the left figure. Clearly, in the case of a single initial donor cell, cell 1 becomes a transconjugant much later than in the case of multiple donor cells initially, but being activated(Q_L >10) as well as can potential perform conjugation with another recipient for a lot longer period as well. In the case of multiple donor cells initially, each of them can participate in conjugation with neighboring recipients and therefore will reach to cell 1 a

lot earlier than the previous case. However, multiple donors initially can also mean quicker buildup of inhibitor I and faster consumption of C, which results in Q_L dropping under the limit for conjugation(Q_L <10). As the activated period shrinks down, probability of that donor cell participating in a conjugation with its neighboring recipient also reduces significantly. From the figure on the right, it is clearly evident that cell 1 in yellow as well as other cells close to the boundary also become donors in this configuration.

In the second spatial configuration, 9 donor cells shown in red are distributed throughout the biofilm. The specific location of each is shown here in Figure 4.4

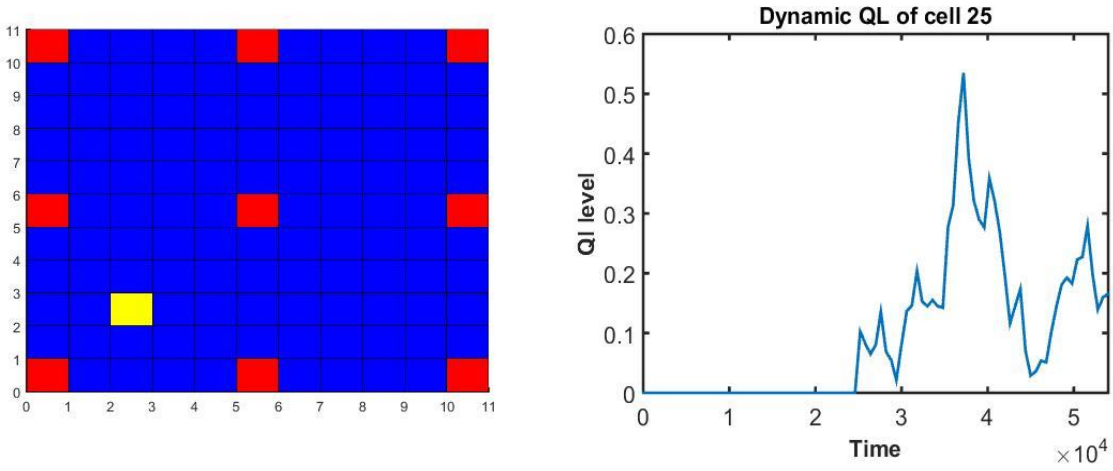

Figure 4.4: Average Ql of cell 25 (a recipient at initial simulation time) in yellow shown in the left

In this case, 9 donor cells shown in red are distributed throughout the biofilm. We selected recipient cell 25 shown in yellow to observe its dynamic behavior. It is important to note that we construct/develop reaction system only for donor cells and can only track the correponding variable once it is a donor cell. At the starting point of simulation, cell 25 is a recipient cell and can be viewed the turning point where it becomes donor once Q_L starts

responding to go above zero. As we observed from the previous case, once a recipient turns into a donor cell, the Q_L level should increase to a relatively much larger number than in this case where Q_L level only fluctuates between 0 and 1. Since the result shown above is computed as an average value from all sample path, the result positively suggests that in almost every sample paths, cell 25 stays as a recipient throughout the entire course of simulation. There is a very slim chance that it may actually be converted to a donor cell. To complete the actual evaluation, let us also take a look at its correponding intracellular/extracellular levels of cCF10 and iCF10:

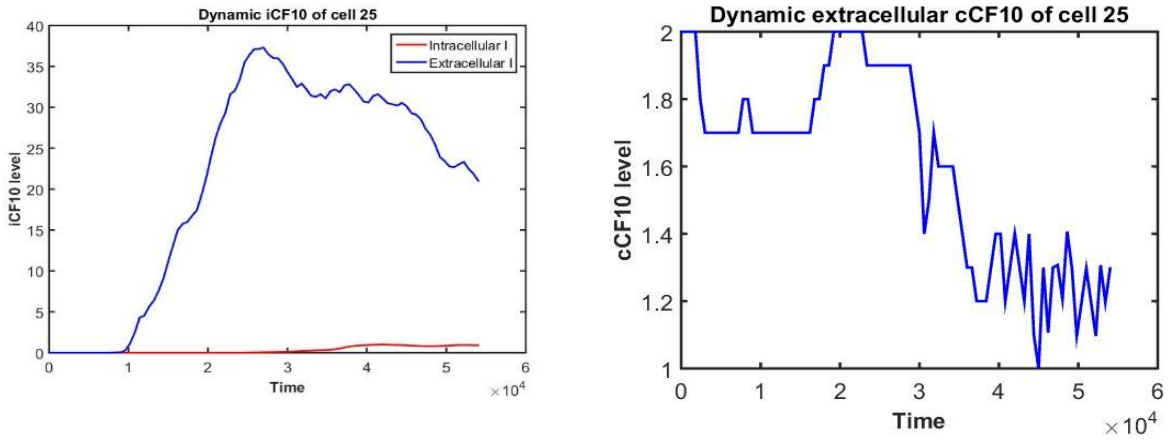

Figure 4.6: Dynamic signaling behavior of Cell 25

The plot on the left includes two curves of intracellular and extracellular inhibitor I associated with cell 25. The only possible way cell 25 can be a donor but has a low level of Q_L is when the intracellular inhibitor is high and no inducer C is around. The figure on the right shows the availability of C in the environment. Moreover, the red curve in the left figure indicates a low level of I which also serves well as more evidence that cell 25 in most sample paths stays as a recipient throughout the entire simulation. Let us explain why

we have a mixture of both donors and recipients at the end of the simulation, which is not observed in the other configuration, even though we have an exact number of donor cells at the beginning of our simulations. We have discussed 3 distinctive cases here in our studies. In the case of the 9 donor cells staying together close to the center, the spreading actually happens a lot quicker but then down more rapidly than in the case of a single donor cell. This illustrates the concept that cells seem to preserve an environment where several species coexist. As more recipient cells become donors, plasmid transfer occurs less frequently. Evidently, the shut-down is slowing down yet is not quick enough to prevent the plasmid transfer from happening. However, in the last configuration, donor cells are located at several different locations and so the spreading starts from multiple directions rather than just one direction from the center to the exterior side observed in the previous cases. Specifically, around cell 25, the local concentration of inhibitor goes up quickly due to various supplies from multiple donor sources nearby. This diffusion effect allows inhibitor I to build up quickly and to be able to overcome the inducer effect from C. The effect of I in this case dominates that of inducer, leading to a great reduction in Q_L of donors that surround cell 25 and math them unable to participate in conjugations with cell 25.

4.4 Conclusions

In this study, we have investigated the biofilm dynamics of drug resistance associated with plasmid transfer using the RDME approach in which reactions and diffusion can be modeled together. Several studies for this system have been done but focus mostly on the planktonic case at both single cell and population levels. This study provides some useful insights on drug transfer dynamics in a much more complex matrix-like system where local

concentration varies throughout the domain of biofilm. In the conditions where we have both donor and recipient cells, one would expect the drug transfer would happen all the way due to the higher survivability of cells if they carry genes which strengthen their ability to survive against antibiotics. We have simulated biofilms in various conditions where generation rates, reaction rate, donor density and spatial configurations are varied. From all the simulations, the results suggest that cells in biofilm seem to prefer the existence of both donors and recipients and do not necessarily want to acquire the ability to resist to antibiotics at all times. In all cases, spreading is slowing down as more recipients become donors. However, the only case which produces final mixture of both donors and recipients at the end of the simulation is when donors are located at multiple different areas. This arrangment allows certain areas in the biofilm to build up inhibitors so quickly they are able to shut down the conjugations and so prevent the full conversion of all recipients to donors. This comes from cell perspective to minimize energy/resourse use. Conjugation and plasmid transfer require a great deal of energy expenditure and so therefore, cells do not necessary need to partake in this process unless there is a threat for their survivability. This finding also provides us a solution to interfere/control the drug resistance transfer. An equivalence of this idea for us is to apply, if possible, is to infuse supply of inhibitor I into the biofilm from various point in space through the surface. Further studies on controling I concentraion as well as selecting the right spatial configurations of cells(i.e. where to supply inhibitors to different points on the biofilms) are needed and can be tested and verified throughout both simulation and experimental tools. Also, carrying out the same studies for cells that are pre-exposed to the antibiotic might also produce different outcomes. Studies of this type will also be a part of extensive future work.

5. A NEW "TAU-LEAP" STRATEGY FOR ACCELERATED STOCHASTIC SIMULATION

5.1 Introduction:

In recent times, stochastic simulations have assumed extraordinary importance in modeling biological and nano systems. The importance of the Monte Carlo simulation methodology arises from its capacity to substitute for the less tractable Master Equation and mean field approaches. This capacity is derived from its basic simplicity which could, however, be seriously compromised if computational times become forbiddingly large because of the repetitive attribute of the methodology. Consequently, considerable effort has come to pass on finding efficient simulation strategies so that computation of statistical averages of stochastic behavior can become facile. The issue of efficiency arises in negotiating accuracy with sacrifice of rigor or through crafty approximations. In stochastic systems, however, the validity of approximations is uncertain because they invariably involve future values of random variables in the system leading to conservative and hence less efficient heuristics. The remedy must lie in the use of mathematical propositions concerning probabilities with which approximations about random variables are true.

It is the purpose of this paper to aid in the quest for efficient stochastic simulation strategies by exploiting an important inequality that could greatly rationalize the process. This is the Chebychev's inequality that is routinely encountered in the treatment of random variables [57]. The inequality is concerned with the probability with which a random variable deviates from its expectation in terms of its variance.

We are concerned here with a stochastic system of chemical transformations involving several species. The stochastic behavior arises out of the randomness in reaction rates and is reflected in the numbers of different reaction species at different times. Simulation of this system can be performed with an *exact* algorithm that has come to be known as SSA (stochastic simulation algorithm). The term "exact" is derived from the fact that the simulation is based on rigorously derived distributions of "waiting times" during which the system remains "quiescent" without any reaction event. While the methodology is usually attributed to Gillespie [29], a basic version of the idea appeared in a paper by Kendall [59] who used the term "quiescent interval" instead of "waiting time". This method was generalized for the simulation of population balance equations in a doctoral thesis by Shah, subsequently published in the chemical engineering literature [23] independently of Gillespie's publication in the geophysical literature. Some further insights as to the connection of the foregoing algorithm to the master equation for population balances are given by Ramkrishna [60]

The virtue of SSA's exactness is computational accuracy but at the cost of large computation times. SSA also provides the starting point for various improvements pursued by researchers in this area to minimize the cost of computation. The term "tau-leap", coined by Gillespie and coworkers, aptly describes the underlying strategy of "leaping" over relatively less significant events (as reflected by minimal changes in the so-called "propensities" or transition rates of individual reactions). For all of the methods discussed in this paper, each leap is checked after each calculation. Anderson's method [61] is a pre-check method, and hence produces a very different mechanism of filtering leaps. For any given selection of ε, post-check methods inevitably encounter some ineffective leap steps

at which the leap condition is violated. Depending upon the number of violations and the extent of each violation, the accuracy of the simulation will be affected. Our method introduces a way to promote the likelihood of acquiring more efficient leap steps by appropriate increase of δ. Anderson's method introduces a procedure to do pre-check before performing calculation of leaps. Only steps where the criteria are satisfied will continue with the calculation of the corresponding leaps. Moreover, only the first step of picking τ will follow the formula of Cao et al.[1] The subsequent steps of evaluating τ strictly follow their procedure. For each new calculated τ, it is then evaluated with the leap condition. Depending upon the value of τ, different scaling factors can be applied to adjust τ. If a leap for $\varepsilon = \varepsilon'$ is rejected, τ will be decreased by multiplying with some $p < 1$. If a leap is accepted for $\varepsilon = \varepsilon'$ but would fail if $\varepsilon = 3\varepsilon'/4$, it will be reduced by multiplying it by some p^* that satisfies $p < p^* < 1$. If a leap is accepted for both cases, τ can be enlarged by raising it to the power of q such that q is between 0 and 1. Thus, the algorithm for pre-check will only accept those τ that satisfy the leap condition. However in stochastic simulations, good results can still be accomplished, as long as the number of unsatisfied leaps is low. Every time a leap is rejected, a new calculation will need to be made; therefore the question is a balance between accuracy and speed. Anderson's approach focuses on checking the leap condition, whereas all other post-check methods focus on the issue of how to select a τ value. Therefore, the method cannot be included in our comparison of post-check methods with the approach of this paper. For that reason, this paper will exclusively focus on comparison among post-check methods. In this paper, we show how the Chebyshev inequality can be used to craft this temporal leap by specifying the probability with which it influences the deviations in the stochastic numbers

of chemical species from their expected values. In so doing, following Gillespie and Petzold[9], we retain the Taylor series expansion of the propensities about the state of the system specified at instant *t* truncated beyond the quadratic terms. A new algorithm is derived simply using different probabilistic criteria for approximations.

5.2 Development of the New "Tau-Leap" Strategy:

As observed earlier, we adhere to the stochastic system formulated by Gillespie and Petzold[22]. Thus, there are N chemical species in numbers represented by the random vector $\mathbf{X}$, involved in M transformations (reactions) with the vector $\boldsymbol{\nu}_j \equiv \{\nu_{ij}, j = 1,2,\mathrm{K}, N\}$ denoting the change in the numbers of species due to each j^{th} reaction event. Further, the reaction propensity vector, conditional on $\mathbf{X} = \mathbf{x}$, is represented by $\{a_j(\mathbf{x}); j = 1,2,\mathrm{K}, M\}$. If at time t, $\mathbf{X}(t) = \mathbf{x}$, then after time τ thereafter (i.e., $\tau > 0$), we have from Gillespie and Petzold[22]

$$\mathbf{X}(t+\tau) - \mathbf{x} \equiv \Lambda(\tau;\mathbf{x}) = \sum_{j=1}^{M} K_j(\tau;\mathbf{x}) \tag{26}$$

where $K_j(\tau;\mathbf{x})$ represents the number of j^{th} reaction events during the interval from t to $t+\tau$. For the formulation of $\tau-$leap, $K_j(\tau;\mathbf{x})$ is assumed to be a Poisson process $\mathcal{P}(a_j(\mathbf{x});\tau)$ with mean and variance given by $a_j(\mathbf{x})\tau$, notwithstanding the change in $\mathbf{X}$ during this time interval. Thus (26) is replaced by

$$\Lambda(\tau;\mathbf{x}) = \sum_{j=1}^{M} \mathcal{P}(a_j(\mathbf{x});\tau)\,\boldsymbol{\nu}_j \tag{27}$$

The "leap condition" of Gillespie and Petzold is given by

$$\left|a_j\left(\mathbf{x}+\Lambda(\tau;\mathbf{x})\right)-a_j(\mathbf{x})\right|\le \varepsilon a_o(\mathbf{x}),\quad a_o(\mathbf{x})\equiv\sum_{j=1}^{N}a_j(\mathbf{x}) \tag{28}$$

where the symbol ε is to be distinguished from that appearing in the Chebyshev inequality. Other replacements considered by these authors for $a_o(\mathbf{x})$ do not affect the treatment of this paper. Defining

$$f_{jj'}(\mathbf{x})\equiv\sum_{i=1}^{N}\frac{\partial a_j(\mathbf{x})}{\partial x_i}\nu_{ij'},\quad j,j'=1,2,\mathrm{K},M, \tag{29}$$

$$\mu_j(\mathbf{x})\equiv\sum_{j'=1}^{M}f_{jj'}(\mathbf{x})a_{j'}(\mathbf{x}),\ \sigma_j^2(\mathbf{x})\equiv\sum_{j'=1}^{M}f_{jj'}^2(\mathbf{x})a_{j'}(\mathbf{x}),\quad j=1,2,\ldots,M \tag{30}$$

The change in propensity during the time interval between t and $t+\tau$ is given by

$$a_j\left(\mathbf{x}+\Lambda(\tau;\mathbf{x})\right)-a_j(\mathbf{x})\equiv\Delta a_j(\tau;\mathbf{x})\approx\sum_{j'=1}^{M}f_{jj'}\mathcal{P}\left(a_{j'}(\mathbf{x});\tau\right) \tag{31}$$

The random variable $\Delta a_j(\tau;\mathbf{x})$, in view of the approximation in (31), has expectation $E\Delta a_j(\tau;\mathbf{x})\equiv\mu_j(\mathbf{x})\tau$, and variance $V\Delta a_j(\tau;\mathbf{x})=\sigma_j^2(\mathbf{x})\tau$. Gillespie and Petzold choose the interval τ in accord with

$$\tau=\min_{j=1,2,\ldots,M}\left\{\tau_j\right\},\quad \tau_j\equiv\min\left(\frac{a_0\varepsilon}{|\mu_j|},\frac{a_0^2\varepsilon^2}{\sigma_j^2}\right) \tag{32}$$

We now consider the application of the Chebyshev inequality to the random variable $\Delta a_j(\tau;\mathbf{x})$ towards determining a suitable criterion that will ensure the leap condition (28). We observe first that the triangular inequality implies that

$$\left|\Delta a_j(\tau;\mathbf{x})\right|\le\left|\Delta a_j(\tau;\mathbf{x})-\mu_j(\mathbf{x})\tau\right|+\left|\mu_j(\mathbf{x})\right|\tau \tag{33}$$

By imposing that

$$\left|\Delta a_j(\tau;\mathbf{x})-\mu_j(\mathbf{x})\tau\right|+\left|\mu_j(\mathbf{x})\right|\tau\le\varepsilon a_o(\mathbf{x}) \tag{34}$$

inequality (33) implies the leap condition (28). From (34), it follows that

$$\left|\Delta a_j(\tau;\mathbf{x})-\mu_j(\mathbf{x})\tau\right|\le\varepsilon a_o(\mathbf{x})-\left|\mu_j(\mathbf{x})\right|\tau \tag{35}$$

The above inequality does require that the right hand side be positive so that

$$\tau<\frac{\varepsilon a_o(\mathbf{x})}{\left|\mu_j(\mathbf{x})\right|} \tag{36}$$

to the right of which is a potential candidate for τ as prescribed by Gillespie and Petzold[24]

Invoking the Chebyshev inequality (2) by replacing the ε which appears there by $\left(\varepsilon a_o(\mathbf{x})-\left|\mu_j(\mathbf{x})\right|\tau\right)$, we obtain

$$\Pr\left\{\left|\Delta a_j(\tau;\mathbf{x})-\mu_j(\mathbf{x})\tau\right|<a_o(\mathbf{x})\varepsilon-\left|\mu_j(\mathbf{x})\right|\tau\right\}>1-\frac{\sigma_j^2(\mathbf{x})\tau}{\left[a_o(\mathbf{x})\varepsilon-\left|\mu_j(\mathbf{x})\right|\tau\right]^2} \tag{37}$$

Since the right hand side of (37) must be non-negative, it is readily found that an upper bound for τ is $a_0^2\varepsilon^2/\sigma_j^2$, thus recovering the second candidate for tau-leap due to Gillespie and Penzold. However, we produce a more effective choice of tau-leap in this development.

If we let the right hand side of (37) be appropriately close to unity, a choice becomes available for τ. Thus setting its value be equal to δ (e.g., $\delta=0.95$), we have (dropping henceforth the argument $\mathbf{x}$ for notational simplicity)

$$\tau^2 - 2\left[\frac{a_0\varepsilon}{|\mu_j|} + \frac{1}{2(1-\delta)}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2\right]\tau + \left[\frac{a_o\varepsilon}{|\mu_j|}\right]^2 = 0 \tag{38}$$

which obtains the following solution for τ_j, the subscript referring to its being specific to the j^{th} reaction.

$$\tau_j = \frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2(1-\delta)}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2 \pm \sqrt{\left[\frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2(1-\delta)}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2\right]^2 - \left[\frac{a_o\varepsilon}{|\mu_j|}\right]^2} \tag{39}$$

The positive solution is in conflict with (36) so that we must seek only the smaller root with the negative sign which is indeed less than $a_o(\mathbf{x})\varepsilon / |\mu_j(\mathbf{x})|$. Thus we have

$$\tau_j = \frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2(1-\delta)}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2 - \sqrt{\left[\frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2(1-\delta)}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2\right]^2 - \left[\frac{a_o\varepsilon}{|\mu_j|}\right]^2} \tag{40}$$

which is of course consistent with (36). We conclude from this analysis that the required τ is given by

$$\tau = \min_{j=1,2,K,M}\{\tau_j\} \tag{41}$$

which assures the required leap condition (28) with probability of *at least* δ.

It is next of interest to compare the development here with the Gillespie-Petzold solution (hereafter referred to as GPS). As (36) represents a common requirement, we only need consider the other alternative $\varepsilon^2 a_o^2 / \sigma_j^2$ for τ_j with that arising from expression (40). In passing, the Cauchy-Schwartz inequality (which states that the absolute value of the inner product between any two vectors is at most equal to the product of their norms) can be used on (30) to show that

$$\left|\mu_j\right|^2 \le \sigma_j^2 a_0 \tag{42}$$

which serves to further calibrate the plot of δ versus τ. We return to compare our prescription for τ_j to that by GPS. Suppose a specific value of ε is chosen. If we begin with choosing

$$\varepsilon > \frac{\sigma_j^2}{a_0\left|\mu_j\right|} \tag{43}$$

which implies that $a_0\varepsilon/\left|\mu_j\right| < \varepsilon^2 a_o^2/\sigma_j^2$, so that from GPS $\tau_j = a_0\varepsilon/\left|\mu_j\right|$. For this ε, Figure 5.1a shows the layout of δ and τ and thus the extent to which they may be negotiated for an efficient simulation as guided by Chebyshev's inequality. In particular, it would appear that τ_j should in fact be safely less than $a_0\varepsilon/\left|\mu_j\right|$ (instead of being equal to it as in GPS) so that δ has a suitably *positive* value. Of course the resulting tau-leap would be lowered but sample paths generated of the process would more reliably satisfy the tau-leap condition (28) which is indicative of higher simulation efficiency. It is noteworthy that the GPS choice is not backed by the probabilistic assurance of satisfying the tau-leap condition (28). If, on the other hand, $\varepsilon < \sigma_j^2/a_0\left|\mu_j\right|$, the GPS prescription would call for $\tau_j = a_0^2\varepsilon^2/\sigma_j^2 < \sigma_j^2/\left|\mu_j\right|^2$; furthermore, it can be shown (Supplementary) that $\tau_j > \tau_0$, where τ_0 is obtained from (39) by setting $\delta = 0$; thus the $\delta - \tau$ scenario would change to that shown in Figure 5.1b.

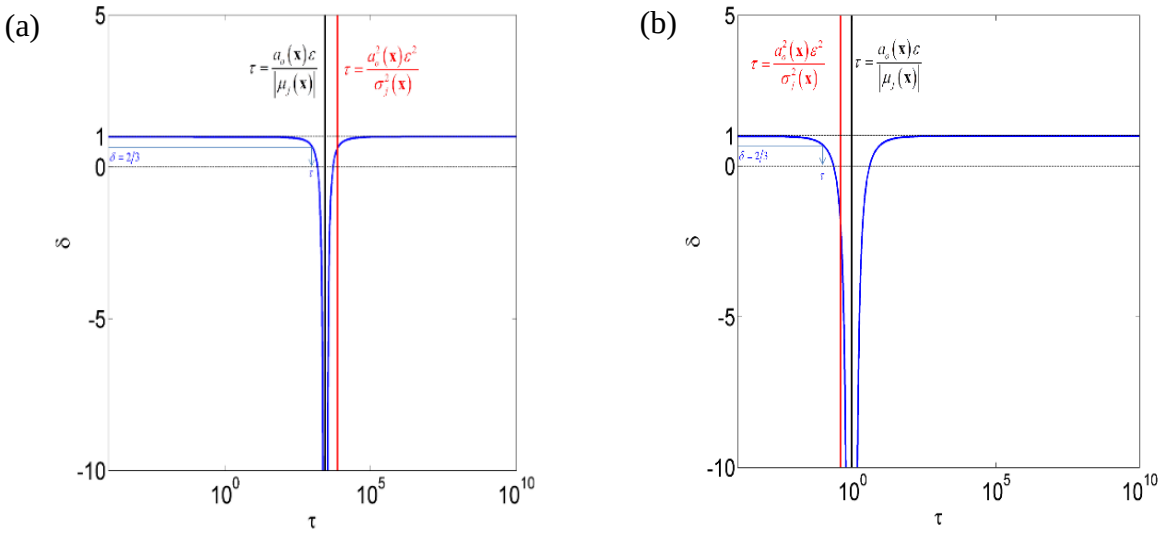

Figure 5.1: δ vs. τ_j (indicated as τ). (a) The case when the tau-leap condition is satisfied for the case $\varepsilon > \sigma_j^2 / a_0 |\mu_j|$. The GSP solution for τ_j appears at the asymptote. (b) the case when the tau-leap condition is satisfied for the case $\varepsilon < \sigma_j^2 / a_0 |\mu_j|$. The GPS obtains τ_j in the region for which $\delta < 0$.

Thus regardless of the choice of τ_j in(32), Figure 5.1b locates it in the domain of *negative* δ thus denying any probabilistic assurance that the $\tau-$leap condition (28) would be satisfied. It does not preclude, however, the generation of meaningful sample paths for the process. As long as τ is chosen to satisfy (36), there is no explicit restriction on the choice of ε, although its smallness would indeed govern the quality of the tau-leap criterion. Throughout the simulation, both the method of Cao et al. as well as the current method encounter a number of defective steps at which the leap condition fails. As the number of these wasteful steps increases, the accuracy will be negatively influenced (i.e. it will require a higher number of sample paths to accomplish a fixed accuracy). Cao et al. proposed

procedure to generate leaps but never proved mathematically how it can be satisfied. Hence, some generated leaps might violate the leap condition badly and affect strongly the overall quality of simulation. One way to avoid this problem and therefore to optimize the process is to increase the probability of getting a good selection of leaps. Two different ways which can accomplish this goal are as follows. Clearly, smaller values of ε will be progressively conservative and hence less effective in reducing computation time. On the other hand, if we fix ε, the use of the Chebyshev inequality may be regarded as a way to reduce the number of wasteful simulations that violate the tau-leap criterion with the stipulated ε. From (41), it is transparent that, for fixed ε, as $\delta \to 1,\ \tau \to 0$, which would be undesirable. Higher values of τ can be chosen by negotiating the value of δ to be suitably less than 1. While it may at first seem that the lower leaps in our approach may, not surprisingly, lead to more accurate solutions, it must be understood that the methodology lies in its efficiency in that a solution of given accuracy is obtained with a smaller number of simulations. The proposed method thrives in the facility to manipulate both the parameters ε and δ to promote efficiency. The computational demonstration to follow would of course confirm the foregoing observations.

The development of this paper is readily adapted to the new $\tau-$leap method presented by Cao et al. [24], who relate the relative change in propensity to that of the stochastic state variables involved. For example, if $a_j(\mathbf{x})$ involves a first order reaction in the i^{th} species (alone) we have

$$\frac{\Delta a_j}{a_j} = \frac{\Delta x_i}{x_i} \tag{44}$$

For the foregoing case, the $\tau-$leap condition in the j^{th} propensity translates to $|\Delta X_i| \le \varepsilon x_i$ in state variable domain. For more general cases, Cao et al.[1] obtain the $\tau-$leap condition in the state variables as

$$|\Delta X_i| \le \max\left\{\frac{\varepsilon x_i}{g_i}, 1\right\} \tag{45}$$

where $g_i\ (\ge 1)$ represents the highest order of reaction with respect to species i. Note that (44) implies a change of at least one molecule; for an explanation of this and other aspects of this algorithm, the reader is referred to Cao et al[24]

The application of our methodology leads to the following expression for τ which will satisfy the leap condition for $\mathbf{X}$ with probability *at least* δ.

$$\tau = \min_{i=1,2,K,N}\left\{\frac{\alpha_i}{|\hat{\mu}_i|} + \frac{1}{2(1-\delta)}\left\{\frac{\hat{\sigma}_i}{|\hat{\mu}_i|}\right\}^2 - \sqrt{\left[\frac{\alpha_i}{|\hat{\mu}_i|} + \frac{1}{2(1-\delta)}\left\{\frac{\hat{\sigma}_i}{|\hat{\mu}_i|}\right\}^2\right]^2 - \left[\frac{\alpha_i}{|\hat{\mu}_i|}\right]^2}\right\} \tag{46}$$

where $\alpha_i \equiv \max\{\varepsilon x_i / g_i, 1\}$, $\hat{\mu}_i(\mathbf{x}) \equiv \sum \nu_{ij} a_j(\mathbf{x})$, and $\hat{\sigma}_i^2(\mathbf{x}) \equiv \sum \nu_{ij}^2 a_j(\mathbf{x})$

The efficacy of the proposed algorithm is shown by comparing it with that of Cao et al.[20]

The stochastic algorithm is usually employed to obtain the average behavior of the system for a chosen time interval by averaging several sample path simulations using the algorithm. The SSA[22], [23] serves as a benchmark in evaluating the accuracy of any algorithm by comparing the histogram obtained by sample path averaging with that of SSA. In this regard, we use the same metric as that used by Cao and Petzold[62]

The comparison between the algorithm presented in this work with others have been made in two different ways. Thus we compare the accuracies obtained for a fixed

computation time, which is more focused on accuracy than computation time. Alternatively, where accuracy is prescribed, computation times can be compared for any two algorithms.

We employ two different examples for our demonstration. They are (i) The Schlogl model noted for its bistability and (ii) A linear consecutive reaction system. The algorithms used for comparison are those due to Cao et al. [20], Gillespie's midpoint Poisson tau-leap method [25], [63], and the bionomial leap method [27], [28] subsequently improved by Peng et al. [26]

5.3 Examples:

Example 1: We consider the consecutive linear reaction system as in (47) below:

$$X1 \xrightarrow{c_1} X2 \xrightarrow{c_2} X3 \tag{47}$$

We apply our tau selecting method with the formula (45) for the generation of τ. With rate constants $c_1 = 1$ and $c_2 = 1$ and initial conditions $X1 = 10^4$, $X2 = 1$ and $X3 = 0$, we calculate $X3$ until time is equal to 0.1.

Example 2*:* Schlogl's chemical reaction model is shown in (48) below. This model is noted for its bistability. B_1 and B_2 are constants with their particle numbers as N_1 and N_2

$$B_1 + 2X \underset{c_2}{\overset{c_1}{\rightleftharpoons}} 3X, \quad B_2 \underset{c_4}{\overset{c_3}{\rightleftharpoons}} X \tag{48}$$

The values of parameters, adapted from Cao et al.[1], are $c_1 = 3\times10^{-7}$, $c_2 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 3.5$, $N_1 = 1\times10^5$, and $N_2 = 2\times10^5$. The initial condition of X is 250 and we simulate the system up to $t = 4$.

5.4 Discussion of Results:

Figure 5.2 shows the probability density for $X3$ at $t = 0.1$ for SSA, the present method and the Binomial method of Peng et al[26], using an $\varepsilon = 0.02$

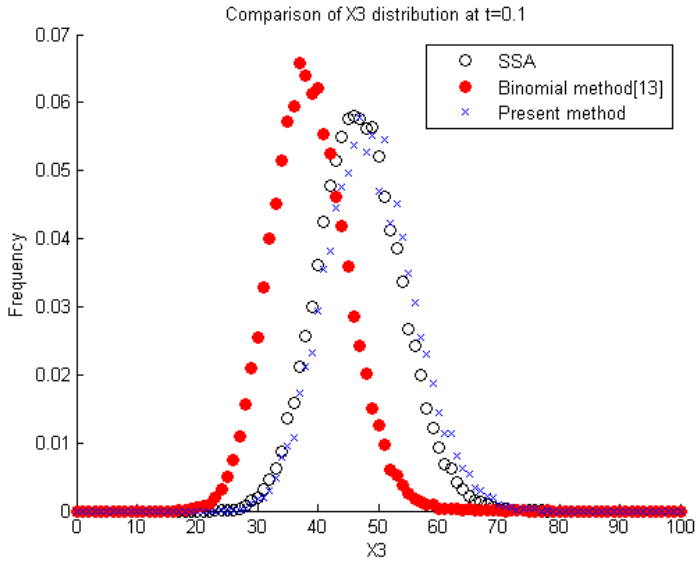

Figure 5.2: Comparison of accuracy for the histogram between two methods: Binomial method [24] and the present method

The higher and close proximity of the distribution by the present method to the SSA is clearly evident. Table 1 shows the number of sample paths used for averaging is only 10,000 for an accuracy notably higher than that from that of the algorithm under comparison. Although the computational time for each sample path for the present method

is higher, a more accurate solution is obtained with 40% higher speed. This table shows all the parameters and results shown in Figure 5.2

Table 5-1: Comparison of results generated from two different methods: binomial method [26] and the present method for the consecutive linear reaction

	Binomial Method	Present Method
Number of trajectories	30,000	10,000
Epsilon	0.02	0.02
Delta	----	0.667
Histogram Error	0.0090	0.0018
Total Simulation Time(s)	5.417e3	3.269e3(40% faster)

Figure 5.3 shows the accuracy of the algorithms in comparison as a function of simulation time at different values of ε. Histogram distance errors are measured by 10^6 samples and 10^5 samples generated from the SSA method and the two tau-leap methods, respectively,

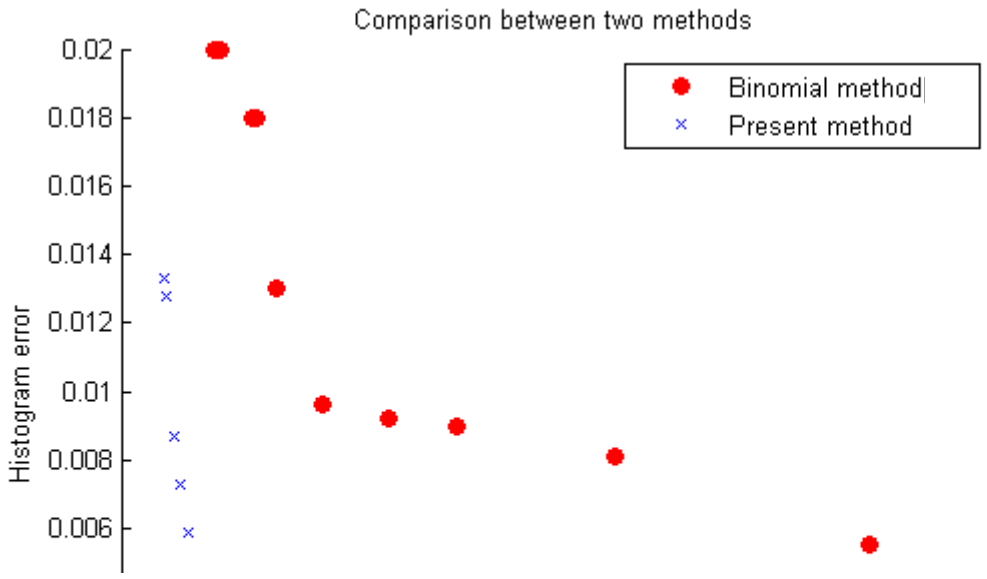

Figure 5.3: Comparison of histogram error corresponding to different simulation times for the two methods being used to model the consecutive linear reaction system. The binomial method of Peng et al. is shown in red whereas ours is shown in blue

at different values of ε. For the same simulation time, the accuracy of the histogram is notably higher for the present method.

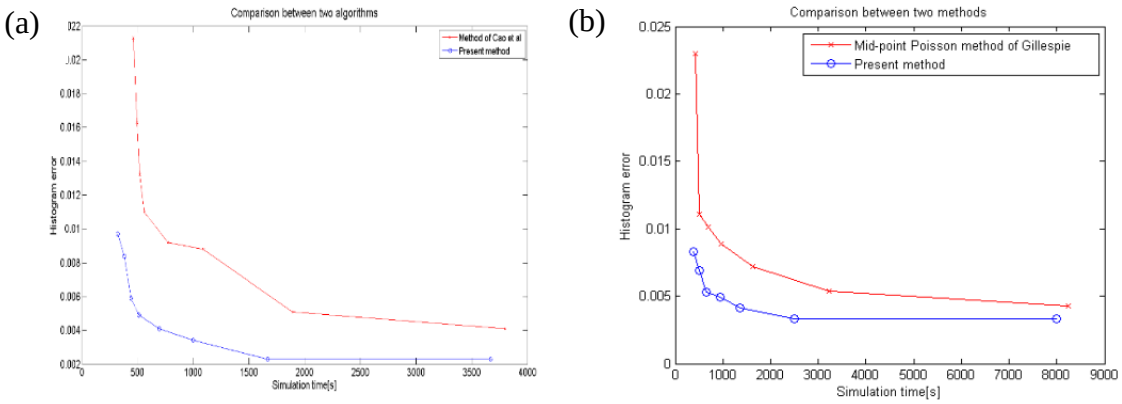

Figure 5.4: Comparison of histogram error with respect to different simulations times in different methods for Schlogl system: (a) Regular Poisson vs. present method and (b) mid-point Poisson vs. present method

Figure 5.4a shows comparison of the histogram error as a function of simulation time for the present method against that by Cao et al.[24] for Example 2. As in Example 1, the notably higher accuracy of the current method is evident for this example also. Figure 5.4b compares the algorithm of Gillespie's midpoint Poisson method with the present method for the same system. In both Figure 5.4a and 5.4b, the number of SSA samples is 10^6. The comparison was made based on a fixed number of simulations for those methods at different values of ε. The simulation time for the former is considerably higher than that for the latter. When higher accuracy is needed, the computational time needed for Gillespie's method is substantially longer

Another verification for our claim is shown in Figure 5.5:

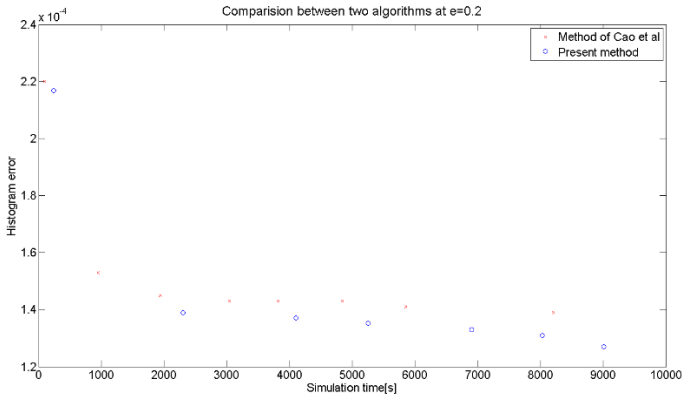

Figure 5.5: Comparison of histogram error with respect to different simulation times two different methods: Gillespie's mid-point Poisson method in red and ours in blue. This plot shows the accuracy level of the two methods for the Schlogl's system at different numbers of samples at $\varepsilon = 0.2$

In this figure, ε is fixed to be 0.2, and results generated using two different methods at different numbers of samples were then compared. Similarly, we observed that our present algorithm improves both speed and accuracy. Although the computation time for each sample path is notably larger for the present work because of smaller time intervals, the higher efficiency of the resulting sample path allows accurate results with significantly smaller number of sample paths as established in Figures 5.3, 5.4a, and 5.4b. Indeed the foregoing results bear testimony to notable improvements with our algorithm arising from the use of Chebyshev's inequality

5.5 Conclusions:

The algorithm presented in this paper has attributes of efficiency earned from being able to account for the likelihood with which approximations for tau-leap are satisfied by

simulated sample paths. The choice of this probability is a rational guideline to Monte Carlo simulations

5.6 Supplementary Materials:

It is shown below that regardless of which of the pair $\left\{\frac{a_0\varepsilon}{|\mu_j|}, \frac{a_0^2\varepsilon^2}{\sigma_j^2}\right\}$ is less than the other, τ_j as obtained from (32) will correspond to $\delta < 0$.

Proof:

Let $\frac{a_0\varepsilon}{|\mu_j|} < \frac{a_0^2\varepsilon^2}{\sigma_j^2}$ (which occurs for the case $\varepsilon > \frac{\sigma_j^2}{a_0|\mu_j|}$) so that $\tau_j = \frac{a_0\varepsilon}{|\mu_j|}$. See Fig.1 for a geometric view of this case. Consider τ_0 as defined by (39) with $\delta = 0$, i.e.,

$$\tau_0 = \frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2 - \sqrt{\left[\frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2\right]^2 - \left[\frac{a_o\varepsilon}{|\mu_j|}\right]^2}$$

Clearly the square root expression is less than $\frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2$ so that $\tau_0 < \frac{a_o\varepsilon}{|\mu_j|} = \tau_j$. In this range $\delta < 0$.

Now assume that $\frac{a_0^2\varepsilon^2}{\sigma_j^2} < \frac{a_0\varepsilon}{|\mu_j|}$ (which occurs for the case $\varepsilon < \frac{\sigma_j^2}{a_0|\mu_j|}$) so that $\tau_j = \frac{a_0^2\varepsilon^2}{\sigma_j^2}$. See Fig. 5.2 for the geometric view of this case. We aim to show that $\tau_0 < \tau_j$.

$$\tau_0 - \tau_j = \frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2 - \frac{a_0^2\varepsilon^2}{\sigma_j^2} - \sqrt{\left[\frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2\right]^2 - \left[\frac{a_o\varepsilon}{|\mu_j|}\right]^2}$$

If we set $A \equiv \frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2 - \frac{a_0^2\varepsilon^2}{\sigma_j^2}$, $B \equiv \sqrt{\left[\frac{a_o\varepsilon}{|\mu_j|} + \frac{1}{2}\left\{\frac{\sigma_j}{|\mu_j|}\right\}^2\right]^2 - \left[\frac{a_o\varepsilon}{|\mu_j|}\right]^2}$, we seek the result that $B > A$. Towards this end, we examine

$$B^2 - A^2 = \left[\frac{a_o \varepsilon}{|\mu_j|} + \frac{1}{2} \left\{ \frac{\sigma_j}{|\mu_j|} \right\}^2 \right]^2 - \left[\frac{a_o \varepsilon}{|\mu_j|} \right]^2 - \left[\frac{a_o \varepsilon}{|\mu_j|} + \frac{1}{2} \left\{ \frac{\sigma_j}{|\mu_j|} \right\}^2 - \frac{a_0^2 \varepsilon^2}{\sigma_j^2} \right]^2$$

$$= -\left(\frac{a_o \varepsilon}{|\mu_j|} \right)^2 - \left(\frac{a_0^2 \varepsilon^2}{\sigma_j^2} \right)^2 + 2 \frac{a_0^2 \varepsilon^2}{\sigma_j^2} \left[\frac{a_o \varepsilon}{|\mu_j|} + \frac{1}{2} \left\{ \frac{\sigma_j}{|\mu_j|} \right\}^2 \right] = -\left(\frac{a_0^2 \varepsilon^2}{\sigma_j^2} \right)^2 + 2 \frac{a_0^3 \varepsilon^3}{\sigma_j^2 |\mu_j|} = \frac{a_0^3 \varepsilon^3}{\sigma_j^2} \left(\frac{2}{|\mu_j|} - \frac{a_o \varepsilon}{\sigma_j^2} \right)$$

Since for this case $\varepsilon < \dfrac{\sigma_j^2}{a_0 |\mu_j|}$, we have $\dfrac{a_0 \varepsilon}{\sigma_j^2} < \dfrac{1}{|\mu_j|}$ so that $\dfrac{1}{|\mu_j|} - \dfrac{a_0 \varepsilon}{\sigma_j^2} > 0$. Thus $B > A$ from which it follows that $\tau_0 < \tau_j$ leading to the conclusion that the tau-leap algorithm of the past is without any probabilistic assurance that the tau-leap condition (28) is satisfied within the approximation of (30).

6. ON SPEEDING UP STOCHASTIC SIMULATIONS BY PARALLELIZATION OF RANDOM NUMBER GENERATION

6.1 Introduction:

The importance of stochastic simulations has risen considerably in recent times both from their applications to biology and investigation of material behavior in the nano state. The repetitive nature of simulations is responsive to simplifications of various kinds. In this paper, we show that the simple strategy of parallelizing random number generations of time subintervals among sample paths can produce notable reductions in computation time.

The idea of the methodology can be communicated in very simple terms although to make a quantitative estimate of the extent of improvement would require an inconvenient amount of effort. Suppose we are interested in computing the behavior of a stochastic process system over a specified time interval. The usual methodology involves exploiting knowledge of the random behavior of the system over successive discrete subintervals by generating random numbers which conform to calculated distributions thus generating a sample path of the process. When many such sample paths are created one after the other, average behavior of the stochastic system as well as fluctuations about the average can be calculated after a suitable number of sample paths have been obtained. The total computational time is clearly governed by the efficiency with which sample paths are created. In what follows, we provide first a simple analysis of the idea to show why the approach is attractive and then demonstrate computational improvements quantitatively with several examples. In showing that a computational procedure has the advantage of being more efficient than an existing one, it is essential to show that for a given

computational time the new procedure produces a distinctly more accurate solution. Alternatively, a solution of a specified accuracy must be shown to accrue by the new method with a considerably lighter computational burden. While the foregoing demonstration would certainly be essential to qualify the new procedure, a clearer understanding can be of the desired comparison by restricting considerations to a simple example, in which it is possible to analytically show why the proposed method is superior. To enable an analytic comparison, we select a simple Poisson process whose properties are well established. In the parallel strategy, will have initiated n sample paths of the process at the outset and allowed to progress simultaneously in time steps. Some paths will progress faster than others. An average time of evolution may be defined (as in Eq.**Error! Reference source not found.** below) to track their concerted motion in time. Those that have transcended the stipulated time will have "dropped off" from the set of n paths. A calculation of the left-over sample paths becomes possible for the Poisson process as also the fluctuations about it. Relating the computation time to the number of steps in the parallel and the sequential strategies, a comparison is enabled. What follows is the translation of this idea in mathematical terms, from which the efficacy of the parallel strategy is elucidated.

6.2 Analysis:

Consider a stochastic process $X(t)$ whose behavior is sought in the interval $I \equiv (0, t_f)$. Given the propensities (transition rates) associated with change, it is possible to define strategies for the simulation of discrete subintervals of time $I_j \equiv (t_{j-1}, t_j);\ j = 1, 2, \mathrm{K}, N$ for the process of interest with the property

$$\bigcup_{j=1}^{N-1} I_j \subset I, \quad \bigcup_{j=1}^{N} I_j \supseteq I, \quad N = 1,2,\mathrm{K}\ .$$

Clearly N is a random integer with an associated probability distribution, say $\pi_k \equiv \Pr\{N = k\}$. The computation of this probability may be tedious but it is not necessary for the ensuing argument. The sample path generated is given by $X(t) = X_j; t \in I_j,\ j = 1,2,\mathrm{K}\ , N$. We presume that n sample paths may be sufficient to obtain reasonable averaging. Further, random number generation is assumed to take *unit* computation time per interval. Thus the computation time taken for the above sample path is N, which results from neglecting possible changes in the computation time for different subintervals. Suppose n sample paths are generated by the usual sequential strategy with the i^{th} sample path involving N_i subintervals of time. Then the computation time for the i^{th} sample path is N_i. The total computation time for the sequential strategy is $\sum_{i=1}^{n} N_i$.

In the parallel strategy we simulate n times, each time subinterval, to produce a *fragment* of the averaged sample path, the computation time for which is n. As we continue this parallel strategy of computing the entire collection of sample paths in fragments, automatically ending those paths that have reached or exceeded the targeted time interval, the number of random variables to be generated can be seen to diminish progressively with a corresponding decrease in computation time.

Suppose for the sake of a preliminary demonstration, we restrict ourselves to processes for which the time interval for the next change at any stage is independent of its current state, (e.g., the Poisson process). We will be concerned with a collection of n sample paths to evolve by simulation in discrete stages over time. In other words, each

simulation will lead to a jump in the process over a time subinterval. Those sample paths that reach or transcend time t_f will have been completed and excluded from further evolution. Thus we begin with a *population* of n sample paths at the outset of the simulation. At the end of the first simulation step, the n different sample paths will have evolved to different times, say $\tau_1, \tau_2, \mathrm{K}, \tau_n$. Since this first simulation step is to be followed by numerous additional ones, the evolution of each sample path after the i^{th} step may be described by $\left\{\tau_1^i, \tau_2^i, \mathrm{K}, \tau_{K_i}^i\right\}$, where K_i is the random number of sample paths left in the collection after $\left(n - K_i\right)$ sample paths have transcended the interval $\left(0, t_f\right)$ during the i simulation steps. Since stochastic simulations of this type are contingent on the existence of a cumulative distribution function $F_T(\tau)$ for the time at which a change occurs, the j^{th} sample path in this collection which has evolved to time τ_j^i before arriving at the i^{th} will have an exit probability of $\mu_j^i \equiv \int_{t_f - \tau_i^j}^{\infty} dF_T(\tau)$.

We seek to identify an equation for the *population* of sample paths in the collection with reference to some *average* time of evolution (rather than their individual times of evolution) to represent *all* the sample paths in a given step. Towards this end, a probability would be required for *any* sample path in the collection to quit by transcending the interval $\left[0, t_f\right]$. We adopt the average time for the i^{th} step denoted $\bar{\tau}_i$ as below which will hold for the sample paths which survive for the next step.

$$\bar{\tau}_i \equiv \bar{\tau}_{i-1} + \int_0^{t_f - \bar{\tau}_{i-1}} \tau dF_T(\tau) \Big/ \int_0^{t_f - \bar{\tau}_{i-1}} dF_T(\tau), \quad \bar{\tau}_0 = 0 \tag{49}$$

The foregoing choice represents the average time at which sample paths in the i^{th} step have expended an average time of $\bar{\tau}_{i-1}$ in the previous step. The probability μ_i that a sample path exits in the i^{th} step from the collection may now be estimated as

$$\mu_i = \int_{t_f - \bar{\tau}_i}^{\infty} dF_T(\tau) \tag{50}$$

If we now let $P_k^i \equiv \Pr\{K_i = k | K_1 = n\}$, then it is readily shown that

$$P_k^{i+1} = (1-\mu_i)^k \sum_{r=0}^{n-k} \binom{k+r}{r} \mu_i^r P_{k+r}^i, \quad P_k^0 = \delta_{k,n} \tag{51}$$

Eq.(51) uses the initial condition that there are n sample paths in all for which the subintervals are generated and may be rewritten as

$$P_k^{i+1} = (1-\mu_i)^k \sum_{m=k}^{n} \binom{m}{k} \mu_i^{m-k} P_m^i \tag{52}$$

The above equation is solved in the Supplementary Material 6.8.1 to obtain the first and second moments of K_{i+1}.

$$EK_{i+1} == n\prod_{j=0}^{i}(1-\mu_j) \tag{53}$$

$$EK_{i+1}^2 = n\prod_{j=0}^{i}(1-\mu_j)^2\left[n + \sum_{k=0}^{i}\mu_k \prod_{j=0}^{k}(1-\mu_j)^{-1}\right] \tag{54}$$

The variance of K_{i+1}, denoted $V(K_{i+1})$, is obtained from (53) and (54)

$$V(K_{i+1}) = EK_{i+1}^2 - (EK_{i+1})^2 = n\prod_{j=0}^{i}(1-\mu_j)^2\left[\sum_{k=0}^{i}\mu_k \prod_{j=0}^{k}(1-\mu_j)^{-1}\right] \tag{55}$$

from which the coefficient of variation, denoted COV_i, is obtained as

$$COV_i = \frac{\sqrt{V\left(K_{i+1}\right)}}{EK_{i+1}} = \sqrt{n^{-1}\sum_{k=0}^{i}\mu_k\prod_{j=0}^{k}\left(1-\mu_j\right)^{-1}} \tag{56}$$

The computation time for the parallel strategy may now be estimated from $\sum_{i=0}^{i_{\max}} EK_{i+1}$, where $i_{\max}$ may be chosen by requiring that the above $EK_{i_{\max}+1}$ is suitably small, which implies that n sample paths have been cleared from the collection. The coefficient of variation (56) serves to verify that the fluctuation associated with this population is negligible.

We now consider the very simple case of a Poisson process adding to a population of individuals at mean rate λ; the change in this process over time is the addition of an individual (in this case independently of the prior population) which has the distribution function given by $F_T\left(\tau\right)=1-e^{-\lambda\tau}$. For this case, we have from (50)

$$\mu_i = \lambda\int_{t_f-\bar{\tau}_i}^{\infty} e^{-\lambda\tau}\,d\tau = e^{-\lambda\left(t_f-\bar{\tau}_i\right)} \tag{57}$$

The expression (57) shows that, as $\bar{\tau}_i$ approaches t_f, μ_i approaches 1 making the sample path increasingly likely to exit the collection. From (49), we obtain the following for the Poisson process.

$$\bar{\tau}_i = \bar{\tau}_{i-1} + \frac{1}{\lambda} - \frac{e^{-\lambda\left(t_f-\bar{\tau}_{i-1}\right)}\left(t_f-\bar{\tau}_{i-1}\right)}{1-e^{-\lambda\left(t_f-\bar{\tau}_{i-1}\right)}}, \quad \bar{\tau}_0 = 0 \tag{58}$$

For a preliminary quantitative demonstration, we consider simulating the Poisson process by the sequential as well as the parallel strategy. We have $F_T\left(\tau\right)=1-e^{-\lambda\tau}$ to generate subintervals.

For the sequential strategy using the Poisson process, it is possible to readily obtain the probability distribution π for the discrete random variable N (see Supplementary Material 6.8.2), the number of subintervals which sum to cover the time interval $\left[0, t_f\right]$. Thus

$$\pi_k \equiv \Pr\left\{Sample\ path\ ends \middle| N = k\right\} = 1 - \frac{1}{2^k}\left(1 - e^{-\lambda t_f}\right)^{k+1}, \quad k = 0,1,2,\mathrm{K}$$

For the parallel strategy, we generate random numbers satisfying the cumulative distribution function $F(\tau) = 1 - e^{-\lambda\tau}$, through simulation of the uniform random variable $X \overset{d}{=} U[0,1]$

6.3 Comparison of Computation Times: Sequential and Parallel Strategy

The expected computation time for the sequential strategy has been shown to be given by

$$E\sum_{k=1}^{n} N_k = \sum_{i=1}^{n} EN_k = nEN \tag{59}$$

which follows from the sample paths possessing the *same* distribution function for generation of subintervals. EN, being the expected number of subintervals in generating an entire sample path, is given by (S.6.9) in the Supplementary Material reproduced below.

$$EN = \frac{\frac{1}{2}K_{\min}\left(K_{\min}+1\right) - 2\left(1-r\right)^{-2}}{K_{\min} - 2r\left(1-r\right)^{-1}}, \quad r \equiv \frac{1}{2}\left(1 - e^{-\lambda t_f}\right) \tag{60}$$

where $K_{\min}$ is as specified by (S.6.11). Thus the expected computation time for the simulation of the Poisson process using the sequential strategy is specified by (60).

Using the parallel strategy, the expected computational time for the Poisson process is given by

$$n\sum_{i=0}^{i_{\max}}\prod_{j=0}^{i}\left(1-e^{-\lambda\left(t_f-\bar{\tau}_j\right)}\right) \tag{61}$$

where $\bar{\tau}_j$ is given (73) in the Supplementary Material. The ratio of (61) to (59) rovides a good quantitative measure of the effectiveness of the parallel strategy relative to the sequential strategy as it is a direct comparison of the expected computation times. Figure 6.1 shows the results of calculations. In this figure, simulation time was plotted as a function of square root of $\left(\lambda * t_f\right)$, and it clearly indicates that the sequential algorithm would cost more CPU time than the parallel algorithm at any given value of $\left(\lambda t_f\right)$.

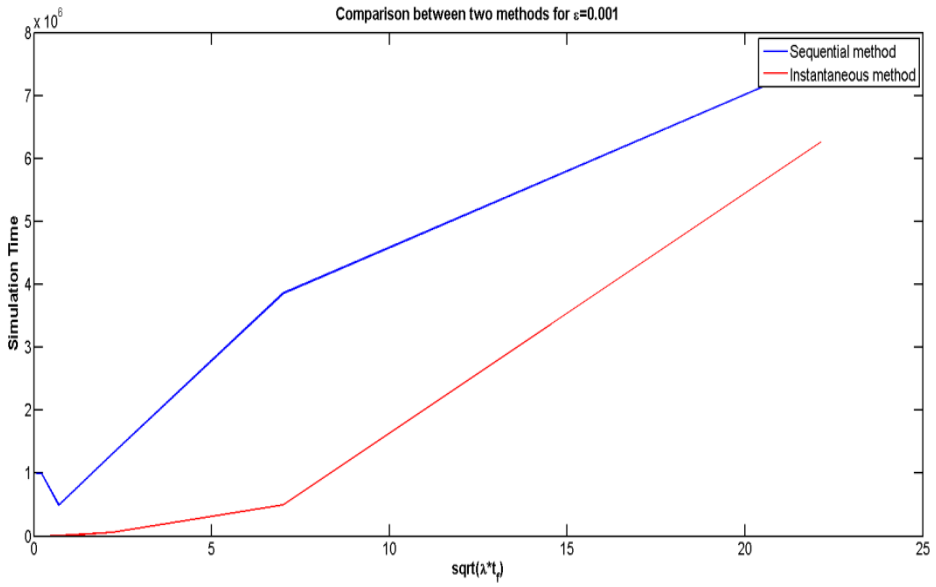

Figure 6.1: Comparison between two methods for eps=0.001

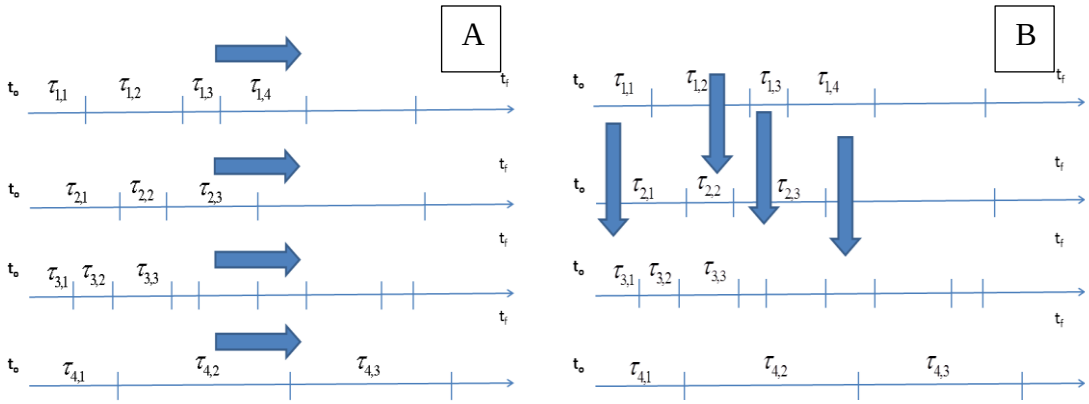

Figure 6.2: Comparison of (A) Sequential Method and (B) Parallel Method

The demonstration above was made for processes in which the distribution function for successive generation of quiescence intervals was the same. For many applications, this is not a realistic assumption so that a demonstration of the effectiveness of the parallel strategy necessarily requires detailed simulations by both strategies. Figure 6.2 below provides a more detailed schematic picture of how each method works.

Let us discuss strategies utilized in each method for a simple scenario of simulations that is composed of 4 sample paths, shown in the figure above. The sequential method simulates leaps sequentially and keeps updating new states using information from the previous step. This procedure is iterated until reaching the final t_f. Upon the completion of one, it then can be applied to the next sample path. The parallel method, on the other hand, will start with generating the first leap for each trajectory independently. Second leap for each sample will then be simulated simultaneously and applied to update variables that correspond to the previous states from the same sample path. This procedure is carried on iteratively. Since generation of various sample paths is independent, some sample paths will reach the mature time before others. Due to that nature, the parallel method can reduce

the number of trajectories that need to be simulated as it approaches the final time. Specifically, in figure 6.2B, it clearly indicates that the sample path 4 can be dropped out of the simulation bath after 4 steps, followed by sample path 1 after another 2 steps. The number sample path will keep decreasing as the simulation evolves with time, hence reducing memory burden and CPU time. The strategy can also be presented in a step-wise manner in the Section 6.3. To further illustrate the key idea, in section 6.4 and 6.5, simulation results corresponding to several examples are shown and discussed

6.4 Parallelizing tau-leaping algorithm

- For a system with n sample paths to evolve, at time zero, set the state of time for each sample path $t_1, t_2, \mathrm{K}, t_n$ equal to 0.
- Set up the initial numbers of molecules for each sample path, say $\mathbf{X}_1(t_1 = 0) = \mathbf{x}_1^o; \mathbf{X}_2(t_2 = 0) = \mathbf{x}_2^o; ...; \mathbf{X}_n(t_n = 0) = \mathbf{x}_n^o$ where the superscript o denotes the initial value. Note that $\mathbf{X}_k, k = 1, 2..., n$ is a vector describing the states of the system for the sample path k.
- Calculate the tau values $\tau_1, \tau_2, \mathrm{K}, \tau_n$ for each sample path and accordingly generate poison random number for all reactions of all sample paths. With information of stoichiometry, the change of states for each sample path could be obtained as $\boldsymbol{\delta}\mathbf{x}_1; \boldsymbol{\delta}\mathbf{x}_2, ..., \boldsymbol{\delta}\mathbf{x}_n$. Note that $\boldsymbol{\delta}\mathbf{x}_k, k = 1, 2..., n$ is a vector describing the state change of the system for the sample path k.
- Update the states for each sample path by

$$X_1(t_1 + \tau_1) = X_1(t_1) + \boldsymbol{\delta}\mathbf{x}_1; X_2(t_2 + \tau_2) = X_2(t_2) + \boldsymbol{\delta}\mathbf{x}_2; ...; X_n(t_n) = \boldsymbol{\delta}\mathbf{x}_n$$

- Update the state of time for each sample path by

  $t_1 \leftarrow t_1 + \tau_1, t_2 \leftarrow t_2 + \tau_2, \mathrm{K}, t_n \leftarrow t_n + \tau_n$

- For sample path k with $t_k < t_f$, repeat step 3 to 5.

- The calculation is done when $t_k \geq t_f$ for all $k \in [1, 2..., n]$

6.5 Examples:

Example 1: (Schlogl's chemical reaction model[24], [29], [31], known for its bistability. B_1 and B_2 are constants with their particle numbers as N_1 and N_2 respectively.

$$B_1 + 2X \underset{c_2}{\overset{c_1}{\rightleftarrows}} 3X, \quad B_2 \underset{c_4}{\overset{c_3}{\rightleftarrows}} X$$

The values of parameters, adapted from Cao et al.[22], are $c_1 = 3\times10^{-7}$, $c_2 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 3.5$, $N_1 = 1\times10^5$, and $N_2 = 2\times10^5$. The initial condition of X is 250 and we simulate the system up to $t_f = 4$.

Example 2*:* Consecutive linear reaction system[24], [26], [28]

$$X_1 \xrightarrow{c_1} X_2 \xrightarrow{c_2} X_3$$

Rate constants $c_1 = 1$ and $c_2 = 1$, and initial conditions $X_1 = 10^4$, $X_2 = 1$ and $X_3 = 0$, we calculate X_3 until time is equal to 0.1.

Example 3*:* Dimer[64] $S_1 \xrightarrow{c_1} \phi, \quad S_2 + S_3 \xrightarrow{c_2} S_1$

At $t = 0$, $S_1 = 3000, \quad S_2 = 3000, \quad S_3 = 10^4$. Moreover, $c_1 = 1$ and $c_2 = 10^{-4}$. Our goal is to compare the results between distributions generated by two algorithms at $t_f = 2$.

Example 4: In this system, a monomer S_1 can dimerize to an unstable form S_2, which in turn coverts to a stable form S_3. Since S_2 is unstable, it can also convert back to S_1 [24], [30]

$$S_1 \xrightarrow{c_1} \phi, \quad S_1 + S_1 \underset{c_3}{\overset{c_2}{\rightleftarrows}} S_2, \quad S_2 \xrightarrow{c_4} S_3$$

Initial conditions: $S_1(0) = 4150, \quad S_2(0) = 39564, \quad S_3(0) = 3445$

Rate constants: $c_1 = 1,\ c_2 = 0.002,\ c_3 = 0.5,\ c_4 = 0.04$

Our goal is to collect the distribution of the dimer at $t_f = 10$.

6.6 Results and discussion

Four examples have been utilized to compare the effectiveness of the proposed parallelization, also referred to here as the simultaneous algorithm. The first example was that of Schologl's system, for which comparison was made of simulations with the τ-leap method involving Poisson distribution. Figure 6.3, shows consistent results for the distribution of X by both methods:

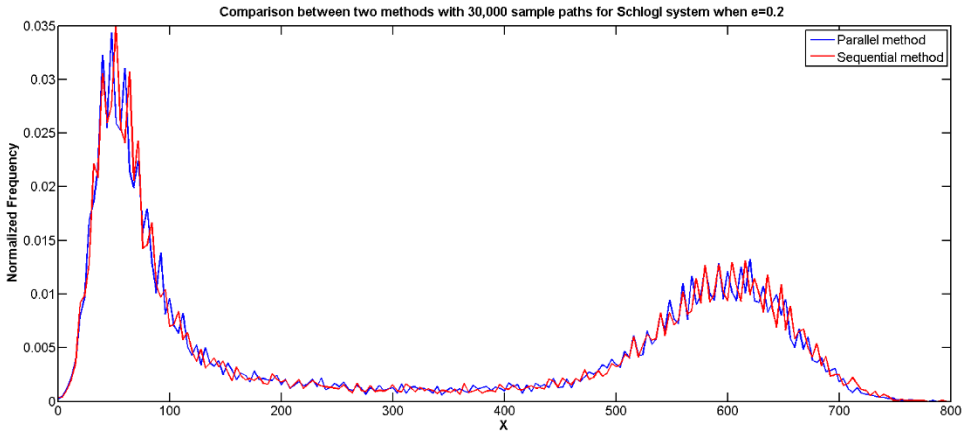

Figure 6.3: Distribution of X

Schlogl's system is noted to exhibits its bistable steady-state distribution[65]–[67]. Distribution of X generated by two different methods are shown in red and blue curves which virtually overlap one another over the entire range of values.

In Figures 6.4 and 6.5, performances of the two algorithms are compared in terms of CPU time.

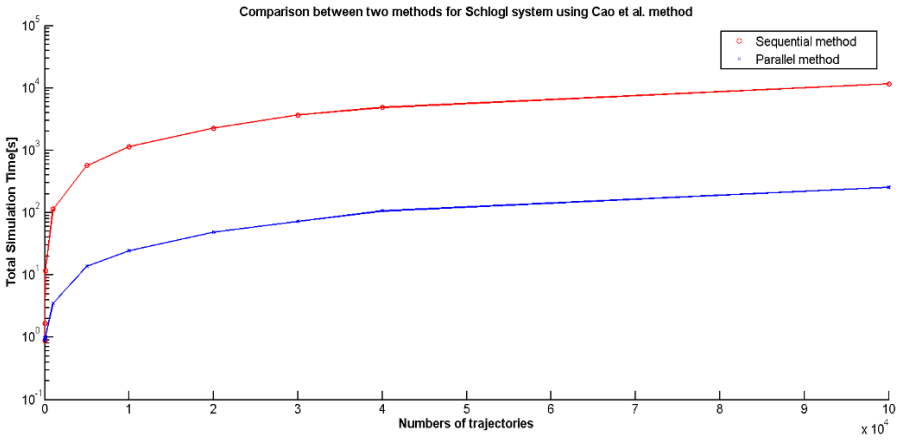

Figure 6.4: Simulation time vs. Number of trajectories

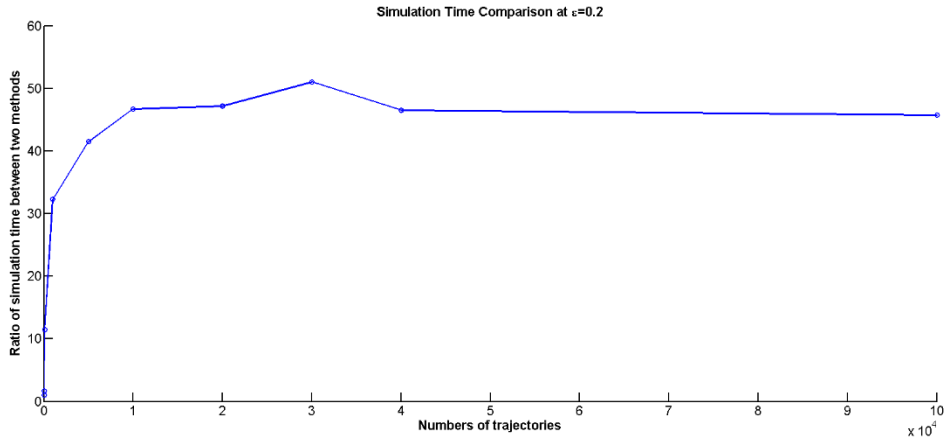

Figure 6.5: Ratio of simulation times between two methods

Clearly, the sequential method requires substantially longer computation times for the simulation, than the simultaneous algorithm. For instance, with 30000 trajectories, the sequential algorithm ran about 50 times slower than the other. In example 2, the binomial τ-leap method was used for comparison, and a similar trend is seen in Figure 6.6 and 6.7.

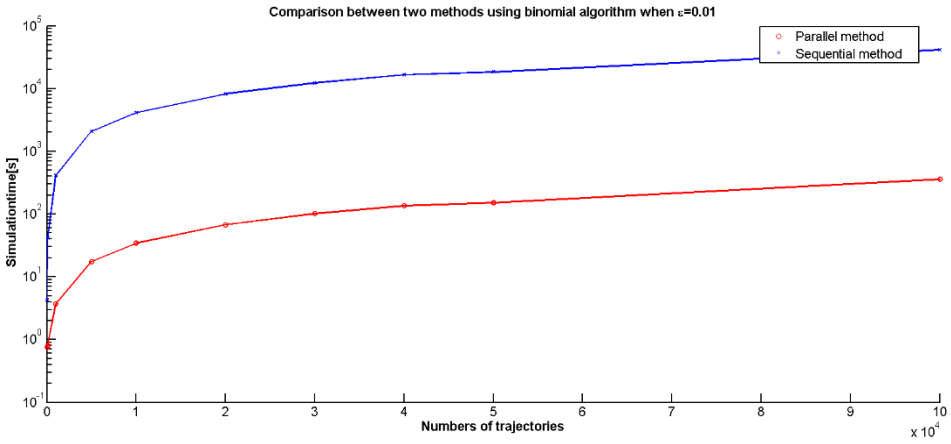

Figure 6.6: Simulation time vs. Number of trajectories using binomial method

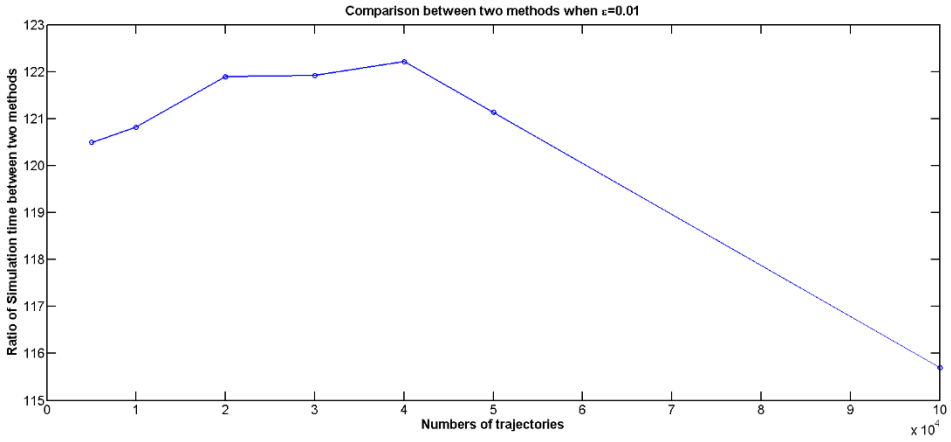

Figure 6.7: Ratio of simulation time between two methods

The simultaneous algorithm outperforms the sequential with a 120 fold improvement in CPU time. Figures 6.8, 6.9 and 6.10 were produced for example 3. Figure 6.8 compared the accuracy of each solution generated by the two algorithms to that produced by SSA with 50000 trajectories.

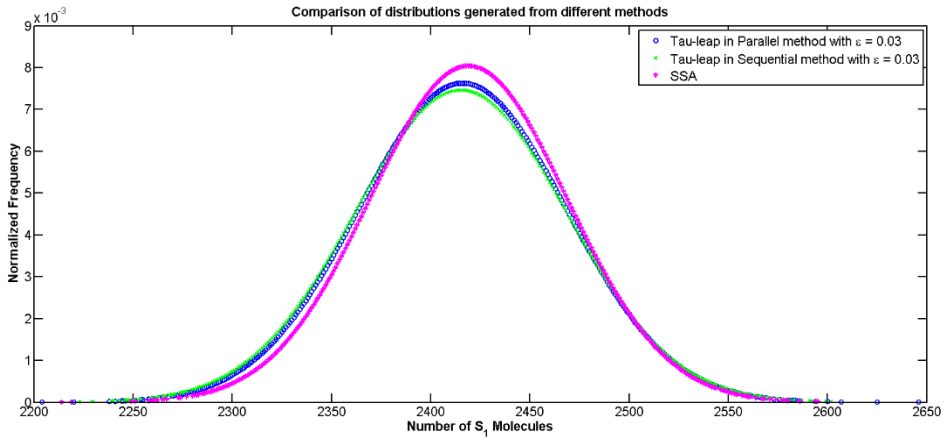

Figure 6.8: Distribution of S1 molecule

To fully investigate the benefit of this method, the performances were compared from two different aspects: in Figure 6.9 epsilon, which represents the measure of accuracy in the tau-leap algorithm[24], [26], [31], was fixed and the number of trajectories was changed and in Figure 6.10 the number of trajectories was fixed and epsilon was varied. In either case, the simultaneous method has prevailed unambiguously over the other.

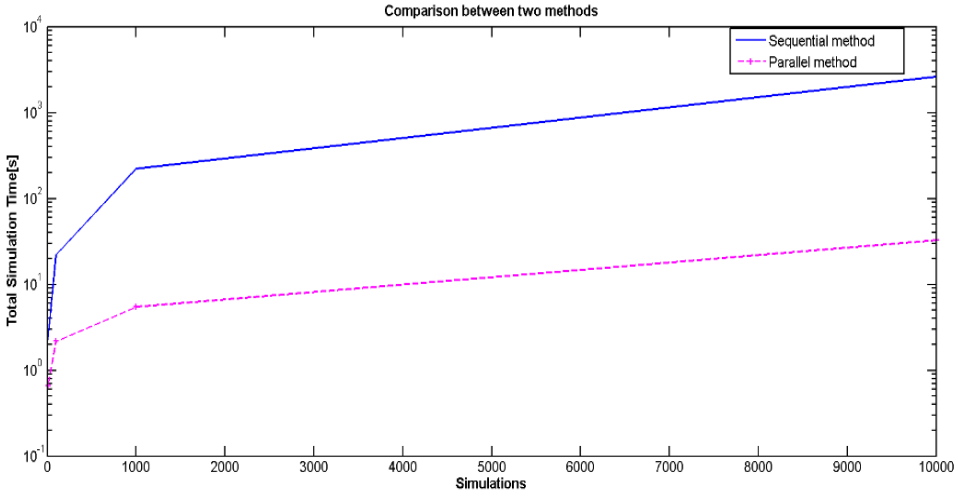

Figure 6.9: Comparison in simulation times between two methods

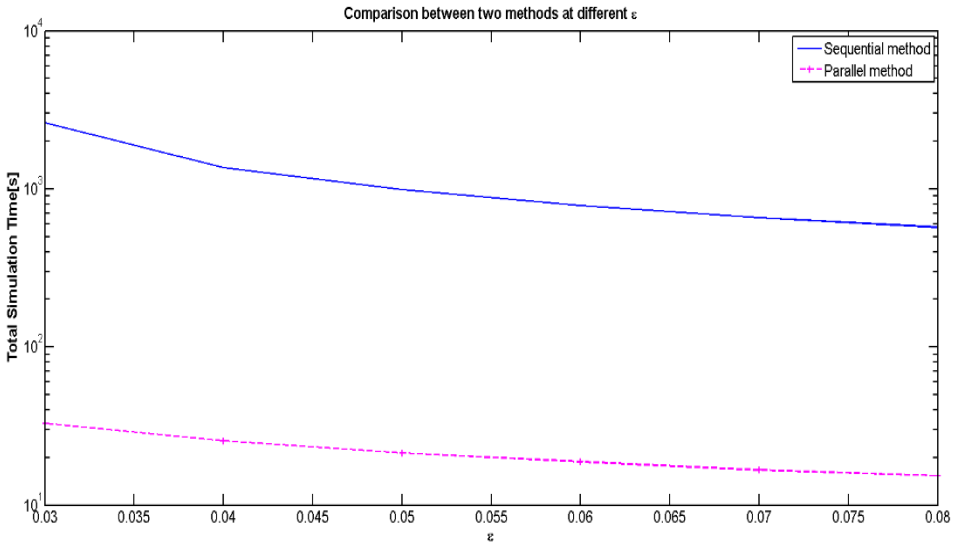

Figure 6.10: Total simulation time

6.7 Conclusions

This paper establishes that the parallel strategy for generating sample paths of stochastic processes is notably superior to the usual sequential strategy followed in stochastic simulations. Since each trajectory is independent of all the others and can vary greatly in the number of steps within each sample path, simultaneous generation of several such leaps for different trajectories eliminates the delay time between each trajectories. As a result, for a given requirement of accuracy, the overall simulation time can be optimized

6.8 Supplemental Materials

6.8.1 Derivation of Moments of Sample Path Populations

We start with Eq. (52) of the manuscript

$$P_k^{i+1} = (1-\mu_i)^k \sum_{m=k}^{n} \binom{m}{k} \mu_i^{m-k} P_m^i$$

Summing over k we have

$$\sum_{k=0}^{n} P_k^{i+1} = \sum_{k=0}^{n} (1-\mu_i)^k \sum_{m=k}^{n} \binom{m}{k} \mu_i^{m-k} P_m^i \tag{62}$$

Changing the order of summation in the right hand side above

$$\sum_{k=0}^{n} P_k^{i+1} = \sum_{m=0}^{n} P_m^i \sum_{k=0}^{m} \binom{m}{k} (1-\mu_i)^k \mu_i^{m-k} = (1)^m \sum_{m=0}^{n} P_m^i = 1 \tag{63}$$

which shows, as it should, the preservation of normalization. Taking the first moment

$$EK_{i+1} \equiv \sum_{k=0}^{n} k P_k^{i+1} = \sum_{m=0}^{n} P_m^i \sum_{k=0}^{m} k \binom{m}{k} (1-\mu_i)^k \mu_i^{m-k} = \sum_{m=0}^{n} P_m^i \sum_{k=1}^{m} \frac{m!}{(m-k)!(k-1)!} (1-\mu_i)^k \mu_i^{m-k}$$

$$EK_{i+1} = (1-\mu_i) \sum_{m=0}^{n} m P_m^i \sum_{k=0}^{m-1} \frac{(m-1)!}{(m-k-1)!k!} (1-\mu_i)^k \mu_i^{m-k-1} = (1-\mu_i) EK_i$$

Thus we have by iteration

$$EK_{i+1} == n \prod_{j=0}^{i} (1-\mu_j) \tag{64}$$

We seek similarly the second moment by evaluating

$$EK_{i+1}^2 \equiv \sum_{k=0}^{n} k^2 P_k^{i+1} = \sum_{m=0}^{n} P_m^i \sum_{k=0}^{m} k^2 \binom{m}{k} (1-\mu_i)^k \mu_i^{m-k}$$

$$= \sum_{m=0}^{n} P_m^i \left[\sum_{k=0}^{m} \binom{m}{k} (1-\mu_i)^k \mu_i^{m-k} \{k(k-1)+k\} \right]$$

$$= (1-\mu_i)^2 \sum_{m=0}^{n} m(m-1) P_m^i + (1-\mu_i) \sum_{m=0}^{n} m P_m^i$$

$$= (1-\mu_i)^2 \sum_{m=0}^{n} m^2 P_m^i + \mu_i (1-\mu_i) \sum_{m=0}^{n} m P_m^i$$

which leads to the formula

$$EK_{i+1}^2 = (1-\mu_i)^2 EK_i^2 + \mu_i (1-\mu_i) EK_i$$

$$EK_1^2 = (1-\mu_0)^2 EK_0^2 + \mu_0 (1-\mu_0) EK_0 = n^2 (1-\mu_0)^2 + n\mu_0 (1-\mu_0)$$

$$EK_2^2 = (1-\mu_1)^2 EK_1^2 + \mu_1 (1-\mu_1) EK_1$$

$$= n^2 (1-\mu_1)^2 (1-\mu_0)^2 + n\mu_0 (1-\mu_0)(1-\mu_1)^2 + n\mu_1 (1-\mu_1)(1-\mu_0)$$

Iterating on the foregoing step leads to the general expression

$$EK_{i+1}^2 = n^2\prod_{j=0}^{i}\left(1-\mu_j\right)^2 + n\mu_0\left(1-\mu_0\right)\prod_{j=1}^{i}\left(1-\mu_j\right)^2 + n\mu_1\left(1-\mu_1\right)\left(1-\mu_0\right)\prod_{j=2}^{i}\left(1-\mu_j\right)^2 + \mathrm{L}$$

$$\mathrm{L}\ \ n + n\mu_i\prod_{j=0}^{i}\left(1-\mu_j\right)$$

$$EK_{i+1}^2 = n\prod_{j=0}^{i}\left(1-\mu_j\right)^2\left[n + \sum_{k=0}^{i}\mu_k\prod_{j=0}^{k}\left(1-\mu_j\right)^{-1}\right] \tag{65}$$

The variance of K_{i+1}, denoted $V\left(K_{i+1}\right)$, is obtained from (65)

$$V\left(K_{i+1}\right) = EK_{i+1}^2 - \left(EK_{i+1}\right)^2 = n\prod_{j=0}^{i}\left(1-\mu_j\right)^2\left[\sum_{k=0}^{i}\mu_k\prod_{j=0}^{k}\left(1-\mu_j\right)^{-1}\right] \tag{66}$$

from which the coefficient of variation, denoted COV_i, is obtained as

$$COV_i = \frac{\sqrt{V\left(K_{i+1}\right)}}{EK_{i+1}} = \sqrt{n^{-1}\sum_{k=0}^{i}\mu_k\prod_{j=0}^{k}\left(1-\mu_j\right)^{-1}} \tag{67}$$

6.8.2 Exit Probability of Sample Path

We seek the probability that a sample path over a time interval $\left[0, t_f\right]$ is completed in N generations of random quiescence intervals. Towards this end, we first focus on $N = 2$.

$$\pi_2 \equiv \Pr\left\{T_1 + T_2 \ge t_f\right\} = \int_{\Omega} dF_{T_1T_2} = \int_{\Omega} dF_{T_1}dF_{T_2} = 1 - \lambda^2\int_0^{t_f} e^{-\lambda\tau_2}d\tau_2\int_0^{\tau_2} d\tau_1 e^{-\lambda\tau_1} = 1 - \lambda\int_0^{t_f} e^{-\lambda\tau_2}\left[1-e^{-\lambda\tau_2}\right]d\tau_2$$

$$1 - \lambda\int_0^{t_f} e^{-\lambda\tau_2}\left[1-e^{-\lambda\tau_2}\right]d\tau_2 = 1 - \left[1 - e^{-\lambda t_f} - \frac{1}{2}\left(1-e^{-2\lambda t_f}\right)\right] = 1 - \frac{1}{2}\left(1-e^{-\lambda t_f}\right)^2$$

Proceeding in this way, we have

$$\pi_k = 1 - \frac{1}{2^{k-1}}\left(1-e^{-\lambda t_f}\right)^k, \quad k = 1, 2, \mathrm{K} \tag{68}$$

which shows that as k increases indiscriminately, $\pi_k \to 1$ thus showing that a sample path must eventually cross the instant t_f with certainty.

6.8.3 Expected Number of Steps in a Sample Path

We define the following propositions.

B: Sample path has evolved to completion (i.e. to time t_f or beyond).

A_k : Sample path has a total of k steps.

$$P(A_k B) = P(A_k | B) P(B), \qquad P(B) = \sum_k P(A_k B)$$

$$p_k \equiv P(A_k | B) = \frac{P(A_k B)}{\sum_j P(A_j B)} = \frac{\pi_k}{\sum_j \pi_j} = \frac{1 - \frac{1}{2^{k-1}}\left(1 - e^{-\lambda t_f}\right)^k}{\sum_j \left[1 - \frac{1}{2^{j-1}}\left(1 - e^{-\lambda t_f}\right)^j\right]} \tag{69}$$

Expected number of steps in a sample path $= \sum_k N_k p_k = \sum_k k\left[1 - \frac{1}{2^{k-1}}\left(1 - e^{-\lambda t_f}\right)^k\right]$. If the sum is taken over integer K , then letting $r \equiv \frac{1}{2}\left(1 - e^{-\lambda t_f}\right)$,

$$EN = \frac{\frac{1}{2}K(K+1) - 2(1-r)^{-2}\left[1 - r^{K-1}\left(K - Kr + r\right)\right]}{K - 2r(1-r)^{-1}\left(1 - r^K\right)} \approx \frac{\frac{1}{2}K_{\min}\left(K_{\min}+1\right) - 2(1-r)^{-2}}{K_{\min} - 2r(1-r)^{-1}}$$

(S.6.9)

where $K_{\min}$ is estimated as the smallest number of steps for which $\pi_{K_{\min}}$ is close to 1 to a desired degree. Thus we let

$$r^{K_{\min}}\left[K_{\min}\frac{(1-r)}{r} + 1\right] = \varepsilon \tag{70}$$

6.8.4 Computation of Average Time for i^{th} Step (Poisson process)

$$\bar{\tau}_i \equiv \bar{\tau}_{i-1} + \int_0^{t_f - \bar{\tau}_{i-1}} \tau dF_T(\tau) \Big/ \int_0^{t_f - \bar{\tau}_{i-1}} dF_T(\tau), \quad \bar{\tau}_0 = 0$$

$$\bar{\tau}_i = \bar{\tau}_{i-1} + \frac{\lambda \int_0^{t_f - \bar{\tau}_{i-1}} \tau e^{-\lambda\tau} d\tau \left[1 - e^{-\lambda\left(t_f - \bar{\tau}_{i-1}\right)}\left(t_f - \bar{\tau}_{i-1} + 1\right)\right]}{1 - e^{-\lambda\left(t_f - \bar{\tau}_{i-1}\right)}}$$

$$\bar{\tau}_i = \bar{\tau}_{i-1} + \left[\frac{1}{\lambda} - \frac{\left(t_f - \bar{\tau}_{i-1}\right) e^{-\lambda\left(t_f - \bar{\tau}_{i-1}\right)}}{1 - e^{-\lambda\left(t_f - \bar{\tau}_{i-1}\right)}} \right] \tag{71}$$

Letting $X_i \equiv \lambda\left(t_f - \bar{\tau}_i\right)$ we obtain the recurrence relationship below:

$$X_i = X_{i-1} - 1 + \frac{\lambda X_{i-1} e^{-X_{i-1}}}{1 - e^{-X_{i-1}}}, \quad X_0 = \lambda t_f \tag{72}$$

From X_i, $\bar{\tau}_i$ is obtained using

$$\bar{\tau}_i = t_f - \lambda^{-1} X_i \tag{73}$$

7. ON FACILITATED COMPUTATION OF MESOSCOPIC BEHAVIOR OF REACTION-DIFFUTION SYSTEMS (BIOFILMS)

7.1 Introduction:

Signaling in cells often involves a small number of molecules so that the reactions among them occur at a random rate [60], [68], [69]. In addition, they follow nonlinear kinetics so that the average behavior does not obey macroscopic kinetics. Thus intracellular balances are (continuous) nonlinear stochastic differential equations whose average behavior must be found by detailed evaluation of sample paths using Monte Carlo simulative algorithms [70], [71]. This process becomes forbiddingly tedious for a sizable population of cells with further exacerbation arising from often encountered stiffness of the differential equations. The Chemical Master Equation (CME) [21], affords a natural avenue for formulation of stochastic reaction systems preserving the discrete nature of the amount of reacting entities. Together with system size expansion techniques [21], [72], CME provides in fact the route to arriving at the continuous stochastic differential equations referred to above. However, CME does not inspire computational methodology towards extracting system behavior. Consequently, researchers have been led to use simulation methods such as Stochastic Simulation Algorithm (SSA) [22] (or equivalently the interval of quiescence used by Shah et al. [23]) suitably embellished by “tau-leap” strategies designed to skip events that do not make notable changes to system dynamics [24]–[26], [28], [31], [32], [63]. In spite of the foregoing armory of techniques, the computation of average behavior of systems of interest

to this paper has been thwarted by overwhelming demands on computational costs primarily due to the large number of stochastic dynamical equations.

A refreshing approach towards resolving the above dilemma appears in a paper by Grima, which raises the possibility of directly obtaining average (mesoscopic) behavior by entirely circumventing the computation of sample paths. The underlying methodology is grounded in quasilinearization leading to a derivation of dynamical equations which represent the potential source of average behavior. A demonstration of this approach appears in the recent publication of Smith et al. [37] from Grima's group through a biological example which shows that the average mesoscopic behavior was dependent on the diffusion coefficients connected with random motion. Such behavior is characteristic of growth in biofilms, a research area which motivates our current effort.

The objective of this paper is to investigate a stochastic reaction system in a discrete number of well-stirred cells which are connected by diffusive transport thus coupling the behavior of all reaction cells. The reactions are represented as

$$s_{1,j}X_1 + s_{2,j}X_2 + \cdots + s_{N,j}X_N \xrightarrow{k_j} r_{1,j}X_1 + r_{2,j}X_2 + \cdots + r_{N,j}X_N, \quad j = 1,2,\ldots,R \quad (74)$$

This scenario is a simplified version of growth in a biofilm in which cells are embedded without accounting for growth or distinction between extra- and intracellular variables. The mass balance equations for each cell account for diffusive mixing from all other cells and a stochastic reaction term in the given cell. Taking expectation of the above equations produces the expectation of the nonlinear reaction terms to which Grima's [36] approximation is made.

While stochastic simulation has been largely restricted to well-mixed stochastic systems, its extension to a system where diffusional effects compete with reaction has been

somewhat sporadic. Experiments show that diffusion can also occur at rates comparable with those of reactions thus invalidating the well-mixed assumption. In contrast to a well-mixed stochastic system, studies on spatial stochastic systems have only begun recently and a clear understanding of how to accurately tract them is yet to emerge. Moreover, generation of sample paths by SSA can be utilized to obtain the average number density of species, the true solution. This true solution is assumed to be almost the same or only slightly different from the solution of the CME. Recent studies from Grima's group showed at the mesoscopic limit, the deviation between these two can be significant. His approach to reaction kinetics for small volumes presents a way to handle reaction in this system. This paper introduces the concept of applying a discrete self-adjoint linear operator whose spectral representation is used to capture the diffusional effect.

In this paper, we attempt to handle this problem in two dimensions by applying two different methods to capture both reactions and diffusion without generating sample paths. The domain of interest is divided into a two-dimensional array of small well-mixed compartments. These compartments are assumed for convenience to be squares of equal size although this assumption is not essential. Reactions occur within each compartment. Concentration of species varies spatially which allows diffusion to occur between neighboring compartments. In the following section, we will briefly discuss the application of the effective mesoscopic rate equations (EMREs) [36] in an effort to approximate the solutions from CME for the reaction system. In the next section, we will provide formulation of the linear operator and its definition, as well as how it can be applied in order to account for the diffusion effect in the system. The results from our current method

can be then compared with the results generated by solving the reaction-diffusion master equation [34], [37], [73]–[75]

7.2 Analysis:

The two dimensional domain is divided into discrete squares of length *l*. Along the "horizontal" and "vertical" coordinate, we have

$i = 0, \pm 1, \pm 2, \mathrm{K}, \pm M \quad (2M+1)$ discrete elements. Denote

$$S^I \equiv \{-M, -M+1, \mathrm{K}, -2, -1, 0, 1, 2, \mathrm{K}, M\}$$

$j = 0, \pm 1, \pm 2, \mathrm{K}, \pm N \quad (2N+1)$ discrete elements. Denote

$$S^J \equiv \{-N, -N+1, \mathrm{K}, -2, -1, 0, 1, 2, \mathrm{K}, N\}$$

There are n variables each representing particle numbers of reaction species in the cell located at (i, j), denoted $Z_{i,j}^{(s)}$, $s = 1, 2, \mathrm{K}, n$. Define $Z_{i,j}^{(s)} \equiv X_{li,lj}^{(s)} / l$ as the ij^{th} component of scaled vector $\mathbf{Z}$ with $\mathbf{X}$ as the vector the actual particle numbers in the physical domain. Collating every cell in the domain, we may define the vector

$$\mathbf{Z}^{(s)} = \left\{ Z_{i,j}^{(s)}; \quad i \in S^I; j \in S^J \right\} \in \mathcal{H} \equiv \Re^{2N+1} \otimes \Re^{2M+1}$$

7.2.1 Effective mesoscopic rate equation[36]

The CME in general cannot be solved analytically; however, the dynamics of the reaction system (74) can be captured by means of the system-size expansion due to Van Kampen [21] using

$$\frac{n_i}{\Omega} = \Phi_i + \Omega^{-1/2} \varepsilon_i \tag{75}$$

where Φ_i is the macroscopic concentration of species i as determined by the regular rate equation (RE) and ε_i is a continuous random variable which represents the system-scaled

fluctuation in concentration. Grima's analysis arrives at the following equation in the average fluctuation:

$$\frac{d\langle \boldsymbol{\varepsilon} \rangle}{dt} = \mathbf{J} \bullet \langle \boldsymbol{\varepsilon} \rangle + \boldsymbol{\Omega}^{-1/2} \boldsymbol{\Delta}(\mathbf{C}) + O(\boldsymbol{\Omega}^{-1}) \tag{76}$$

$$\frac{d\mathbf{C}}{dt} = \mathbf{J} \bullet \mathbf{C} + \mathbf{C} \bullet \mathbf{J}^T + \mathbf{D} + O(\boldsymbol{\Omega}^{-1/2}) \tag{77}$$

where $\mathbf{J}$ is the Jacobian matrix, $\langle \boldsymbol{\varepsilon} \rangle \equiv \left[\langle \varepsilon_1 \rangle \quad \langle \varepsilon_2 \rangle \quad \mathrm{L} \quad \langle \varepsilon_n \rangle \right]^T$ is the vector of the first moment, $\boldsymbol{\Delta}$ is a column vector whose s^{th} entry is the coefficients of $\Omega^{-1/2}$, $\mathbf{C}$ is a symmetric matrix with entries in the i^{th} row and j^{th} column given by $\langle \varepsilon_i \varepsilon_j \rangle$, and angle bracket denotes the mean.

$\boldsymbol{\Delta}\mathbf{C} \equiv (\sum_{i,j=1}^{N} \frac{\partial J_{ki}}{\partial \phi_j} \langle \varepsilon_i \varepsilon_j \rangle - \sum_{i=1}^{N} \phi_i \frac{\partial J_{ki}}{\partial \phi_i}) \mathbf{e}_k$, $\mathbf{e}_k$ is the k^{th} unit vector.

From these equations, the average value of each species can be calculated directly without usage of the SSA method for sample paths.

7.2.2 Linear operator for the diffusion

7.2.2.1 Reactions:

We assume R reactions involving the n species. The propensity of the r^{th} reaction occurring in the cell at (i, j) is denoted as $a_r\left(\mathrm{Z}_{i,j}\right)$. Next, we define stoichiometric matrix $\{\beta_{s,r}; s = 1,2,\mathrm{K}\,,n; r = 1,2,\mathrm{K}\,,R\}$ so that the rate of change of concentration of each species may be related to the reaction rate vector. Thus we may write the rates of change

of concentration of the s^{th} extracellular species in rectangular domain at (i, j) as $\sum_{r=1}^{R} \beta_{s,r} a_r \left(Z_{i,j} \right)$.

7.2.2.2 Mass Balances:

We have a set of stochastic processes $Z_{i,j}^{(s)}$ that must be simulated so that their average values $\left\langle Z_{i,j}^{(s)} \right\rangle$ can be obtained. The average concentrations must satisfy the differential equations

$$\frac{d\left\langle Z_{i,j}^{(s)} \right\rangle}{dt} = D_l \left(\left\langle Z_{i-1,j}^{(s)} \right\rangle + \left\langle Z_{i+1,j}^{(s)} \right\rangle + \left\langle Z_{i,j-1}^{(s)} \right\rangle + \left\langle Z_{i,j+1}^{(s)} \right\rangle - 4\left\langle Z_{i,j}^{(s)} \right\rangle \right) + \left\langle \sum_{r=1}^{R} \beta_{s,r} a_r \left(Z_{i,j} \right) \right\rangle, \quad (78)$$

$$s = 1, 2, \text{K}, n; \quad i = 0, \pm 1, \pm 2, \text{K}, \pm M - 1; \; j = 0, \pm 1, \pm 2, \text{K}, \pm N - 1$$

D_l is the diffusion coefficient for the l^{th} species through the biofilm. The last term on the right hand side is the change in concentration resulting from reactions. For the situation where no variable can escape through the boundaries of the biofilm, we may write the equations for the boundaries as shown below.

$$\frac{d\left\langle Z_{-M,j}^{(s)} \right\rangle}{dt} = D_l \left(\left\langle Z_{-M+1,j}^{(s)} \right\rangle + \left\langle Z_{-M,j-1}^{(s)} \right\rangle + \left\langle Z_{-M,j+1}^{(s)} \right\rangle - 3\left\langle Z_{-M,j}^{(s)} \right\rangle \right) + \left\langle \sum_{r=1}^{R} \beta_{s,r} a_r \left(Z_{-M,j} \right) \right\rangle, \quad (79)$$

$$s = 1, 2, \text{K}, n, \quad j \in S^J$$

$$\frac{d\left\langle Z_{M,j}^{(s)} \right\rangle}{dt} = D_l \left(\left\langle Z_{M-1,j}^{(s)} \right\rangle + \left\langle Z_{M,j-1}^{(s)} \right\rangle + \left\langle Z_{M,j+1}^{(s)} \right\rangle - 3\left\langle Z_{M,j}^{(s)} \right\rangle \right) + \left\langle \sum_{r=1}^{R} \beta_{s,r} a_r \left(Z_{M,j} \right) \right\rangle, \quad (80)$$

$$s = 1, 2, \text{K}, n, \quad j \in S^J$$

$$\frac{d\left\langle Z_{i,-N}^{(s)} \right\rangle}{dt} = D_l \left(\left\langle Z_{i-1,-N}^{(s)} \right\rangle + \left\langle Z_{i+1,-N}^{(s)} \right\rangle + \left\langle Z_{i,-N+1}^{(s)} \right\rangle - 3\left\langle Z_{i,-N}^{(s)} \right\rangle \right) + \left\langle \sum_{r=1}^{R} \beta_{s,r} a_r \left(Z_{i,-N} \right) \right\rangle, \quad (81)$$

$$s = 1, 2, \text{K}, n, \quad i \in S^I$$

$$\frac{d\left\langle Z_{i,N}^{(s)} \right\rangle}{dt} = D_l \left(\left\langle Z_{i-1,N}^{(s)} \right\rangle + \left\langle Z_{i+1,N}^{(s)} \right\rangle + \left\langle Z_{i,N-1}^{(s)} \right\rangle - 3\left\langle Z_{i,N}^{(s)} \right\rangle \right) + \left\langle \sum_{r=1}^{R} \beta_{s,r} a_r \left(Z_{i,N} \right) \right\rangle, \quad (82)$$

$$s = 1, 2, \mathrm{K}\ , n, \quad i \in S^I$$

7.2.2.3 Operator Formulation:

Eqs. (79) through (82) for each s may be succinctly written by defining the following operators.

$$\mathbf{L}^I : \Re^{2M+1} \to \Re^{2M+1}, \quad \mathbf{L}^J : \Re^{2N+1} \to \Re^{2N+1}, \quad \mathbf{I}^I \equiv \text{ Identity on } \Re^{2M+1}, \mathbf{I}^J : \text{Identity on } \Re^{2N+1}$$

$$\mathbf{T} \equiv \left(\mathbf{L}^I \otimes \mathbf{I}^J + \mathbf{I}^I \otimes \mathbf{L}^J\right) : \left(\Re^{2M+1} \otimes \Re^{2N+1}\right) \to \left(\Re^{2M+1} \otimes \Re^{2N+1}\right)$$

where

$$\mathbf{L}^I \equiv \begin{bmatrix} -2 & 1 & 0 & \mathrm{L} & & & & 0 \\ 1 & -2 & 1 & 0 & \mathrm{L} & & & 0 \\ 0 & 1 & -2 & 1 & 0 & \mathrm{L} & & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & & & & & \mathrm{M} \\ & & & & & & & 0 \\ 0 & \mathrm{L} & & & & 1 & -2 & 1 \\ 0 & & & & \mathrm{L} & 0 & 1 & -2 \end{bmatrix} \quad \mathbf{L}^J \equiv \begin{bmatrix} -2 & 1 & 0 & \mathrm{L} & & & & 0 \\ 1 & -2 & 1 & 0 & \mathrm{L} & & & 0 \\ 0 & 1 & -2 & 1 & 0 & \mathrm{L} & & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & & & & & \mathrm{M} \\ & & & & & & & 0 \\ 0 & \mathrm{L} & & & & 1 & -2 & 1 \\ 0 & & & & \mathrm{L} & 0 & 1 & -2 \end{bmatrix}$$

$$(2M+1) \times (2M+1) \qquad\qquad (2N+1) \times (2N+1)$$

The above are symmetric, tridiagonal (Jacobi) matrices with real eigenvalues and orthogonal eigenvectors. Note further that if we had chosen different sizes for the discrete reaction domains, we will have still had a self-adjoint operator [76]. Denoting the eigenvalues and eigenvectors of $\mathbf{L}^I$ and $\mathbf{L}^J$ by

$$\left\{\lambda_i^I, \mathbf{z}_i^I; \quad i \in S^I\right\}, \left\{\lambda_j^J, \mathbf{z}_j^J; \quad j \in S^J\right\}$$

they are identified (see Amundson[27]) as

$$\lambda_i^I = -2\left(1 + \cos\frac{\pi(i+M)}{2(M+1)}\right), \quad i \in S^I$$

$$\mathbf{z}_i^I = \left[\sum_{k=1}^{2M+1}\left(\frac{\sin\left[\pi(i+M+1)k/(2M+2)\right]}{\sin\left[\pi(i+M+1)/(2M+2)\right]}\right)^2\right]^{1/2} \times \begin{bmatrix} \dfrac{\sin\left[\pi(i+M+1)(2M+1)/(2M+2)\right]}{\sin\left[\pi(i+M+1)(2M+2)\right]} \\ -\dfrac{\sin\left[\pi(i+M+1)(2M)/(2M+2)\right]}{\sin\left[\pi(i+M+1)(2M+2)\right]} \\ \dfrac{\sin\left[\pi(i+M+1)(2M-1)/(2M+2)\right]}{\sin\left[\pi(i+M+1)(2M+2)\right]} \\ \mathrm{M} \\ \mathrm{M} \\ (-1)^{2M} \end{bmatrix}$$

$$\lambda_j^J = -2\left(1+\cos\frac{\pi(j+N)}{2(N+1)}\right), \quad j \in S^J$$

$$\mathbf{z}_j^J = \left[\sum_{k=1}^{2N+1}\left(\frac{\sin\left[\pi(j+N+1)k/(2N+2)\right]}{\sin\left[\pi(j+N+1)/(2N+2)\right]}\right)^2\right]^{1/2} \times \begin{bmatrix} \dfrac{\sin\left[\pi(j+N+1)(2N+1)/(2N+2)\right]}{\sin\left[\pi(j+N+1)(2N+2)\right]} \\ -\dfrac{\sin\left[\pi(j+N+1)(2N)/(2N+2)\right]}{\sin\left[\pi(j+N+1)(2N+2)\right]} \\ \dfrac{\sin\left[\pi(j+N+1)(2N-1)/(2N+2)\right]}{\sin\left[\pi(j+N+1)(2N+2)\right]} \\ \mathrm{M} \\ \mathrm{M} \\ (-1)^{2N} \end{bmatrix}$$

The operator $\mathbf{T}$ is readily shown to be self-adjoint relative to the regular inner product in $\mathcal{H}$ given by $\left\langle \mathbf{u}^I \otimes \mathbf{u}^J, \mathbf{v}^I \otimes \mathbf{v}^J \right\rangle = \left\langle \mathbf{u}^I, \mathbf{v}^I \right\rangle_I \left\langle \mathbf{u}^J, \mathbf{v} \right\rangle_J$, where subscripts I and J are used to represent the regular inner products in $\Re^{2M+1}$ and $\Re^{2N+1}$ respectively. The self-

adjointness of $\mathbf{T}$ follows from that of $\left(\mathbf{L}^I \otimes \mathbf{I}^J\right)$ and $\mathbf{I}^I \otimes \mathbf{L}^J$. We show the self-adjointness of $\left(\mathbf{L}^I \otimes \mathbf{I}^J\right)$ below.

$$\left\langle \left(\mathbf{L}^I \otimes \mathbf{I}^J\right)\left(\mathbf{u}^I \otimes \mathbf{u}^J\right), \mathbf{v}^I \otimes \mathbf{v}^J \right\rangle = \left\langle \left(\mathbf{L}^I \mathbf{u}^I \otimes \mathbf{u}^J\right), \mathbf{v}^I \otimes \mathbf{v}^J \right\rangle = \left\langle \mathbf{L}^I \mathbf{u}^I, \mathbf{v}^I \right\rangle_I \left\langle \mathbf{u}^J, \mathbf{v}^J \right\rangle_J$$
$$= \left\langle \mathbf{u}^I, \mathbf{L}^I \mathbf{v}^I \right\rangle_I \left\langle \mathbf{u}^J, \mathbf{v}^J \right\rangle_J = \left\langle \left(\mathbf{u}^I \otimes \mathbf{u}^J\right), \left(\mathbf{L}^I \mathbf{v}^I \otimes \mathbf{v}^J\right) \right\rangle = \left\langle \mathbf{u}^I \otimes \mathbf{u}^J, \left(\mathbf{L}^I \otimes \mathbf{I}^J\right)\left(\mathbf{v}^I \otimes \mathbf{v}^J\right) \right\rangle$$

The self-adjoint projections of $\mathbf{T}$ are given by

$$\left(\mathbf{P}_i^I \otimes \mathbf{P}_j^J\right)(\ ) = \left\langle \mathbf{z}_i^I, (\ ) \right\rangle_I \left\langle \mathbf{z}_j^J, (\ ) \right\rangle_J$$

7.2.2.4 Spectral Representation of T and its Functions:

$$\mathbf{T} = \sum_{i=-M}^{M} \sum_{j=-N}^{N} \left(\lambda_i^I + \lambda_j^J\right)\left(\mathbf{P}_i^I \otimes \mathbf{P}_j^J\right)$$

Since in the sequel, it would be necessary to obtain $e^{\mathbf{T}}$, we identify it as

$$e^{\mathbf{T}} = \sum_{i=-M}^{M} \sum_{j=-N}^{N} e^{\left(\lambda_i^I + \lambda_j^J\right)}\left(\mathbf{P}_i^I \otimes \mathbf{P}_j^J\right)$$

7.2.2.5 Towards Stochastic Simulation:

The differential equations (79)-(82) can be concisely written as

$$\frac{d\langle \mathbf{Z} \rangle}{dt} = \mathbf{DT}\langle \mathbf{Z} \rangle + \mathbf{Ba}(t) \tag{83}$$

where $\mathbf{B} \equiv \{\beta_{sr}; s = 1,2,\mathrm{K}\ ,n; r = 1,2,\mathrm{K}\ ,R\}$; $\mathbf{a} \equiv \left[a_1 \quad a_2 \quad \mathrm{L} \quad a_R\right]^T$. It is readily shown that the operator $\mathbf{DT}$ is self-adjoint with respect to the inner product $[\mathbf{u}, \mathbf{v}] \equiv \left\langle \mathbf{D}^{-1}\mathbf{u}, \mathbf{v} \right\rangle$ where $\langle , \rangle$ is the inner product with respect to which $\mathbf{D}$ and $\mathbf{T}$ are both self-adjoint.

The foregoing differential equation is to be integrated from $t = \tau_p$, where τ_p is the p^{th} discrete time. Eq. (71) may be integrated subject to the value of $\left\langle \mathbf{Z}\left(\tau_p\right)\right\rangle$. The solution may be written as

$$\left\langle \mathbf{Z}(t)\right\rangle = e^{\left(t-\tau_p\right)\mathbf{DT}}\left\langle \mathbf{Z}\left(\tau_p\right)\right\rangle + \int_{\tau_p}^{t} e^{\left(t'-\tau_p\right)\mathbf{DT}}\left\langle \mathbf{Ba}(t')\right\rangle dt' \tag{84}$$

The first term on the right side of the above equation captures the change in species due to diffusion occurring between neighboring compartments and the second term describes the change in particle numbers as the result of reactions. The second term on the right hand side of equation (84) becomes difficult to solve as the system involves high order reactions. Grima's EMRE method[1], briefly discussed in the earlier section, provides a tool to compute this average change and hence complete the full calculations for this reactive-diffusive system.

7.3 Examples

7.3.1 Diffusion augmented Schlogl's system:

The well-known system Schlogl has been well studied and used widely to illustrate new simulation algorithms. The system involves two reversible stochastic relations and the environment can be assumed to be well-mixed. Here in this paper, we attempted to expand the system and allow diffusion to occur within the domain of interest. We also perturb the conditions slightly so that we can still relate our simulation results for the diffusive reactive system with that of the original single cell case. The Schlogl's chemical reaction model is shown in [63]. This model is noted for its bistability. B_1 and B_2 are constants with their particle numbers as N_1 and N_2, respectively

$$B_1 + 2X \underset{c_2}{\overset{c_1}{\rightleftarrows}} 3X, \quad B_2 \underset{c_4}{\overset{c_3}{\rightleftarrows}} X$$

The values of (dimensionless) parameters, adapted from Cao et al.[41], are $c_1 = 3\times10^{-7}$, $c_2 = 10^{-4}$, $c_3 = 10^{-3}$, $c_4 = 3.5$, $N_1 = 1\times10^5$, and $N_2 = 2\times10^5$. The initial condition of X is 250 and we simulate the system up to $t = 4$.

For this application, we allow the same set of reactions to occur within each squared sub-domain and the diffusion between different neighbor sub-units. Also the particle numbers of B_1 and B_2 in the bulk surrounding the domain is maintained to be the same as those in the single cell case. We set the particle numbers of all species in all compartments to be zero and set diffusivity to be 0.1.

7.3.2 Bimolecular reactions-two model problems:

This is a heteroreaction reaction which involves two species A and B. The example is subjected to two reactions below:

$$A + B \xrightarrow{k_1} B, \qquad \varnothing \xrightarrow{k_2} A$$

In the first reaction, A is decayed whereas B acts as a catalyst. The second reaction is coupled with the first and represents the generation of A. If we denote a the area of the cell. The values for parameters are $k_1 / a = 0.2s^{-1}$, $k_2 a = 1s^{-1}$, $A(0) = 5$, and $B(0) = 1$

7.4 Discussion:

Two main examples have been utilized to illustrate how effectively the proposed method works. The first example is the diffusion augmented Schlogl's system. Figure 5.1a shows the distribution of X at $t = 4$ by applications of two different approaches: Langevin's equation [77]–[79] and Master equation with 30,000 sample paths each. The figure shows

a well-known feature of this system, the bi-modal distribution. In figure 5.1b, we have shown the distribution X in the central compartment of the domain using RDME.

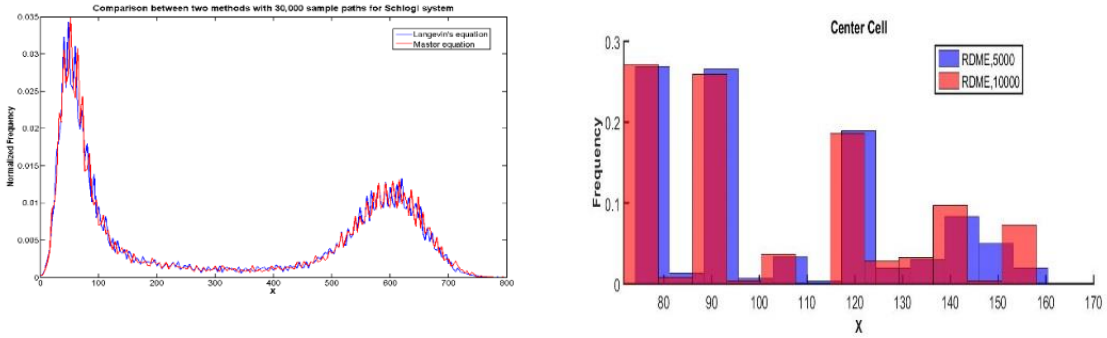

Figure 7.1: Schlogl's system (a) Distribution of X in the case of single cell simulation (on left), and (b) Distribution of X in the center compartment in reactive diffusive system (on right)

From the figure, there is not much difference between two distributions, coming from 5000 sample-path vs. 10000 sample-path simulations. The number of sample paths is significantly lower than that in the case of the single cell as the time step is chosen to be the smallest of those generated for all the compartments. This lower time step reduces the error in the propensity function due to its being fixed during the time interval. This comparison suggests that beyond 5000 sample paths, simulation produces no significant difference in X distribution. The first interesting feature from this figure is that the particle number is a lot lower as compared to that in the single cell case. All inner compartments have all species set to be zeros initially. As a result of that, there is only diffusion for some period of time through which the materials are transferred from the bulk to the inner cells. Reactions first take place at the outer regions of the domain and subsequently in the inner regions as diffusion occurs and transfers materials from the periphery to the interior.

Specifically, in the center compartment, there is no species initially thus nothing occurs until much later, and this explains why the particle number of X is much lower in this case. Another aspect of this plot is that even though the range of X is reduced, the bimodal nature of the distribution of X (observed in the single cell case) is retained. This is a consequence of our choice of boundary and initial conditions. In other words, fixing the bulk concentrations in the exterior, provides for some uniformity with time because of diffusion thus approaching the single cell circumstance. The center cell is also located at the position that is most symmetric spatially, hence receives the most steady supply of particles from all directions because of diffusion. Consequentially, X distribution in this case retains the bimodality observed in the single cell case. To illustrate further, consider Figure 7.2b which shows the X distribution in a compartment located in the outer-most layer.

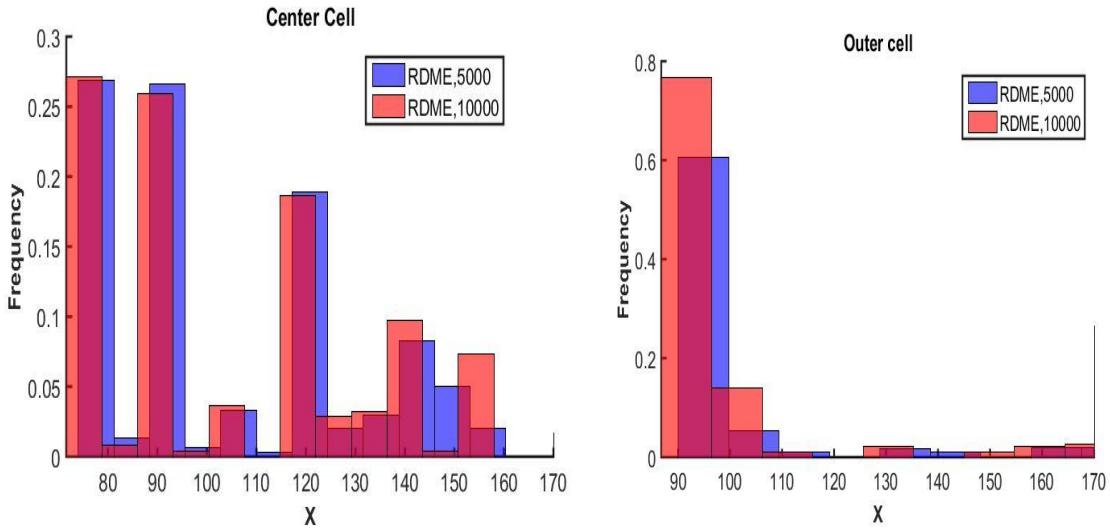

Figure 7.2: Distribution of X in (a) center compartment (on left), and (b) outer compartment (on right)

Unlike the previous case shown in Figure 7.2a, the distribution shows no bimodal distribution for which our hypothesis is as follows. Due to the gradient in concentration between the bulk (source) and inner compartments (sink), there is a continuous flow-in and flow-out which gives rise to a very different set of conditions, as compared to the single cell case. On top of it, concentration of species in neighboring compartments is not uniform, differentiating this case from the previous case of the center compartment where the concentration of species around the compartment is relatively uniform. In Figure 5.3, we compare the averaged value of X in the center compartment at different time points generated from two different method: RDME method (10000 sample paths) and our method.

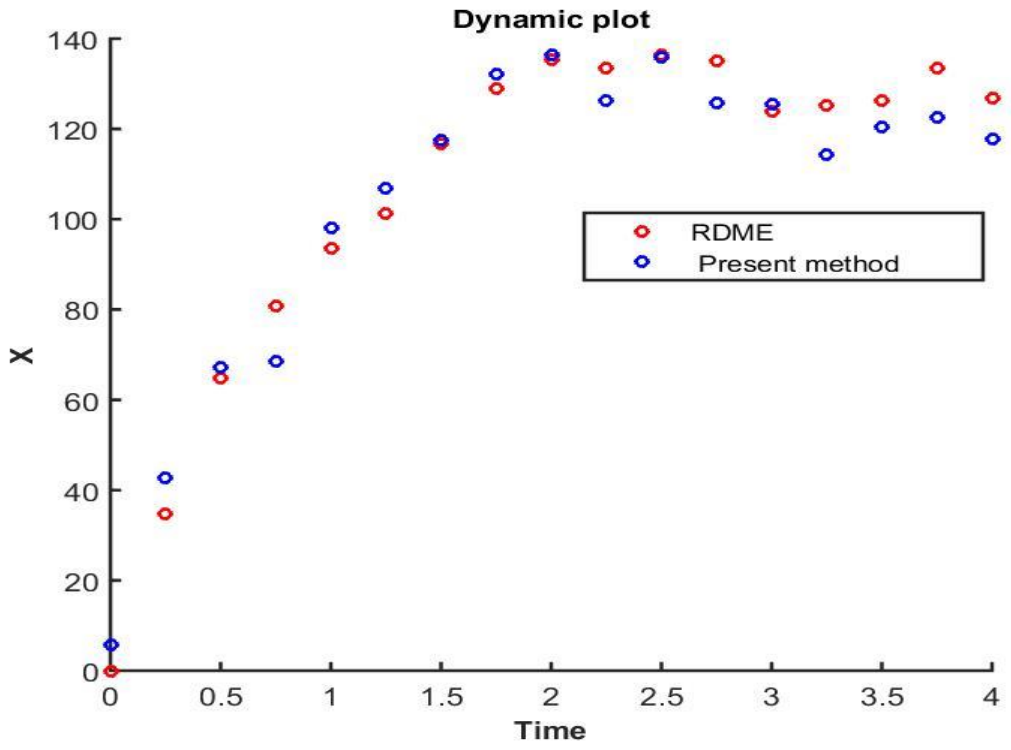

Figure 7.3: Dynamic plot of averaged X in the center compartment

The average value of X can be calculated as the sum of all products between X value collected from all realization and its corresponding frequency. The data points generated from our methods show close prediction to those simulated by RDME. The averaged relative error from all these points is only 0.12. To further investigate the validity of our approach, we now fix the final time point $t = 4$ and vary the positions. Figure 7.4 shows the average value of X at different compartments within the whole domain.

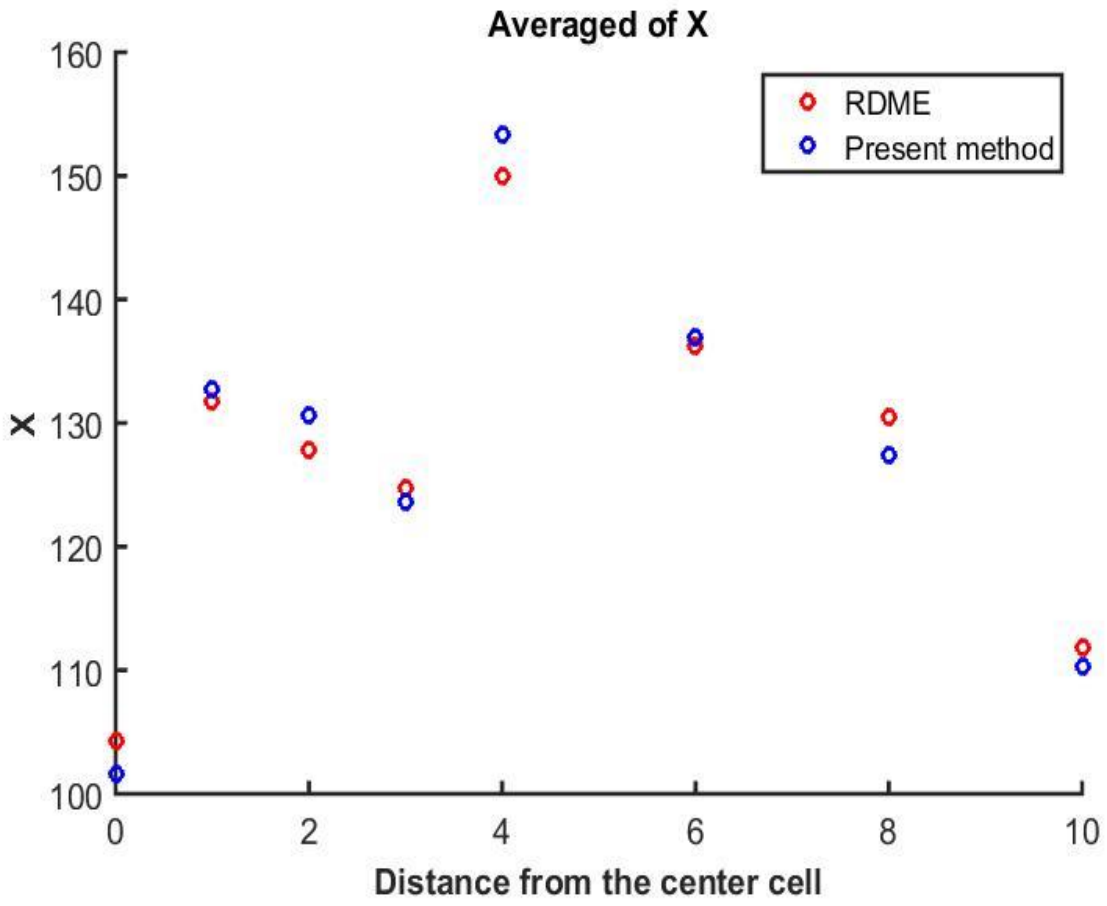

Figure 7.4: Average value of X at different locations

Horizontal axis indicates how many cell units that are between the compartment of interest and the center compartment. The two methods produce close predictions for the averaged value of X at various compartments within the domain. The averaged relative error is calculated to be 0.09. The current algorithm outperforms the RDME in term of CPU time. The simulation time for the present method is equivalent with the amount of time it would take the RDME to simulate three trajectories.

The second example is adopted from[80] except that we modify the rate constant unit and diffusivity unit for reactions and diffusion since our results are simulated for the domain in two dimensions. Similar to the derivation in[80] the solution for the master equation in the well-mixed case can be solved analytically in the similar manner and arrives as:

$$\phi(n) = \frac{1}{n!}(\frac{k_2 a^2}{k_1 B_0})^n \exp[-\frac{k_2 a^2}{k_1 B_0}],$$

where the term on the left indicates the probability that there are n molecules of A in the system. Figure 7.5 shows the distribution of A from generating 10000 sample paths in SSA method and it is confirmed to have the same results as from equation (72) [80]

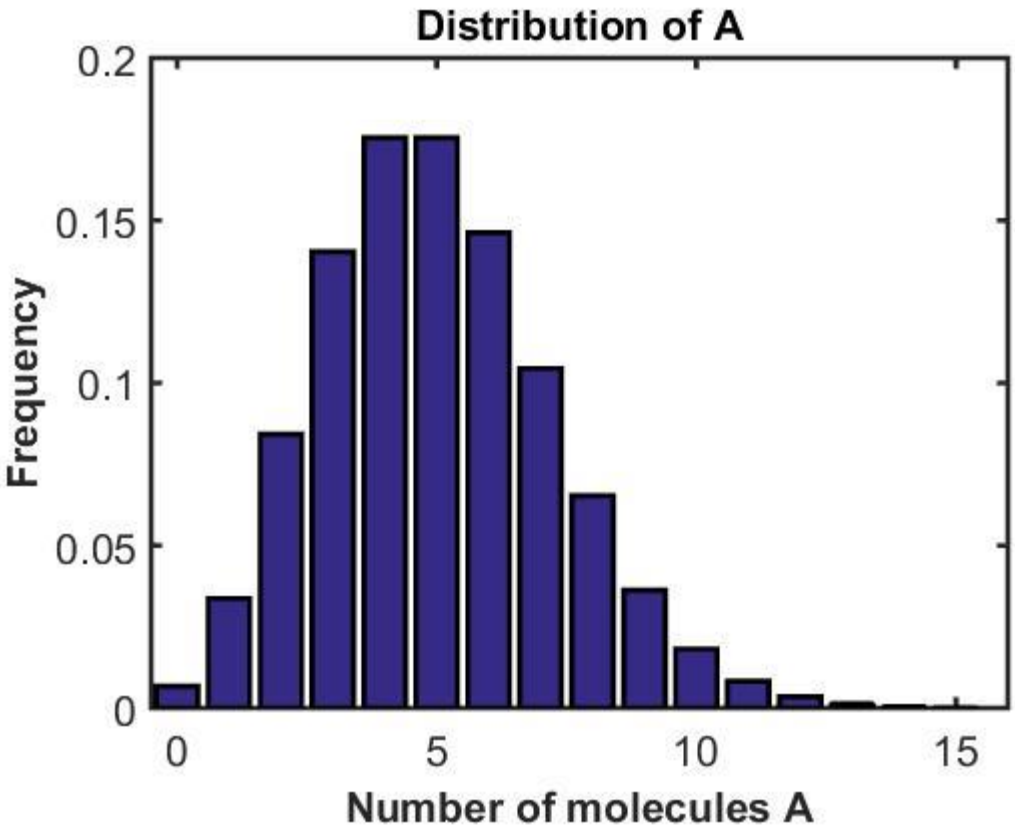

Figure 7.5: Stationary distribution of A

For example 2, the whole domain is composed of 21x21 compartments. $D_A = D_B = 1\ \mu m^2\ s^{-1}$, L=1$\mu m$, $k_1 = 0.2(\mu m)^{-2}\ s^{-1}$, $k_2 = 1(\mu m)^{-2}\ s^{-1}$. Within any compartment (i, j), the propensity function for the reactions can be computed according to: $\alpha_{ij,1}(t) = k_1 A_{ij}(t) B_{ij}(t) / l^2$, $\alpha_{ij,2}(t) = k_1 l^2$.

where l^2 is the area of each compartment. In a similar manner, we can write the propensity function for the diffusion of A and B as $A_{ij} D_A / l$ and $B_{ij} D_B / l$. Also, we find the ratio between the propensity function of diffusion to that of reaction to be about 100, indicating that diffusion is a lot quicker than reactions in this case. In Figure 7.6, we present the three

main distributions: the stationary distribution generated from equation (72), the distribution generated from the usage of SSA when the domain is divided into 21x21 compartments, and the distribution generated by our present method. We denote them as distribution (I), distribution (II) and distribution (III), respectively.

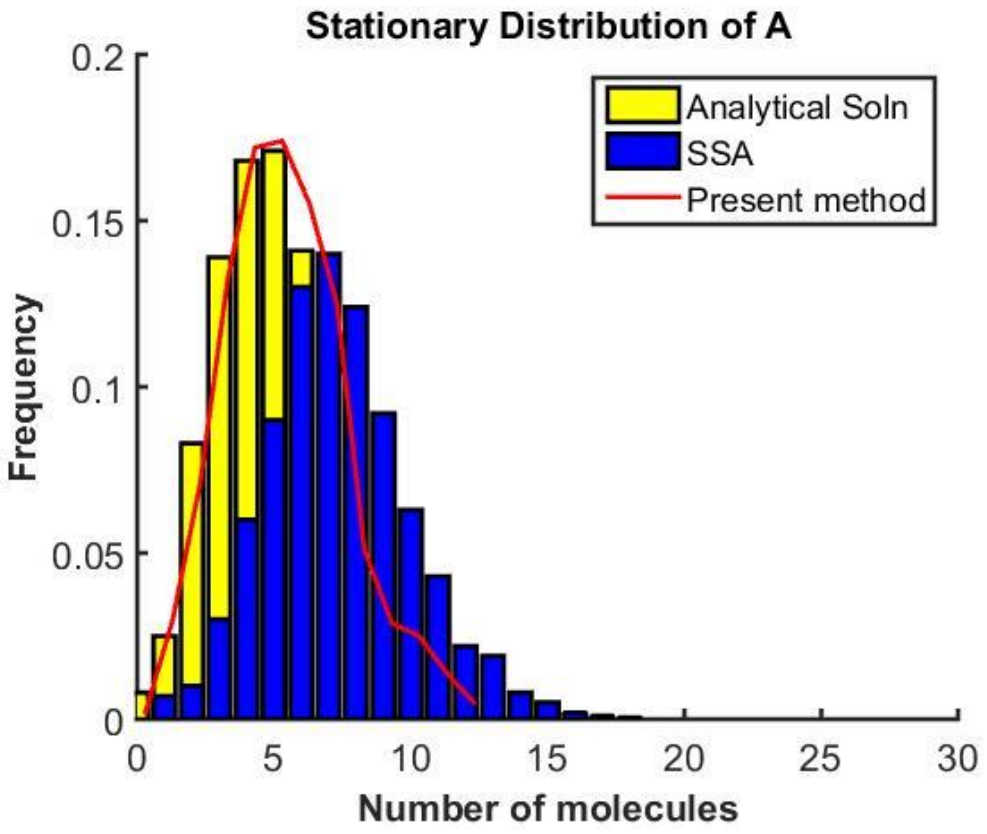

Figure 7.6: Stationary distribution of A

Distribution (II) shifts to the right as compared to the first distribution. This movement has been discussed in the work of Isaacson [35]: in the theoretical limit $l \to 0$, the bimolecular reaction $A + B \to \phi$ is lost and the compartment model can only recover the diffusion part. In the example, the second reaction is a zero-th order reaction and so can be computed in term of a total production of A in the whole domain, which is independent of choice l. The bimolecular reaction, on the other hand, is a second order reaction and hence its rate of

consumption of A decreases as l is reduced. These two effects result in a larger amount of A at the end of the process in the domain. Generally, there is an agreement in the literature that there is a bound on l so that $l \gg k_1 / \left(D_A + D_B \right)$ [35], [75]. That also explains why when l becomes small, the regular SSA might not necessarily predict accurately the behavior of this system. However, distribution III generated by our method produces similar prediction to the stationary distribution I. Figure 7.6 indicates that our method can overcome the problem associated with the selection of compartment size. In this example, the simulation time for the present algorithm is as small as it would take for the SSA to complete on two independent trajectories.

7.5 Conclusions:

This paper introduces an algorithm that is capable of capturing the average behaviors of chemical/biological species in a reactive-diffusive system without generating multitudes of sample paths. The method combines EMRE [36] and Linear Operator techniques to describe reaction and diffusion effects simultaneously. Two examples have been used to illustrate the validity of the method. Predictions generated by this algorithm show good agreement with the SSA and require a much lower CPU time for simulations. Extension of this method to larger systems involving many species appears to be feasible.

8. DIRECT AVERAGE CALCULATIONS FOR STOCHASTIC PROCESSES

8.1 Introduction

Modeling and simulation of chemical/biological systems is an active area of research towards as it holds the key to their quantitative understanding. Evolution of those systems has traditionally relied on deterministic rate laws. However, many contemporary applications involve small systems that display inherently stochastic dynamics as a consequence of their comprising randomly behaving constituents. A common example is that of a system harboring a small number of molecular entities that chemically react in a random manner. Such systems arise routinely in signaling processes in cells, synthesis of nano materials and on so. The system variables are present in low molecule numbers and consequently the intrinsic noise in beharios is significant [68], [69], [81]–[88]. The Chemical Master Equation can be written to formulate the behavior of these systems. However, its solution is combinatorically complex so that the system behavior must necessarily be obtained by stochastic simulation. In this regard, the exact method of Gillespie [22] or the equivalent quiescence interval approach of Shah et al. [23] provide an alternative route to solving for the system behavior. This simulation method relies on creating every constituent event by generating random numbers satisfying exactly calculated distribution for the time interval between events. However, the strategy of capturing every single event in the exact method results in a substantial computation time. Moreover, biochemical systems are generally large and complex networks, which are composed of a large number of species, at a wide range of molecule numbers, undergoing

reactions at different time-scales. As a result, the usage of the exact algorithm can become computationally expensive for obtaining solutions for those systems. A modified version of exact method was introduced by Gibson and Bruck [41]. Several approximation strategies, such as tau-leaping methods [19], [24], [27], [28], have been proposed in subsequent years in an attempt to reduce CPU time without sacrificing much accuracy. The tau-leaping strategy at any instant is contingent upon satisfying an approximation of the process at a future time. Since any specific realization of the process may, however, be in conflict with the proposed approximation, Ramkrishna et al. [31] incorporated the Chebyshev inequality to produce a modified tau-leap strategy to assure the approximation with a suitably specified probability. As this significantly reduces the number of "delinquent" sample paths the simulation is made more efficient. A further attempt to speed up simulations was accomplished in [32] by simulating over each infinitesimal interval multiple trajectories in parallel. Since simulation is performed over a prescribed time interval, this modified strategy, which allows some trajectories to advance faster than others, provides for termination of those sample paths that have met the time constraint. This parallelization results in the reduction of CPU time because the number of sample paths diminishes in course of time.

The foregoing simulation methods generate the sample paths of the entire process over a chosen period of time, which approximate the probability space of the stochastic process being analyzed. The collection of sample paths serves as the source of the probabilities of any information associated with the system. Modelers, however, are often satisfied with the temporal evolution of the system in terms of only its average behaviors and the average fluctuation(such as the standard deviation) of the system over a period of time. This paper

is concerned with enabling the calculation of the aforementioned averages without undergoing the enormous computational burden of computing entire sample paths. This approach has been demonstrated for a reaction diffusion system [38] by combining a method of Grima [36] based on quasilinearization for reactions with an operator-theoretic formation of discretized diffusion.

Unlike the exact and tau-leaping methods, one can also approximate the solution of the master equation by formulating the equations in a continuous version (i.e. Stochastic Differential Equations(SDE)) [79], [89]–[92]. SDE has shown its potential in describing system behaviors accurately in various applications in chemistry, physics and biology [19], [91]–[95]. Two main approaches for solving these types of equations are known as explicit and implicit numerical methods. In the explicit approach [70], [94], [98], [99] approximate variables at each time point can be computed using the previous time-point values. This strategy is easy to implement and works well for non-stiff problems. However, due to the poor stability property, explicit methods are required to reduce step sizes significantly when applied to systems associated with stiff behaviors. To solve this issue, various types of implicit methods have appeared in the literature [100]–[105]. These versions have a higher stability property and hence can be used to capture stiff systems more accurately. However, the implicit methods involve solving a high number of algebraic equations at each time step and therefore can also result in adding to CPU time. Yin et al. [106] proposed a modified version of the Milstein method, which avoids solving nonlinear algebra equations. The explicit equation is coupled with another correction equation in which a correct term is introduced to reduce the error associated with the explicit approximation. The method also shows good mean-square stability.

Each sample path therefore contains little information towards the actual average behavior of the system. On the other hand algorithms for solving SDE's invariably are crafted over intervals small enough to permit truncation of Taylor series expansion of nonlinearities in the system to retain at most the second order terms to produce the random process state at the $(n+1)^{st}$ discrete instant conditional on specification of the process at the n^{th} instant and random variables(which arise from stochastic integration with known expectation for their average and variance). Thus one obtains algebraic equations for the process average and its standard deviation from it at the $(n+1)^{st}$ discrete instant in terms of those at the n^{th} instant. By avoiding simulating a large number of trajectories, the method can speed up the simulations significantly.

8.2 Milstein's method and its advance version for stiff systems

Generic $d-$dimensional SDE has the following form:

$$dX(t) = f(t, X(t))dt + \sum_{j=1}^{m} g^j(t, X(t))dW^j(t), \qquad X(t_0) = X_0 \quad t \in [t_0, T] \tag{85}$$

where $f : \mathbb{R}^d \to \mathbb{R}^d$ is the drift coefficient, $g^j, j = 1, 2, ..., m : \mathbb{R}^d \to \mathbb{R}^d$ is the diffusion coefficient and $W_j(t)$ is the standard Wiene process with the property of $\Delta W_j(t) = W_j(t + \Delta t) - W_j(t)$ is a Gaussian random variable The SDE interpreted in the Ito sense, for the case $d = 1$, has three main schemes:

- The explicit Milstein method:

$$X_{n+1} = X_n + f(t_n, X_n)h + g(t_n, X_n)\Delta W_n + \frac{1}{2}(g(t_n, X_n)g'(t_n, X_n))(\Delta W_n^2 - h) \tag{86}$$

- The semi-implicit Milstein method:

$$X_{n+1} = X_n + f(t_{n+1}, X_{n+1})h + g(t_{n+1}, X_{n+1})\Delta W_n + \frac{1}{2}(g(t_n, X_n)g'(t_n, X_n))(\Delta W_n^2 - h) \quad (87)$$

- The implicit Milstein method:

$$X_{n+1} = X_n + f(t_{n+1}, X_{n+1})h + g(t_{n+1}, X_{n+1})\Delta W_n + \frac{1}{2}(g(t_{n+1}, X_{n+1})g'(t_{n+1}, X_{n+1}))(\Delta W_n^2 - h) \quad (88)$$

The improved Milstein method for stiff systems:

$$Z_{n+1} = \overline{Z}_{n+1} + (1 - hf'(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n)) \quad (89)$$

$$\overline{Z}_{n+1} = Z_n + hf(Z_n) + g(Z_n)\Delta W_n + \frac{1}{2}g'(Z_n)g(Z_n)(\Delta W_n^2 - h) \quad (90)$$

where Z_n is the approximation of the exact solution $X(t)$ at time $t_n = nh$. The term $(1 - hf'(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n))$ is added as a correction term and $\overline{Z}_{n+1}$ is computed using the classical explicit Milstein method.

Expanding the formula to the vector case when $d > 1$ we then have the classical Milstein method as follows:

$$X_{n+1} = X_n + f(t_n, X_n)h + g(t_n, X_n)\Delta W_n + \sum_{j_2=1}^{m}\sum_{j_1=1}^{m} L^{j_1} g^{j_2}(X_n) I_{(j_1, j_2)}, \quad (91)$$

where $$L^{j_1} = \sum_{i=1}^{d} g^{i, j_1} \frac{\partial}{\partial x^i}, \quad (92)$$

$$I_{(j_1, j_2)} = \int_{t_n}^{t_{n+1}} \int_{t_n}^{s_2} dW^{j_1}(s_1) dW^{j2}(s_2) \quad (93)$$

With x^i and g^{i, j_2} are the $i-th$ element of the vector functions x and g^{j_1}

And so the formula to approximate Z is:

$$Z_{n+1} = \overline{Z}_{n+1} + (I - hf^{'}(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n)) \tag{94}$$

$$\overline{Z}_{n+1} = Z_n + hf(Z_n) + \sum_{j=1}^{m} g^j(Z_n)\Delta W_n^j + \sum_{j_2=1}^{m}\sum_{j_1=1}^{m} L^{j_1} g^{j_2}(Z_n) I_{(j_1,j_2)} \tag{95}$$

where I is the $d-$dimensional identity matrix and $f^{'}, g^{'}$ stand for Jacobian matrix of the vector-valued function f, g

8.3 Development of direct-average calculation:

Let us first apply Taylor's expansion on equations (94) followed by taking expected operator to both sides of equation (94) and (95):

$$EZ_{n+1} = E\overline{Z}_{n+1} + E(I - hf^{'}(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n)) \tag{96}$$

$$\overline{EZ}_{n+1} = EZ_n + Ehf(Z_n) + E\sum_{j=1}^{m} g^j(Z_n)\Delta W_n^j + E\sum_{j_2=1}^{m}\sum_{j_1=1}^{m} L^{j_1} g^{j_2}(Z_n) I_{(j_1,j_2)} \tag{97}$$

Derivation of the entire development can be found in Supplemental material part. The final forms of (96) and (97) are as follows:

$$\begin{aligned}
EZ_{n+1}^{j} = {} & E\overline{Z}_{n+1}^{j} + h\{f^j(E\overline{Z}_{n+1}) + 1/2E[(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^T Hf^j(Z)|_{\overline{EZ}_{n+1}} (\overline{Z}_{n+1} - E\overline{Z}_{n+1})] + \\
& - f^j(EZ_n) - 1/2(Z_n - EZ_n)^T Hf^j(Z)|_{EZ_n} (Z_n - EZ_n)\} + \\
& + h^2\{f^j(\overline{EZ}_{n+1}) D(f^{j_2})^{j_3}(\overline{Z}_{n+1})|_{\overline{EZ}_{n+1}} + \\
& + 1/2E\{(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^T \{(DD(f^{j_2})^{j_3}(Z) Df^j(Z)^T|_{\overline{EZ}_{n+1}} + D(f^{j_2})^{j_3}(Z) Hf^j(Z)^T|_{\overline{EZ}_{n+1}} + \\
& + Df^j(Z) DD(f^{j_2})^{j_3}(Z)]|_{\overline{EZ}_{n+1}} + f^j(E\overline{Z}_{n+1}) HD(f^{j_2})^{j_3}(Z)]|_{\overline{EZ}_{n+1}}\}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})\}\} - \\
& - h^2\{f^{j_1}(EZ_n) D(f^{j_2})^{,j_3}(Z)|_{\overline{EZ}_{n+1}} \\
& + 1/2E\{f^{j_1}(EZ_n)(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^T HD(f^{j_2})^{,j_3}(Z)|_{\overline{EZ}_{n+1}} (\overline{Z}_{n+1} - E\overline{Z}_{n+1}))\} \\
& + E\{(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^T Df^{j_1}(Z)|_{\overline{EZ}_{n+1}} DD(f^{j_2})^{,j_3}(Z)|_{\overline{EZ}_{n+1}} (Z_n - EZ_n)\} \\
& + 1/2\{(Z_n - EZ_n)^T Hf^{j_1}(Z)|_{\overline{EZ}_{n+1}} D(f^{j_2})^{,j_3}(Z)|_{\overline{EZ}_{n+1}} (Z_n - EZ_n))\}\}
\end{aligned} \tag{98}$$

$$E\overline{Z}_{n+1}^{j} = EZ_n^j + hf^j(Z_n) + E\sum_{j_2=1}^{m}\sum_{j_1=1}^{m} g^{j_1} D(g^{j_2})^j(Z_n) EI_{(j_1,j_2)} \quad (99)$$

$$= EZ_n^j + hf^j(Z_n) + g^{j_1}(EZ_n) D(g^{j_2})^j(Z)|_{EZ_n}$$

$$+1/2E\{(Z_n - EZ_n)^T \{(DD(g^{j_2})^j(Z) Dg^{j_1}(Z)^T|_{EZ_n}$$

$$+ D(g^{j_2})^j(Z) Hg^{j_1}(Z)^T|_{EZ_n} +$$

$$+\mathrm{D}g^{j_1}(Z) DD(g^{j_2})^j(Z)]|_{EZ_n} + g^{j_1}(EZ_n) HD(g^{j_2})^j(Z)]|_{EZ_n}\}(Z_n - EZ_n)\}*$$

$$* E(\int_{t_n}^{t_{n+1}} \int_{t_n}^{s_2} dW^{j_1}(s_1) dW^{j_2}(s_2))^{\cdot j}$$

Where Z_{n+1}^j is the $j-$th component of Z_{n+1}. $H, D,$ and I are Hessian matrix, matrix of partial derivative and identity matrix, respectively.

Every term in the right equations (98) and (99) can be evaluated directly at the mean value of the previous point, therefore we can compute directly EY_{n+1} once we can compute $E(\overline{Z}_{n+1}^{j_1} - \overline{EZ}_{n+1}^{j_1})(\overline{Z}_{n+1}^{j_2} - \overline{EZ}_{n+1}^{j_2})$. In order to evaluate $E(\overline{Z}_{n+1}^{j_1} - \overline{EZ}_{n+1}^{j_1})(\overline{Z}_{n+1}^{j_2} - \overline{EZ}_{n+1}^{j_2})$, at each current time point we generate a sample of 100 points at next time step using equation (89) and (90)

8.4 On usage of the method to current biological systems:

For a chemical/biological system which is composed of d species undergoing m reactions:

$$s_{1,j}X_1 + s_{2,j}X_2 + \cdots + s_{d,j}X_N \xrightarrow{k_j} r_{1,j}X_1 + r_{2,j}X_2 + \cdots + r_{d,j}X_d, \quad j = 1, 2, \ldots, m$$

f g can be represented using regular macroscopic rate law (or written in propensity functions when converted into particle numbers):

$$f = \sum_{j=1}^{m} \nu_j a_j(X)$$

The diffusion coefficient term can also be computed as follows:

$$g = \sum_{j=1}^{m} \nu_j \sqrt{a_j(X)}$$

Specifically for our system of plasmid transfer, most variables exists in low numbers and hence it explains for the usage of power series shown in the Supplemental Materials.

8.5 Example:

To illustrate how the method works, the system of *Enterrococus faecalis* is selected. The system is composed of 17 variables and 45 reactions. *Enterococcus faecalis* utilizes the mechanism of conjugation to transfer antibiotic resistance from plasmid-harboring antibiotic-resistant donors to plasmid-free antibiotic-sensitive recipients. The plasmid carrying the tetracycline resistance is known as pCF10. Two types of signaling molecules which regulate conjugative transfer of the plasmid pCF10 are inhibitor iCF10 and inducer cCF10. cCF10 molecules, responsible for inducing conjugation, are generated by recipient cells. Donor cells, on the other hand, produce iCF10, whose role is to repress conjugation. When the inducer concentration is high, several cascade reactions occur and result in activation of conjugations. Q_L, one of the key variables, indicates the level of conjugations. Therefore, Q_L level increases when more inducer is present and decreases when concentration of inhibitor is high.

8.6 Results and Discussion:

The system that is not only composed of complex reaction network but also includes variables with stiff behaviors is indeed a good example to test the validity of this method. We simulate and compare the dynamics of key variables under different conditions using different method by using SSA as the benchmark. Results generated by utilizing the regular

sample path method and the proposed direct-average method are together. Figure 8.1 below shows Q_L dynamics when the concentration of extracellular inducer is high

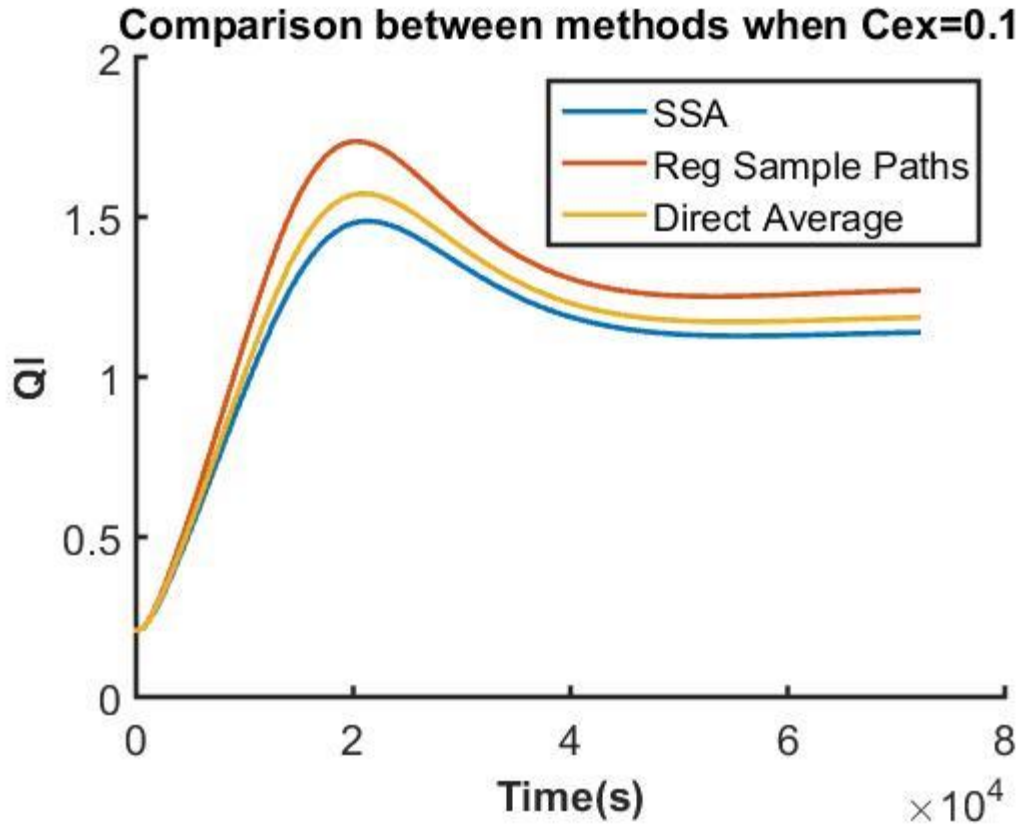

Figure 8.1: Q_L dynamics at low extracellular inducer concentration

In Figure 8.1, the three curves show similar expected results for Q_L response. At a low concentration of inducer in the environment, Q_L level stays low, indicating that the cells stay inactivated and are not ready for any conjugation process. From modeling perspective, the curve generated by the direct method lies so much closer to that generated by the SSA method. The two curves associated with SSA method and the regular sample path method are generated by averaging results from 100,000 independent sample paths. On the other hand, to evaluate the direct average of variables at each following time step in the direct method, 100 sample points are generated to approximate both variance and covariance

shown in both equations (98) and (99). The argument to justify for our sampling of only 100 points to approximate variance and covariance comes from the fact the flunctuation associated with the sample points is very small. At any given time point, the calculation is done based on the previous average time point directly. Because of that, even though each variable can fluctuate, it cannot fluctuate too much away from the actual average value. Unlike this proposed method, in both SSA or the regular sample path methods, each trajectory is independent from one another. Calculation at each time point depends only to the previous point in that same trajectory. As time progresses, this fluctuation can accumulate and can potentially drift the the calculated values far away from its actual average value. This phenonmenon explains why usage of SSA or the regular sample path method requires a high number of trajectories in order to obtain accurate results. This system is also known to be stiff and so an investigation of a scenario where Q_L changes dratically is needed for a complete evaluation for this method. Figure 8.2 below represents Q_L response at a high concentration of inducer in the environment:

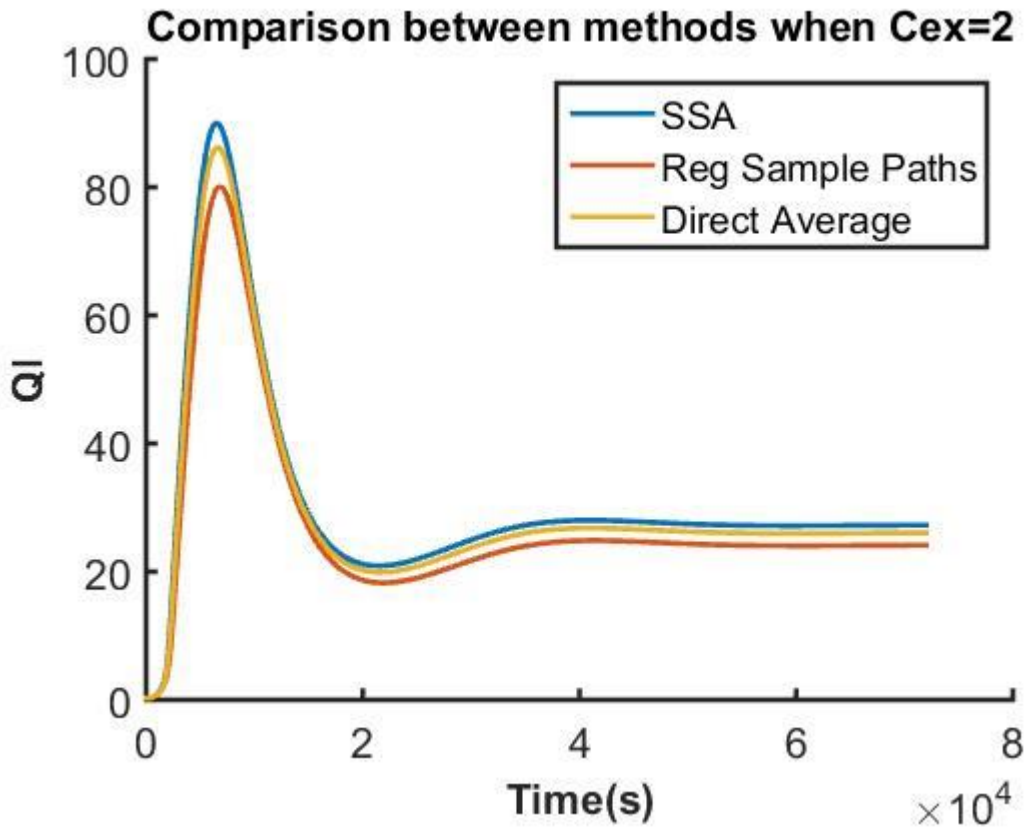

Figure 8.2: Q_L level at a high level of inducer

Figure 8.2 shows good agreement among results generated from all three methods. A same number of trajectories is utilized to compute the average in the SSA and the regular sample path methods. A high concentration of inducer immediately results in a stiff increase of Q_L, indicating a high sensitivity level of Q_L with respect to the inducer. In both cases shown in Figure 8.1 and 8.2, the curves generated by the direct average method show less deviation from the "actual" solution (generated by the SSA method). The small deviation is a result of computing variable at each time point directly using the average values from the previous time point. The errors associated with the results generated by the regular sample path method and the direct average methods using SSA as the benchmark are 16% and 7%

in the first case and 11% and 6% in the second case, respectively. Figure 8.3 below shows the full-scaled dynamic simulation for Q_L at various inducer levels in the environment

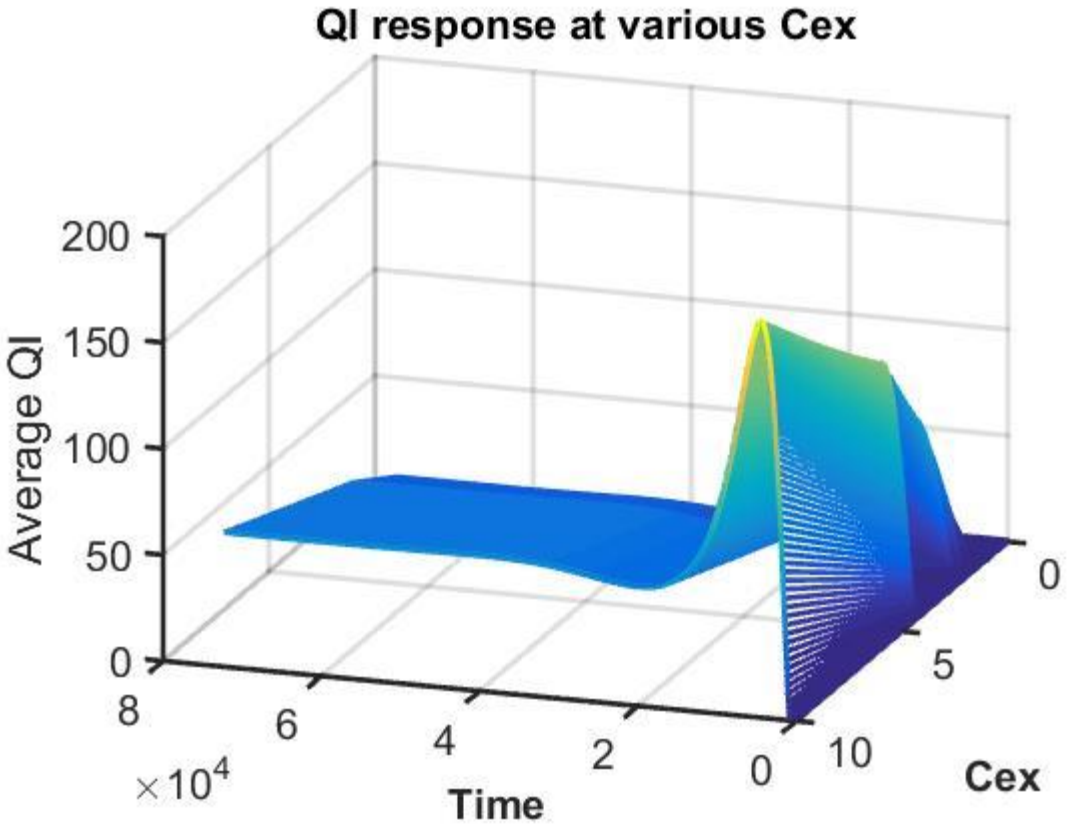

Figure 8.3: Ql response at various cCF10

This full-scaled calculation can serve to provide an overall picture of how the system would response at different conditions. These results can also be used to provide inputs for experimentalists to develop the experimental design space in a most effective way. To complete the evaluation of current method, simulation time is utilized in comparison among methods. The average step size of SSA method is selected to be a base unit from which the step sizes for other methods can be chosen. Time steps for the regular sample path and the direct average methods are both pre-selected to be ten folds larger than the

base unit. Figure 8.3 below provides comparison in simulation times required by the three methods:

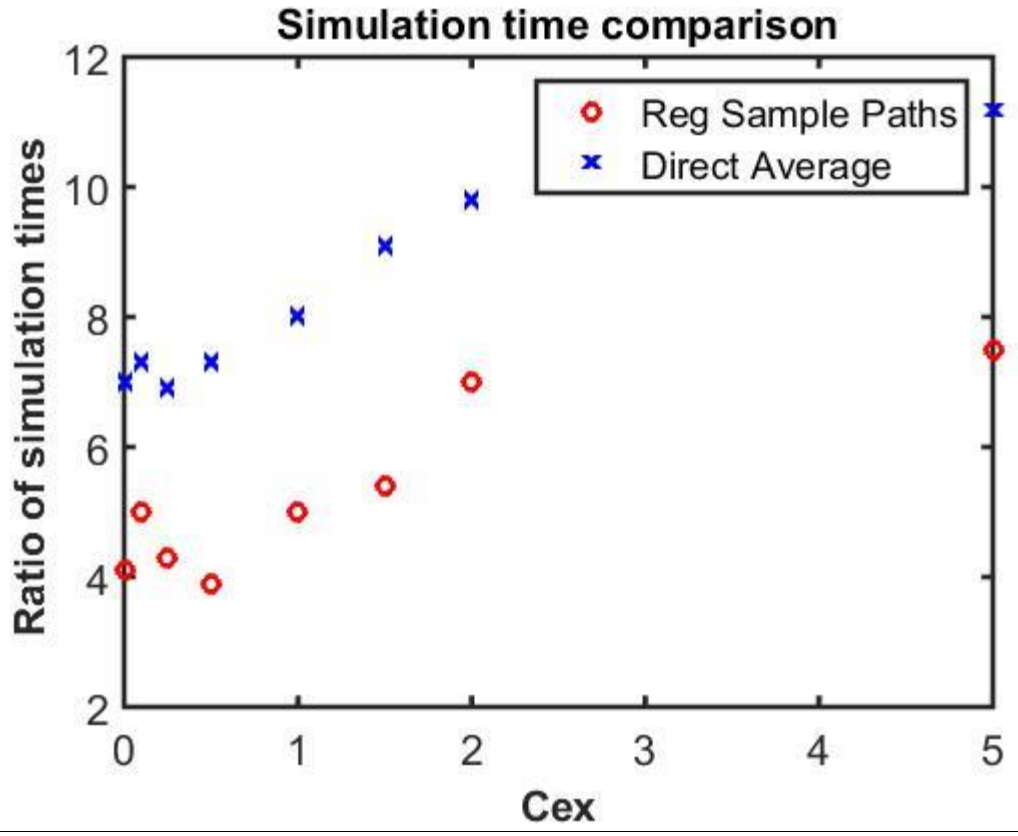

Figure 8.4: Ratio of simulation time between two methods and SSA

Figure 8.4 has y-axis indicate the ratio between required simulation times for the direct method as well as the regular sample path method and the SSA. In various conditions of cCF10, the method of generating sample path requires about 3-4 folds less in CPU time as compared to that in the SSA. The direct average method shows its advantage over the regular sample path strategy by halting the simulation time by about half. Both methods become more effective as the concentration of cCF10 increases. The explanation comes from how the system or Q_L specifically in this case is sensitive with an increase of inducer in the environment. As the stiffness increases, the change in variables become more

drastically, which in turn results in reducing time steps for SSA and hence increasing the simulation time for SSA.

8.7 Conclusions:

Stochastic differential equations are widely used as approximated methods to obtain solutions for the master equation. The advantage of utilizing this type of equation is to smooth out solution behaviors (continuous responses) and to reduce the time steps in simulations. However, the explicit versions of SDE show little capability of capturing systems with stiff behaviors. Moreover, due to the unbounded features, it is also known that the methods can sometimes inaccurately predict results in some systems. Implicit SDE methods can describe stiff behaviors with a high efficacy but involve solving a large number of algebraic equations at each time point, resulting in a large increase in the CPU time. Recent work of Yin et al. [106] proposed an improved Milstein method for stiff systems. The method has the advantage of capturing solutions for systems with stiff behaviors. However, it still requires so much time to obtain the average simulated results by generating independent sample paths. Moreover, from the experimental perspective, average dynamic behaviors of variables in a stochastic system are much more important and needed for cross-validations. Our group proposed a strategy through which the average behaviors can be directly computed without generating a large number of sample paths. We also illustrate the advantage of this method by testing on a very large complex biological system. The results generated by the proposed methods show high accuracy with a less simulation time as compared to both SSA and the regular sample path method. This strategy can also extend to other methods and can potentially offer the most efficient way to obtain the average with a shorter CPU time. The development shown in this paper

involved the usage of Taylor's series expansion and power expansion up to the second order term only. Testing this method on many other examples are needed and can provide us an insight on how far down in the Taylor expansion the calculations need to be done for a given requirement of an accuracy level.

8.8 Supplemental materials

8.8.1 Derivation of the direct average formula

Let us consider the equation (84)

$$Z_{n+1} = \overline{Z}_{n+1} + (I - hf'(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n))$$

h is the time step and is picked to be very small. f is the regular rate law and hence has the polynomial form. The system that we work on has low numbers in particles for most species. Therefore, $hf' \ll 1$. Applying power expansion, we have:

$$(I - hf'(\overline{Z}_{n+1}))^{-1} \approx I + hf'(\overline{Z}_{n+1}) + h^2[f'(\overline{Z}_{n+1})]^2$$

Hence,

$$(I - hf'(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n)) \simeq \{I + hf'(\overline{Z}_{n+1}) + h^2[f'(\overline{Z}_{n+1})]^2\} h(f(\overline{Z}_{n+1}) - f(Z_n)) \simeq$$
$$\simeq hf(\overline{Z}_{n+1}) - hf(Z_n)) + h^2 f'(\overline{Z}_{n+1}) f(\overline{Z}_{n+1}) - h^2 f'(\overline{Z}_{n+1}) f(Z_n)$$

Also using Taylor's expansion we have:

$$f^j(\overline{Z}_{n+1}) \simeq f^j(E\overline{Z}_{n+1}) + Df^j(Z)|_{E\overline{Z}_{n+1}} (\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$
$$+ 1/2(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^T Hf^j(Z)|_{E\overline{Z}_{n+1}} (\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$

$$Df^{j}(\overline{Z}_{n+1}) \simeq D(f^{j_1})^{,j_2}(Z)|_{E\overline{Z}_{n+1}} + DD(f^{j_1})^{,j_2}(Z)|_{E\overline{Z}_{n+1}}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$
$$+1/2(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} HD(f^{j_1})^{,j_2}(Z)|_{E\overline{Z}_{n+1}}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$

$$f^{j_1}(\overline{Z}_{n+1})D(f^{j_2})^{j_3}(\overline{Z}_{n+1}) \simeq f^{j_1}(E\overline{Z}_{n+1})D(f^{j_2})^{j_3}(Z)|_{E\overline{Z}_{n+1}} +$$
$$+[(D(f^{j_2})^{j_3}(Z)Df^{j_1}(Z)^{T}|_{E\overline{Z}_{n+1}} + f^{j_1}(E\overline{Z}_{n+1})DD(f^{j_2})^{j_3}(Z)](\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$
$$+1/2(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T}\nabla[(D(f^{j_2})^{j_3}(Z)Df^{j_1}(E\overline{Z}_{n+1})^{T}|_{E\overline{Z}_{n+1}}$$
$$+ f^{j_1}(E\overline{Z}_{n+1})DD(f^{j_2})^{j_3}(Z)]|_{E\overline{Z}_{n+1}}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$
$$\simeq f^{j_1}(E\overline{Z}_{n+1})D(f^{j_2})^{j_3}(Z)|_{E\overline{Z}_{n+1}} +$$
$$+[(D(f^{j_2})^{j_3}(Z)Df^{j_1}(Z)|_{E\overline{Z}_{n+1}} + f^{j_1}(E\overline{Z}_{n+1})DD(f^{j_2})^{j_3}(Z)|_{E\overline{Z}_{n+1}}](\overline{Z}_{n+1} - E\overline{Z}_{n+1}) +$$
$$+1/2(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T}\{(DD(f^{j_2})^{j_3}(Z)Df^{j_1}(Z)^{T}|_{E\overline{Z}_{n+1}} + D(f^{j_2})^{j_3}(Z)Hf^{j_1}(Z)^{T}|_{E\overline{Z}_{n+1}} +$$
$$+\mathrm{D}f^{j_1}(Z)DD(f^{j_2})^{j_3}(Z)]|_{E\overline{Z}_{n+1}} + f^{j_1}(\overline{EZ}_{n+1})HD(f^{j_2})^{j_3}(Z)]|_{E\overline{Z}_{n+1}}\}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$

$$f^{j}(Z_n) \simeq f^{j}(EZ_n) + Df^{j}(Z)|_{E\overline{Z}_{n+1}}(Z_n - EZ_n) + 1/2(Z_n - EZ_n)^{T} Hf^{j}(Z)|_{E\overline{Z}_{n+1}}(Z_n - EZ_n)$$

$$f^{j_1}(Z_n)D(f^{j_2})^{j_3}(\overline{Z}_{n+1}) = (f^{j_1}(EZ_n) + Df^{j_1}(Z)|_{E\overline{Z}_{n+1}}(Z_n - EZ_n)$$
$$+1/2(Z_n - EZ_n)^{T} Hf^{j_1}(Z)|_{E\overline{Z}_{n+1}}(Z_n - EZ_n))$$
$$+(D(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}} + DD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$
$$+1/2(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} HD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}(\overline{Z}_{n+1} - E\overline{Z}_{n+1}))$$
$$\simeq f^{j_1}(EZ_n)D(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}$$
$$+[f^{j_1}(EZ_n)DD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}](\overline{Z}_{n+1} - E\overline{Z}_{n+1})$$
$$+[D(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}} Df^{j_1}(Z)|_{E\overline{Z}_{n+1}}](Z_n - EZ_n)$$
$$+1/2f^{j_1}(EZ_n)(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} HD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}(\overline{Z}_{n+1} - E\overline{Z}_{n+1}))$$
$$+(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} Df^{j_1}(Z)|_{E\overline{Z}_{n+1}} DD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}(Z_n - EZ_n)$$
$$+1/2(Z_n - EZ_n)^{T} Hf^{j_1}(Z)|_{E\overline{Z}_{n+1}} D(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}}(Z_n - EZ_n))$$

Applying the expected operator to both sides of the equation, we have:

$$
\begin{aligned}
EZ_{n+1}^{j} &= E\overline{Z}_{n+1}^{j} + E(I - hf^{'}(\overline{Z}_{n+1}))^{-1} h(f(\overline{Z}_{n+1}) - f(Z_n))^{,j} \\
&= E\overline{Z}_{n+1}^{j} + E[hf(Z_n) - hf(Z_n)) + h^2 f^{'}(Z_n)|_{\overline{Z}_{n+1}} \; f(\overline{Z}_{n+1}) - h^2 f^{'}(Z_n)|_{\overline{Z}_{n+1}} \; f(Z_n)]^{,j} \\
&= E\overline{Z}_{n+1}^{j} + h\{f^{j}(E\overline{Z}_{n+1}) + 1/2E[(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} Hf^{j}(Z)|_{E\overline{Z}_{n+1}} (\overline{Z}_{n+1} - E\overline{Z}_{n+1})] + \\
&\quad - f^{j}(EZ_n) - 1/2(Z_n - EZ_n)^{T} Hf^{j}(Z)|_{EZ_n} (Z_n - EZ_n)\} + \\
&\quad + h^2\{f^{j}(\overline{EZ}_{n+1}) D(f^{j_2})^{j_3}(\overline{Z}_{n+1})|_{E\overline{Z}_{n+1}} + \\
&\quad + 1/2E\{(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T}\{(DD(f^{j_2})^{j_3}(Z) Df^{j}(Z)^{T}|_{E\overline{Z}_{n+1}} + D(f^{j_2})^{j_3}(Z) Hf^{j}(Z)^{T}|_{E\overline{Z}_{n+1}} + \\
&\quad + \mathrm{D}f^{j}(Z) DD(f^{j_2})^{j_3}(Z)]|_{E\overline{Z}_{n+1}} + f^{j}(E\overline{Z}_{n+1}) HD(f^{j_2})^{j_3}(Z)]|_{E\overline{Z}_{n+1}}\}(\overline{Z}_{n+1} - E\overline{Z}_{n+1})\}\} - \\
&\quad - h^2\{f^{j_1}(EZ_n) D(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}} \\
&\quad + 1/2E\{f^{j_1}(EZ_n)(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} HD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}} (\overline{Z}_{n+1} - E\overline{Z}_{n+1}))\} \\
&\quad + \mathrm{E}\{(\overline{Z}_{n+1} - E\overline{Z}_{n+1})^{T} Df^{j_1}(Z)|_{E\overline{Z}_{n+1}} DD(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}} (Z_n - EZ_n)\} \\
&\quad + 1/2\{(Z_n - EZ_n)^{T} Hf^{j_1}(Z)|_{E\overline{Z}_{n+1}} D(f^{j_2})^{,j_3}(Z)|_{E\overline{Z}_{n+1}} (Z_n - EZ_n))\}\}
\end{aligned}
$$

Similarly, we can apply the same method to equation (85):

$$
\begin{aligned}
g^{j_1}(Z_n) D(g^{j_2})^{j_3}(Z_n) &\simeq g^{j_1}((EZ_n) D(g^{j_2})^{j_3}((Z)|_{EZ_n} + \\
&\quad + [(D(g^{j_2})^{j_3}(Z) Df^{j_1}(Z)^{T}|_{EZ_n} + g^{j_1}(EZ_n) DD(g^{j_2})^{j_3}(Z)](Z_n - EZ_n) + \\
&\quad + 1/2(Z_n - EZ_n)^{T} \nabla[(D(g^{j_2})^{j_3}(Z) Dg^{j_1}(Z)^{T}|_{E\overline{Z}_{n+1}} \\
&\quad + g^{j_1}(EZ_n) DD(g^{j_2})^{j_3}(Z)]|_{E\overline{Z}_{n+1}} (Z_n - EZ_n) \\
&\simeq g^{j_1}(EZ_n) D(g^{j_2})^{j_3}(Z)|_{EZ_n} + \\
&\quad + [(D(g^{j_2})^{j_3}(Z) Dg^{j_1}(Z)^{T}|_{EZ_n} + g^{j_1}(EZ_n) DD(g^{j_2})^{j_3}(Z)](Z_n - EZ_n) + \\
&\quad + 1/2(Z_n - EZ_n)^{T}\{(DD(g^{j_2})^{j_3}(Z) Dg^{j_1}(Z)^{T}|_{EZ_n} + D(g^{j_2})^{j_3}(Z) Hg^{j_1}(Z)^{T}|_{EZ_n} + \\
&\quad + \mathrm{D}g^{j_1}(Z) DD(g^{j_2})^{j_3}(Z)]|_{EZ_n} + g^{j_1}(EZ_n) HD(g^{j_2})^{j_3}(Z)]|_{EZ_n}\}(Z_n - EZ_n)
\end{aligned}
$$

Applying the expected operator to both sides of the equation, we have:

$$
\begin{aligned}
E\overline{Z}_{n+1}^{j} &= EZ_n^{j} + hf^{j}(Z_n) + E\sum_{j_2=1}^{m}\sum_{j_1=1}^{m} g^{j_1} D(g^{j_2})^{j}(Z_n) EI_{(j_1, j_2)} \\
&= EZ_n^{j} + hf^{j}(Z_n) + g^{j_1}(EZ_n) D(g^{j_2})^{j}(Z)|_{EZ_n} \\
&\quad + 1/2E\{(Z_n - EZ_n)^{T}\{(DD(g^{j_2})^{j}(Z) Dg^{j_1}(Z)^{T}|_{EZ_n} + D(g^{j_2})^{j}(Z) Hg^{j_1}(Z)^{T}|_{EZ_n} + \\
&\quad + \mathrm{D}g^{j_1}(Z) DD(g^{j_2})^{j}(Z)]|_{EZ_n} \\
&\quad + g^{j_1}(EZ_n) HD(g^{j_2})^{j}(Z)]|_{EZ_n}\}(Z_n - EZ_n)\} E(\int_{t_n}^{t_{n+1}}\int_{t_n}^{s_2} dW^{j_1}(s_1) dW^{j_2}(s_2))^{,j})
\end{aligned}
$$

We can also evaluate the last term from the above equation

$$
E(\int_{t_n}^{t_{n+1}}\int_{t_n}^{s_2} dW^{j_1}(s_1) dW^{j_2}(s_2)) = \begin{cases} -h & , \text{if } j_1 \neq j_2 \\ 0 & , \text{if } j_1 = j_2 \end{cases}
$$

8.8.2 Reaction kinetics and parameters:

Table 8-1: Reactions and kinetic parameters

	Reactions	Kinetic parameters	Reaction type
1	$X_4 + 4i \rightarrow X_4 i_4$	8.016x10^7 (M^{-1}s^{-1})	1st order on i, 1st order on x_4
2	$X_4 + 4c \rightarrow X_4 c_4$	1.377x10^8 (M^{-1}s^{-1})	1st order on c, 1st order on x_4
3	$B + X_4 \underset{\leftarrow}{\rightarrow} BX_4$	F: **10^6** (M^{-1}s^{-1}) R: **10^{-1}**(s^{-1})	(parameters base on the lacI)
4	$B + X_4 i_4 \underset{\leftarrow}{\rightarrow} BX_4 i_4$	F: **10^8** (M^{-1}s^{-1}) R: **10^{-3}**(s^{-1})	
5	$B + X_4 c_4 \underset{\leftarrow}{\rightarrow} BX_4 c_4$	F: **10^9** (M^{-1}s^{-1}) R: **10^{-2}**(s^{-1})	
6	$\xrightarrow{Pq-Induced} Q_{pre} \quad (Induced : B)$	0.1 (s^{-1})	1st order on B
7	$\xrightarrow{Pq-Induced} Q_{pre} \quad (Uninduced : BX_4 c_4)$	0.1(s^{-1})	1st order on $BX_4 c_4$
8	$\xrightarrow{Pq-Uninduced} Q_{pre} \quad (Uninduced : BX_4)$	0.000723 (s^{-1})	1st order on BX_4
9	$\xrightarrow{Pq-Uninduced} Q_{pre} \quad (Uninduced : BX_4 i_4)$	0.000723 (s^{-1})	1st order on $BX_4 i_4$
10	$\xrightarrow{Px-Induced} Q_a \quad (Induced : B)$	0.00121(s^{-1})	1st order on B
11	$\xrightarrow{Px-Uninduced} Q_a \quad (Uninduced : BX_4 c_4)$	0.00121(s^{-1})	1st order on $BX_4 c_4$
12	$\xrightarrow{Px-Uninduced} Q_a \quad (Uninduced : \mathrm{B}X_4)$	0.00823(s^{-1})	1st order on BX_4
13	$\xrightarrow{Px-Uninduced} Q_a \quad (Uninduced : BX_4 i_4)$	0.00823(s^{-1})	1st order on $BX_4 i_4$
14	$Q_{pre} \longrightarrow Q_L *$	1(s^{-1})	(the time for transcript the length from Px to pass IRS1, roughly 1s)
15	$Q_{pre} + Q_a \longrightarrow Q_s$	4.43x10^9(M^{-1}s^{-1})	From previous Kq = 4.43(nM^{-1}); 4.43x(6 x10^8)/0.6, the latter is one reaction per second PS. 1nM ≈ 0.6 particle per cell for cell volume
16	$\xrightarrow{Q_s + Q_L} I_{ex}$	**0.5(s^{-1}) x delta**	**delta is volume conversion factor,**

			x 10^{-3} for 10^9 cells/ml
17	$I_{ex} \xrightarrow{uptake} i$	0.001(s^{-1})	
18	$C_{ex} \xrightarrow{uptake} c$	**0.001(s^{-1})**	
19	$\xrightarrow{Px-Induced} prgX \quad (Induced : B)$	0.000121(s^{-1})	1st order on B
20	$\xrightarrow{Px-Induced} prgX \quad (Uninduced : BX_4c_4)$	0.000121(s^{-1})	1st order on BX_4c_4
21	$\xrightarrow{Px-Uninduced} prgX \quad (Uninduced : BX_4)$	0.001021(s^{-1})	1st order on BX_4
22	$\xrightarrow{Px-Induced} prgX \quad (Uninduced : BX_4i_4)$	0.001021(s^{-1})	1st order on BX_4i_4
23	$\xrightarrow{prgX} X_2$	0.005(s^{-1})	
24	$2X_2 \rightleftarrows X_4$	F: **1×10^5** ($M^{-1}s^{-1}$) R: **0.01**(s^{-1})	

Table 8-2: Degradation rates for different species

Species	Degradation Rate (1/s)
X_4	1×10^{-5}
X_4i_4	1×10^{-5}
i	1×10^{-5}
X_4c_4	1×10^{-5}
c	1×10^{-5}
B	-
BX_4	-
BX_4i_4	-
BX_4c_4	-
Q_{pre}	-
Q_a	1×10^{-3}
Q_L	1×10^{-4}
Q_s	2×10^{-3}
I_{ex}	1×10^{-5}
C_{ex}	1×10^{-5}
$prgX$	2×10^{-4}
X_2	1×10^{-5}

9. CONCLUSIONS AND FUNTURE DIRECTIONS

9.1 Conclusions

Bacteria of the genus *Enterococcus*, commonly found in the intestinal tract, are the main cause of antibiotic-resistant infections that are acquired in hospitals. Signaling molecules (inducers and inhibitors) regulate the conjugation process through which the gene that is responsible for the drug resistance can be transferred from donor cells to recipient cells. The bacteria can exist in either free floating form (planktonic) or the matrix-like structure (biofilms). Due to the differences in the structure and the mechanism between the two environments, separate equations and models are developed. Several publications have modeled different scenarios related to the system. However, recent experiments reveal some new discoveries that invalidate many assumptions that were used in the previous modeling development. These updates require new efforts on modifying reaction kinetics as well as corresponding modeling approaches. The deterministic models were first used to obtain the average results as well as to fit some unavailable experimental parameters. Cells can exist either in planktonic or biofilm environments. Unlike the previous work that focused on the planktonic case, this present work attempts to model the system behaviors in the biofilm arrangement. It has shown that cells in biofilm can behave similarly but not always as similar to the planktonic case. It is found that spatial configurations of species (donors/recipients) in biofilms can lead to very different outcomes. Due to the well-mixed conditions assumed in the planktonic case, a sufficient amount of inducer C results in a full conversion of recipients to donors. In the biofilm case, some spatial configurations reveal the coexistence of both donor and recipient cells in the community. This outcome can be

explained by the concept of energy reservation in cells. Unless either threat to survivability is present or there is a need of cell modification for adaptation, cells prefer to not undergo any changes in order to minimize energy expenditure.

As compared to the deterministic models, stochastic modeling can describe systems associated with noise to a much higher level of accuracy yet is much more complicated and requires a long simulation time. Therefore, improved stochastic modeling efficiency is indeed critical in obtaining solutions in a reasonable time manner. Two main approaches for modeling stochastic systems include solving master equations and solving stochastic differential equations. The exact method can be used to obtain the solutions for the master equation but shows limitation on CPU time when the systems involve high numbers of reactions and species. Several versions of tau-leaping method are proposed to approximate the actual solutions of the master equations by capturing multiple events (as opposed to capturing a single event in the exact method). In order to ensure the accuracy of the method, a leap condition is introduced to guarantee that no significant change in propensity functions occurs during the chosen time step. Our group later introduced an alternative leaping condition using the Chebyshev' inequality [31]. Through this choice of selecting tau, a high number of useful "sample paths" can be obtained. This leads to a reduction in the required simulation size and hence improves the CPU time. Moreover, the group also published another paper later to further save CPU time by a "parallel' method. In the classical stochastic sample path generation, trajectories are simulated sequentially, so one does not start until the previous one finishes. Due to the nature of stochastic simulations, each trajectory is independent from one another and can vary greatly in length. This classical approach posts a huge "waiting time" between

trajectories. Our method simply allows them to simultaneously start and proceed independently. Whenever some trajectories are reaching the target point, the memory associated with those can be then reallocated for the unfinished trajectories. This approach avoids the waiting time and hence results in an average reduction of 40 folds in CPU time.

Those techniques have provided a way to acquire the solutions for the stochastic systems. These methods work well for the single cell case. However, in the biofilm case, the domain of interest is highly heterogeneous and interaction among cells are present at all time. The scope of this type of system is so much larger and more complex, presenting a much more computationally challenging problem for modelers. Even with all the advanced methods mentioned above, the need for further development still exists. A refreshing approach towards resolving the above dilemma appears in a paper by Grima, which raises the possibility of directly obtaining average (mesoscopic) behavior by entirely circumventing the computation of sample paths. The underlying methodology is grounded in quasilinearization leading to a derivation of dynamical equations which represent the potential source of average behavior. We then published a paper that introduces the concept of applying a discrete self-adjoint linear operator whose spectral representation is used to capture the diffusional effect. Coupling the two methods, we are able to capture systems, which involved both reaction and diffusion. This application allows us to compute the direct average without generating a large number of trajectories. In the final chapter, we also introduce another similar approach to calculate the direct average of variables modeled in SDE. Both methods have illustrated a higher efficiency than the classical sample path generating methods.

9.2 Future directions:

In all the modeling that we have built, some parameters have been measured directly using experimental results. However, the others are selected by using the kinetic parameters either associated with the similar systems or by using the parameter fitting techniques. Fortunately, the models are built in such a way that they can be updated immediately. Tuning and updating the parameters to improve the modeling performance will be ongoing work. Moreover, the experimental group has not been able to set up the experiments for the biofilm scenario. However the group has been working hard on getting it ready and the results of this type of experiment can also help cross-validate the biofilm model. Throughout my work, two main approaches that have been utilized to describe the *Enterrococcus faecalis* include solving the master equation and solving the stochastic differential equations (SDE). Both methods have their own advantages and drawbacks that need to be worked on. In the case of modeling the plasmid transfer between donor and recipient cells in the biofilm structure, I have applied a discrete technique, known as RDME, to describe the phenomenon under influence of both reactions and diffusion. Even though our modified "tau-leap" [31] and "parallel" [32] methods have illustrated a significant improvement in the CPU time, they are still not quite sufficient enough to produce results in a reasonable amount of time. In RDME approach, the domain is divided into smaller subunits. In addition to numerous interactions occurring in each subunit through reactions (40 reactions and 20 species), subunits also constantly interact with others by diffusion. These many levels of complexity therefore require even more advance modeling techniques. In many cases, average behaviors are the main focus in stochastic simulations, we proposed an algorithm [38] that combines the Grima's method [36] and the application of linear operator to capture both reactions and diffusion in the biofilms. The method has

been tested on multiple systems with smaller sizes in reactions and species. We intend to apply the method to the current system. The version of Grima's method included in the proposed algorithm can handle reactions up to the second order. Some modifications and improvement might need to be implemented to completely capture large systems with high order reactions that also include our problem of drug resistance transfer. On the other hand, in the planktonic environment, we also attempt to solve the SDE to capture the process in the well-mixed condition. New experimental results invalidate some assumptions that were made in previous models and suggest updates on the reaction kinetics. The model with the updated formulation suggests stiffness in behaviors and therefore requires special care on the approach. Implicit methods are made specifically for obtaining accurate results in stiff systems but require users to solve an extremely large system of nonlinear algebraic equations at each time step. As the numbers of reactions and species increase, this type of approach can indeed result in a huge burden on the CPU time. We indeed decided to utilize a modified version of Milstein method, in which a correct term is added to the explicit Milstein method. This method has fixed limitations associated with both explicit and implicit strategies. In addition, we proceeded even further to obtain the average calculations without simulating the sample paths in a traditional way. Taylor's expansion and power series expansion have both been utilized in simplifying the equations before the expected operator is applied to the equations. The general approach of this strategy can also extend to other types of SDEs. The only drawback of this method is that as we utilize higher order terms in Taylor expansion, higher derivatives for matrices will convert them into high order tensors that can also result in an enormous computation effort and lose the

attracting feature of this method in saving CPU time. More studies and testing need to be done to complete the development for this method.

Another current trend with high potential in combining strengths of various strategies is development of those hybrid models. Hybrid algorithms [93], [107]–[115] are distinctively useful in solving systems that follow multiple time scales. These models have been shown to reduce the computational cost very effectively.

Acknowledgments:

This work was supported by National Institutes of Health Grants GM081888 (to W.-S.H.)

REFERENCES

[1] P.-R. Hsueh *et al.*, "Pandrug-resistant Acinetobacter baumannii causing nosocomial infections in a university hospital, Taiwan.," *Emerg. Infect. Dis.*, vol. 8, no. 8, pp. 827–32, Aug. 2002.

[2] L. M. Mundy, D. F. Sahm, and M. Gilmore, "Relationships between enterococcal virulence and antimicrobial resistance.," *Clin. Microbiol. Rev.*, vol. 13, no. 4, pp. 513–22, Oct. 2000.

[3] A. Chatterjee *et al.*, "Antagonistic self-sensing and mate-sensing signaling controls antibiotic-resistance transfer," *Proc. Natl. Acad. Sci.*, vol. 110, no. 17, pp. 7086–7090, Apr. 2013.

[4] G. M. Dunny and C. M. Johnson, "Regulatory circuits controlling enterococcal conjugation: lessons for functional genomics," *Curr. Opin. Microbiol.*, vol. 14, no. 2, pp. 174–180, Apr. 2011.

[5] A. Chatterjee *et al.*, "Convergent transcription confers a bistable switch in Enterococcus faecalis conjugation.," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 108, no. 23, pp. 9721–6, Jun. 2011.

[6] A. M. T. Barnes, K. S. Ballering, R. S. Leibman, C. L. Wells, and G. M. Dunny, "Enterococcus faecalis produces abundant extracellular structures containing DNA in the absence of cell lysis during early biofilm formation.," *MBio*, vol. 3, no. 4, pp. e00193-12, Aug. 2012.

[7] V. C. Thomas, Y. Hiromasa, N. Harms, L. Thurlow, J. Tomich, and L. E. Hancock, "A fratricidal mechanism is responsible for eDNA release and contributes to biofilm development of Enterococcus faecalis.," *Mol. Microbiol.*, vol. 72, no. 4, pp. 1022–36, May 2009.

[8] S. K. Hahl and A. Kremling, "A Comparison of Deterministic and Stochastic Modeling Approaches for Biochemical Reaction Systems: On Fixed Points, Means, and Modes.," *Front. Genet.*, vol. 7, p. 157, 2016.

[9] H. M. Tsuchiya, A. G. Fredrickson, and R. Aris, "Dynamics of Microbial Cell Populations," *Adv. Chem. Eng.*, vol. 6, pp. 125–206, Jan. 1966.

[10] D. Ramkrishna, “Statistical models of cell populations,” Springer, Berlin, Heidelberg, 1979, pp. 1–47.

[11] A. Kremling, K. Bettenbrock, and E. Gilles, “Analysis of global control of Escherichia coli carbohydrate uptake,” *BMC Syst. Biol.*, vol. 1, no. 1, p. 42, Sep. 2007.

[12] G. Shinar, R. Milo, M. R. Martínez, and U. Alon, “Input output robustness in simple bacterial signaling systems.,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 104, no. 50, pp. 19931–5, Dec. 2007.

[13] J. J. Tyson and K. B. Hannsgen, “The distributions of cell size and generation time in a model of the cell cycle incorporating size control and random transitions,” *J. Theor. Biol.*, vol. 113, no. 1, pp. 29–62, Mar. 1985.

[14] L. Edelstein-Keshet, *Mathematical Models in Biology*. Society for Industrial and Applied Mathematics, 2005.

[15] L. S. Tsimring, “Noise in biology.,” *Rep. Prog. Phys.*, vol. 77, no. 2, p. 26601, 2014.

[16] C. C. Shu, A. Chatterjee, G. Dunny, W. S. Hu, and D. Ramkrishna, “Bistability versus bimodal distributions in gene regulatory processes from population balance,” *PLoS Comput. Biol.*, vol. 7, no. 8, 2011.

[17] M. Henson and F. H. Arnold, “PLOS ONE Role of Intracellular Stochasticity in Biofilm Growth . Insights from Population Balance Dear Jorg :”

[18] M. Rathinam, L. R. Petzold, Y. Cao, and D. T. Gillespie, “Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method,” *J. Chem. Phys.*, vol. 119, no. 24, pp. 12784–12794, Dec. 2003.

[19] M. Rathinam, L. R. Petzold, Y. Cao, and D. T. Gillespie, “Consistency and Stability of Tau-Leaping Schemes for Chemical Reaction Systems,” *Multiscale Model. Simul.*, vol. 4, no. 3, pp. 867–895, Jan. 2005.

[20] Y. Cao, D. T. Gillespie, and L. R. Petzold, “Accelerated stochastic simulation of the stiff enzyme-substrate reaction,” *J. Chem. Phys.*, vol. 123, no. 14, p. 144917, Oct. 2005.

[21] N. G. van. Kampen, *Stochastic processes in physics and chemistry*. North-Holland, 1992.

[22] D. T. Gillesple and D. T. Gillespie, “Exact Stochastic Simulation of Coupled Chemical Reactions.”

[23] B. H. Shah, D. Ramkrishna, and J. D. Borwanker, “Simulation of particulate systems using the concept of the interval of quiescence,” *AIChE J.*, vol. 23, no. 6, pp. 897–904, Nov. 1977.

[24] Y. Cao, D. T. Gillespie, and L. R. Petzold, “Efficient step size selection for the tau-leaping simulation method.,” *J. Chem. Phys.*, vol. 124, no. 4, p. 44109, Jan. 2006.

[25] Y. Cao, D. T. Gillespie, and L. R. Petzold, “Avoiding negative populations in explicit Poisson tau-leaping,” *J. Chem. Phys.*, vol. 123, no. 5, p. 54104, Aug. 2005.

[26] X. Peng, W. Zhou, and Y. Wang, “Efficient binomial leap method for simulating chemical kinetics,” *J. Chem. Phys.*, vol. 126, no. 22, p. 224109, Jun. 2007.

[27] A. Chatterjee, D. G. Vlachos, and M. A. Katsoulakis, “Binomial distribution based τ-leap accelerated stochastic simulation,” *J. Chem. Phys.*, vol. 122, no. 2, p. 24112, Jan. 2005.

[28] T. Tian and K. Burrage, “Binomial leap methods for simulating stochastic chemical kinetics,” *J. Chem. Phys.*, vol. 121, no. 21, pp. 10356–10364, Dec. 2004.

[29] D. T. Gillespie, “Approximate accelerated stochastic simulation of chemically reacting systems,” *J. Chem. Phys.*, vol. 115, no. 4, pp. 1716–1733, Jul. 2001.

[30] D. T. Gillespie and L. R. Petzold, “Improved leap-size selection for accelerated stochastic simulation,” *J. Chem. Phys.*, vol. 119, no. 16, p. 8229, 2003.

[31] V. Ramkrishna, Doraiswami; Shu,Che-Chi; Tran, “New ‘Tau-Leap’ Strategy for Accelerated Stochastic Simulation,” *I&EC*, 2014. [Online]. Available: http://pubs.acs.org/doi/pdf/10.1021/ie502929q. [Accessed: 17-Feb-2015].

[32] C. C. Shu, V. Tran, J. Binagia, and D. Ramkrishna, “On speeding up stochastic simulations by parallelization of random number generation,” *Chem. Eng. Sci.*, vol. 137, pp. 828–836, 2015.

[33] *Physica A: Theoretical and statistical physics.* North Holland, 1975.

[34] R. Erban, J. Chapman, and P. Maini, “A practical guide to stochastic simulations of reaction-diffusion processes,” Apr. 2007.

[35] S. A. Isaacson, “The Reaction-Diffusion Master Equation as an Asymptotic Approximation of Diffusion to a Small Target,” *SIAM J. Appl. Math.*, vol. 70, no. 1, pp. 77–111, Jan. 2009.

[36] R. Grima, “An effective rate equation approach to reaction kinetics in small volumes: Theory and application to biochemical reactions in nonequilibrium steady-state conditions,” *J. Chem. Phys.*, vol. 133, no. 3, p. 35101, Jul. 2010.

[37] S. Smith, C. Cianci, and R. Grima, “Analytical approximations for spatial stochastic gene expression in single cells and tissues.,” *J. R. Soc. Interface*, vol. 13, no. 118, p. 20151051, May 2016.

[38] V. Tran and D. Ramkrishna, “On facilitated computation of mesoscopic behavior of reaction-diffusion systems,” *AIChE J.*, vol. 63, no. 12, pp. 5258–5266, Dec. 2017.

[39] G. N. Milshtein, “Approximate Ingeration of stochastic differential equations,” *SIAM*, pp. 557–562, 1972.

[40] S. Ilie, W. H. Enright, and K. R. Jackson, “NUMERICAL SOLUTION OF STOCHASTIC MODELS OF BIOCHEMICAL KINETICS,” *Can. Appl. Math. Q.*, vol. 17, no. 3, 2009.

[41] M. a. Gibson and J. Bruck, “Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels,” *J. Phys. Chem. A*, vol. 104, no. 9, pp. 1876–1889, Mar. 2000.

[42] M. H. Antiporta and G. M. Dunny, “ccfA, the genetic determinant for the cCF10 peptide pheromone in Enterococcus faecalis OG1RF.,” *J. Bacteriol.*, vol. 184, no. 4, pp. 1155–62, Feb. 2002.

[43] T. Bae and G. M. Dunny, “Dominant-negative mutants of prgX: evidence for a role for PrgX dimerization in negative regulation of pheromone-inducible conjugation,” *Mol. Microbiol.*, vol. 39, no. 5, pp. 1307–1320, Feb. 2004.

[44] T. Bae, B. Kozlowicz, and G. M. Dunny, “Two targets in pCF10 DNA for PrgX binding: their role in production of Qa and prgX mRNA and in regulation of pheromone-inducible conjugation,” *J. Mol. Biol.*, vol. 315, no. 5, pp. 995–1007, Feb. 2002.

[45] R. P.-A. Berntsson, G. K. Schuurman-Wolters, G. Dunny, D.-J. Slotboom, and B. Poolman, “Structure and mode of peptide binding of pheromone receptor PrgZ.,” *J. Biol. Chem.*, vol. 287, no. 44, pp. 37165–70, Oct. 2012.

[46] E. Caserta *et al.*, “In vivo and in vitro analyses of regulation of the pheromone-responsive prgQ promoter by the PrgX pheromone receptor protein.,” *J. Bacteriol.*, vol. 194, no. 13, pp. 3386–94, Jul. 2012.

[47] B. K. Kozlowicz, T. Bae, and G. M. Dunny, “Enterococcus faecalis pheromone-responsive protein PrgX: genetic separation of positive autoregulatory functions from those involved in negative regulation of conjugative plasmid transfer,” *Mol. Microbiol.*, vol. 54, no. 2, pp. 520–532, Aug. 2004.

[48] B. K. Kozlowicz, K. Shi, Z.-Y. Gu, D. H. Ohlendorf, C. A. Earhart, and G. M. Dunny, “Molecular basis for control of conjugation by bacterial pheromone and inhibitor peptides,” *Mol. Microbiol.*, vol. 62, no. 4, pp. 958–969, Nov. 2006.

[49] C. S. Laspidou and B. E. Rittmann, “Modeling the development of biofilm density including active bacteria, inert biomass, and extracellular polymeric substances.,” *Water Res.*, vol. 38, no. 14–15, pp. 3349–61, 2004.

[50] W. Gujer, “A Multispecies Biofilrn Model 0.,” 1985.

[51] E. Arvin and P. Harremoës, “Concepts and Models for Biofilm Reactor Performance,” *Water Sci. Technol.*, vol. 22, no. 1–2, 1990.

[52] C. Picioreanu, M. C. M. van Loosdrecht, and J. J. Heijnen, “A new combined differential-discrete cellular automaton approach for biofilm modeling: Application for growth in gel beads,” *Biotechnol. Bioeng.*, vol. 57, no. 6, pp. 718–731, Mar. 1998.

[53] C. Picioreanu, M. C. M. Van Loosdrecht, and J. J. Heijnen, “Mathematical Modeling of Biofilm Structure with a Hybrid Differential-Discrete Cellular Automaton Approach,” *Biotechnol Bioeng*, vol. 58, pp. 101–116, 1998.

[54] J. W. T. Wimpenny and R. Colasanti, “A unifying hypothesis for the structure of microbial biofilms based on cellular automaton models,” *FEMS Microbiol. Ecol.*, vol. 22, no. 1, pp. 1–16, Jan. 2006.

[55] R. Dillon, L. Fauci, A. Fogelson, and D. Gaver III, “Modeling Biofilm Processes Using the Immersed Boundary Method,” *J. Comput. Phys.*, vol. 129, no. 1, pp. 57–73, Nov. 1996.

[56] J.-U. Kreft, C. Picioreanu, M. C. M. van Loosdrecht, and J. W. T. Wimpenny, “Individual-based modelling of biofilms,” *Microbiology*, vol. 147, no. 11, pp. 2897–2912, Nov. 2001.

[57] W. Feller, *An introduction to probability theory and its applications*. John Wiley & Sons, Inc, 1957.

[58] S. Smith and R. Grima, “The breakdown of the reaction-diffusion master equation with non-elementary rates,” Jan. 2016.

[59] D. G. Kendall, “Random Fluctuations in the Age-Distribution of a Population Whose Development is Controlled by the Simple "Birth-and-Death" Process,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 12. WileyRoyal Statistical Society, pp. 278–285, 1950.

[60] D. Ramkrishna and M. R. Singh, “Population Balance Modeling: Current Status and Future Prospects,” *Annu. Rev. Chem. Biomol. Eng.*, vol. 5, no. 1, pp. 123–146, Jun. 2014.

[61] D. F. Anderson, “Incorporating postleap checks in tau-leaping.,” *J. Chem. Phys.*, vol. 128, no. 5, p. 54103, Feb. 2008.

[62] Y. Cao and L. Petzold, “Accuracy limitations and the measurement of errors in the stochastic simulation of chemically reacting systems,” *J. Comput. Phys.*, vol. 212, no. 1, pp. 6–24, 2006.

[63] Y. Cao, D. T. Gillespie, and L. R. Petzold, “Adaptive explicit-implicit tau-leaping method with automatic tau selection,” *J. Chem. Phys.*, vol. 126, no. 22, p. 224101, Jun. 2007.

[64] X. Cai and Z. Xu, “K-leap method for accelerating stochastic simulation of coupled chemical reactions,” *J. Chem. Phys.*, vol. 126, no. 7, p. 74102, Feb. 2007.

[65] M. Vellela and H. Qian, “Stochastic dynamics and non-equilibrium thermodynamics of a bistable chemical system: the Schlögl model revisited.,” *J. R. Soc. Interface*, vol. 6, no. 39, pp. 925–40, Oct. 2009.

[66] R. G. Endres, “Bistability: Requirements on Cell-Volume, Protein Diffusion, and Thermodynamics,” *PLoS One*, vol. 10, no. 4, p. e0121681, Apr. 2015.

[67] C. Escher and J. Ross, “Multiple ranges of flow rate with bistability and limit cycles for Schlögl’s mechanism in a CSTR,” *J. Chem. Phys.*, vol. 79, no. 8, pp. 3773–3777, Oct. 1983.

[68] W. J. Blake, M. KÆrn, C. R. Cantor, and J. J. Collins, “Noise in eukaryotic gene expression,” *Nature*, vol. 422, no. 6932, pp. 633–637, Apr. 2003.

[69] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain, “Stochastic gene expression in a single cell.,” *Science*, vol. 297, no. 5584, pp. 1183–6, Aug. 2002.

[70] N. J. Rao ’, J. D. Borwankar, and D. Ramkrishna, “NUMERICAL SOLUTION OF ITO INTEGRAL EQUATIONS,” vol. 12, no. 1, 1974.

[71] E. Pardoux and D. Talay, “Discretization and simulation of stochastic differential equations,” *Acta Appl. Math.*, vol. 3, no. 1, pp. 23–47, Jan. 1985.

[72] J. Schnakenberg and K.-L. Kompa, “Gardiner: Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences/Bass und Stitch: Laser Handbook, Vol. 5,” *Phys. J.*, vol. 42, no. 8, pp. 307–307, Aug. 1986.

[73] R. J. Ellis, “Macromolecular crowding: an important but neglected aspect of the intracellular environment,” *Curr. Opin. Struct. Biol.*, vol. 11, no. 1, pp. 114–119, Feb. 2001.

[74] S. Engblom, L. Ferm, A. Hellander, and P. Lötstedt, “Simulation of Stochastic Reaction-Diffusion Processes on Unstructured Meshes,” *SIAM J. Sci. Comput.*, vol. 31, no. 3, pp. 1774–1797, Jan. 2009.

[75] S. A. Isaacson and C. S. Peskin, “Incorporating Diffusion in Complex Geometries into Stochastic Chemical Kinetics Simulations,” *SIAM J. Sci. Comput.*, vol. 28, no. 1, pp. 47–74, Jan. 2006.

[76] D. Ramkrishna and N. R. (Neal R. Amundson, *Linear operator methods in chemical engineering with applications to transport and chemical reaction systems*. Prentice-Hall, 1985.

[77] T. Asano, T. Wada, M. Ohta, and N. Takigawa, “Langevin equation as a stochastic differential equation in nuclear physics,” in *AIP Conference Proceedings*, 2007, vol. 891, pp. 435–438.

[78] B. Kafash, R. Lalehzari, A. Delavarkhalafi, and E. Mahmoudi, “Application of Stochastic Differential System in Chemical Reactions via Simulation.”

[79] D. T. Gillespie, “The multivariate Langevin and Fokker–Planck equations,” *Am. J. Phys.*, vol. 64, no. 10, pp. 1246–1257, Oct. 1996.

[80] R. Erban and S. J. Chapman, “Stochastic modelling of reaction-diffusion processes: algorithms for bimolecular reactions,” Mar. 2009.

[81] A. Arkin, J. Ross, and H. H. McAdams, “Stochastic Kinetic Analysis of Developmental Pathway Bifurcation in Phage λ-Infected Escherichia coli Cells,” *Genetics*, vol. 149, no. 4, 1998.

[82] M. B. Elowitz and S. Leibler, “A synthetic oscillatory network of transcriptional regulators,” *Nature*, vol. 403, no. 6767, pp. 335–338, Jan. 2000.

[83] M. S. Ko, “Induction mechanism of a single gene molecule: stochastic or deterministic?,” *Bioessays*, vol. 14, no. 5, pp. 341–6, May 1992.

[84] H. H. McAdams and A. Arkin, “Stochastic mechanisms in gene expression,” *Proc. Natl. Acad. Sci.*, vol. 94, no. 3, 1997.

[85] H. H. McAdams and A. Arkin, “It’s a noisy business! Genetic regulation at the nanomolar scale,” *Trends Genet.*, vol. 15, no. 2, pp. 65–69, Feb. 1999.

[86] J. Paulsson, “Summing up the noise in gene networks,” *Nature*, vol. 427, no. 6973, pp. 415–418, Jan. 2004.

[87] P. S. Swain, M. B. Elowitz, and E. D. Siggia, “Intrinsic and extrinsic contributions to stochasticity in gene expression.,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 99, no. 20, pp. 12795–800, Oct. 2002.

[88] M. Thattai and A. van Oudenaarden, “Intrinsic noise in gene regulatory networks,” *Proc. Natl. Acad. Sci.*, vol. 98, no. 15, 2001.

[89] D. T. Gillespie, “The chemical Langevin equation,” *J. Chem. Phys.*, vol. 113, no. 1, p. 297, Jun. 2000.

[90] J. M. G. Vilar, H. Y. Kueh, N. Barkai, and S. Leibler, “Mechanisms of noise-resistance in genetic oscillators.,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 99, no. 9, pp. 5988–92, Apr. 2002.

[91] D. J. Wilkinson, “Stochastic modelling for quantitative description of heterogeneous biological systems,” *Nat. Rev. Genet.*, vol. 10, no. 2, pp. 122–133, Feb. 2009.

[92] T. E. Turner, S. Schnell, and K. Burrage, “Stochastic approaches for modelling in vivo reactions,” *Comput. Biol. Chem.*, vol. 28, no. 3, pp. 165–178, Jul. 2004.

[93] E. L. Haseltine and J. B. Rawlings, “Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics,” *J. Chem. Phys.*, vol. 117, no. 15, pp. 6959–6969, Oct. 2002.

[94] D. J. Higham., “An Algorithmic Introduction to Numerical Simulation of Stochastic Differential Equations,” *SIAM Rev.*, vol. 43, no. 3, pp. 525–546, Jan. 2001.

[95] B. A. Bodo, M. E. Thompson, and T. E. Unny, “A review on stochastic differential equations for applications in hydrology,” *Stoch. Hydrol. Hydraul.*, vol. 1, no. 2, pp. 81–100, Jun. 1987.

[96] E. Beretta, V. Kolmanovskii, and L. Shaikhet, “Stability of epidemic model with time delays influenced by stochastic perturbations,” *Math. Comput. Simul.*, vol. 45, no. 3–4, pp. 269–277, Feb. 1998.

[97] N. Bruti-Liberati and E. Platen, “Strong approximations of stochastic differential equations with jumps,” *J. Comput. Appl. Math.*, vol. 205, no. 2, pp. 982–1001, Aug. 2007.

[98] B. fiisiro-Maruyama, “48 CONTINUOUS MARKOV PROCESSES AND STOCHASTIC EQUATIONS.”

[99] W. Rüemelin, “Numerical Treatment of Stochastic Differential Equations,” *SIAM J. Numer. Anal.*, vol. 19, no. 3, pp. 604–613, Jun. 1982.

[100] “BALANCED METHODS FOR STIFF STOCHASTIC SYSTEMS.”

[101] J. Alcock and K. Burrage, “A note on the Balanced method,” *BIT Numer. Math.*, vol. 46, no. 4, pp. 689–710, Dec. 2006.

[102] K. Burrage and T. Tian, “The composite Euler method for stiff stochastic differential equations,” *J. Comput. Appl. Math.*, vol. 131, no. 1–2, pp. 407–426, Jun. 2001.

[103] M. A. Omar, A. Aboul-Hassan, and S. I. Rabia, “The composite Milstein methods for the numerical solution of Stratonovich stochastic differential equations,” *Appl. Math. Comput.*, vol. 215, no. 2, pp. 727–745, Sep. 2009.

[104] X. Wang, S. Gan, and D. Wang, “A family of fully implicit Milstein methods for stiff stochastic differential equations with multiplicative noise,” *BIT Numer. Math.*, vol. 52, no. 3, pp. 741–772, Sep. 2012.

[105] T. Tian and K. Burrage, “Implicit Taylor methods for stiff stochastic differential equations,” *Appl. Numer. Math.*, vol. 38, no. 1–2, pp. 167–185, Jul. 2001.

[106] Z. Yin and S. Gan, “An improved Milstein method for stiff stochastic differential equations,” *Yin Gan Adv. Differ. Equations*, 2015.

[107] W. E, D. Liu, and E. Vanden-Eijnden, “Nested stochastic simulation algorithms for chemical kinetic systems with multiple time scales,” *J. Comput. Phys.*, vol. 221, no. 1, pp. 158–180, Jan. 2007.

[108] A. Alfonsi, E. Cancès, G. Turinici, B. Di Ventura, and W. Huisinga, “Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems,” *ESAIM Proc.*, vol. 14, pp. 1–13, Sep. 2005.

[109] A. Hellander and P. Lötstedt, “Hybrid method for the chemical master equation,” *J. Comput. Phys.*, vol. 227, no. 1, pp. 100–122, Nov. 2007.

[110] T. R. Kiehl, R. M. Mattheyses, and M. K. Simmons, “Hybrid simulation of cellular behavior,” *Bioinformatics*, vol. 20, no. 3, pp. 316–322, Feb. 2004.

[111] S. MacNamara, K. Burrage, and R. B. Sidje, “Multiscale Modeling of Chemical Kinetics via the Master Equation,” *Multiscale Model. Simul.*, vol. 6, no. 4, pp. 1146–1168, Jan. 2008.

[112] J. Puchałka and A. M. Kierzek, “Bridging the Gap between Stochastic and Deterministic Regimes in the Kinetic Simulations of the Biochemical Reaction Networks,” *Biophys. J.*, vol. 86, no. 3, pp. 1357–1372, Mar. 2004.

[113] C. V. Rao and A. P. Arkin, “Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the Gillespie algorithm,” *J. Chem. Phys.*, vol. 118, no. 11, pp. 4999–5010, Mar. 2003.

[114] H. Salis and Y. Kaznessis, “Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions,” *J. Chem. Phys.*, vol. 122, no. 5, p. 54103, Feb. 2005.

[115] A. Samant and D. G. Vlachos, “Overcoming stiffness in stochastic simulation stemming from partial equilibrium: A multiscale Monte Carlo algorithm,” *J. Chem. Phys.*, vol. 123, no. 14, p. 144114, Oct. 2005.