UNIVERSITY OF CALIFORNIA

Santa Barbara

The Finite State Projection Approach for the Solution of the Master Equation and its Applications to Stochastic Gene Regulatory Networks

A Dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy
in Mechanical Engineering

by

Brian E. Munsky

Committee in charge:

Professor Mustafa Khammash, Chair

Professor João Hespanha

Professor David Low

Professor Jeffrey Moehlis

Professor Linda Petzold

June 2008

The dissertation of Brian E. Munsky is approved.

João Hespanha

David Low

Jeffrey Moehlis

Linda Petzold

Mustafa Khammash, Committee Chair

May 2008

The Finite State Projection Approach for the Solution of the Master Equation

and its Applications to Stochastic Gene Regulatory Networks

by

Brian E. Munsky

I dedicate this work to all of the friends and family that have helped me through the exciting challenges I faced while attending UCSB.

Acknowledgments

This work couldn't have been possible without the many collaborations I have had on these topics with researchers in and out of UCSB. David Low, Aaron Hernday, Bruce Braaten, and Brooke Trinh have contributed heavily to the development of the Pap system model in Chapters 13 and 16. Slaven Pelěs contributed to the Slow Manifold FSP method in Chapter 7. Conversations with Eric Klavins helped frame much of the model matching method presented in Chapter 11.2.3. I want to thank the many Systems Biology faculty and students, especially Sandra Dandach, Frank Doyle, Hana el Samad, Laura Giarré, João Hespanha, Josh Johnson, Teri Lampoudi, Gabriele Lillacci, Jeffrey Moehlis, Linda Petzold, and Patrick Sheppard, whose feedback has helped me to polish many of the research ideas contained herein. I would also like to thank the many great teachers at UCSB, especially Petar Kokotovic, Karl Astrom and Mihai Putinar, all of whom have been exceptionally generous with their time and energy beyond the classroom and throughout my studies. I am also very appreciative to all of the ME staff, especially Laura Reynolds, Lauren Gleason, Julie Dunson, and Jeff Oaks who have been very patient with my innumerable administrative and technological requests. I want to express my gratitude to my adviser Mustafa Khammash who provided me with great guidance, encouragement and feedback throughout the course of this research.

Thanks to my friends and family, I have thoroughly enjoyed my time at UCSB. I especially want to thank Mark Zielke, Mary Dinh, Eric Sandoz, Thomas John, Don Freeborn, and Brendon Hall for the many sailing, surfing, camping, and climbing trips that helped to keep me sane throughout this endeavor and also for the use of their couches in my occasional periods of homelessness. I want to thank my parents and brothers for their unerring support. Most of all, I want to thank Lynn for her love and friendship.

Finally, I want to thank the funding authorities that have enabled me to pursue this research–much of this work has been supported by the UCSB Chancellor's Fellowship, the National Science Foundation under Grant NSF-ITR CCF-0326576 and the Institute for Collaborative Biotechnologies through Grant DAAD19-03-D-0004 from the U.S. Army Research Office.

Vita of Brian E. Munsky
June, 2008

EDUCATION

- **The University of California at Santa Barbara**, Santa Barbara, CA
  Center for Control Dynamical Systems and Computation
  Ph.D. in Mechanical Engineering; Adviser: Mustafa Khammash
  Sept. 2003–June 2008; GPA: 3.86/4.00
  Dissertation topic: *Modeling and Analysis of Stochastic Networks in Biological Systems.*

- **The Pennsylvania State University**, University Park, PA.
  M.S. in Aerospace Engineering, Adviser: Farhan Gandhi
  Aug. 2000–Aug. 2002; GPA: 3.88/4.00
  Thesis topic: *Fluid/Structural/Acoustic Analyses of Helicopter Blade-Vortex Interactions.*

- **The Pennsylvania State University**, University Park, PA.
  B.S. in Aerospace Engineering, Advisers: Farhan Gandhi and Ed Smith
  August 1996–May 2000; Junior/Senior GPA: 3.92/4.00
  Honors thesis topic: *Active/Passive Damping Treatments to Alleviate Resonant Oscillations.*

HONORS AND AWARDS

- Selected to present one of six student talks at the Eighth International Conference on Systems Biology, Long Beach, CA (Oct. 2007)
- Selected to present one of ten contributed talks at the first q Bio Conference on Cellular Information Processing, Santa Fe, NM (Aug. 2007)
- UCSB Chancellor's Fellowship (Sept. 2003–present)
- UCSB Dept. of Mechanical Engineering Graduate Fellowship (Sep. 2003–present)
- National Defense Science and Engineering Graduate Fellowship (2001–02)
- American Helicopter Society Vertical Flight Foundation Award (1999, 2000)
- Penn State University College of Engineering Graduate Fellowship (2000–01)
- Mary Ilgen Memorial Scholarship (1999–2000)
- Graduated with Honors from the Penn State Schreyer Honors College (May 2000)
- Penn State Deans List (7 times)
- Schreyer Honors College Academic Excellence Award (1996-2000)

RESEARCH EXPERIENCE

- ◇ **Graduate Research Assistant**, Dept. of Mechanical Engineering, UC-Santa Barbara.
  Working closely with experimental biologists, I have developed and validated a stochastic model of the Pap Pili epigenetic switch in *E. coli.* During the course of this research, I have developed an array of new analytical tools to solve the master equations for discrete state, continuous time Markov processes. (Jan. 2004 to Present).
- ◇ **Graduate Research Assistant**, Dept. of Aerospace Engineering, Penn State University.
  As a research assistant at the Rotorcraft Center of Excellence (RCOE), I developed a Finite Element aeroelastic model to simulate a helicopter in forward flight. Working closely with other students, I coupled this model with a free wake aerodynamic analysis and an aero-acoustic prediction code. I used this model to study of the effects of helicopter flight trajectory on the noise due to Blade-Vortex Interactions. (May 2000–Aug. 2002)
- ◇ **Undergrad Research Assistant**, Dept. of Aerospace Engineering, Penn State University.
  Using Finite Element Analysis, I performed parametric studies to ascertain the optimal design of an Active Constrained Layer Damping treatment with viscoelastic materials for the alleviation of resonant oscillations. (May, 1999–May, 2000)
- ◇ **Undergrad Research Assistant**, Dept. of Aerospace Engineering, Penn State University.
  As a research assistant at the RCOE, I played a pivotal role in the analysis, design, and construction of a piezoelectric actuator for rotor blade trailing edge flaps. (Jan. 1999–Aug. 1999)

TEACHING AND TUTORING EXPERIENCE

- ◇ **Course Development and Teaching**, Dept. of Mechanical Engr., UC-Santa Barbara.
  Together with my graduate adviser (Mustafa Khammash), I co-developed and co-taught a new graduate level course on the modeling and analysis of stochastic gene regulatory networks. Duties included researching material, preparing and giving lectures, and assigning and grading homeworks and exams (Spring, 2007). We condensed this course to a three hour tutorial session titled: "Stochastic Gene Expression in Systems Biology," which we presented at the *Eighth International Conference on Systems Biology* (October 1, 2007).
- ◇ **Student Lectures**, First q-bio Summer School on Cellular Information Processing, Los Alamos National Laboratory. As part of a summer school for graduate students and postdocs in the field of systems biology, I gave a one hour student lecture on Finite State Projection based approaches for the solution of the chemical master equation.

While at UCSB, I have given several lectures and presentations on stochastic modeling of gene regulatory networks for groups such as the NSF Integrative Graduate Education and Research Traineeship (IGERT) program, the Institute for Collaborative Biology (ICB), and the UCSB Theoretical Ecology Seminar.

- ◇ **Graduate Teaching Assistant**, Dept. of Aerospace Engineering, Penn State University.
  As a teaching assistant of a course on the numerical analysis of aerospace structures, I prepared and delivered weekly review sessions; assigned, solved, and graded biweekly homework assignments; and provided group and one-on-one tutoring to students. (Spring, 2001)
- ◇ **Writing Tutor**, University Learning Resource Center, Penn State University.
  As a trained peer tutor for writing, I assisted undergraduate, graduate and ESL (English as a Second Language) students at the Penn State Writing Center. (Jan. 1997–Dec. 1997)

PUBLICATIONS AND PRESENTATIONS

Systems Biology

- ◇ Munsky, B. and Khammash, M., Using Noise Transmission Properties to Identify Stochastic Gene Regulatory Networks, *Submitted to* 46th *IEEE Conference on Decision and Control*, Cancun, Mexico, Dec. 2008.
- ◇ Munsky, B. and Khammash, M., Computation of Switch Time Distributions in Stochastic Gene Regulatory Networks, *Proc. 2008 American Control Conference*, Seattle, WA, June 2008.
- ◇ Munsky, B. and Khammash, M., Transient Analysis of Stochastic Switches and Trajectories with Applications to Gene Regulatory Networks, *To appear in IET Systems Biology*, 2008.
- ◇ Munsky, B. and Khammash, M., The FSP Approach for the Analysis of Stochastic Noise in Gene Networks, *IEEE Trans. Automat. Contr./IEEE Trans. Circuits and Systems: Part 1*, **52**, No. 1, Jan. 2008, pp 201-214.
- ◇ Inglesias, P., Khammash, M., Munsky, B., Sontag, E. and Del Vecchio, D., Systems Biology and Control – A Tutorial, 45th *IEEE Conference on Decision and Control*, New Orleans, LA, Dec 2007.
- ◇ Munsky, B. and Khammash, M., A Multiple Time Interval Finite State Projection Algorithm for the Solution to the Chemical Master Equation, *J. Comp. Phys.*, **226**, No. 1, Sept. 2007, pp. 818-835.
- ◇ Munsky, B. and Khammash, M., Analysis of Noise Induced Stochastic Fluctuations in Gene Regulatory Networks, *J. SICE*, **46**, No. 5, May 2007, pp. 405-411.
- ◇ Khammash, M. and Munsky, B., Systems Theory Applications in Biology: From Stochastic Chemical Kinetics to Deterministic Model Invalidation, *Invited Paper–European Control Conference*, Kos, Greece, July 2007.

◇ Munsky, B., Peleš, S. and Khammash, M., Stochastic Analysis of Gene Regulatory Networks Using Finite State Projection and Singular Perturbation, *Invited Paper–Proc. of the* 26th *American Control Conference*, New York, NY, July 2007, pp. 1323-1328.

◇ Munsky, B. and Khammash, M., A Reduced Model Solution for the Chemical Master Equation Arising in Stochastic Analyses of Biological Networks, *Proc. of the* 45th *IEEE Conference on Decision and Control*, San Diego, CA, Dec. 2006, pp. 25-30.

◇ Peleš, S., Munsky, B. and Khammash, M., Reduction and Solution of the Chemical Master Equation Using Time Scale Separation and Finite State Projection, *J. of Chemical Physics*, **125**, No. 20, 204104, Nov. 2006.

◇ Munsky, B. and Khammash, M. Modeling and Analysis of a Bacterial Stochastic Switch, *Proc. of the* 14th *Mediterranean Conference on Control and Automation*, Ancona, Italy, pp. 1-6, June 2006.

◇ Munsky, B. and Khammash, M. The Finite State Projection Algorithm for the Solution of the Chemical Master Equation, *J. of Chemical Physics*, **124**, No. 4, 044104, Jan. 2006.

◇ Munsky, B., Hernday, A., Low, D., and Khammash, M. Stochastic Modeling of the Pap Pili Epigenetic Switch, *Proc. of Foundations of Systems Biology in Engineering*, Santa Barbara, CA, August 2005, pp. 145-148.

Structural Dynamics

◇ Munsky, B., Gandhi, F., and Tauszig, L., Analysis of Helicopter Blade-Vortex Interaction Noise with Flight Path or Attitude Modification, *J. American Helicopter Society*, **50**, No. 2, April 2005, pp. 123-137. Earlier version available as Munsky, B., Gandhi, F., and Tauszig, L., *Proc. of the 58th Annual Forum of the AHS, Acoustic Session*, **2**, Montreal, Canada, June 2002, pp. 1531-1551.

◇ Gandhi, F. and Munsky, B., Effectiveness of Active Constrained Layer Damping Treatments in Attenuating Resonant Oscillations, *J. of Vibration and Controls*, **8**, No. 6, 2002, pp. 747-775.

◇ Gandhi, F. and Munsky, B., Comparison of Damping Augmentation Mechanisms with Position and Velocity Feedback in Active Constrained Layer Treatments, *J. of Intelligent Material Systems and Structures*, **13**, No. 5, May 2002, pp. 259-326. Earlier version available as: Gandhi, F. and Munsky, B., Comparison of the Mechanism and Effectiveness of Position and Velocity Feedback in Active Constrained Layer Damping Treatments, *Proc. of SPIE Smart Structures and Materials*, **3989**, Orange County, CA, March 2000, pp. 61-72.

◇ Centolanza, L.R., Smith, E.C., and Munsky, B.E. Design, Fabrication, and Experimental Testing of an Induced-Shear Piezoelectric Actuator for Rotor Blade Trailing Edge Flaps, *Smart Materials and Structures*, **11**, Feb. 2002, pp. 24-35.

SKILLS

- ◇ **S**ystems and Control theory based modeling and analysis techniques for biological systems. These include tools for sensitivity and robustness analysis, model reduction, and parameter identification for complex bio-networks.
- ◇ **N**umerical methods for stochastic systems: stochastic simulations, τ leaping, hybrid methods, Stochastic Differential Equations, Finite State Projection reductions for master equations, and others.
- ◇ **N**umerical analyses of PDEs using finite difference, finite element, Raleigh-Ritz, and other approaches.
- ◇ **C**/C++, Fortran, Matlab, Unix, Linux, MS-Windows, Mac OS X, LaTeX, Maple, and others.

PROFESSIONAL ACTIVITIES

- ◇ Referee for Physics Letters A, IEEE Conference on Decision and Control, IET Systems Biology, Biotechnology Progress, and Smart Materials and Structures
- ◇ Member of the Institute of Electrical and Electronics Engineers

Abstract

The Finite State Projection
for the Solution of the Master Equation
and its Applications to Stochastic Gene Regulatory Networks

by

Brian Munsky
Doctor of Philosophy in Mechanical Engineering

University of California, Santa Barbara

Mustafa Khammash, Chair

This dissertation discusses the Finite State Projection (FSP) method for the direct computational analysis of probability distributions arising from discrete state Markov Processes. While the methods contained herein apply to a wide range of scientific inquiries, this study focuses on the treatment of chemically reacting biological systems. The probability distributions of such systems evolve according to a set of linear ordinary differential equations known as the chemical master equation (CME) or forward Kolmogorov equation. If the CME describes a system that has a finite number of distinct configurations, then the FSP method provides an exact analytical expression for its solution. When an infinite or extremely large number of variations is possible, the state space is truncated, and the FSP method provides a certificate of accuracy for how closely the FSP approximation matches the true solution. The proposed FSP algorithm systematically increases the projection space to meet any pre-specified error tolerance in the probability distribution. For any system in which a sufficiently accurate FSP solution exists, the FSP algorithm is shown to converge in a finite number of steps. The FSP approach is enhanced by taking advantage of well-known tools from modern control and dynamical systems theory such as minimal realizations, balanced truncation, linear perturbation theory, and coarse gridding approaches. Each such reduction has successfully improved the efficiency and applicability of the FSP, and more are envisioned to be possible.

The power of the FSP is illustrated on a few important genetic regulatory networks including a toy model of the heat shock mechanism in *E. coli* and a detailed analysis of a genetic toggle switch. The FSP method is also applied to a detailed model of the Pap pili epigenetic switch in *E. coli.* The Pap model predicts the switching behavior of the Pap system under varying levels of various regulatory molecules and under the influence of various gene insertions and mutations. When possible, predictions have been validated against experimental observations. In all cases, the current model matches the observed qualitative behavior of the Pap switch and provides an excellent starting point for future Pap modeling endeavors.

Mustafa Khammash
Dissertation Committee Chair

Table of Contents

Dedication **iv**

Acknowledgments **v**

Curriculum Vitæ **vii**

Abstract **xii**

List of Figures **xvii**

List of Tables **xx**

1 Introduction **1**
1.1 Outline of this Dissertation 4

2 The Chemical Master Equation **6**

3 Monte Carlo Solutions to the CME **10**
3.1 The Stochastic Simulation Algorithm 10
3.1.1 System partitioning methods 13
3.1.2 τ leap methods 14
3.1.3 Chemical Langevin Equation 16
3.1.4 StochSim 16
3.2 Solving the CME with Monte Carlo Algorithms 17

4 Moment Analyses for the CME **19**
4.1 Linear Noise Approximation 20
4.2 Moment Closure and Moment Matching Approaches 21

5 The Finite State Projection Method (FSP) **23**
5.1 Basic FSP Algorithm 29
5.2 Initializing $\mathbf{X}_{J_0}$ 30

5.3 Expansion through N-step Reachability 31
5.4 Expansion through Probability Tracking 34

6 Minimal Realizations for the FSP Method 37
6.1 Aggregation of Unobservable Configurations (OA-FSP) 37
6.1.1 Estimating the probability of important states 39

7 The Slow Manifold FSP (SM-FSP) 46
7.1 Simple SM-FSP Example 55

8 The FSP for Non-Sparse Initial Distributions (NS-FSP) 59

9 The Multiple Time Interval FSP Method (MTI-FSP) 66
9.1 The FSP τ Leap Approach (τ-FSP) 71

10 Interpolation Based FSP Reduction (I-FSP) 75
10.1 Non-Linear Shape Functions (NL-FSP) 80
10.1.1 Poisson counting process 84
10.1.2 Birth-death process 85

11 FSP for the Analysis of Stochastic Switches and Trajectories 89
11.1 Pathway Bifurcation analysis with the FSP 92
11.2 Analyzing switch statistics with the FSP 95
11.2.1 Input-output description of connected Markov chains 98
11.2.2 Numerical convolution to compute trajectory times 98
11.2.3 Probabilities of Specific Trajectories 100

12 Sensitivity Analysis and Identification of Stochastic Models 105
12.1 Sensitivity Analyses of Stochastic Processes 106
12.2 Identification of Stochastic Processes 108
12.2.1 Moment analysis of a simple gene regulatory network . . . 108
12.2.2 Identifying transcription parameters 111
12.2.3 Identifying transcription and translation parameters 113
12.2.4 Non-linear optimization based identification 118
12.2.5 Transcription and translation identification examples . . . 121
12.2.6 Effect of measurement errors on identification 124

13 Applying the FSP to a Toy Model of the Pap Switch 130
13.1 Exact Solution for Finite State Problem 131
13.2 Approximate Solution Infinite State Problem 135

14 Applying the FSP to the Toy Heat Shock Model 147

15 Applying the FSP to the Genetic Toggle Switch 159
15.1 FSP Switch Rate Analysis . . . 167
15.2 Sensitivity Analysis of the Toggle Switch . . . 176
15.3 Identifying Gene Toggle Parameters . . . 176

16 Case Study: The Pap Pili Epigenetic Switch in *E. Coli* 190
16.1 The Core Mechanics of the Pap Switch . . . 192
16.1.1 The *pap* operon . . . 193
16.1.2 Leucine-Responsive regulatory Protein (Lrp) . . . 194
16.1.3 DNA Adenine Methylase (DAM) . . . 196
16.1.4 The PapI and PapB local regulatory proteins . . . 201
16.1.5 Modeling of multiple generations . . . 204
16.2 Analysis of the Pap Switch . . . 206
16.2.1 Wild-type Pap analysis . . . 208
16.2.2 Effect of DNA Adenine Methylase . . . 210
16.2.3 Comparison with experimental results . . . 215
16.2.4 Effect of Leucine Responsive regulatory Protein . . . 215
16.2.5 Effect of PapI . . . 221
16.2.6 Effect of various *pap* mutations . . . 222

17 Conclusions and Future Work 225

Bibliography 230

List of Figures

1.1 Schematic representation of gene transcription, translation and regulation. 4

3.1 Convergence of error for Monte Carlo coin-toss experiment. . . . 18

5.1 Intuitive schematic representation of the basic Finite State Projection approach. 27

5.2 Schematic representation of the concept of N-step reachability. . 31

6.1 Schematic representation of the Observability Aggregated FSP (OAFSP) approach. 45

7.1 Schematic representation of the Slow Manifold FSP (SM-FSP) approach. 49

7.2 Comparing the SM-FSP and the full FSP on a simple example. . 57

7.3 Comparison of the SM-FSP with the full FSP for a random set of examples with varying degrees of time scale separation. 58

9.1 Schematic representation of the Multiple Time Interval FSP (MTI-FSP) method. 67

10.1 Schematic representation of the Interpolation reduced FSP (I-FSP) method for a one dimensional lattice. 80

10.2 Comparison of the I-FSP and full FSP methods for the solution of a simple one dimensional Markov process. 81

10.3 Interpolation scheme for a two dimensional Markov lattice. . . . 81

11.1 Schematic description of the use of the FSP to analyze stochastic decisions and switch times. 93

11.2 Schematic description of the FSP based computation of round trip times for discrete state Markov processes. 96

12.1 Identification of system parameters based upon observation of the statistical moments of gene transcripts and proteins. 123
12.2 Identification of system parameters based upon observation of the statistical moments of proteins only. 124
12.3 Effect of noisy measurements on estimation error versus number and period of measurements. 128

13.1 Schematic of the simplest toy model of the Pap pili epigenetic switch. 132
13.2 Solution of the simplest Pap switch model using FSP, SSA and τ leaping approaches. 135
13.3 Probability density vector for possible configurations of the Pap switch. 141
13.4 Probability density of PapI for each gene configuration of the Pap switch. 142
13.5 Probability density of PapI in the Pap switch. 143
13.6 Predictions of **OFF** to **ON** switching rate for various computational methods. 145

14.1 Schematic representation of the Toy Heat shock model. 148
14.2 Schematic representing the possible configurations of the toy heat shock model, and a description of the application of FSP reduction to the model. 149
14.3 Multiple Time Interval Tradeoffs. 155
14.4 Probability distribution of σ_{32}-RNAP complexes at different times as computed using various solutions of the toy heat shock model. 158

15.1 Schematic of the two inhibitor toggle model. 160
15.2 Projection used for the genetic toggle model. 162
15.3 Contour plots of the probability distributions for the Genetic toggle regulatory network. 164
15.4 Alternate representation of the probability distributions of the Genetic toggle regulatory network. 165
15.5 Probability distribution for the genetic toggle switch for different parameters. 167
15.6 Probability distributions of toggle switch and trajectory completion times. 174
15.7 Sensitivity analysis of the toggle switch 177
15.8 Toggle distributions based system identification 181
15.9 Effect of UV and MMC on degradation of u 182
15.10 The distribution of u with different parameter sets 184
15.11 The distribution of u at different times and UV levels. 186
15.12 The distribution of v at different times and UV levels. 187

16.1 Organization of the *pap* operon. 193
16.2 Schematic of the Possible *pap*-Lrp binding configurations. 196
16.3 Schematic of the 16 possible *pap* methylation configurations. . . 199
16.4 Configurations of ON and OFF states of the *pap* operon. 200
16.5 Schematic of the 64 possible *pap* operon configurations 201
16.6 Probability distribution of Pap molecules in wild-type *E. coli* . . 209
16.7 Stationary probability distribution of Pap molecules in wild-type *E. coli*. 210
16.8 Probability of wild-type*pap* methylation patterns. 211
16.9 Probability of wild-type*pap* Lrp binding patterns. 211
16.10 Effect of DAM levels on the Pap switch. 212
16.11 Effect of DAM population on *pap* methylation patterns. 213
16.12 Effect of DAM population on *pap* Lrp binding patterns. 214
16.13 Comparison of model predictions and experimental observations. 216
16.14 Effect of Lrp levels on the Pap switch. 217
16.15 Tradeoff between DAM and Lrp in the expression of Pap. 218
16.16 Effect of Lrp population on *pap* methylation patterns. 219
16.17 Effect of Lrp population on *pap* Lrp binding patterns. 220
16.18 Effect of PapI feedback on Pap switch. 222
16.19 Effect of mutations on Pap switch. 223

List of Tables

12.1 Results of various identification strategies on a gene transcription/translation process. 129

13.1 Parameters for the simplest toy model of the Pap switch. 136
13.2 Reactions and parameters for a toy Pap model. 146
13.3 Comparison of accuracy and efficiency for various solutions of the toy Pap model. 146

14.1 Accuracy and efficiency of various computations of the master equation for the toy heat shock model. 157

15.1 Computational efficiency and accuracy of various solutions of the chemical master equation for the stochastic genetic toggle model. 166
15.2 Accuracy and efficiency of various computations of the master equation for the stochastic gene toggle model. 175
15.3 Three Parameter Sets for the Stochastic Toggle Model 189

16.1 Reaction rate parameters for the Lrp association and dissociation events. 197
16.2 Pap replication stoichiometries for various assumptions. 207
16.3 Experimentally observed Pap switching behavior for wild-type *E. coli* and four constructed mutations. 224

Chapter 1

Introduction

In cellular biology, much remains unexplained. While modern genetic and molecular biology techniques have successfully revealed elaborate regulatory networks that ultimately control various biological responses, the experiments needed to explore these systems are expensive, time-consuming or otherwise difficult to perform. With the right tools, computational models can help fill the gaps in our understanding, enable us to design better experiments, and increase the yield of experimental procedures. A major goal of Systems Biology is to combine experimental procedures and computational models to explain how individual regulatory steps integrate to produce observable responses. Such models could assist biologists and biochemists to (1) gain new understanding into complicated regulatory phenomena, (2) pinpoint key regulatory features and alter those features to achieve desired outcomes, and (3) achieve better understanding of how and why regulatory systems have evolved in different species. One of the biggest hurdles in this endeavor is dealing with the inherent stochasticity of cellular processes.

The cellular environment is abuzz with noise [60, 27, 95, 44, 76, 29, 51]. The origin of this noise is attributed to the random events that govern the motion of cellular constituents at the molecular level. Cellular noise not only results in random fluctuations within individual cells, but it is also a source of phenotypic variability among clonal cellular populations [6]. In some instances, these fluctuations are suppressed downstream through intricate dynamical networks that act to filter the noise [22], much like a low pass filter attenuates high frequency signals. Yet in other instances, noise induced fluctuations are exploited to the cell's advantage. Researchers are only now beginning to understand that the richness of stochastic phenomena in biology depends directly upon these interactions of dynamics and noise and upon the mechanisms through which these interactions occur. Intriguing examples of mechanisms that rely on noise include stochastic switches [6, 66, 97], coherence resonance in oscillators [57], and stochastic focusing for the amplification of signals [77].

Given the importance of noise induced stochastic fluctuations in the cell, the quantitative modeling and analysis of these fluctuations is of paramount importance for the understanding and synthesis of biological networks. While mathematical models of genetic networks often represent gene expression and regulation as deterministic processes with continuous variables, the stochastic nature of cellular noise necessitates an approach that models these variables as discrete and stochastic. The continuous and deterministic approach makes sense when large numbers of molecules justify a continuous valued concentration description using mass-action kinetics. In this case, chemical reactions are modeled as reaction diffusion processes, and their dynamics can be found with partial differential equations (PDEs). When the reacting chemical solutions are well-mixed, these PDEs can then be well approximated with ordinary differential equations (ODEs). On the other hand, the cellular milieu is often home to key molecules that can be found in very small integer populations [29]. Indeed in a typical living cell, it is not uncommon for some of the key molecules have ten or fewer copies. Clearly, in these instances the concentration description is meaningless, and a discrete stochastic model of the chemical species is essential. The choice between the two modeling approaches in not always clear. What is clear, however, is that as the size of the system of interacting species decreases, intrinsic noise becomes increasingly important (a relative change of one molecule is very important when there are only ten to begin with). At the sub-cellular level where gene regulatory networks reside, crucial chemical species such as DNA, RNA, and regulatory proteins may be present in only one or two copies per cell [60]. In these networks, which affect all aspects of life, stochastic effects have been found to play a significant and often a detrimental role in various aspects of cell function.

As a simple example, Fig. 1.1 represents a generic gene regulatory network comprised of only three mechanisms: transcription, translation, and regulatory feedback. With intrinsic noise, even this simple system can exhibit a rich variety of behaviors. For example, consider an open-loop system where transcription is slow, but translation is very fast. Such a strategy, which may be used to conserve energy [59], can result in systems where the transcripts may be entirely absent from the cell most of the time. However, because of efficient translation, one of these rare transcripts may occasionally result in large bursts of proteins [59, 76]. Because such events can happen in some cells and not in others, they may account for huge variation in phenotype despite isogenic populations [76]. Conversely, if transcription were much faster and translation slower, the same average amount of protein may be found, but the variation could be far less [59].

Chemical regulators may also induce phenotypical variation despite homogenous genotypes, as will be seen in great detail in the examination of the the Pap

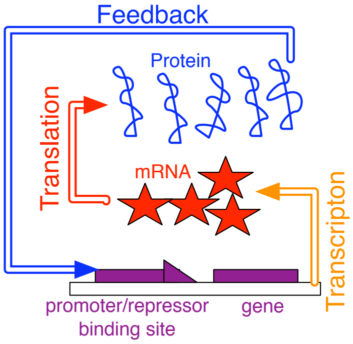


Figure 1.1: Schematic representation of gene transcription, translation and regulation. When in an "on" configuration the gene will transcribe mRNA molecules (stars). These, in turn, are translated to produce regulatory proteins, which can regulate the gene, turning it "off" in the case of negative feedback or "on" in the case of positive feedback.

Pili epigenetic switch in *E. coli* in Chapter 16 of this dissertation. In the *pap* system, DNA adenine methylase (DAM) applies irremovable methyl groups at some key regulatory regions of the DNA. In one location, these methyl groups can help activate the *pap* gene, it another location, the methyl group will deactivate the gene [46]. The system is further affected by intrinsic noise due to a transcriptional feedback mechanism similar to that illustrated in Fig. 1.1. In this case, the *pap*-encoded protein PapI works in conjunction with Leucine-responsive regulator protein (Lrp) to block Dam from methylating the sites which turn the gene expression off.

1.1 Outline of this Dissertation

This dissertation is comprised of three parts. The first part provides a brief review of previous tools for the stochastic analysis of gene regulatory networks. Chapter 2 reviews the discrete chemical reaction problem on the mesoscopic scale and derives what is commonly referred to as the Chemical Master Equation (CME). Next, Chapter 3 reviews a few of the recent Kinetic Monte Carlo approaches for generating sample trajectories of systems described by the CME. Chapter 4 discusses some recently proposed analytical techniques for solving for the evolution of moment distributions of stochastic processes.

The second part provides new analytical tools for the mathematical modeling and analysis of discrete stochastic systems. The vast majority of Chapters 5 through 11 focus on these new approaches for computing the solution to the CME. This technique, called the Finite State Projection (FSP) method, involves the projection of the solution of the CME onto finite, solvable subsets. In addition to presenting and explaining the theory underlying the FSP approach, Chapters 6 through 10 describe some system theory based modifications and enhancements that enable large reductions and increased efficiency with little to no loss in accuracy in the FSP solution.

While the FSP approach is applicable to any discrete state Markov process, its development has been driven by the study of the gene regulatory systems. As such, the third part of this dissertation illustrates the FSP methods on such cellular networks. Chapter 13 considers a toy model of the gene regulatory network that controls the expression of Pap pili in *E. coli.* Chapter 14 examines a toy model of the Heat Shock mechanism in *E. coli.* Chapter 15 concentrates on a detailed analysis of a stochastic model of Gardner's genetic toggle switch [31]. The main case study in Chapter 16 then describes a much more detailed model of the Pap switch, analyzes it using the various FSP tools, and validates it against experimental studies in the literature. Finally, Chapter 17 summarizes the main results of this work and outline a few directions for future work.

Chapter 2

The Chemical Master Equation

Gillespie's 1992 paper [36] provides a good background on the stochastic chemical kinetics problem and its major result: the forward Chapman Kolmogorov equation, commonly referred to as the chemical master equation (CME). For convenience, this chapter provides a much simplified and less rigorous outline of his argument. Consider two molecules s_1 and s_2 moving around in a system of volume V. Suppose that molecule s_1 moves with the speed u, but in randomly changing directions. Suppose that a reaction $s_1 + s_2 \rightarrow s_3$ will occur when the center of molecule s_1 comes within a distance r of the center of molecule s_2. In some small fraction of time, dt, the molecule s_1 will cover a distance udt and will sweep a region dV whose volume is approximately $\pi r^2 udt$. If the center of s_2 is in dV then a reaction will occur; otherwise it will not. Since the system is well mixed, the probability that s_2 is in that region and that a reaction will occur is $\pi r^2 u V^{-1} dt$. If there were ξ_1 molecules of s_1 and ξ_2 molecules of s_2, then the probability that *any* such reaction will occur is given by $\xi_1 \xi_2 \pi r^2 u V^{-1} dt$.

For a chemical solution of N species, $\{s_1, \ldots, s_N\}$, one can define the system state as $\mathbf{x} = [\xi_1, \ldots, \xi_N]$. Each μ^{th} reaction is a transition from some state $\mathbf{x}_i$ to some other state $\mathbf{x}_j = \mathbf{x}_i + \nu_\mu$, where ν_μ is known as the *stoichiometric vector*.

Following the methodology above, each reaction also has a *propensity function*, $w_\mu(\mathbf{x})dt$, which is the probability that the μ^{th} reaction will happen in a time step of length dt. For example, the reaction $s_1 + s_2 \rightarrow s_3$ discussed above has the stoichiometric vector $\nu = [-1, -1, 1]^T$, and a propensity $w(\mathbf{x})dt = \xi_1\xi_2\pi r^2 u V^{-1} dt$.

The stoichiometry and propensity functions for each of the M possible reactions fully define the system dynamics and are sufficient to find sample trajectories with the Monte Carlo methods of Chapter 3. However, for many interesting gene regulatory problems individual system trajectories are not the best description. Instead, it is desirable to analyze the dynamics in terms of probability distributions. For this it is useful to derive the chemical master equation (CME).

Suppose that one knows the probability of all states $\mathbf{x}_i$ at time t, then the probability that the system will be in the state $\mathbf{x}_i$ at time, $t + dt$, is equal to the sum of (i) the probability that the system begins in the state $\mathbf{x}_i$ at t and remains there until $t + dt$, and (ii) the probability that the system is in a different state at time t and will transition to $\mathbf{x}_i$ in the considered time step, dt. This probability can be written as:

$$p(\mathbf{x}_i; t + dt) = p(\mathbf{x}_i; t)\left(1 - \sum_{\mu=1}^{M} w_\mu(\mathbf{x})dt\right) + \sum_{\mu=1}^{M} p(\mathbf{x}_i - \nu_\mu; t) w_\mu(\mathbf{x}_i - \nu_\mu)dt. \tag{2.0.1}$$

From Eqn 2.0.1 it is relatively simple to derive the differential equation known as the Chemical Master Equation, or CME [35]:

$$\dot{p}(\mathbf{x}; t) = -p(\mathbf{x}; t)\sum_{\mu=1}^{M} w_\mu(\mathbf{x}) + \sum_{\mu=1}^{M} p(\mathbf{x} - \nu_\mu; t) w_\mu(\mathbf{x} - \nu_\mu). \tag{2.0.2}$$

This time derivative of the probability density of state $\mathbf{x}$ can also be written in vector form as:

$$\dot{p}(\mathbf{x};t) = \begin{bmatrix} -\sum_{\mu=1}^{M} w_\mu(\mathbf{x}) \\ w_1(\mathbf{x}-\nu_1) \\ w_2(\mathbf{x}-\nu_2) \\ \vdots \\ w_M(\mathbf{x}-\nu_M) \end{bmatrix}^T \begin{bmatrix} p(\mathbf{x};t) \\ p(\mathbf{x}-\nu_1);t) \\ p(\mathbf{x}-\nu_2);t) \\ \vdots \\ p(\mathbf{x}-\nu_M);t) \end{bmatrix}. \tag{2.0.3}$$

Fix a sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots$ of elements in $\mathbb{N}^N$ and define $\mathbf{X} := [\mathbf{x}_1, \mathbf{x}_2, \ldots \ \ ]^T$ as the set of all possible configurations. The particular sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots$ may be chosen to visit every element of the entire space $\mathbb{N}^N$. In this case, the choice of $\mathbf{X}$ corresponds to a particular enumeration of the space $\mathbb{N}^N$. Once $\mathbf{X}$ is selected, Eqn 2.0.3 can be rewritten as a single linear expression:

$$\dot{\mathbf{P}}(\mathbf{X};t) = \mathbf{A} \cdot \mathbf{P}(\mathbf{X};t), \tag{2.0.4}$$

where $\mathbf{P}(\mathbf{X};t) := [p(\mathbf{x}_1,t), p(\mathbf{x}_2,t), \ldots \ \ ]^T$, is the complete probability density state vector at time t, and $\mathbf{A}$ is the infinitesimal generator of the discrete stochastic process. For convenience, let the notation $\mathbf{P}(t) := \mathbf{P}(\mathbf{X};t)$ denote the distribution on the entire set $\mathbf{X}$. The columns and rows of $\mathbf{A}$ are uniquely defined by the system's stoichiometry and the choice of $\mathbf{X}$. Beginning at any state, $\mathbf{x}_i$, there can be a maximum of M possible reactions; each reaction leads to a different state: $\mathbf{x}_j = \mathbf{x}_i + \nu_\mu$. Thus, $\mathbf{A}$ is typically a very sparse matrix.

The infinitesimal generator contains information regarding every reaction, each weighted by the corresponding propensity function, and the elements of $\mathbf{A}$ are given as:

$$\mathbf{A}_{ji} = \left\{ \begin{array}{cc} -\sum_{\mu=1}^{M}(w_\mu(\mathbf{x}_i)) & \text{for } (i=j) \\ w_\mu(\mathbf{x}_i) & \text{for all } j \text{ such that } (\mathbf{x}_j = \mathbf{x}_i + \nu_\mu) \\ 0 & \text{Otherwise} \end{array} \right\}. \tag{2.0.5}$$

A has the properties that it is independent of t; all of its diagonal elements are non-positive; all its off-diagonal elements are non-negative; and all its columns sum to exactly zero. The solution to the linear ODE beginning at $t = 0$ and ending at $t = t_f$ in Eqn 2.0.4 is the expression:

$$\mathbf{P}(t_f) = \mathbf{\Phi}(0, t_f) \cdot \mathbf{P}(0). \tag{2.0.6}$$

In the case where there are only a finite number of reachable states, the operator, $\mathbf{\Phi}(0, t_f)$, is the exponential of $\mathbf{A}t_f$, and one can, in principle, compute the solution: $\mathbf{P}(t_f) = \exp(\mathbf{A}t_f)\mathbf{P}(0)$. Of course, for many systems, **X** may be infinite dimensional, or at least very large, and the corresponding analytical solution may be very difficult, or even impossible, to compute. For such systems, researchers have developed a number of different techniques, which will be discussed in the following chapters.

Chapter 3

Monte Carlo Solutions to the CME

Because the CME is often infinite dimensional, it is usually impossible to solve exactly. For this reason, the majority of analyses at the mesoscopic scale have been conducted using kinetic Monte Carlo (MC) algorithms. The most widely used of these algorithms is Gillespie's Stochastic Simulation Algorithm (SSA) [34, 35], for which there are large numbers of variants [33] and approximations [82, 15, 43, 86, 38, 96, 19, 17, 84, 37, 83, 81]. These are discussed in the following subsections.

3.1 The Stochastic Simulation Algorithm

Gillespie Stochastic Simulation Algorithm (SSA) [34, 35] is the most common tool in use for stochastic analyses at the mesoscopic level. This is to be expected, because once one defines the propensity functions and the stoichiometry for each of the M reactions, the SSA is very easy to apply. Each step of the SSA begins at a random state $\mathbf{x}$ and a time t and is comprised of three tasks, (i) generate the time until the next reaction, (ii) determine which reaction happens at that time, and (*iii*) update the time and state to reflect the previous two choices. There are two common methods typically used to accomplish tasks (*i*) and (*ii*) above; these are referred to as the "direct" and "next reaction" methods and are considered below.

For a single reaction with propensity function, $w(\mathbf{x})$, the random time of the next reaction, $\tilde{\tau}$, is an exponentially distributed random variable with mean $w^{-1}(\mathbf{x})$:

$$P_{\tilde{\tau}}(\tau) = \frac{1}{w(\mathbf{x})} \exp\left(-\frac{\tau}{w(\mathbf{x})}\right),$$

where the notation $P_{\tilde{\tau}}(\tau)$ denotes the probability that the random variable $\tilde{\tau}$ is equal to τ. For M different possible reactions with propensities $\mathbf{w}(x) = [w_1(\mathbf{x}), \ldots, w_M(\mathbf{x})]$, $\tilde{\tau}$ is the minimum of M such random variables. In the "direct" SSA method, one utilizes the fact that the minimum of a finite number of exponential random variables is itself an exponential random variable. Thus, when there are multiple reaction channels, the random time $\tilde{\tau}$ follows the distribution:

$$\begin{aligned} P_{\tilde{\tau}}(\tau) &= \frac{1}{\sum_{\mu=1}^{M} w_\mu(\mathbf{x})} \exp\left(-\frac{\tau}{\sum_{\mu=1}^{M} w_\mu(\mathbf{x})}\right), \\ &= \frac{1}{|\mathbf{w}(\mathbf{x})|_1} \exp\left(-\frac{\tau}{|\mathbf{w}(\mathbf{x})|_1}\right). \end{aligned}$$

In practice, $\tilde{\tau}$ is found by first generating a uniform random number $\tilde{r}_1$ in the interval $(0, 1)$ and applying the relation:

$$\tilde{\tau} = \frac{1}{|\mathbf{w}(\mathbf{x})|_1} \log \frac{1}{\tilde{r}_1}.$$

To determine which of the M reactions occurs at $t + \tilde{\tau}$, one must generate a second random variable, $\tilde{\mu}$, from the set $\{1, 2, \ldots, M\}$ with the probability distribution given by:

$$P_{\tilde{\mu}}(\mu) = \frac{w_\mu(\mathbf{x})}{|\mathbf{w}(\mathbf{x})|_1}.$$

In practice, $\tilde{\mu}$ is typically generated by using a second uniform random variable $\tilde{r}_2$ on the interval (0,1) and the relation:

$$\tilde{\mu} = \min(k) \text{ such that } \sum_{\mu=1}^{k} \frac{w_\mu(\mathbf{x})}{|\mathbf{w}(\mathbf{x})|_1} > r_2.$$

Thus, in the direct method of the SSA, the random time and type of each reaction is generated with a set of exactly two uniform random numbers.

In the "next reaction" method of [33], the authors show that when there is a very large number of different reaction types, then an individual reaction may not affect the majority of the remaining propensity functions. Such an abundance of reaction channels is common in reaction diffusion processes where a reaction in one spatial cell has no effect on reaction channels in other spatial cells. In such cases, it may be advantageous to generate the individual times for each of the M reaction channels rather than the time of the first reaction as in the direct method. Many of these reaction times can then be updated and reused over many subsequent time steps. In the limit of an infinite number of reaction types or completely uncoupled reaction rates, such an approach will require only one random variable per reaction. However, in many cases the computational cost of storing and updating reaction times from one time step to the next may overshadow the benefit seen by requiring fewer random numbers [18]. In [81] the next reaction method has also been used in conjunction with the τ leaping strategy described below.

In either the direct or the next reaction methods, once $\tilde{\tau}$ and $\tilde{\mu}$ have been chosen, the system is updated to $t = t + \tilde{\tau}$ and $\mathbf{x} = \mathbf{x} + \nu_{\tilde{\mu}}$, and the process continues until the final time of interest is reached. Both SSA approaches are exact in the sense that they generate a random trajectory, $\tilde{\mathbf{x}}(t)$, with a probability distribution exactly equal to the solution of the corresponding CME at each point in time. However, each run of the SSA provides only a single, not necessarily representative, trajectory. Should one actually wish to reproduce the probability distribution, the SSA must be run many times. For this reason, many accelerated approximations have been proposed to improve the efficiency of the SSA.

3.1.1 System partitioning methods

In the first type of approximation to the SSA, the system is partitioned into slow and fast portions [82, 15, 16, 94]. This partitioning has been approached in a number of different manners. In [82] the system is separated into slow "primary" and fast "intermediate" species. This method uses three random variables at each step: first, the primary species' populations are held constant, and the population of the intermediate species is generated as a random variable from its quasi-steady-state (QSS) distribution. The dynamics of the "primary" species are then found with two more random variables, similar to the SSA above but with propensity functions depending upon the chosen populations of the intermediates species. The more recently developed Slow-Scale SSA (ssSSA) [15, 16] is very similar in that the system is again separated into sets of slow and fast species. The ssSSA differs in that it does not explicitly generate a realization for the fast species, but instead uses the QSS distribution to scale the propensities of the slow reactions. In [15, 16] the QSS approximation is made by solving a relatively simple algebraic equation for the quasi-steady distribution of the fast species. In some cases this distribution does not have such a simple form, and alternative approaches must be taken to determine this distribution. In [107] and [87] the authors run a short run on an inner SSA loop for the fast reactions in order to compute the average rate of the slow reactions. As we will see later in Chapter 7, the methods in [15, 16, 107, 87] effectively work by generating trajectories for a new master equation that corresponds to a slow manifold projection of the original master equation.

Hybrid Methods

So-called hybrid methods such as [43] and [86] also separate the system into separate frequent and infrequent partitions, but these methods do not then rely upon a QSS approximation. Instead, the fast reactions are approximated with deterministic ODEs or as continuous valued Markov processes using Langevin equations, and the slow reactions are treated in a manner similar to the SSA except now with time varying propensity functions. Such approaches are very useful when there is a huge separation in the population numbers of different species.

3.1.2 τ leap methods

The second approach to accelerating the SSA assumes that propensity functions are constant over small time intervals. With this "τ leap assumption" one can model each of the M reaction channels as an independent Poisson random process [38]. Beginning at time t and state $\mathbf{x}(t)$, the state at the end of a time step of length τ is approximated as

$$\mathbf{x}(t+\tau) = \mathbf{x}(t) + \sum_{\mu=1}^{M} \tilde{k}_\mu \nu_\mu, \tag{3.1.1}$$

where each $\tilde{k}_\mu$ is a random variable chosen from the Poisson distribution:

$$P_{\tilde{k}_\mu}(k) = \frac{\lambda^k \mathrm{e}^{-\lambda}}{k!},$$

where $\lambda = w_\mu(\mathbf{x})\tau$. The accuracy of τ leaping methods depends only upon how well the τ leap assumption is satisfied. Naturally, the τ leap assumption is best satisfied when all species have sufficiently large populations and all propensities functions are relatively smooth. Otherwise, small changes in populations could result in large relative changes in propensities. Ignoring these changes can easily lead to unrealistic predictions of negative populations and/or numerical stiffness. One may avoid negative populations by using a Binomial τ leap strategy [96, 19] or by adaptively choosing the size of each τ leap [17]. One can also ameliorate the problem of numerical stiffness using implicit methods such as that in [84]. While these approaches relieve the possibility of negative populations, one must be careful that they do not artificially satisfy the τ leap assumption by changing the system. In particular, if the true system has propensity functions that change quickly in comparison to the length of the τ leap, then no τ leap strategy will satisfy the τ leap assumption. Nearly all of the examples considered in this work fall into this category. For example, the majority of the reactions of the Pap models of Chapters 13 and 16 have propensity functions that change between positive values and zero with *almost every* reaction. Similarly, the reactions of the toy heat shock model and the toggle switch in Chapters 14 and 15 change very quickly. In each of these, the time leap can be no longer that the average step of the SSA.

3.1.3 Chemical Langevin Equation

When the populations are very large, and the propensity functions are very smooth, the chemical species may be more easily modeled with continuous variables using the *chemical Langevin equation* [37, 39, 5]. In this solution scheme, one assumes that many reactions will occur in the *macroscopic infinitesimal* time step dt without violating the τ leap assumption. At large numbers, the Poisson random variable $\tilde{k}_\mu$ in (3.1.1) can be replaced with a much more easily generated continuous valued Gaussian random variable, $\tilde{y}_\mu$:

$$P_{\tilde{y}_\mu}(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{(y-\sigma^2)^2}{2\sigma^2}\right),$$

where $\sigma^2 = w_\mu(\mathbf{x})dt$ is the mean number of reactions in the time step dt. The final result after this approximation is that the process is treated as a stochastic differential equation (SDE) driven by white noise whose variance is equal to its mean [37, 39].

3.1.4 StochSim

In addition to the SSA, one other common Monte Carlo algorithm is StochSim [64, 65]. This algorithm is an object oriented approach that considers each individual molecule as they interact with one another. The main advantage of this type of approach occurs when the number of molecular species and reactions is extremely large in comparison to the actual population levels. This can be the case when the reacting molecules exhibits a vast number of chemically distinct configurations due to various methylation, phosphorylation or other distinct binding patterns.

3.2 Solving the CME with Monte Carlo Algorithms

Monte Carlo algorithms such as SSA and its various approximations can provide excellent sample trajectories of the process whose distribution evolves according to the chemical master equation. In many cases, these trajectories may be all that is needed in the analysis of a stochastic chemical process. In many other situations, however, a single trajectory or set of trajectories does not provide enough information regarding the overall behavior of the reacting system. In these cases, one may wish to actually solve the CME in order to find the probability of certain traits at certain instances in time. For these, one needs to run many MC simulations. As more runs of the MC algorithm are performed, the error will converge to zero with order $\mathcal{O}(N^{-1/2})$, where N is the number of runs. In other words to diminish the error by a factor of ten, one will require a hundred times as many MC runs. For high precision requirements, the number of MC runs can be prohibitive. For example, three MC analyses of a simple coin toss experiment will predict the probability of heads is 0.500457, 0.500370, and 0.499724 after 10^6 tosses yielding relative errors that of 0.000914 , 0.000740, and 0.000552, respectively. As expected these errors are on the order of $\sqrt{10^{-6}} = 10^{-3}$. For further comparison, Fig. 3.1 shows the convergence of the error with increasing numbers of coin tosses.

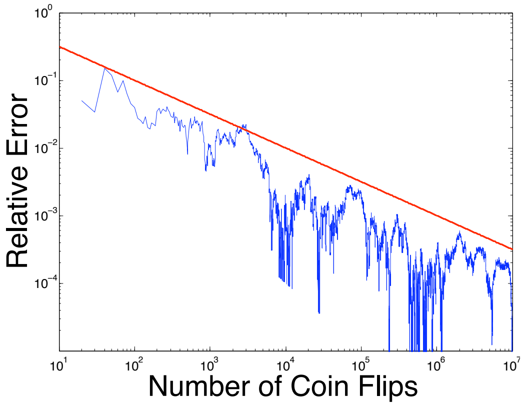


Figure 3.1: Convergence of the error for a simple Monte Carlo Algorithm coin toss experiment. The error, $(\#heads/N - 0.5)$ is plotted versus the number of coin tosses, N. This error converges slowly to zero with $\mathcal{O}(N^{-1/2})$ (compare blue curve to red line).

Chapter 4

Moment Analyses for the CME

Instead of simulating the chemical master equation with a large set of numerical Monte Carlo simulations, one may instead choose to solve or approximate the CME in terms of the evolution of its statistical means and higher order moments of the multi-variate distribution. If $\mathbf{w}(\mathbf{x})$ denotes the propensity functions and $\mathbf{S} = [\nu_1, \ldots, \nu_M]$ denotes the stoichiometry matrix of the M different reactions, then the expected change in $\mathbf{x}$ over the time interval dt can be written as:

$$E\{\mathbf{x}(t+dt)\} = E\{\mathbf{x}(t)\} + \mathbf{S}E\{\mathbf{w}(\mathbf{x}(t))\}dt, \tag{4.0.1}$$

and the mean evolves according to the relatively simple ODE:

$$\frac{d}{dt}E\{\mathbf{x}\} = \mathbf{S}E\{\mathbf{w}(\mathbf{x}).\} \tag{4.0.2}$$

For systems with affine propensity functions, $\mathbf{w}(\mathbf{x}) = \mathbf{F}\mathbf{x} + \mathbf{K}$, this approach is relatively straightforward. In this case, the expected value of the propensity function is simply $E\{\mathbf{w}(\mathbf{x})\} = \mathbf{F}E\{\mathbf{x}\} + \mathbf{K}$, and the equation for the first moment is:

$$\frac{d}{dt}E\{\mathbf{x}\} = \mathbf{S}\left(\mathbf{F}E\{\mathbf{x}\} + \mathbf{K}\right).$$

Similarly, the second, un-centered moment can be shown [53, 56] to evolve according to the equation:

$$\frac{d}{dt}E\{\mathbf{x}\mathbf{x}^T\} = E\{\mathbf{x}\mathbf{w}^T(\mathbf{x})\mathbf{S}^T + \mathbf{S}\mathbf{w}(\mathbf{x})\mathbf{x}^T + \mathbf{S}diag[\mathbf{w}(\mathbf{x})]\mathbf{S}^T\}, \tag{4.0.3}$$

where $diag[.]$ refers to a square diagonal matrix with the elements of [.] on the diagonals. Applying the affine linear formulation for the propensity function yields:

$$\begin{aligned}\frac{d}{dt}E\{\mathbf{x}\mathbf{x}^T\} &= E\{\mathbf{x}(\mathbf{F}\mathbf{x}+\mathbf{K})^T\mathbf{S}^T + \mathbf{S}(\mathbf{F}\mathbf{x}+\mathbf{K})\mathbf{x}^T + \mathbf{S}diag[\mathbf{F}\mathbf{x}+\mathbf{K}]\mathbf{S}^T\},\\ &= E\{\mathbf{x}(\mathbf{x}^T\mathbf{F}^T+\mathbf{K}^T)\mathbf{S}^T + \mathbf{S}(\mathbf{F}\mathbf{x}+\mathbf{K})\mathbf{x}^T + \mathbf{S}diag[\mathbf{F}\mathbf{x}+\mathbf{K}]\mathbf{S}^T\},\\ &= E\{\mathbf{x}\mathbf{x}^T\}\mathbf{F}^T\mathbf{S}^T + \mathbf{S}\mathbf{F}E\{\mathbf{x}\mathbf{x}^T\}\\ &\quad + E\{\mathbf{x}\}\mathbf{K}^T\mathbf{S}^T + \mathbf{S}\mathbf{K}E\{\mathbf{x}^T\} + E\{\mathbf{S}diag[\mathbf{F}\mathbf{x}+\mathbf{K}]\mathbf{S}^T\}.\end{aligned}$$

Similar equations can be found for the covariance and auto covariance matrices (see for example [53, 56]). The important observation to make is that in each case, when the propensity functions are all linear or affine linear, then these moment equations do not depend upon higher order moments. Thus, the equations are finite dimensional and easily solved or simulated. However, when the propensity function are higher order in $\mathbf{x}$ then equations 4.0.2 and 4.0.3 each depend upon all higher order moments, and further approximations are necessary. Some of these approximations are covered in the following paragraphs.

4.1 Linear Noise Approximation

The first and most common approximation of the moment dynamics is the Linear Noise Approximation (LNA) [105, 26, 98], also known as van Kampen's approximation. In the LNA, one expands the solution of the master equation in a Taylor series about the macroscopic trajectory. The first order terms correspond to the macroscopic rate equations, and the second order terms approximate the system noise. The end result is a first order Fokker Planck equation, which is far more readily solved than the CME. In [41] a similar approach is taken except that the computation of the mean is coupled with that of the variances; this mass fluctuations kinetics (MFK) approach allows one to capture fluctuations where the mean deviates from the macroscopic equation. This is particularly important for systems that exhibit stochastic focusing [77]. Because the LNA and MFK approaches both assume that the distribution is Gaussian, they will typically be unable to describe processes that are non-gaussian in nature. In particular, these methods will fail in the analysis of processes with multimodal distributions, such as is the case for systems exhibiting stochastic switching as in [6, 66, 97].

4.2 Moment Closure and Moment Matching Approaches

In a similar approach, the dynamics of each *uncentered* moment of the CME can be shown to depend linearly upon the rest to form an *infinite dimensional moment dynamics* linear ODE equivalent to the CME [47, 92]. By assuming that the distributions are normal [108], lognormal [50, 91], Poisson and binomial [72], or another common form, one can approximate higher order moments in terms of the lower moments and effectively truncate the dynamics. Singh and Hespanha review a few of these approaches for the *stochastic logistic model* in population biology [92]. Each of these shapes may work well in different situations. Without prior knowledge of the shape of the distribution, however, it is impossible to know which distribution will work best for which system. In the same paper, Singh and Hespanha also introduce an effective moment closure technique, which does not make an a priori assumption on the distribution shape, but instead defines a moment closure scheme in which they match the time derivatives of the truncated moment dynamics to the full moment dynamics at the initial time t_0 [92, 93]. The separable structure of this derivative matching approach ensures that the approximate system of equations has a unique, real and positive steady-state solution. Furthermore, the derivative matching guarantees a good approximation during short periods of time. The authors of [93] provide explicit formulas to construct the moment closure functions for arbitrary n^{th}-order truncation, and it is observed that higher values of n lead to better moment dynamics approximations. By examining other moment closure functions, the authors showed that without achieving derivative matching, closure techniques typically fail to closely approximate the exact moment solution. By extending these closure functions to enforce derivative matching, one can improve the accuracy of many previously proposed moment closure functions.

Problems with a single macroscopic steady state often result in unimodal distributions and can be expressed with only the first few moments. For these, the above techniques are very well suited. However, problems that exhibit multimodal distributions, such as switching systems, will require many higher order moments, and the applicability of these methods may quickly degrade.

Chapter 5

The Finite State Projection Method (FSP)

Chapter 2 above shows how one can derive the chemical master equation in the form of (2.0.4):

$$\dot{\mathbf{P}}(t) = \mathbf{A} \cdot \mathbf{P}(t),$$

for which the solution was given in (2.0.6) as:

$$\mathbf{P}(t_f) = \mathbf{\Phi}(0, t_f) \cdot \mathbf{P}(0).$$

In the case where there are only a finite number of reachable states, the operator, $\mathbf{\Phi}(0, t_f)$, is the exponential of $\mathbf{A}t_f$, and one can in principle compute the solution: $\mathbf{P}(t_f) = \exp(\mathbf{A}t_f)\mathbf{P}(0)$.

For a few examples in this study, we are interested only in the probability density at the final time, t_f. This information is simply obtained by computing the exponential of $(\mathbf{A}t_f)$ directly and multiplying the resulting matrix by the initial probability density vector. Moler and Van Loan provide many methods for performing this computation in their celebrated 1978 paper "Nineteen Dubious Ways to Compute the Exponential of a Matrix" [61] and its revisited edition of

2003 [62]. The choice of approach obviously depends upon the particular numerical study. For many examples, exponentials will be computed using the `expm` function in MathWorks `Matlab`. This built-in routine is based upon a scaling and squaring algorithm with a Pade approximation. Other cases will use Roger Sidje's `Expokit`– a powerful matrix exponential package, which solves the system of equations in (2.0.4) using a Krylov subspace approximation [90]. In some situations, one may wish to obtain the probability density at many intermediate times as well as the final time. For this it may be more efficient not to directly calculate the matrix exponential, but instead use a numerical stiff ODE solver such as one of `Matlab`'s `ode15s` or `ode23s`. As will be seen in the following chapters, each of the approaches may be advantageous in different circumstances.

In practice there may be many simple chemical systems for which the exponential representation will produce an exact solution (see the example in Section 13.1). Such cases include any system in which the number of molecules in each species is bounded through considerations such as the conservation of mass. However, when $\mathbf{A}$ is infinite dimensional or extremely large, the corresponding analytic solution is unclear or vastly difficult to compute. Even in these cases, however, one may devise a systematic means of approximating the full system using finite dimensional sub-systems. This systematic truncation approach is known as the Finite State Projection method [67].

The presentation of the FSP method first requires the introduction of some convenient notation. Let $J = \{j_1, j_2, j_3, \ldots\}$ denote an ordered index set corresponding to a specific set of states, $\{x_{j_1}, x_{j_2}, x_{j_3}, \ldots\}$. For any matrix, let $\mathbf{A}_{IJ}$ denote a sub-matrix of $\mathbf{A}$ such that the rows have been chosen and ordered according to I and the columns have been chosen and ordered according to J. For example, if $\mathbf{A}$ is given by:

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix},$$

and I and J are defined as $\{3, 1, 2\}$ and $\{1, 3\}$, respectively; then the sub-matrix, $\mathbf{A}_{IJ}$ is given as:

$$\mathbf{A}_{IJ} = \begin{bmatrix} 7 & 9 \\ 1 & 3 \\ 4 & 6 \end{bmatrix}.$$

Similarly let $\mathbf{A}_J$ denote the principle sub-matrix of $\mathbf{A}$, in which both rows and columns have been chosen and ordered according to J. We will use the notation J' to denote the complement of the set J on the entire set, $\mathbf{X}$. Define the sequence $\{J_k\}$ as a sequence of nested sets such that $J_1 \subseteq J_2 \subseteq J_3 \subseteq \cdots$. In addition to the set notation, the vector $\mathbf{1}$ will be used to denote a column of all ones such that for any vector, $\mathbf{v}$, the product $\mathbf{1}^T\mathbf{v}$ is the sum of the elements in $\mathbf{v}$.

Let $\mathcal{M}$ denote a Markov chain on the configuration set $\mathbf{X}$, such as that shown in Fig. 5.1a, whose master equation is $\dot{\mathbf{P}}(t) = \mathbf{AP}(t)$, with initial distribution $\mathbf{P}(0)$. Let $\mathcal{M}_J$ denote a reduced Markov chain, such as that in Fig. 5.1b, comprised of the configurations indexed by J plus a single absorbing state. The master equation of $\mathcal{M}_J$ is given by

$$\begin{bmatrix} \dot{\mathbf{P}}_J^{FSP}(t) \\ \dot{G}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{0} \\ -\mathbf{1}^T\mathbf{A}_J & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ G(t) \end{bmatrix}, \tag{5.0.1}$$

with initial distribution,

$$\begin{bmatrix} \mathbf{P}_J^{FSP}(0) \\ G(0) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_J(0) \\ 1 - \sum \mathbf{P}_J(0) \end{bmatrix}.$$

Because this master equation is finite dimensional, its solution can be found:

$$\begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ G(t) \end{bmatrix} = \begin{bmatrix} \exp(\mathbf{A}_J t_f) & \mathbf{0} \\ -\mathbf{1}^T \exp(\mathbf{A}_J t) & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(0) \\ \mathbf{G}(0) \end{bmatrix}, \qquad (5.0.2)$$

At this point it is crucial to have a very clear understanding of how the process $\mathcal{M}_J$ relates to $\mathcal{M}$ and in particular the definitions of the terms $\mathbf{P}_J^{FSP}(t)$ and $G(t)$. First, the scalar $G(0)$ is the *exact* probability that the system begins in the set $\mathbf{X}_{J'}$ at time $t = 0$, and $G(t)$ is the *exact probability* that the system has been in the set $\mathbf{X}_{J'}$ at *any* time $\tau \in [0, t]$. Second, the vector $\mathbf{P}_J^{FSP}(0)$ contains the *exact probabilities* that the system begins in the set $\mathbf{X}_J$ at time $t = 0$, and $\mathbf{P}_J^{FSP}(t)$ are the *exact joint probabilities* that the system (*i*) is in the corresponding states $\mathbf{X}_J$ at time t, and (*ii*) the system has remained in the set $\mathbf{X}_J$ *for all* $\tau \in [0, t]$. Note that $\mathbf{P}_J^{FSP}(t)$ also provides a finite dimensional approximation of the solution to the CME, as is clearly seen in the following reformulation of the original FSP theorems (The proofs presented here are highly modified from their original presentation in [67]):

Lemma 5.0.1. *For any index set J and any initial distribution* $\mathbf{P}(0)$,

$$\mathbf{P}_J(t) \geq \mathbf{P}_J^{FSP}(t) \geq \mathbf{0}.$$

Proof. $\mathbf{P}_J^{FSP}(t)$ is a more restrictive joint distribution than $\mathbf{P}_J(t)$. □

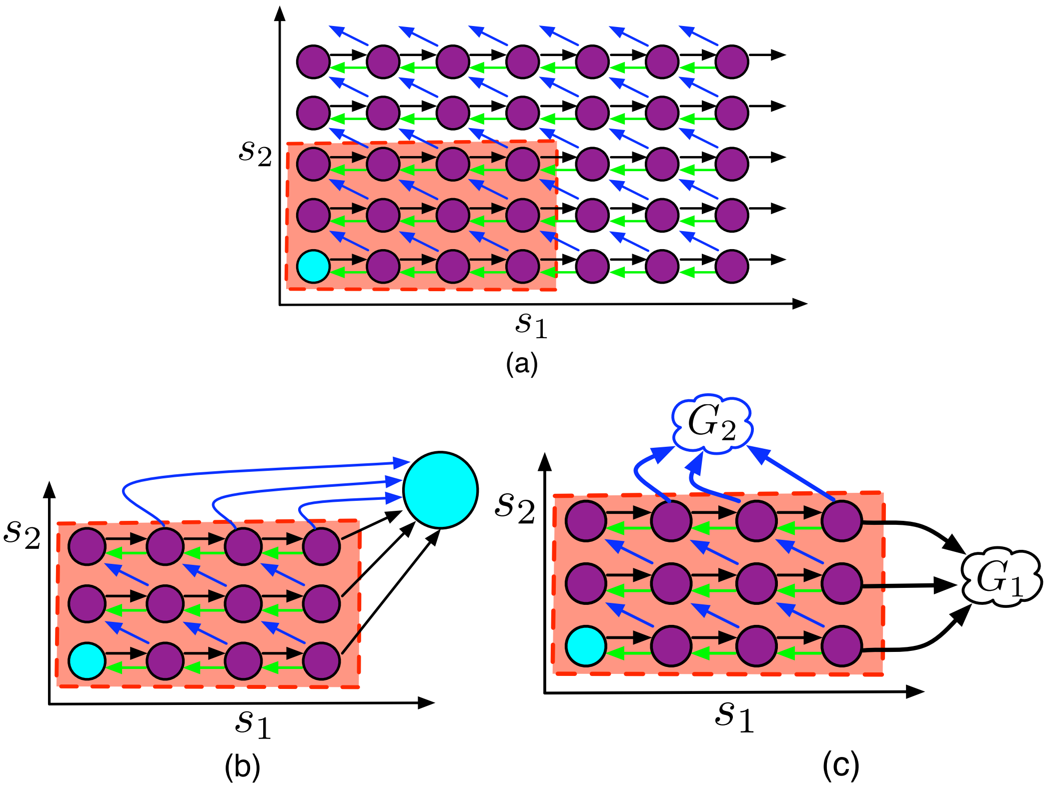


Figure 5.1: (a) Two dimensional lattice of configurations for a chemically reacting system with two species. The system begins in the configuration shaded in grey and undergoes three reactions: The first reaction $\emptyset \rightarrow s_1$ results in a net gain of one s_1 molecule and is represented by right arrows. The second reaction $s_1 \rightarrow \emptyset$ results in a net loss of one s_1 molecule and is represented by a left arrow. The third reaction $s_1 \rightarrow s_2$ results in a loss of one s_1 molecule and a gain of one s_2 molecule. The dimension of the Master equation is equal to the total number of configurations, and is too large to solve exactly. (b) In the original FSP algorithm a configuration subset is chosen and all remaining configurations are projected to a single absorbing point, G. This results in a small dimensional solvable master equation, where the total error is given by the probability that has leaked into G. (c) Instead of considering only a single absorbing point, transitions out of the finite projection can be sorted as to how they leave the projection space. In this case, G_1 and G_2 absorb the probability that has leaked out through reactions 1 and 2, respectively. This information can then be used to expand the configuration set in later iterations of the FSP algorithm (See Section 5.4).

Theorem 5.0.2. *Consider any Markov chain $\mathcal{M}$ and its reduced Markov chain $\mathcal{M}_J$. If $G(t_f) = \varepsilon$, then*

$$\left| \left[\begin{array}{c} \mathbf{P}_J(t_f) \\ \mathbf{P}_{J'}(t_f) \end{array} \right] - \left[\begin{array}{c} \mathbf{P}_J^{FSP}(t_f) \\ \mathbf{0} \end{array} \right] \right|_1 = \varepsilon. \tag{5.0.3}$$

Proof. The left side of (5.0.3) can be expanded to:

$$LHS = \left|\mathbf{P}_J(t_f) - \mathbf{P}_J^{FSP}(t_f)\right|_1 + \left|\mathbf{P}_{J'}(t_f)\right|_1.$$

Applying the Lemma 5.0.1 yields

$$LHS = \left|\mathbf{P}_J(t_f)\right|_1 - \left|\mathbf{P}_J^{FSP}(t_f)\right|_1 + \left|\mathbf{P}_{J'}(t_f)\right|_1.$$

Since $\mathbf{P}(t_f)$ is a probability distribution $\left|\mathbf{P}_J(t_f)\right|_1 + \left|\mathbf{P}_{J'}(t_f)\right|_1 = \left|\mathbf{P}(t_f)\right|_1 = 1$ and the LHS can be rewritten:

$$LHS = 1 - \left|\mathbf{P}_J^{FSP}(t_f)\right|_1.$$

Because the pair $\{G(t_f), \mathbf{P}_J^{FSP}(t_f)\}$ are a probability distribution for $\mathcal{M}_J$, one can see that the right hand side is precisely equal to $\left|G(t_f)\right|_1$ and the proof is complete. □

Lemma 5.0.1 and Theorem 5.0.2, which will hereafter be referred to as the Finite State Projection theorems, tell us two very important pieces of information. First, Lemma 5.0.1 shows that as we increase the size of the finite projection space, the approximation result monotonically increases. Second, Theorem 5.0.2 guarantees that the approximate solution never exceeds the actual solution and gives us certificate of how close the approximation is to the true solution. The interested reader should note that these theorems and their respective proofs apply *with no modification* to a far wider class of problems than the simple time-invariant, discrete state Markov processes considered here. In fact, the FSP Theorems apply to any time varying stochastic process.

5.1 Basic FSP Algorithm

Together, the two FSP theorems above suggest a systematic procedure to to evaluate stochastic processes such as those described by the Chemical Master Equation. Basically, this procedure works by examining a sequence of finite projections of the CME. For each projection set, one can obtain an accuracy guarantee using Theorem 5.0.2. If this accuracy is insufficient, more configurations can be added to the projection set, thereby monotonically improving the accuracy as guaranteed by Lemma 5.0.1. The Finite State Projection algorithm, can be stated as follows:

The Finite State Projection Algorithm

Step 0 Define the propensity functions and stoichiometry for all reactions.
Choose the initial probability distribution, $\mathbf{P}(0)$.
Choose the final time of interest, t_f.
Specify the total amount of acceptable error, $\varepsilon > 0$.
Choose an initial finite set of states, $\mathbf{X}_{J_o}$, for the FSP.
Initialize a counter, $i = 0$.

Step 1 Use propensity functions and stoichiometry to form $\mathbf{A}_{J_i}$.
Compute $\Gamma_{J_i} = |\exp(\mathbf{A}_{J_i} t_f)\mathbf{P}_{J_i}(0)|_1$.

Step 2 If $\Gamma_{J_i} \geq 1 - \varepsilon$, **Stop**.
$\exp(\mathbf{A}_{J_i} t_f)\mathbf{P}_{J_i}(0)$ approximates $\mathbf{P}_{J_i}(t_f)$ to within a total error of ε.

Step 3 Add more states to find $\mathbf{X}_{J_{i+1}}$.
Increment i and return to **Step 1**.

In Steps 0 and 3 of the above algorithm, the choice of how to initialize the set of states for the finite state projection and the approach to adding new states to the FSP has not been explicitly stated. While Lemma 5.0.1 guarantees that adding new states can only improve the accuracy of the approximate solution, it does not state which additions are most beneficial. In practice there may be many methods of choosing how to add states to the projection, and the efficiency of each method may depend upon the class of problem. In general, the best methods will utilize knowledge of the stoichiometry of the chemical reactions and avoid including unreachable states. The following sections illustrate a few such methods to initialize and expand the FSP.

5.2 Initializing $\mathbf{X}_{J_0}$

In the zeroth step of the FSP algorithm, the initial projection set $\mathbf{X}_{J_0}$ can be an arbitrarily chosen set of configurations reachable from the initial condition. The most obvious choice is simply to choose $\mathbf{X}_{J_0}$ to contain only the initial configuration: $\mathbf{X}_{J_0} = \{\mathbf{x}(0)\}$. Instead of choosing $\mathbf{X}_{J_0}$ offline or arbitrarily, it is better to run the SSA [35] a few times and record every configuration reached in those simulations. The set of states reached in those simulations can then be used as the initial projection configuration space, $\mathbf{X}_{J_0}$. If one uses more SSA runs, $\mathbf{X}_{J_0}$ will likely be larger and therefore retain a larger measure of the probability distribution in the specified time interval. Therefore, fewer iterations should be necessary until the FSP algorithm converges.

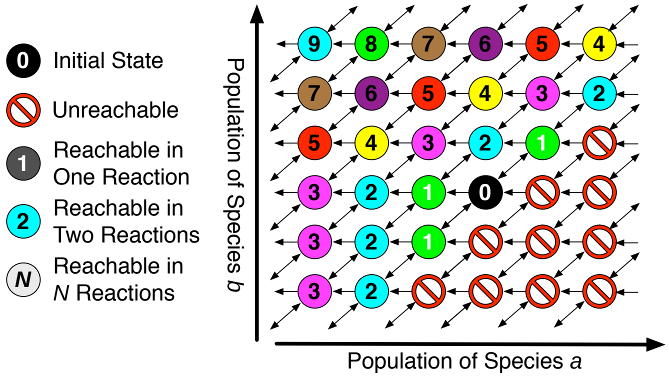


Figure 5.2: Schematic of a two dimensional integer lattice representing the infinite states of a discrete valued Markov process. Each integer valued state vector [a,b] is represented by a circle and the directionality of transitions between states are shown by the connecting arrows.

5.3 Expansion through N-step Reachability

In order to properly introduce the process of expanding the state space through the idea of reachability, it helps to introduce some additional concepts. Consider the generic two-dimensional infinite state space lattice shown in Figure 5.2. In general any chemically reacting system can be represented by an N-dimensional integer lattice, where N is the number of reacting species, and where every node on the lattice is unique and can be enumerated. In Figure 5.2, each circle represents a specific population vector $x^T = [a, b]$, and the initial condition is shaded in black. Reactions are shown with arrows connecting the states. For this specific system, the diagonal oriented reactions are reversible, or bidirectional, while the horizontal reactions are irreversible.

Let I_k denote the set of all states that can be reached from the initial condition in k or fewer chemical reactions. For instance, in Figure 5.2, I_o consists of only the initial condition, which is labeled with the number zero. Similarly, I_1 includes the initial condition and all the green nodes containing the number 1. In general

I_k contains all states in I_{k-1} combined with all states that can be reached via a single reaction beginning in I_{k-1}. Consider any finite set of states, I_R, which are reachable from the initial set, I_0. It is not difficult to see that there will always exist a finite integer, k_R, such that $I_k \supseteq I_R$ for all $k \geq k_R$. For this method of including sequentially reachable states, the following result guarantees that if a finite state projection exists that satisfies the stopping criterion, then the FSP algorithm will converge in a finite number of steps.

Proposition 5.3.1. *Suppose that there exists a finite set of states indexed by S for which the FSP meets the stopping criterion:*

$$\left|\exp(\boldsymbol{A}_S t_f)\boldsymbol{P}_S(0)\right|_1 \geq 1 - \varepsilon. \tag{5.3.1}$$

Then there exists a number of reactions, m, such that the set of reachable states, I_k, also satisfies 5.3.1 for all $k \geq m$.

Proof. The finite set, S can be separated into the reachable subset, R, and the unreachable subset, U. Without loss of generality, the state reaction matrix, $\mathbf{A}_S$ can be written as:

$$\mathbf{A}_S = \begin{bmatrix} \mathbf{A}_R & \mathbf{B} \\ \mathbf{C} & \mathbf{A}_U \end{bmatrix},$$

and the initial condition, which must be contained in the reachable space, can be written as:

$$\mathbf{P}_S(0) = \begin{bmatrix} \mathbf{P}_R(0) \\ \mathbf{P}_U(0) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_R(0) \\ \mathbf{0} \end{bmatrix}.$$

Since the states in U are unreachable from the states in R, the matrix $\mathbf{C}$ is zero.

Through series expansion, the exponential of $(\mathbf{A}_S t_f)$ can be written as:

$$\exp(\mathbf{A}_S t_f) = \left(\begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} + \begin{bmatrix} \mathbf{A}_R & B \\ \mathbf{0} & \mathbf{A}_U \end{bmatrix} t_f + \frac{1}{2} \begin{bmatrix} \mathbf{A}_R^2 & \mathbf{A}_R B + B\mathbf{A}_U \\ \mathbf{0} & \mathbf{A}_U^2 \end{bmatrix} t_f^2 + \dots \right).$$

Combining terms allows one to write the matrix exponential as:

$$\exp(\mathbf{A}_S t_f) = \begin{bmatrix} \exp(\mathbf{A}_R t) & Q \\ \mathbf{0} & \exp(\mathbf{A}_U t) \end{bmatrix},$$

where Q is a positive matrix. Substituting this expression into Eqn 5.3.1 gives:

$$1 - \varepsilon \leq \left|\exp(\mathbf{A}_S t_f)\mathbf{P}_S(0)\right|_1 = \left|\exp(\mathbf{A}_R t_f)\mathbf{P}_R(0)\right|_1 . \qquad (5.3.2)$$

Choose m large enough such that $J_m \supseteq R$, then the set indexed by J_k satisfies Eqn 5.3.1 for all $k \geq m$, completing the proof. □

Proposition 5.3.1 requires that there exists a finite set of states in which the system remains (with probability $1 - \varepsilon$) for the entire time interval, $t \in (0, t_f)$. If this assumption is satisfied, then the N-step FSP algorithm will produce an acceptable approximation within a finite number of steps. If the population of the system is bounded (i.e. by conservation of mass or volume), then such a set will obviously exist. However, one can construct some pathological examples, where the population becomes unbounded for some $t \in (0, t_f)$ (with probability greater than ε). For such examples, the FSP will fail to find a sufficient approximation to the entire probability density vector. Such pathological examples cannot exist in biology, but if such an example did exist, all other methods (SSA, τ leaping and others) would similarly fail.

5.4 Expansion through Probability Tracking

In the previous subsection, the FSP expansion was performed using the concept of $N-$step reachability, where each set $\{\mathbf{X}_{J_N}\}$ included all configurations that are reachable from $\mathbf{X}_{J_0}$ in N reactions or fewer. Proposition 5.3.1 guarantees that if there exists a sufficiently accurate FSP solution, then the FSP algorithm with such an expansion routine is guaranteed to converge in a finite number of steps. This section documents an improved version of the N-step reachability routine. In the original FSP approach all configurations outside the set $\mathbf{X}_J$ have been projected to a single point. Many alternative projections are possible. In particular, one can choose M absorbing points $\{G_1, \ldots, G_M\}$ where each $G_\mu(t)$ corresponds to the probability that the system has left the set $\mathbf{X}_J = \{\mathbf{x}_{j_1}, \mathbf{x}_{j_2}, \ldots\}$ via a μ^{th} reaction. Fig. 5.1c illustrates such a projection choice. For this choice, one arrives at a new master equation:

$$\begin{bmatrix} \dot{\mathbf{P}}_J^{FSP}(t) \\ \dot{\mathbf{G}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{0} \\ \mathbf{Q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ \mathbf{G}(t) \end{bmatrix}, \tag{5.4.1}$$

where $\mathbf{G} = [G_1, \ldots, G_M]^T$ and the matrix $\mathbf{Q}$ is given by

$$\mathbf{Q}_{\mu k} = \left\{ \begin{array}{cc} a_\mu(\mathbf{x}_{j_k}) & \text{if } (\mathbf{x}_{j_k} + \nu_\mu) \notin \mathbf{X}_J \\ 0 & \text{Otherwise} \end{array} \right\}.$$

The solution of (5.4.1) at a time t_f has the form

$$\begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ \mathbf{G}(t) \end{bmatrix} = \begin{bmatrix} \exp(\mathbf{A}_J t_f) & \mathbf{0} \\ \int_0^{t_f} \mathbf{Q} \exp(\mathbf{A}_J \tau) d\tau & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(0) \\ \mathbf{G}(0) \end{bmatrix}, \tag{5.4.2}$$

and yields all of the same information as (5.0.2), but it now provides additional useful knowledge. Specifically, each column of the operator in (5.4.2) corresponds to a specific $\mathbf{x}_i \in \mathbf{X}_J$. Each of the last M elements of the column corresponding to $\mathbf{x}_i$ gives the exact probability that a trajectory beginning in $\mathbf{x}_i$ at time $t = 0$ will have exited the full set $\mathbf{X}_J$ via a specific reaction channel before the time $t = t_f$. This knowledge is easily incorporated into Step 3 of the above algorithm. If most of the probability measure left via one particular reaction, it is reasonable to expand $\mathbf{X}_J$ in the corresponding direction. Conversely, if very little of the probability measure leaks out via a given reaction, it would be useless to expand the projection in that direction.

For the basic FSP algorithm with these or any other expansion routine, if one wishes to find a solution that is accurate to within ε at a time t_f, he or she must find a finite set of configurations such that the probability of ever leaving that set during the time interval $[0, t_f]$ is less than ε. For many problems, including the examples shown in [67, 66], this set of configurations may be small enough that one can easily compute a single matrix exponential to approximate the solution to the CME. However, in other situations the configuration space required for a one matrix solution may be exorbitantly large. The following chapters present a number of means in which the FSP can be extended to handle much more involved systems. Chapter 6 uses concepts of observability and reachability to reduce the system to its minimal realization; Chapter 7 reduces the system of ODEs using time scale separation based approximations. In Chapters 8 and 9, the FSP is improved to exploit properties of superposition and time invariance, respectively. In Chapter 10, the FSP problem is reduced by making an assumption that the distribution of the system on the full integer lattice can be interpolated from among points of a much coarser lattice.

Chapter 6

Minimal Realizations for the FSP Method

The finite state projection works by providing a bulk reduction to the chemical master equation. This reduction transforms an infinite dimensional system into a finite dimensional system, but often the reduced system remains excessively high in its dimensionality. In these cases further reduction are required. The first such reduction to be considered in this chapter is based upon well established results from linear system's theory, particularly the concepts of controllability and observability.

6.1 Aggregation of Unobservable Configurations (OA-FSP)

Consider a master equation, $\dot{\mathbf{P}}(t) = \mathbf{AP}(t)$, for which the initial probability density vector (pdv) is supported only on the set indexed by U; in other words $p_i(0) = 0$ for all $i \notin U$. For this system, the initial value problem is equivalent to the impulse response problem:

$$\dot{\mathbf{P}}(t) = \mathbf{AP}(t) + \mathbf{b}\delta(t),$$

where $\mathbf{b} = \mathbf{P}(0)$.

Suppose that one wishes only to compute the statistical quantity $\mathbf{y}(t) = \mathbf{CP}(t)$. As one example, if one were interested in estimating the mean or variance of the population of the m^{th} molecular species, then $\mathbf{C}$ would simply be the row vector

$$\mathbf{C}_{mean} = [\ \mathbf{x}_{1m}, \quad \mathbf{x}_{2m}, \quad \ldots\ ],$$

or

$$\mathbf{C}_{var} = [\ (\mathbf{x}_{1m}^2 - \mathbf{x}_{1m}), \quad (\mathbf{x}_{2m}^2 - \mathbf{x}_{2m}), \quad \ldots\ ],$$

respectively, where $\mathbf{x}_{im}$ is the m^{th} component of the integer vector $\mathbf{x}_i$. Alternatively, as in the next subsection, one may choose the output to correspond to the probability density on a portion of the configuration set. For any $\mathbf{C}$, the resulting problem now takes on a familiar form:

$$\begin{aligned} \dot{\mathbf{P}}(t) &= \mathbf{AP}(t) + \mathbf{b}\delta(t); \\ \mathbf{y} &= \mathbf{CP}(t). \end{aligned} \tag{6.1.1}$$

For systems on a finite configuration set, or for systems that have been projected onto a finite configuration set, this standard representation is open to a host of computational tools already available for linear time invariant systems (for examples, see [4]). These tools have been developed for arbitrary control inputs but work reasonable well for the analysis of master equation ODEs in which the input is a single unit impulse at the initial time. Additional model reductions may be possible to take advantage of this more specific formulation. In cases where one wishes to analyze the system with many different initial conditions, one can replace $\mathbf{b}$ with a matrix $\mathbf{B}$ whose columns span all of the various initial conditions.

The following subsection illustrates how one may use concepts closely related to observability and reachability to easily improve upon the efficiency of the FSP. Later, Subsection 11.2.1 will also illustrate how Hankel norm based balanced truncation can be used to reduce the order of the FSP analysis.

6.1.1 Estimating the probability of important states

In many cases, one is not interested in the probabilities of every possible configuration, but instead one wishes only to know the probabilities of certain important configurations. Suppose that the system begins with the known population vector, $\mathbf{x}_u$, and we want only the probability distribution on the configuration subset $\mathbf{X}_K = \{\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \ldots\}$. In other words, we wish to compute $\mathbf{y}(t) = \mathbf{P}_K(t)$. For example, $\mathbf{X}_K$ may correspond to configurations that exhibit a specific biological trait, such as the expression of a certain gene. As above, define the vector $\mathbf{b} = \mathbf{P}(0) = \{\mathbf{b}_i\}_{i=1}^{\infty}$. In this case $\mathbf{b}_i = 1$ for $i = u$ and zero otherwise. For this $\mathbf{b}$ and the impulse response in (6.1.1), let $\mathbf{X}_R$ be the subset of all configuration points $\mathbf{x}_i$ such that $p_i(t) > 0$ at any $t \geq 0$. This subset is indexed by R to denote that it is the *reachable configuration subset*; its complement $\mathbf{X}_{R'}$ is the unreachable configuration subset. Define the *observable configuration subset*, $\mathbf{X}_O$, as the set of all $\mathbf{x}_i$ such that $p_i(t_0) > 0$ at time t_0 guarantees that $|\mathbf{y}| > \mathbf{0}$ at some $t \geq t_0$. We will call the complement, $\mathbf{X}_{O'}$, the *unobservable* configuration subset. Note that our definitions of reachability and observability are slightly less restrictive than the traditional usage. While using the usual concepts of observability and reachability would often allow bigger reductions in the order of the problem (See Section 11.2.1), it is often much easier–and less computationally intensive–to categorize the system as shown here.

Now that the configuration set has been decomposed into subsets, we can introduce the following theorem:

Theorem 6.1.1. *Consider a process whose distribution evolves according to the linear ODE:*

$$\begin{bmatrix} \dot{\boldsymbol{P}}_{I_1}(t) \\ \dot{\boldsymbol{P}}_{I_2}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_{I_1} & \boldsymbol{0} \\ \boldsymbol{A}_{I_2 I_1} & \boldsymbol{A}_{I_2} \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{I_1}(t) \\ \boldsymbol{P}_{I_2}(t) \end{bmatrix}, \tag{6.1.2}$$

where I_1 and I_2 are disjoint index sets.

If for some finite index set $J \subseteq I_1$, $\varepsilon > 0$, and $t_f \geq 0$,

$$\mathbf{1}^T exp \begin{bmatrix} \boldsymbol{A}_J t_f & 0 \\ \mathbf{1}^T \boldsymbol{A}_{I_2 J} t_f & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_J(0) \\ \mathbf{1}^T \boldsymbol{P}_{I_2}(0) \end{bmatrix} \geq 1 - \varepsilon, \tag{6.1.3}$$

then

$$\exp(\boldsymbol{A}_J t_f) \boldsymbol{P}_J(0) \leq \boldsymbol{P}_J(t_f), \text{ and} \tag{6.1.4}$$

$$\left| \boldsymbol{P}_J(t_f) - \exp(\boldsymbol{A}_J t_f) \boldsymbol{P}_J(0) \right|_1 \leq \varepsilon. \tag{6.1.5}$$

Proof. We begin by proving (6.1.4). Let J' denote the complement of J on the set I_1. The evolution of the full probability density vector is governed by the permuted ODE:

$$\begin{bmatrix} \dot{\mathbf{P}}_J \\ \dot{\mathbf{P}}_{J'} \\ \dot{\mathbf{P}}_{I_2} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{A}_{JJ'} & \mathbf{0} \\ \mathbf{A}_{J'J} & \mathbf{A}_{J'} & \mathbf{0} \\ \mathbf{A}_{I_2J} & \mathbf{A}_{I_2J'} & \mathbf{A}_{I_2} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J \\ \mathbf{P}_{J'} \\ \mathbf{P}_{I_2} \end{bmatrix}, \tag{6.1.6}$$

where the submatrices $\mathbf{A}_{JJ'}$ and $\mathbf{A}_{I_2J'}$ are nonnegative since $\mathbf{A}$ has no negative off-diagonal terms. We now sum all of the rows corresponding to the set I_2:

$$\begin{bmatrix} \dot{\mathbf{P}}_J \\ \dot{\mathbf{P}}_{J'} \\ \mathbf{1}^T\dot{\mathbf{P}}_{I_2} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{A}_{JJ'} & \mathbf{0} \\ \mathbf{A}_{J'J} & \mathbf{A}_{J'} & \mathbf{0} \\ \mathbf{1}^T\mathbf{A}_{I_2J} & \mathbf{1}^T\mathbf{A}_{I_2J'} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J \\ \mathbf{P}_{J'} \\ \mathbf{P}_{I_2} \end{bmatrix}, \tag{6.1.7}$$

where we have used the fact that all columns of $\mathbf{A}$, particularly those indexed by I_2, sum to zero: $\mathbf{1}^T\mathbf{A}_{I_2} = \mathbf{0}$.

Let $p_{agg} := \mathbf{1}^T\mathbf{P}_{I_2}$. The aggregated probability density is now governed by the finite linear ODE:

$$\begin{bmatrix} \dot{\mathbf{P}}_J \\ \dot{p}_{agg} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{0} \\ \mathbf{1}^T\mathbf{A}_{I_2J} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J \\ \mathbf{P}_{I_2} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_{JJ'} \\ \mathbf{1}^T\mathbf{A}_{I_2J'} \end{bmatrix} \mathbf{P}_{J'}.$$

The solution of this forced ODE is

$$\begin{bmatrix} \mathbf{P}_J(t_f) \\ p_{agg}(t_f) \end{bmatrix} = \exp \begin{bmatrix} \mathbf{A}_J t_f & \mathbf{0} \\ \mathbf{1}^T\mathbf{A}_{I_2J} t_f & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J(0) \\ p_{agg}(0) \end{bmatrix} + \\ \int_0^{t_f} \exp \begin{bmatrix} \mathbf{A}_J(t_f-\tau) & \mathbf{0} \\ \mathbf{1}^T\mathbf{A}_{I_2J}(t_f-\tau) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{JJ'} \\ \mathbf{1}^T\mathbf{A}_{I_2J'} \end{bmatrix} \mathbf{P}_{J'}(\tau) d\tau.$$

Since $\mathbf{A}_{JJ'}$, $\mathbf{A}_{I_2J'}$, $\mathbf{P}_{J'}(t)$, and $\exp\begin{bmatrix} \mathbf{A}_J t & \mathbf{0} \\ \mathbf{1}^T\mathbf{A}_{I_2J} t & \mathbf{0} \end{bmatrix}$ are all nonnegative for $t \geq 0$, we obtain the inequality in (6.1.4) as the top part of

$$\begin{bmatrix} \mathbf{P}_J(t_f) \\ p_{agg}(t_f) \end{bmatrix} \geq \exp\begin{bmatrix} \mathbf{A}_J t_f & \mathbf{0} \\ \mathbf{1}^T\mathbf{A}_{I_2J} t_f & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J(0) \\ p_{agg}(0) \end{bmatrix}. \tag{6.1.8}$$

Using (6.1.2) and the fact that the probability distribution the distribution on the J and I_2-indexed sets must be non-negative and have a combined sum of no more than one we get:

$$\left| \exp\begin{bmatrix} \mathbf{A}_J t_f & 0 \\ \mathbf{1}^T\mathbf{A}_{I_2J} t_f & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_J(0) \\ \mathbf{1}^T\mathbf{P}_{I_2}(0) \end{bmatrix} \right|_1 \geq \left\| \begin{bmatrix} \mathbf{P}_J(t_f) \\ p_{agg}(t_f) \end{bmatrix} \right\|_1 - \varepsilon, \tag{6.1.9}$$

Finally, applying (6.1.8) and rearranging terms yields:

$$\left\| \begin{bmatrix} \mathbf{P}_J(t_f) \\ p_{agg}(t_f) \end{bmatrix} - \exp\begin{bmatrix} \mathbf{A}_J t_f & 0 \\ \mathbf{1}^T\mathbf{A}_{I_2J} t_f & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_J(0) \\ \mathbf{1}^T\mathbf{P}_{I_2}(0) \end{bmatrix} \right\|_1 \varepsilon, \tag{6.1.10}$$

and completes the proof. □

By our definition of reachable, the probability density vector on the configuration subset $\mathbf{X}_{R'}$ is zero, and a permutation can reorder the remaining rows of (2.0.4) as:

$$\begin{bmatrix} \dot{\mathbf{P}}_{RO} \\ \dot{\mathbf{P}}_{RO'} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{RO} & \mathbf{A}_{RORO'} \\ \mathbf{A}_{RO'RO} & \mathbf{A}_{RO'} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{RO} \\ \mathbf{P}_{RO'} \end{bmatrix}, \tag{6.1.11}$$

where $RO := R \cap O$ indexes the reachable/observable configuration subset, and $RO' := R \cap O'$ indexes the reachable/unobservable configuration subset. Also by definition, no configuration in $\mathbf{X}'_O$ can transition into the configuration subset $\mathbf{X}_O$, which results in the identity: $\mathbf{A}_{RORO'} = \mathbf{0}$, and the system reduces to:

$$\begin{bmatrix} \dot{\mathbf{P}}_{RO} \\ \dot{\mathbf{P}}_{RO'} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{RO} & \mathbf{0} \\ \mathbf{A}_{RO'RO} & \mathbf{A}_{RO'} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{RO} \\ \mathbf{P}_{RO'} \end{bmatrix}. \tag{6.1.12}$$

Applying Theorem 3.1 yields the following corollary:

Corollary 3.2. Consider any Markov process in which the probability density state vector evolves according to (6.1.12). Let J be a finite subset of the index set RO. If for $\varepsilon > 0$, and $t_f \geq 0$

$$\mathbf{1}^T \exp \begin{bmatrix} \mathbf{A}_J t_f & 0 \\ \mathbf{1}^T \mathbf{A}_{RO'J} t_f & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_J(0) \\ \mathbf{1}^T \mathbf{P}_{RO'}(0) \end{bmatrix} \geq 1 - \varepsilon, \tag{6.1.13}$$

then

$$\exp(\mathbf{A}_J t_f) \mathbf{P}_J(0) \leq \mathbf{P}_J(t_f), \text{ and} \tag{6.1.14}$$

$$\left| \mathbf{P}_J(t_f) - \exp(\mathbf{A}_J t_f) \mathbf{P}_J(0) \right|_1 \leq \varepsilon. \tag{6.1.15}$$

The proof of Corollary 3.2 follows directly from Theorem 3.1 where $I_1 = RO$ and $I_2 = RO'$. To illustrate the underlying intuition of Corollary 3.2, Fig. 6.1(top) illustrates a two dimensional state lattice for a two chemical reacting system. The system begins with an initial configuration $\mathbf{x}_u$ at time $t = 0$, and we are interested in calculating the probability that the system has configuration, $\mathbf{x}_y$, at the time $t = t_f \geq 0$. The configuration set can be separated into three disjoint subsets: the unreachable region, $\mathbf{X}_{R'}$; the unobservable region, $\mathbf{X}_{O'}$; and the reachable/observable region $\mathbf{X}_{RO}$. Using the OAFSP, we remove the $\mathbf{X}_{R'}$ from the system and aggregate $\mathbf{X}_{O'}$ to a single point, as shown in Fig. 6.1(bottom left). We then project $\mathbf{X}_{RO}$ onto a finite configuration subset $\mathbf{X}_J$. The projected system is shown in Fig. 6.1(bottom right), where the subsets $\mathbf{X}_{J'}$ and $\mathbf{X}_{RO'}$ have each been aggregated to a single point. Because the projected system is finite dimensional, its solution can be computed using the matrix exponential function or by using a standard ODE solver. Theorem 2.1 shows that as the subset $\mathbf{X}_J$ increases, fewer trajectories are lost to $\mathbf{X}_{J'}$ and the probability of remaining in $\mathbf{X}_J \cup \mathbf{X}_{RO'}$ increases. Corollary 3.2 shows that the probability that the system is currently in $\mathbf{X}_J \cup \mathbf{X}_{RO'}$ must be at least as large as the probability that the system has been in $\mathbf{X}_J \cup \mathbf{X}_{RO'}$ for all times $t = 0$ to $t = t_f$.

The OA-FSP Algorithm

The results above and our previous work on the FSP [67] suggest a systematic procedure for solving the chemical kinetic problem as posed in (6.1.1). This algorithm, which we refer to as the Observability Aggregated FSP algorithm, can be stated as follows:

```
Step 0 Define reaction propensities and stoichiometry.
       Choose the initial pdv, $\mathbf{P}(0)$.
       Choose the final time of interest, $t_f$.
       Specify the total acceptable error, $\varepsilon > 0$.
       Define configuration subsets: $\mathbf{X}_{RO}$ and $\mathbf{X}_{RO'}$.
       Choose initial finite index set, $J_o \subseteq RO$.
       Initialize a counter, $i = 0$.
Step 1 Use propensities and stoichiometry to compute
       $\Gamma_{J_i} = \mathbf{1}^T \exp \begin{bmatrix} \mathbf{A}_J t_f & \mathbf{0} \\ \mathbf{1}^T \mathbf{A}_{RO'J} t_f & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_{J_i}(0) \\ \mathbf{1}^T \mathbf{P}_{RO'}(0) \end{bmatrix}.$
Step 2 If $\Gamma_{J_i} \geq 1 - \varepsilon$, Stop.
       $\exp(\mathbf{A}_{J_i} t_f) \mathbf{P}_{J_i}(0)$ is within $\varepsilon \mathbf{1}$ error from $\mathbf{P}_{J_i}(t_f)$.
Step 3 Add more configurations to find $\mathbf{X}_{J_{i+1}}$.
       Increment $i$ and return to Step 1.
```

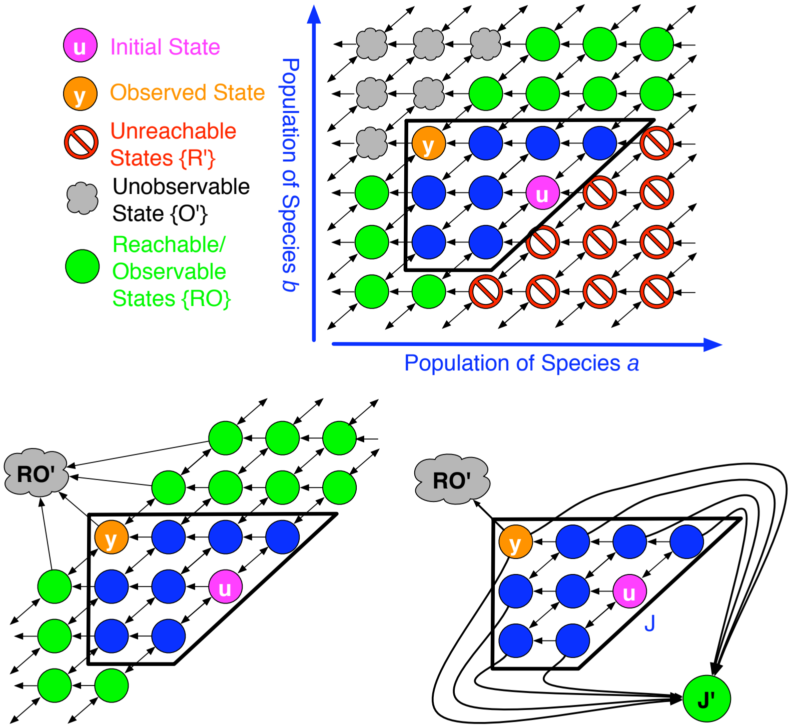


Figure 6.1: Top: schematic of a two dimensional integer lattice representing the configuration set of a two species chemical reaction. Each configuration point [a,b] is represented by a circle and transitions (reactions) are shown by the connecting arrows. Bottom: aggregation of the unobservable configuration subset (left), and projection of the observable/reachable configuration subset onto a finite configuration subset: $\mathbf{X}_J \in \mathbf{X}_{RO}$ (right).

Chapter 7

The Slow Manifold FSP (SM-FSP)

In many biological models, certain reactions occur much faster and more frequently than others. For KMC algorithms like the SSA, the majority of the simulated reactions correspond to those with large propensities. In the case of the CME or its projection, this separation of time scales results in numerical stiffness. As discussed in Section 3.1.1, there has been significant progress in developing approximate KMC algorithms to deal with these concerns. In these, the fast dynamics are essentially averaged, and the slow dynamics are simulated assuming the fast dynamics have instantaneously reached thermal equilibrium. We have shown that the FSP algorithm is also amenable to time-partitioning approximation schemes that speed up computation at a small cost to the accuracy [78, 71]. In those papers, the time scale separation is carried out using a singular perturbation approach similar to that in [32]. In the control community, perturbation methods have also had long use as described in [55]. This chapter takes a linear systems theory approach to such problems.

In the configuration space, some subsets of configuration points are often interconnected[1] by fast reactions and separated from each other by slow reactions. One such example is the 4-configuration Markov process illustrated in Fig. 7.1(a), where the fast reactions (solid lines) have propensities equal to one, and the slow reactions (dashed lines) have propensities equal to ε. The master equation for this particular process has the generator

$$\mathbf{A} = \begin{bmatrix} -r-\varepsilon & r & \varepsilon & 0 \\ r & -r-\varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & -r-\varepsilon & r \\ 0 & \varepsilon & r & -r-\varepsilon \end{bmatrix}.$$

If one groups together the fast interconnected configurations (possibly requiring a permutation of the configuration set), one can separate the system into fast and slow parts: $\mathbf{A} = \mathbf{H} + \varepsilon\mathbf{G}$, where $\mathbf{H}$ is block diagonal with each block representing a fast interconnected configuration set. For the schematic in Fig. 7.1 this separation gives

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_2 \end{bmatrix} = \begin{bmatrix} -r & r & 0 & 0 \\ r & -r & 0 & 0 \\ 0 & 0 & -r & r \\ 0 & 0 & r & -r \end{bmatrix},$$

[1] Here the term "interconnected" is used to mean that the configurations form a non-separable Markov process. Any finite dimensional interconnected system can readily be shown to have a simple eigenvalue at zero.

and

$$\varepsilon\mathbf{G} = \begin{bmatrix} -\varepsilon & 0 & \varepsilon & 0 \\ 0 & -\varepsilon & 0 & \varepsilon \\ \varepsilon & 0 & -\varepsilon & 0 \\ 0 & \varepsilon & 0 & -\varepsilon \end{bmatrix}.$$

It is easily seen that each $\mathbf{H}_i$ is the generator matrix for the i^{th} fast cluster, and $\varepsilon\mathbf{G}$ is the generator matrix of the reactions that take the system from one cluster to another.

For an N dimensional finite state projection with m fast interconnected configuration sets, the master equation can be written

$$\dot{\mathbf{P}}(t) = (\mathbf{H} + \varepsilon\mathbf{G})\mathbf{P}(t), \tag{7.0.1}$$

where $\mathbf{H}$ can be written $\mathbf{H} = \text{diag}\{\mathbf{H}_1, \mathbf{H}_2, \ldots, \mathbf{H}_m\}$. Because they are generators, each $\mathbf{H}_i$ has a single eigenvalue equal to zero, and its corresponding left and right eigenvectors are $\mathbf{u}_i = \mathbf{1}^T$ and $\mathbf{v}_i$, respectively. We define the following matrices.

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & 0 & \ldots \\ 0 & \mathbf{u}_2 & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix}, \text{ and } \mathbf{V} = \begin{bmatrix} \mathbf{v}_1 & 0 & \ldots \\ 0 & \mathbf{v}_2 & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix}.$$

Let $\mathbf{S} = [\ \mathbf{V} \quad \mathbf{R}\ ]$ be a square matrix in which the columns of $\mathbf{R}$ are the remaining $N - m$ right eigenvectors of $\mathbf{H}$. The inverse of $\mathbf{S}$ is given by $\mathbf{S}^{-1} = \begin{bmatrix} \mathbf{U}^T & \mathbf{L}^T \end{bmatrix}^T$

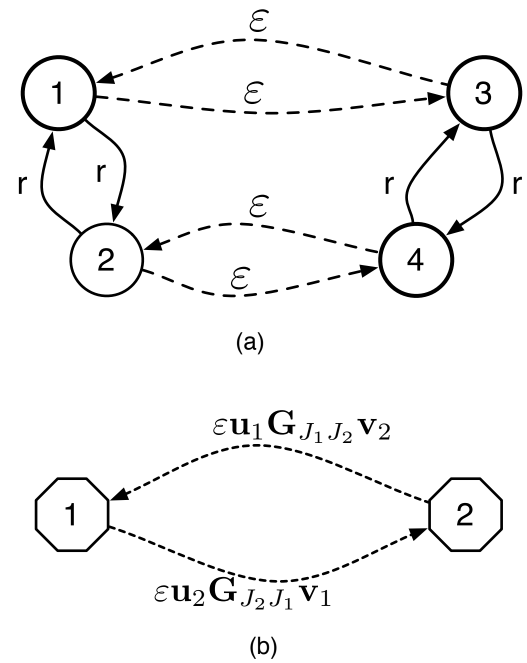

ε
1
3
ε
r
r
r
r
ε
2
4
ε
(a)
$\varepsilon\mathbf{u}_1\mathbf{G}_{J_1J_2}\mathbf{v}_2$
1
2
$\varepsilon\mathbf{u}_2\mathbf{G}_{J_2J_1}\mathbf{v}_1$
(b)

Figure 7.1: (a) A four configuration Markov process that undergoes reactions on two different time scales. For $r \gg \varepsilon$, the fast reactions are represented by solid lines, and slow reactions are represented by dashed lines. There are two sets of strongly connected configurations: $J_1 = \{1, 2\}$ and $J_2 = \{3, 4\}$. The fast reactions form two infinitesimal generators, $\mathbf{H}_1$ and $\mathbf{H}_2$, and the slow reactions form a single generator for the whole system, $\varepsilon\mathbf{G}$. (b) In the reduced model, each fast interconnected set becomes a single configuration. The strength of the slow reactions from the i^{th} to the j^{th} set is given by $\varepsilon\mathbf{u}_j\mathbf{G}_{J_jJ_i}\mathbf{v}_i$, where $\mathbf{u}_i$ and $\mathbf{v}_i$ are the left and right zero-eigenvectors of $\mathbf{H}_i$, and $\mathbf{G}_{J_jJ_i}$ is the sub-matrix of $\mathbf{G}$ with columns corresponding to J_i and rows corresponding to J_j.

such that we have the following similarity transformation for $\mathbf{H}$:

$$\mathbf{S}^{-1}\mathbf{H}\mathbf{S} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda} \end{bmatrix}, \Lambda = \operatorname{diag}(\lambda_{m+1}, \ldots, \lambda_N).$$

where the first m diagonal elements correspond to the zero eigenvalues of the $\mathbf{H}_i$ blocks. With out loss of generality, the non-zero eigenvalues of $\mathbf{H}$ can be ordered so that $0 > \operatorname{Re}\{\lambda_{m+1}\} \geq \operatorname{Re}\{\lambda_{m+2}\}, \ldots \geq \operatorname{Re}\{\lambda_N\}$. Applying the coordinate transformation $\begin{bmatrix} \mathbf{y}_1^T(t) & \mathbf{y}_2^T(t) \end{bmatrix}^T = \mathbf{S}^{-1}\mathbf{P}(t)$, (7.0.1) becomes:

$$\begin{bmatrix} \dot{\mathbf{y}}_1(t) \\ \dot{\mathbf{y}}_2(t) \end{bmatrix} = \begin{bmatrix} \varepsilon\mathbf{UGV} & \varepsilon\mathbf{UGR} \\ \varepsilon\mathbf{LGV} & \mathbf{Q} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1(t) \\ \mathbf{y}_2(t) \end{bmatrix}, \tag{7.0.2}$$

where the matrix $\mathbf{Q} = \mathbf{\Lambda} + \varepsilon\mathbf{LGR}$.

There are two important observations to make regarding this transformed system. First, the matrix $\mathbf{UGV}$ is itself a generator for a Markov process in that it satisfies the two sufficient conditions: *(i)* its columns sum to zero, and *(ii)* its off-diagonal elements are non-negative. To show that, note that $\mathbf{1}^T\mathbf{U} = \mathbf{1}^T$ and therefore $\mathbf{1}^T\mathbf{UG} = \mathbf{1}^T\mathbf{G} = \mathbf{0}$. Furthermore,

$$[\mathbf{UGV}]_{ij} = \mathbf{u}_i\mathbf{G}_{J_iJ_j}\mathbf{v}_j,$$

where $\mathbf{u}_i$ and $\mathbf{v}_j$ are non-negative for any (i, j) and the sub-matrix $\mathbf{G}_{J_iJ_j}$ is non-negative for any $i \neq j$. Hence the off-diagonal elements of $\mathbf{UGV}$ are indeed non-negative. The second observation that one can make is that for $\varepsilon \ll |\operatorname{Re}\{\lambda_{m+1}\}|$, linear perturbation theory assures us that the matrix $\mathbf{Q}$ is Hurwitz, and its eigenvalues are close to $\{\lambda_{m+1}, \lambda_{m+2}, \ldots, \lambda_N\}$. In particular if we let $\tilde{\lambda}$ denote the real part of the least stable eigenvalue of $\mathbf{Q}$, we know that $\tilde{\lambda} \approx \mathrm{Re}\{\lambda_{m+1}\}$.

With these observations in mind, one can now examine the forced dynamics of $\mathbf{y}_2(t)$:

$$\dot{\mathbf{y}}_2(t) = \varepsilon \mathbf{L}\mathbf{G}\mathbf{V}\mathbf{y}_1(t) + \mathbf{Q}\mathbf{y}_2(t),$$

which has a solution comprised of a zero-state and a zero-input response:

$$\mathbf{y}_2(t) = \mathbf{y}_2^{zs}(t) + \mathbf{y}_2^{zi}(t).$$

Because $\mathbf{Q}$ is Hurwitz, with eigenvalues all having real parts less than or equal to $\tilde{\lambda}$, the zero-input response, $\mathbf{y}_2^{zi}(t)$, is bounded by the exponentially decaying expression. Therefore, there exists a constant K_1 such that

$$\left|\mathbf{y}_2^{zi}(t)\right|_1 \leq K_1 \exp(\tilde{\lambda} t), \qquad \forall t \geq 0.$$

By the definition of our transformation

$$|\mathbf{y}_1(t)|_1 = |\mathbf{U}\mathbf{P}(t)|_1 = 1,$$

and $|\mathbf{L}\mathbf{G}\mathbf{V}\mathbf{y}_1(t)|_1$ is bounded. Since $\mathbf{Q}$ is Hurwitz and the input is $O(\varepsilon)$, we are guaranteed that the zero-state solution, $\mathbf{y}_2^{zs}(t)$ satisfies

$$|\mathbf{y}_2^{zs}(t)|_1 = O(\varepsilon), \qquad \forall t \geq 0.$$

Combining the two solutions, we have the following bounds on $\mathbf{y}_2(t)$

$$|\mathbf{y}_2(t)|_1 \leq K_1 \exp(\tilde{\lambda} t) + O(\varepsilon), \tag{7.0.3}$$

for all times $t \geq 0$.

The forced dynamics of $\mathbf{y}_1(t)$ given by the

$$\dot{\mathbf{y}}_1(t) = \varepsilon \mathbf{UGV} \mathbf{y}_1(t) + \varepsilon \mathbf{UGR} \mathbf{y}_2(t),$$

has a solution at the chosen final time t_f:

$$\mathbf{y}_1(t_f) = \exp(\varepsilon \mathbf{UGV} t_f) \mathbf{y}_1(0) + \varepsilon \int_0^{t_f} \exp(\varepsilon \mathbf{UGV}(t_f - \tau)) \mathbf{UGR} \mathbf{y}_2(\tau) d\tau. \quad (7.0.4)$$

Note that since $\mathbf{UGV}$ is a infinitesimal generator of a Markov process, every column of $\exp(\mathbf{UGV}t)$ has a sum of exactly one for any $t \geq 0$, and

$$||\exp(\varepsilon \mathbf{UGV}(t - \tau))||_1 = 1,$$

for all $\varepsilon \geq 0$ and $t \geq \tau$. Therefore

$$|\mathbf{y}_1(t_f) - \exp(\varepsilon \mathbf{UGV} t_f) \mathbf{y}_1(0)|_1 \leq \varepsilon \int_0^{t_f} |\mathbf{UGR} \mathbf{y}_2(\tau)|_1 \, d\tau.$$

Combining this with (7.0.3) and defining the constant $K_2 = K_1 \, ||\mathbf{UGR}||_1$, one obtains the following bound on the error of $\mathbf{y}_1$ at $t = t_f$:

$$\begin{aligned} |\mathbf{y}_1(t_f) - \exp(\varepsilon \mathbf{UGV} t_f) \mathbf{y}_1(0)|_1 &\leq \varepsilon \int_0^{t_f} K_2 \exp(\tilde{\lambda}\tau) + O(\varepsilon) d\tau \\ &\leq \varepsilon K_2 \frac{1}{|\tilde{\lambda}|} + t_f O(\varepsilon^2). \end{aligned}$$

Therefore, for any fixed $t_f \geq 0$,

$$|\mathbf{y}_1(t_f) - \exp(\varepsilon \mathbf{UGV} t_f) \mathbf{y}_1(0)|_1 = O(\varepsilon). \quad (7.0.5)$$

Combining (7.0.3) and (7.0.5) gives the following bounds on the approximation error:

$$\left|\left| \begin{bmatrix} \mathbf{y}_1(t_f) \\ \mathbf{y}_2(t_f) \end{bmatrix} - \begin{bmatrix} \exp(\varepsilon \mathbf{UGV} t_f)\mathbf{y}_1(0) \\ \mathbf{0} \end{bmatrix} \right|\right|_1 \leq K_1 \exp(\tilde{\lambda} t_f) + O(\varepsilon).$$

Substituting the initial condition,

$$\begin{bmatrix} \mathbf{y}_1(0) \\ \mathbf{y}_2(0) \end{bmatrix} = \mathbf{S}^{-1}\mathbf{P}(0) = \begin{bmatrix} \mathbf{UP}(0) \\ \mathbf{LP}(0) \end{bmatrix},$$

and performing the reverse similarity transformation, $\mathbf{P}(t_f) = \mathbf{V}\mathbf{y}_1(t_f) + \mathbf{R}\mathbf{y}_2(t_f)$, yields:

$$|\mathbf{P}(t_f) - \mathbf{V}\exp(\varepsilon \mathbf{UGV} t_f)\mathbf{UP}(0)|_1 \leq K_1 \, ||\mathbf{R}||_1 \, |\mathbf{LP}(0)|_1 \exp(\tilde{\lambda} t_f) + O(\varepsilon).$$

Thus, this reduced model differs from the full system by at most an exponentially decreasing transient term plus a term of order ε.

In the toy example in Fig. 7.1, the blocks $\mathbf{H}_1$ and $\mathbf{H}_2$ were identical, with eigenvalues of zero and $-2r$. The left and right eigenvectors for the zero eigenvalue are $\mathbf{u}_i = \begin{bmatrix} 1 & 1 \end{bmatrix}$ and $\mathbf{v}_i^T = \begin{bmatrix} 1/2 & 1/2 \end{bmatrix}$, respectively. The generator for the reduced system (as shown in Fig. 7.1(b)) is

$$\mathbf{UGV} = \begin{bmatrix} \mathbf{u}_1 \mathbf{G}_{J_1} \mathbf{v}_1 & \mathbf{u}_1 \mathbf{G}_{J_1 J_2} \mathbf{v}_2 \\ \mathbf{u}_2 \mathbf{G}_{J_2 J_1} \mathbf{v}_1 & \mathbf{u}_2 \mathbf{G}_{J_2} \mathbf{v}_2 \end{bmatrix},$$

where the the index set for the first and second blocks are $J_1 = \{1, 2\}$ and $J_2 = \{3, 4\}$, respectively.

Applying this model reduction approach to the original FSP algorithm yields the following algorithm which we name the Slow-Manifold FSP algorithm:

The Slow-Manifold FSP Algorithm

```
Inputs  Propensities and stoichiometries for all reactions.
        Initial probability density vector, $\mathbf{P}(0)$.
        Final time of interest, $t_f$.
        Target FSP error, $\delta > 0$.
Step 0  Choose initial set of states, $\mathbf{X}_{J_o}$, for the FSP.
        Initialize a counter, $k = 0$.
Step 1  Use fast reactions connecting states within $\mathbf{X}_{J_k}$ to
            form $\mathbf{H}_{J_k} = \text{diag}\{\mathbf{H}_1, \ldots, \mathbf{H}_m\}$.
        Use remaining reactions to form $\mathbf{G}_{J_k}$.
Step 2  Find eigenvalues and vectors of each $\mathbf{H}_i$ and
            build matrices $\mathbf{U}$ and $\mathbf{V}$.
        Estimate $\varepsilon = ||\mathbf{G}_{J_k}\mathbf{V}||_1 \,/\, |\lambda_{m+1}|$.
        Compute $\gamma = |\mathbf{S}_L\mathbf{P}(0)|_1 \exp(\lambda_{m+1} t_f)$.
Step 3  Find $\mathbf{P}^{FSP}_{J_k}(t_f) = \mathbf{V}\exp(\mathbf{U}\mathbf{G}_{J_k}\mathbf{V}t_f)\mathbf{U}\mathbf{P}_{J_k}(0)$
            and compute $\Gamma_{J_k} = \mathbf{1}^T\mathbf{P}^{FSP}_{J_k}(t_f)$.
Step 4  If $\Gamma_{J_k} \geq 1 - \delta$, Stop.
        $\mathbf{P}^{FSP}_{J_k}(t_f)$ is within $\delta + \gamma + O(\varepsilon)$ of $\mathbf{P}_{J_k}(t_f)$.
Step 5  Add more states to find $\mathbf{X}_{J_{k+1}}$.
        Increment $k$ and return to Step 1.
```

Here, the non-traditional error estimate notation $\delta + \gamma + O(\varepsilon)$ is used to mean the following. If δ is largest, then the dominant error is most likely the result of the projection, and the slow manifold truncation error can be ignored. If γ is largest then the time t_f is too short for the transient dynamics to sufficiently diminish and additional eigenvectors must be included in the truncation. Finally, if ε is larger than δ and γ, then there is insufficient separation between the slow and fast dynamics and an alternative reduction scheme may be required.

The next section illustrates the slow manifold approach on a simple example and later Chapters 14 applies this method to a toy model of the heat shock response in *E. coli.* For a third example, the reader is encouraged to see [71].

7.1 Simple SM-FSP Example

This section illustrates the Slow Manifold FSP solution technique with a very simple example. The system has two weakly coupled sets of three different configurations each. The master equation for the fast reactions, $\mathbf{H}$, is a block diagonal matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & 0 \\ 0 & \mathbf{H}_2, \end{bmatrix}, \tag{7.1.1}$$

with blocks

$$\mathbf{H}_1 = \begin{bmatrix} -4 & 2 & 4 \\ 1 & -2 & 0 \\ 3 & 0 & -4 \end{bmatrix} \quad \text{and} \quad \mathbf{H}_2 = \begin{bmatrix} -6 & 3 & 2 \\ 2 & -3 & 0 \\ 4 & 0 & -2 \end{bmatrix}. \tag{7.1.2}$$

As generators, the blocks $\mathbf{H}_1$ and $\mathbf{H}_2$ have one zero eigenvalue apiece, with corresponding right eigenvectors $\mathbf{v}_1 = (4, 2, 3)$ and $\mathbf{v}_2 = (3, 2, 6)$. From these eigenvectors, one can assemble the matrix $\mathbf{V}$,

$$\mathbf{V} = \begin{bmatrix} 4/9 & 2/9 & 3/9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3/11 & 2/11 & 6/11 \end{bmatrix}^T. \tag{7.1.3}$$

The matrix composed of left eigenvectors of $\mathbf{H}_1$ and $\mathbf{H}_2$ is similarly used to form $\mathbf{U}$,

$$\mathbf{U} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}. \tag{7.1.4}$$

The generator of the slow reactions that couples the fast configurations sets to one another is

$$\mathbf{G} = \begin{bmatrix} -8 & 0 & 0 & 5 & 3 & 2 \\ 0 & -5 & 0 & 2 & 3 & 1 \\ 0 & 0 & -12 & 4 & 6 & 2 \\ 4 & 2 & 3 & -11 & 0 & 0 \\ 1 & 2 & 5 & 0 & -12 & 0 \\ 3 & 1 & 4 & 0 & 0 & -5 \end{bmatrix}. \tag{7.1.5}$$

To get the equations for the slowly changing variables (7.0.4), calculate

$$\mathbf{UGV} = \begin{bmatrix} -87/11 & 78/11 \\ 29/3 & -26/3 \end{bmatrix}. \tag{7.1.6}$$

from which one can obtain the approximate solution as

$$\mathbf{P}(t) = \mathbf{V}\exp(\varepsilon\mathbf{UGV}t)\mathbf{UP}(0). \tag{7.1.7}$$

As an illustration of the effectiveness of this reduction, Fig. 7.2 shows components $P_1(t)$ and $P_2(t)$ of the solution above for the initial condition $P_i(0) = \delta_{2i}$, and $\varepsilon = 0.01$. One can see that after a short transient time has elapsed, there is an excellent agreement between the exact and the approximate solution to this example problem.

As a second example, we have considered a large set of randomly generated master equations, each with a near block-diagonal structure. For each system, we have and compared found the exact and the slow manifold solutions. Figure 7.3 shows that the approximation error is indeed strongly controlled by the small parameter ε.

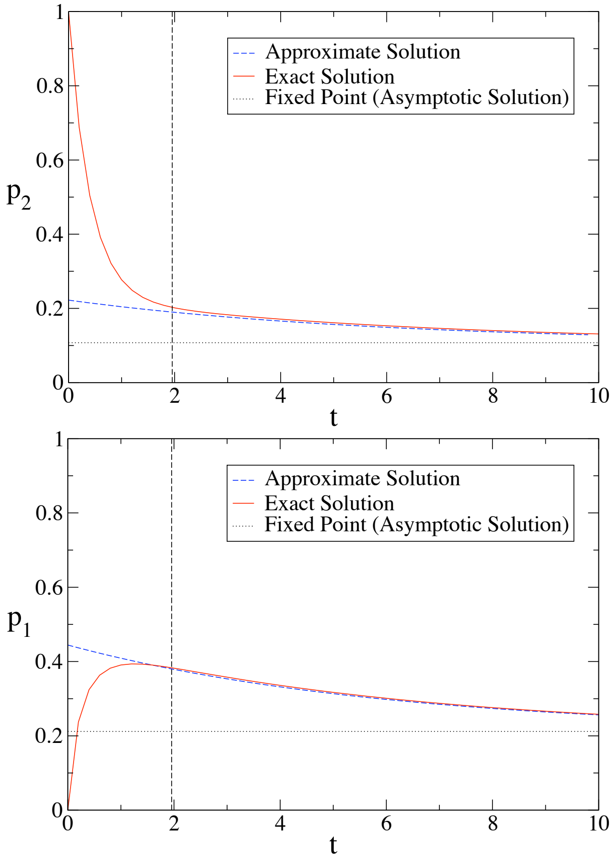


Figure 7.2: Comparison of the approximate and the exact solution to the master equation in Section 7.1. The initial probability distribution is $P_i(0) = \delta_{2i}$. The transient time is estimated to be $T(\varepsilon) = \ln \varepsilon/\lambda_3 = 1.96$ for $\varepsilon = 0.01$, and is denoted by the vertical line on the graph.

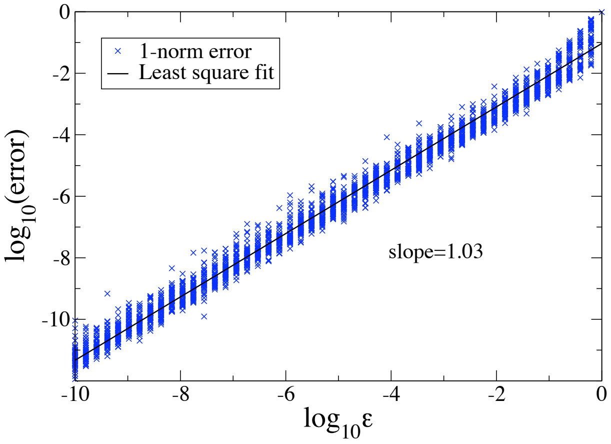


Figure 7.3: 1-norm error in probability distribution for the truncated solution versus ε. For each value of ε we have randomly generated 50 matrices $\mathbf{H}$ and $\mathbf{G}$, so that every $\mathbf{H} + \varepsilon\mathbf{G}$ defines a proper master equation. Each matrix $\mathbf{H}$ has between 2 and 6 blocks and each block has size between 2 and 21. The elements of $\mathbf{H}$ and $\mathbf{G}$ are randomly generated from a uniform distribution between 0 and 1. The probability distributions were calculated after time $t = 2T(\varepsilon) = 2 \log \varepsilon / \lambda_{m+1}$.

Chapter 8

The FSP for Non-Sparse Initial Distributions (NS-FSP)

Although the FSP method is valid for any initial probability distribution, most examples in the literature so far [67, 66, 68, 14, 78, 71] begin with a specific known initial configuration; if the system begins in configuration $\mathbf{x}_k$, the initial probability distribution for the CME was written, $p_i(0) = \delta_{ik}$, where δ_{ik} is the Kronecker delta. Suppose now that the initial distribution is given not by the Kronecker delta but by a vector with many non-zero elements. For example, suppose that the initial distribution is specified by the solution at the end of a previous time interval. From Theorem 5.0.2, in order for the original FSP algorithm to converge, one must be able to find a set of states, $\mathbf{X}_J$, that satisfies the stopping criterion:

$$\left|\exp(\mathbf{A}_J t_f)\mathbf{P}_J(0)\right|_1 \geq 1 - \varepsilon.$$

Since the sum of the FSP solution at t_f cannot exceed the sum of the truncated initial pdv, $\mathbf{P}_J(0)$, one must always include at least as many states in the FSP solution as is required such that $\left|\mathbf{P}_J(0)\right|_1 \geq 1 - \varepsilon$. For a sparse pdv, such as that generated by δ_{ik}, this restriction on the size of the FSP solution is trivial: J need only include k. However, when the initial pdv has broad support, the size of the FSP solution may be much larger and therefore require the inefficient calculation of very high-dimensional matrix exponentials. Fortunately, one can use the property of super-positioning guaranteed by the linearity of the FSP to mitigate this concern and recover some computational efficiency as shown in the following proposition.

This analysis keeps the indexing notation from the original FSP in Chapter 5, but also requires an embedding operator, $\mathcal{D}_J\{.\}$ as follows. Given any vector $\mathbf{v}$ and its J indexed sub-vector $\mathbf{v}_J$, the vector $\mathcal{D}_J\left\{\mathbf{v}_J\right\}$ has the same dimension as $\mathbf{v}$ and its only non-zero entries are the elements of $\mathbf{v}_J$ distributed according to the indexing set J. Also, let the vector $\mathbf{e}_i$ denote a column vector whose i^{th} element is one and the rest of whose elements are zero.

Proposition 8.0.1. *Superposition of FSP Solutions*

Consider any Markov process in which the distribution evolves according to the linear ODE:

$$\dot{\boldsymbol{P}}(t) = \boldsymbol{A}\boldsymbol{P}(t).$$

Let $\gamma < 1$, $\eta < 1$ and $t_f \geq 0$. If there is an index set I such that:

$$\left|\boldsymbol{P}_I(0)\right|_1 \geq \gamma, \tag{8.0.1}$$

and if for every $i \in I$, there is a corresponding index set J_i containing i such that

$$\left|\exp(\boldsymbol{A}_{J_i} t_f)\boldsymbol{e}^i_{J_i}\right|_1 \geq \eta, \tag{8.0.2}$$

then,

$$\sum_{i \in I} p_i \mathcal{D}_{J_i}\left\{\exp(\boldsymbol{A}_{J_i} t_f)\boldsymbol{e}^i_{J_i}\right\} \leq \mathbf{P}(t_f), \tag{8.0.3}$$

and

$$\left| \boldsymbol{P}(t_f) - \sum_{i \in I} p_i \mathcal{D}_{J_i} \left\{ \exp(\boldsymbol{A}_{J_i} t_f) \boldsymbol{e}^i_{J_i} \right\} \right|_1 \leq 1 - \gamma\eta. \tag{8.0.4}$$

Proof. We begin by proving (8.0.3). If we define the index set $I_f = \bigcup_{i \in I} J_i$, then we have the relation,

$$\mathcal{D}_{I_f} \left\{ \exp(\mathbf{A}_{I_f} t_f) \mathbf{P}_{I_f}(0) \right\} = \sum_{i \in I_f} p_i(0) \mathcal{D}_{I_f} \left\{ \exp(\mathbf{A}_{I_f} t_f) \mathbf{e}^i_{I_f} \right\}, \tag{8.0.5}$$

Since $I \subseteq I_f$, we are guaranteed that

$$\mathcal{D}_{I_f} \left\{ \exp(\mathbf{A}_{I_f} t_f) \mathbf{P}_{I_f}(0) \right\} \geq \sum_{i \in I} p_i(0) \mathcal{D}_{I_f} \left\{ \exp(\mathbf{A}_{I_f} t_f) \mathbf{e}^i_{I_f} \right\}.$$

Furthermore, since for every i, $J_i \subseteq I_f$ and $p_i(0) \geq 0$, Theorem 5.0.1 guarantees that,

$$\mathcal{D}_{I_f} \left\{ \exp(\mathbf{A}_{I_f} t_f) \mathbf{P}_{I_f}(0) \right\} \geq \sum_{i \in I} p_i(0) \mathcal{D}_{J_i} \left\{ \exp(\mathbf{A}_{J_i} t_f) \mathbf{e}^i_{J_i} \right\}. \tag{8.0.6}$$

Furthermore, using the result from Theorem 5.0.1 that $\exp(\mathbf{A}_J t_f)$ is non-negative for any index set J, and applying conditions (8.0.1) and (8.0.2) yields

$$\begin{aligned} \left| \mathcal{D}_{I_f} \left\{ \exp(\mathbf{A}_{I_f} t_f) \mathbf{P}_{I_f}(0) \right\} \right|_1 &\geq \left| \sum_{i \in I} p_i(0) \mathcal{D}_{J_i} \left\{ \exp(\mathbf{A}_{J_i} t_f) \mathbf{e}^i_{J_i} \right\} \right|_1 \\ &\geq \eta \left| \mathbf{P}_I(0) \right|_1 \\ &\geq \eta\gamma. \end{aligned} \tag{8.0.7}$$

Theorem 5.0.2 tells us that

$$\mathcal{D}_{I_f}\left\{\exp(\mathbf{A}_{I_f}t_f)\mathbf{P}_{I_f}(0)\right\} \leq \mathbf{P}(t_f),$$

and then from Eqn (8.0.6) we show that

$$\sum_{i\in I_0} p_i(0)\mathcal{D}_{J_i}\left\{\exp(\mathbf{A}_{J_i}t_f)\mathbf{e}^i_{J_i}\right\} \leq \mathbf{P}(t_f), \tag{8.0.8}$$

which is Eqn. (8.0.3).

Combining the fact that $|\mathbf{P}(t_f)|_1 = 1$ and inequality (8.0.7) gives:

$$\left|\sum_{i\in I} p_i(0)\mathcal{D}_{J_i}\left\{\exp(\mathbf{A}_{J_i}t_f)\mathbf{e}^i_{J_i}\right\}\right|_1 \geq \left(|\mathbf{P}(t_f)|_1 - 1\right) + \eta\gamma. \tag{8.0.9}$$

Rearranging this result and applying (8.0.8) yields inequality (8.0.4)

$$\left|\mathbf{P}(t_f) - \sum_{i\in I} p_i(0)\mathcal{D}_{J_i}\left\{\exp(\mathbf{A}_{J_i}t_f)\mathbf{e}^i_{J_i}\right\}\right|_1 \leq 1 - \eta\gamma, \tag{8.0.10}$$

and completes the proof. □

The result of Proposition 8.0.1 now enables one to modify the original FSP algorithm to better handle situations in which the initial probability distribution is non-sparse. Before stating this new algorithm, however, it is important to make a few notes to explain the choice of notation. First, although this algorithm can be useful on its own, it will be seen below that it is most effective as part of a multiple time interval solution scheme. For this reason, the initial time is labeled t_k and the final time is labeled $t_{k+1} = t_k + \tau$. Second, the total error of the current approach is separated into two components, $\varepsilon = 1 - \eta\gamma$, where both γ

and η are numbers slightly less than 1 and will be considered as independent inputs to the algorithm. Here γ refers to the required sum of the truncated probability distribution at t_k, and η refers to the relative accuracy requirement for the solution at t_{k+1} compared to the accuracy at t_k. Third, for added convenience the notation $\mathbf{E}_i = \mathcal{D}_{J_i}\left\{\exp(\mathbf{A}_{J_i}\tau)\mathbf{e}^i_{J_i}\right\}$ denotes the J_i indexed FSP approximation of the distribution at t_{k+1} conditioned upon the i^{th} configuration at t_k. Each matrix exponential, $\exp(\mathbf{A}_{J_i}\tau)$ provides not only $\mathbf{E}_i$ but also approximations to $\mathbf{E}_j$ for every $j \in J_i$. Once these matrix exponentials are computed, one can store every $\mathbf{E}_j = \mathcal{D}_{J_i}\left\{\exp(\mathbf{A}_{J_i}\tau)\mathbf{e}^j_{J_i}\right\}$ and its corresponding index set $J_j = J_i$ that meets the accuracy requirement $|\mathbf{E}_j|_1 \geq \eta$. Note that each vector $\mathbf{E}_i$ is an approximation of the i^{th} column of the operator $\mathbf{\Phi}(\tau)$ in Equation 2.0.6, and the one norm error in this approximation is exactly $(1 - \mathbf{1}^T\mathbf{E}_i)$. This means we are effectively storing a few columns of $\mathbf{\Phi}(\tau)$ at a time. These can later be reused to reduce the total number of matrix computations for a given initial probability distribution $\mathbf{P}(0)$. In addition, one can reuse $\mathbf{\Phi}(\tau)$ for any initial distribution that is supported on the set for which these columns of $\mathbf{\Phi}(\tau)$ have already been computed. With this notation, one can now state the following algorithm:

The FSP Algorithm for Non-Sparse Initial PDV's

Inputs Propensity functions and stoichiometry for all reactions.
Error Parameters, $0 \leq \gamma < 1$ and $0 \leq \eta < 1$.
Initial probability distribution, $\mathbf{P}(t_k)$, where $1 \geq |\mathbf{P}(t_k)|_1 \geq \gamma$.
Length of time interval, τ.

Step 0 Choose a finite set of states, $\mathbf{X}_{I_k}$ such that $|\mathbf{P}_{I_k}(0)|_1 \geq \gamma$.
Initialize a counter, i, as the first element in I_k.
Initialize the FSP solution index set: $I_f = \{i\}$.
Initialize the FSP solution summation to zero: $\mathbf{P}^{FSP}_{I_f}(t_f) = 0$.

Step 1 If $\mathbf{E}_i$ has *not* already been calculated:

Use original FSP algorithm to find J_i and $\exp(\mathbf{A}_{J_i}\tau)$ such that $\left|\exp(\mathbf{A}_{J_i}\tau)\mathbf{e}^i_{J_i}\right|_1 \geq \eta$.
For every $j \in J_i$, if $\left|\exp(\mathbf{A}_{J_i}t_f)\mathbf{e}^j_{J_i}\right|_1 \geq \eta$, then record $\mathbf{E}_j = \mathcal{D}_{J_i}\left\{\exp(\mathbf{A}_{J_i}t_f)\mathbf{e}^j_{J_i}\right\}$ and $J_j = J_i$.

Step 2 Update the FSP solution index set: $I_f = I_f \bigcup J_i$.
Update the FSP solution summation: $\mathbf{P}^{FSP}_{I_f} = \mathbf{P}^{FSP}_{I_f} + p_i\mathbf{E}_i$.

Step 3 If i is the last element in I_0, **Stop**.
$\mathcal{D}_{I_f}\left\{\mathbf{P}^{FSP}_{I_f}(t_f)\right\}$ approximates $\mathbf{P}(t_f)$ to within $\varepsilon = 1 - \gamma\eta$.

Step 4 Increment i to the next element in I_0 and return to **Step 1**.

As discussed above in Sections 5.3 and 5.4, there may be many choices for initializing and expanding the projection during the call to the FSP algorithm in Step 1. Here, the initial projection is chosen using a few SSA runs only on the first time that Step 1 is executed, but the initial projections for subsequent executions of Step 1 are found a little differently. In the previous step, we already computed a set $\mathbf{X}_{J_i}$ that is sufficient for an initial configuration $\mathbf{x}_i$, and we now wish to find a projection that is sufficient for a different initial configuration $\mathbf{x}_j$. As a first guess for $\mathbf{X}_{J_j}$, we take the set $\mathbf{X}_{J_i}$ and translate it by the amount $\mathbf{x}_j - \mathbf{x}_i$. In some cases, this may lead to unrealistic choices for the initial set, such as negative populations, but these are unreachable configurations that are automatically removed from the configuration set. Once this initial projection has been chosen, the expansion routine is the same as above in Section 5.4.

These alterations in the FSP algorithm enable one to handle problems in which the initial probability density vector is not sparse. On its own, this may be convenient when one wishes to study systems that begin somewhere within a range of possible initial configurations. However, as the next chapter illustrates, the non-sparse FSP algorithm has its greatest use when it is integrated into a multiple time interval FSP algorithm.

Chapter 9

The Multiple Time Interval FSP Method (MTI-FSP)

Suppose that one requires that the FSP solution be precise to a 1-norm error of ε for the entire time interval $(0, t_f)$. This requires that the system remains with probability (1-ε) within a finite set $\mathbf{X}_J$ for all times $t \in (0, t_f)$. One can envision many simple cases where such a restriction can require an exorbitantly large space $\mathbf{X}_J$. Suppose that the system begins with an initial condition at $t = 0$ far from the support of the distribution at the later time t_6 as illustrated in Fig. 9.1a. In this case the probability distribution is likely to evolve along some path connecting the initial condition to the final solution. To achieve acceptable accuracy at all times, the projection region must contain not only the initial condition and the final solution, but also every point likely to be reached during the intervening time. In such a circumstance, it can help to break the time interval into pieces and require only that the FSP criteria are satisfied only during each sub-interval. In effect, one seeks a changing projection space that follows the support of the distribution as it evolves. To do this, one can utilize the linearity and time invariance properties of the chemical master equation.

Suppose the system starts with a known initial probability distribution, $\mathbf{P}(0)$,

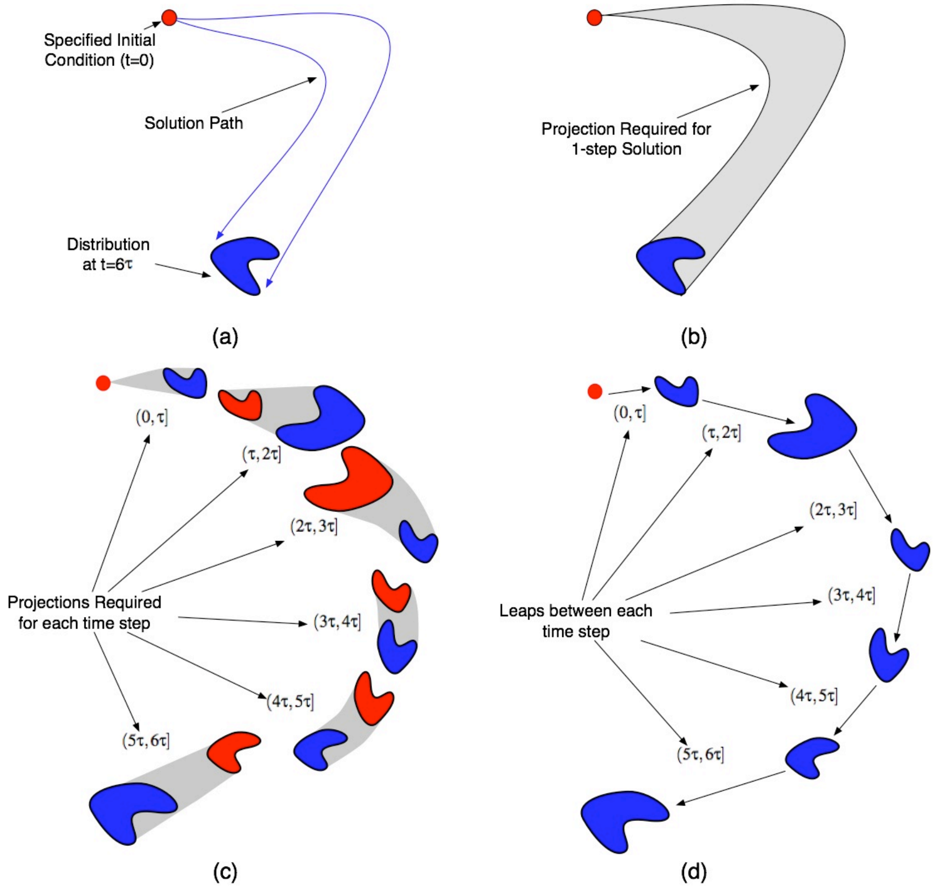

Specified Initial Condition (t=0)
Solution Path
Distribution at t=6τ
(a)
Projection Required for 1-step Solution
(b)
(0, τ]
(τ, 2τ]
(2τ, 3τ]
Projections Required for each time step
(3τ, 4τ]
(4τ, 5τ]
(5τ, 6τ]
(c)
(0, τ]
(τ, 2τ]
(2τ, 3τ]
Leaps between each time step
(3τ, 4τ]
(4τ, 5τ]
(5τ, 6τ]
(d)

Figure 9.1: Schematic of the Multiple Time Interval FSP method. (a) We are given a Markov process that begins at a known initial point in the configuration space. As the probability distribution evolves, it follows a long path in the configuration space such that at time t_6 the distribution is supported in a region far from the initial condition. (b) In order to find a sufficiently accurate FSP solution for all times in the interval $[0, 6\tau]$, the FSP must include not only the initial condition and the final distribution, but also all points along the path. (c) To save computational effort, one can discretize the time interval into smaller intervals and find overlapping projections that need only satisfy the accuracy requirements during those shorter periods of time. Here the final distribution of each time interval (shown in blue) becomes the initial distribution for the next time interval (shown in red). (d) The end result is a discrete map taking the distribution from one instant in time to the next.

and one wishes to approximate the solution to the CME in k time intervals of equal length τ. Using the algorithm in Chapter 8, one can specify a positive $\eta < 1$ and require that transition vectors $\{\mathbf{E}_i\}$ satisfy $|\mathbf{E}_i|_1 \geq \eta$ for all i. For the first time interval, suppose that we simply specify $\gamma_1 = \eta$ and use the non-sparse FSP algorithm (from Chapter 8) to find an approximation of the distribution at $t_1 = \tau$ such that

$$\mathbf{0} \leq \mathcal{D}_{I_1}\left\{\mathbf{P}_{I_1}^{FSP}(t_1)\right\} \leq \mathbf{P}(t_1) \text{ and } \left|\mathbf{P}_{I_1}^{FSP}(t_1)\right|_1 \geq \gamma_1 \eta = \eta^2.$$

For the second time interval, we use $\mathbf{P}_{I_1}^{FSP}(t_1)$ as the initial distribution. If we use the same η, we can save some effort by reusing some of the $\mathbf{E}_i$'s already computed. However, since our solution at the end of the previous interval has a guaranteed sum of only η^2, we must choose a different γ_2. A very reasonable choice is simply to use the guarantee from the previous interval: $\gamma_2 = \eta^2$. With this choice, we can again apply the non-sparse FSP algorithm to find an FSP solution at the end of the second time interval such that

$$\mathbf{0} \leq \mathcal{D}_{I_2}\left\{\mathbf{P}_{I_2}^{FSP}(t_2)\right\} \leq \mathbf{P}(t_2) \text{ and } \left|\mathbf{P}_{I_2}^{FSP}(t_2)\right|_1 \geq \eta^3.$$

Following this example, at each k^{th} step, if we use $\gamma_k = \eta^k$, then we will recover a solution such that

$$\mathbf{0} \leq \mathcal{D}_{I_k}\left\{\mathbf{P}_{I_k}^{FSP}(t_k)\right\} \leq \mathbf{P}(t_k) \text{ and } \left|\mathbf{P}_{I_k}^{FSP}(\tau)\right|_1 \geq \eta^{k+1}.$$

If we apply the fact that $|\mathbf{P}(t_k)|_1 = 1$, we have

$$\left|\mathbf{P}_{I_k}^{FSP}(\tau)\right|_1 \geq (|\mathbf{P}(t_k)|_1 - 1) + \eta^{k+1},$$

which after some rearranging yields

$$\left|\mathbf{P}(t_k) - \mathcal{D}_{I_k}\left\{\mathbf{P}_{I_k}^{FSP}(\tau)\right\}\right|_1 \leq 1 - \eta^{k+1}.$$

Suppose that we wish to find a solution that is within ε of the exact solution of the CME at $t_f = K\tau$. Following the ideas above, we would choose η according to the relation $\varepsilon = 1 - \eta^{K+1}$, or $\eta = (1-\varepsilon)^{\frac{1}{K+1}}$. This procedure is stated more formally in the following algorithm.

The Multiple Time Interval FSP Algorithm (MTI-FSP)

Inputs Propensity functions and stoichiometry for all reactions.
Initial probability distribution, $\mathbf{P}(t_0)$.
Final time of interest, t_f.
Total error, $\varepsilon > 0$.

Step 0 Choose the number of time intervals, K, and calculate $\tau = t_f/K$.
Compute the required sum for each $\mathbf{E}_i$, $\eta = (1-\varepsilon)^{\frac{1}{K+1}}$.
Initialize time step counter: $k = 0$.
Choose initial time index set, I_0, such that $\left|\mathbf{P}_{I_0}(t_0)\right|_1 \geq \eta$.
Initialize the FSP approximate solution at t_0, $\mathbf{P}_{I_0}^{FSP}(t_0) = \mathbf{P}_{I_0}(t_0)$.

Step 1 Run the Non-Sparse FSP algorithm with the initial condition $\mathbf{P}_{I_k}^{FSP}(t_k)$, and error parameters η and $\gamma_k = \eta^{k+1}$ and get $\mathbf{P}_{I_{k+1}}^{FSP}(t_{k+1})$.

Step 2 If $k+1 = K$, then **Stop**.
$\mathcal{D}_{I_K}\left\{\mathbf{P}_{I_K}^{FSP}(t_K)\right\}$ approximates $\mathbf{P}_{I_k}(t_f)$ to within ε.

Step 3 Increment k and return to **Step 1**.

Fig. 9.1 illustrates the possible benefit obtained from this modification to the FSP algorithm. Suppose that one is interested in finding the distribution at time $t = 6\tau$ of a Markov process that begins in the known initial configuration represented by the black dot. Even though the distributions at each of the times $\{0, \tau, 2\tau, \ldots, 6\tau\}$ are supported on only a small portion of the configuration space, the one shot FSP solution must include the whole region of the configuration space that is swept by the distribution between 0 and 6τ (see Fig. 9.1b). Therefore, the one step FSP algorithm requires a large matrix exponential computation. By subdividing the full interval into six subintervals as shown in Fig. 9.1c, one requires more exponential computations, but since each of these computations will be much smaller, the total computational effort may be much less. Recently, Burrage et. al have utilized a similar approach to solve the FSP problem over a set of small time intervals in their Krylov-based FSP algorithm [14]. Their approach is more efficient than the original FSP in that they no longer solve for the full operator $\mathbf{\Phi}(t_f, t_0)$, but instead restrict their efforts to directly computing $\mathbf{P}^{FSP}(t_f) = \mathbf{\Phi}(t_f, t_0)\mathbf{P}(t_0)$. The approach here is far different. Rather than sacrifice the original FSP's ability to handle different initial distributions, as must be done in order to use the Krylov reductions in [14], we instead exploit this flexibility. By restricting all time intervals to the same length, τ, time invariance of the CME guarantees that much of $\mathbf{\Phi}(\tau) = \mathbf{\Phi}(t + \tau, t)$ can be reused from one time interval to the next.

In order to estimate the computational complexity of the new algorithm, one must make make a few assumptions. First, assume that in every call to the original FSP algorithm, the initially chosen projection $\mathbf{X}_{J_0}$ is sufficient to meet the desired accuracy tolerance. This allows one to analyze the complexity separate from the choice of FSP initialization and expansion routines. Let n denote the number of configurations necessary to solve the FSP in a single time interval. The cost of this solution is dominated by the matrix exponential computation on the order of $O(n^3)$. Suppose that the current multiple time interval version of the FSP can solve the same problem with K time intervals while using z matrix exponential computations of equal size s. Furthermore, assume that the sets $\{\mathbf{X}_{I_k}\}$ needed to support the probability distribution at the beginning of each time interval all have exactly w elements. In this case the cost of computing the z exponentials is $O(zs^3)$. The remaining overhead cost is broken into two terms: first, the cost of storing the n columns of $\mathbf{\Phi}(\tau)$ each with s non-zero elements is $O(ns)$. Second, the cost of all K matrix-vector updates is dominated by the cost of multiplying a $w \times s$ matrix by a w element vector or $O(Ksw)$. The total complexity of the Multiple Time Interval FSP algorithm is then $O(zs^3) + O(ns) + O(Ksw)$. As K increases, smaller matrices will be necessary, but the rate at which s decreases will vary from one chemical system to the next. In general, for a small number of time intervals, s is large and the total cost is dominated by the exponential computations (first term). Conversely, for a large number of time intervals, the cost is dominated by the overhead (second two terms). Below, Chapter 14 illustrates the use of this MTI-FSP algorithm through a simplified model of the heat shock response in *E. coli*.

9.1 The FSP τ Leap Approach (τ-FSP)

The previous section showed how time-scale based system partitioning methods can significantly speed up the computation of the FSP. The benefit of such an approach is that one only need consider a part of the configuration space during each time step. This section shows how this approach can be extended by incorporating some ideas of time leaping. The τ leaping methods discussed in Section 3.1.2 above make the assumption that many reactions may occur in a period of time without causing significant changes in the propensity functions. This assumption enables the one to make two related assumptions: first, that each reaction channel can be considered independently of the others, and second, that each reaction channel can be regarded as a simple pure birth, or Poisson, process. In a Monte Carlo simulation scheme, this assumption enables the researcher to approximate the number of times each reaction fires over a given amount of time.

Recall that the SSA is simply a realization of the Chemical Master Equation, which is an infinite set of ODEs describing the evolution of probabilities for every possible population configuration. It is reasonable for one to expect that any valid approximation of the SSA must correspond to similar approximation to the Chemical Master Equation. For example, the system partitioning methods reviewed in 3.1.1 is essentially a Monte Carlo analysis solution of the slow manifold of the CME as discussed in Chapter 7. Similarly, this section shows how the assumption of τ leaping effectively transforms the continuous time CME into an approximate discrete time system.

The original CME is given by the infinite dimensional ODE:

$$\dot{\mathbf{P}}(t) = \mathbf{A}\mathbf{P}(t).$$

As stated above, this system has the solution:

$$\mathbf{P}(t+\tau) = \mathbf{\Phi}(\tau)\mathbf{P}(t).$$

Because the system obeys the rules of superposition, one can consider evolution of the probability from each configuration of the initial distribution independently (see also Chapter 8):

$$\mathbf{P}(t+\tau) = \sum_{i=1}^{\infty} \mathbf{E}_i(\tau) P_i(t),$$

where $\mathbf{E}_i$ is the i^{th} column of the state transition matrix $\mathbf{\Phi}$. If the τ leap assumption holds, and the propensity functions for each of the M reaction channels do not change from their initial value over a time step of length τ, then the number of times the μ^{th} reaction channel fires is a Poisson random variable:

$$\mathcal{P}_{\tilde{k}_{\mu,i}}(k) = \left\{ \begin{array}{cc} \frac{\lambda_{\mu,i}^k \mathrm{e}^{-\lambda_{\mu,i}}}{k!} & ; \text{for } k \geq 0 \\ 0 & ; \text{for } k < 0 \end{array} \right\},$$

where $\lambda_{\mu,i} = a_\mu(\mathbf{x}_i)\tau$ is the propensity function of the μ^{th} reaction evaluated at the i^{th} configuration. The probability of transformation from i to j in the time period τ can be approximated by the sum:

$$\mathbf{\Phi}_{ji}(\tau) \approx \tilde{\mathbf{\Phi}}_{ji}(\tau) = \sum \frac{\left(a_\mu(\mathbf{x}_i)\tau\right)^k \mathrm{e}^{-a_\mu(\mathbf{x}_i)\tau}}{k!} \text{ for all } (k,\mu) \text{ such that } \mathbf{x}_j = \mathbf{x}_i + k\nu_\mu. \tag{9.1.1}$$

In practice, one will typically step through time using the algorithm in the previous section and compute only the columns of $\tilde{\mathbf{\Phi}}$ that are required for the probability distribution at the beginning of each time step. For the readers' conveniences, this algorithm can be summarized as follows:

The τ-FSP Algorithm

Inputs Propensity functions and stoichiometry for all reactions.
Initial probability distribution, $\mathbf{P}(t_0)$.
Final time of interest, t_f.
Total error, $\varepsilon > 0$.

Step 0 Choose the number of time intervals, K, and calculate $\tau = t_f/K$.
Compute the required sum for each $\tilde{\mathbf{\Phi}}_i \approx \mathbf{E}_i$, $\eta = (1-\varepsilon)^{\frac{1}{K+1}}$.
Initialize time step counter: $k = 0$.
Choose initial time index set, I_0, such that $|\mathbf{P}_{I_0}(t_0)|_1 \geq \eta$.
Initialize the FSP approximate solution at t_0, $\mathbf{P}_{I_0}^{FSP}(t_0) = \mathbf{P}_{I_0}(t_0)$.

Step 1 Run the Non-Sparse FSP algorithm with the initial condition $\mathbf{P}_{I_k}^{FSP}(t_k)$,

and error parameters η and $\gamma_k = \eta^{k+1}$ and get $\mathbf{P}^{FSP}_{I_{k+1}}(t_{k+1})$.
Instead of computing matrix exponentials to get $\mathbf{E}$, the columns of $\tilde{\mathbf{\Phi}}$ are estimated using the Poisson τ leap assumption (9.1.1).

Step 2 If $k+1 = K$, then **Stop**.
$\mathcal{D}_{I_K}\left\{\mathbf{P}^{FSP}_{I_K}(t_K)\right\}$ approximates $\tilde{\mathbf{P}}_{I_k}(t_f)$ to within ε, where $\tilde{\mathbf{P}}(t_f)$ is the exact solution to the master equation under τ leap assumption.

Step 3 Increment k and return to **Step 1**.

The end result is a new algorithm that approximates the time leaps as depicted in Fig. 9.1d without actually computing any of the matrix exponentials. The main two differences between this algorithm and that in the previous section are (*i*) the approximate $\tilde{\mathbf{\Phi}}$ rather than $\mathbf{E}$ is used as the probability transformation operator, and (*ii*) the error is given in terms of the difference between the acquired approximation and the exact solution of an artificial process that has been restricted to obey the τ leap assumption. As a result, this approximation is only as valid as the τ leap assumption allows. For processes in which the propensity functions are constant, the two algorithms are equivalent, and the current approach may be far more efficient. For processes in which the propensity functions change rapidly, this algorithm will introduce significant error.

Chapter 10

Interpolation Based FSP Reduction (I-FSP)

In the previous reductions schemes, knowledge of the system is exploited to provide smaller order models while maintaining known bounds on the error of the achieved approximation. This chapter presents a simpler reduction scheme, which can be very effective, but which no longer provides accuracy guarantees.

Suppose that one wishes to find a vector $\mathbf{q}(t) \in \mathbb{R}^m$, for some known interpolation operator $\Xi \in \mathbb{R}^{n\times m}$ such that $\Xi\mathbf{q}(t)$ provides an approximation of $\mathbf{P}(t)$. We assume that $\mathbf{q}(t)$ has linear dynamics and can be expressed by $\mathbf{q}(t) = \exp(\mathcal{A}t)\mathbf{q}(0)$ for some choice of $\mathbf{q}(0)$ and $\mathcal{A}$, and we pose the following problem

$$\min_{\mathbf{q}(0),\mathcal{A}} \left|\mathbf{P}(t) - \Xi\exp(\mathcal{A}t)\mathbf{q}(0)\right|.$$

Performing a Taylor series expansion, the cost of the minimization becomes

$$\left|(\mathbf{P}(0) - \Xi\mathbf{q}(0)) + (\mathbf{AP}(0) - \Xi\mathcal{A}\mathbf{q}(0))\,t + O(t^2)\right|.$$

Minimizing the first term in the least squares sense yields $\mathbf{q}(0) = \Xi^{-L}\mathbf{P}(0)$, and minimizing the second gives $\mathcal{A} = \Xi^{-L}\mathbf{A}\Xi$, where Ξ^{-L} is the left inverse of Ξ.

As an aside, with the correct choice of Ξ, all previous projections shown here can also be derived with this same formalism. The original FSP uses $\Xi_{FSP} = \mathbf{I}_{XJ}$; the original OAFSP uses the projection $\Xi_{OAFSP} = \left[\begin{array}{cc} \mathbf{I}_{JX} & \mathcal{D}_{RO'}\{\mathbf{1}_{RO'}^T\} \end{array} \right]$; in the standard controllability or observability reduction, the columns of Ξ form a basis for the range of the minimal model; and in the multiple time scale reduction, Ξ is simply the matrix of right eigenvectors: $\mathbf{V}$. In the above minimization, problem one could also explore Krylov based methods of simultaneously choosing Ξ as well as $\mathcal{A}$ and $\mathbf{q}(0)$, but these are left to future work.

To illustrate this interpolation based projection technique, we first consider a Markov process evolving along a one dimensional lattice such as that involving a single chemically reacting species, a. We begin with the full lattice, which we project to a finite subset as illustrated in Fig. 10.1(a,b). We choose a smaller subset of interpolation points as shown in Fig. 10.1(c). When the number of a molecules is small, we need greater precision and these points must be closer together, but when the number is larger, a coarser grid is more likely to suffice. Each two consecutive values $q_i(t)$ and $q_{i+1}(t)$ approximate the probability distribution at the points indexed by integers L_i and R_i, respectively. We assume that the probability distribution varies linearly between these two points; and we interpolate the distribution for any intervening point according to:

$$p_j(t) = \left[\begin{array}{cc} \left(1 - \frac{j-L_i}{R_i-L_i}\right) & \frac{j-L_i}{R_i-L_i} \end{array} \right] \left[\begin{array}{c} q_i(t) \\ q_{i+1}(t) \end{array} \right].$$

From this formulation, if we use m nodes to represent a distribution with n elements, we can obtain the projection operator, $\mathbf{\Xi} \in \mathbb{R}^{n\times m}$ as

$$[\Xi_{ji}, \Xi_{j(i+1)}] = \left[\left(1 - \frac{j - L_i}{R_i - L_i}\right), \frac{j - L_i}{R_i - L_i}\right],$$

for all j.

As an example, suppose that the 1 dimension lattice in Fig. 10.1 represents a one species chemical reaction with the following two reactions

$$x \to 2x, \text{ and } 2x \to x,$$

where the propensity of the first reaction is $a_1(x) = 3x$, the propensity of the second is $a_2(x) = x(x-1)$, and the initial condition is $x_{t=0} = 1$. By choosing to include only the first ten configurations of the system $J = \{1, 2, \ldots, 10\}$, one can obtain the finite state projection $\dot{\mathbf{P}}_J^{FSP}(t) = \mathbf{A}_J\mathbf{P}_J^{FSP}(t)$, where the elements of $\mathbf{A}$ are given by

$$A_{ij} = \left\{ \begin{array}{cc} -j^2 - 2j & \text{for } i = j \\ 3j & \text{for } i = j+1 \\ j^2 - j & \text{for } i = j-1 \end{array} \right\}.$$

and the initial distributions is given as $\mathbf{P}_J^{FSP}(0) = [1, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$. One may choose to the interpolate the distribution among the points in the smaller 6 element set $\{1, 2, 4, 6, 8, 10\}$ which corresponds to the using the projection operator

$$\mathbf{\Xi} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Applying the reduction yields

$$\begin{aligned} \mathcal{A} &= \mathbf{\Xi}^{-L}\mathbf{A}\mathbf{\Xi} \\ &\approx \begin{bmatrix} -3.0000 & 2.0000 & 0 & 0 & 0 & 0 \\ 2.4853 & -5.2965 & 5.8865 & -2.8133 & 0.9546 & -0.3091 \\ -0.4264 & 3.4823 & -8.4323 & 14.0664 & -4.7729 & 1.5454 \\ 0.0732 & -0.5976 & 4.7073 & -16.5854 & 27.6829 & -8.9634 \\ -0.0126 & 0.1030 & -0.8116 & 5.4458 & -28.3246 & 52.2351 \\ 0.0025 & -0.0206 & 0.1623 & -1.0892 & 6.2649 & -79.4470 \end{bmatrix}, \end{aligned}$$

and $\mathbf{q}(0) = \mathbf{\Xi}^{-L}\mathbf{P}_J^{FSP}(0) = [1, 0, 0, 0, 0, 0]$. Fig. 10.2 shows the probability distribution at $t_f = 1s$ for the 10-state FSP solution, $\mathbf{P}_J^{FSP}(t_f) = \exp(\mathbf{A}_J t_f)\mathbf{P}_J(0)$, as well as the reduced 6-state solution, $\mathbf{P}_J^I(t_f) = \mathbf{\Xi}\exp(\mathcal{A}t_f)\mathbf{q}(0)$. From the figure, one can see that the two solutions are in relatively good agreement.

For a lattice of two or more dimensions, the process is essentially the same, but the interpolation is slightly more involved and must be approached with more care. For the case of two species, each point (a_j, b_j) on the lattice is interpolated between the four corners of the mesh rectangle in which that point resides: (B_j, L_j), (B_j, R_j), (T_j, L_j), (T_j, R_j), where (B_j, R_j) is the grid point to the bottom-right side of lattice point indexed by j, (T_j, R_j) is the grid point lying to its top-right side, and so on. The probability at time t at each of these grid points is given by $p_{BL(j)}(t)$, $p_{BR(j)}(t)$, $p_{TL(j)}(t)$, and $p_{TR(j)}(t)$. In our approximation scheme, these variables will be approximated by $q_{BL(j)}(t)$, $q_{BR(j)}(t)$, $q_{TL(j)}(t)$, and $q_{TR(j)}(t)$, whose dynamics evolve in a lower dimensional space than the original system. To assign an approximation for $p_j(t)$ where j is the index of lattice point surrounded by the mesh rectangle, we interpolate the four computed q variables, i.e.

$$p_j(t) \approx \mathcal{N}(j)\mathbf{q}_j(t) = \begin{bmatrix} (1-\alpha)(1-\beta) \\ \alpha(1-\beta) \\ (1-\alpha)\beta \\ \alpha\beta \end{bmatrix}^T \begin{bmatrix} q_{LB(j)}(t) \\ q_{RB(j)}(t) \\ q_{LT(j)}(t) \\ q_{RT(j)}(t) \end{bmatrix},$$

where

$$\alpha = \frac{a_j - L_j}{R_j - L_j} \text{ and } \beta = \frac{b_j - T_j}{T_j - B_j}.$$

As in the one dimensional case, these Finite-Element-Method-like “shape functions,” and our chosen enumeration will directly provide the operator Ξ:

$$[\Xi_{j,LB(j)}, \Xi_{j,RB(j)}, \Xi_{j,LT(j)}, \Xi_{j,RT(j)}] = \mathcal{N}(j), \qquad \forall j.$$

Below, Chapters 14 and 15 illustrate this reduction method on a few example

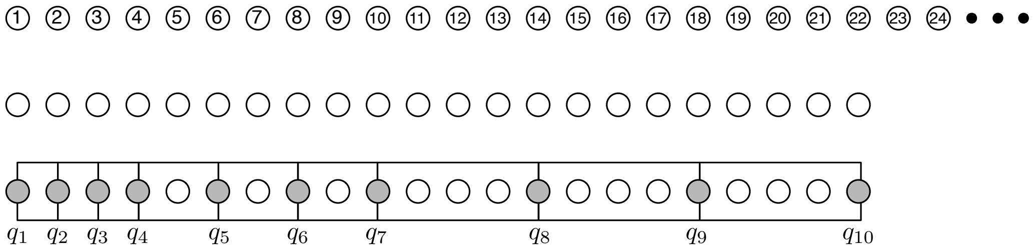


Figure 10.1: One dimensional lattice Markov process. (a) The full infinite dimensional configuration space, (b) The finite state projection, (c) The finite lattice after it has been broken into 9 lattice elements with ten nodes (shaded). In this projection distributions at the unshaded points are interpolated from the distributions approximated at the nodes.

gene regulatory networks.

10.1 Non-Linear Shape Functions (NL-FSP)

The previous section considers interpolating the probability distribution from a subset of configuration points using a linear shape function. For greater reductions in the model order, one may wish to use a nonlinear shape function to describe the distribution. For example, suppose that the system can be assumed to maintain a Poisson, Normal or other common distribution. In such a case, one can write the distribution in terms of only a few variables. If we approximate the distribution as being Poisson,

$$p_i(t) \approx \frac{q^i(t)}{i!}e^{-q(t)}, \text{ for } i = 0, 1, 2, \ldots,$$

then it is described in terms of a single variable, $q(t)$, which is the mean of the Poisson distributed random variable. Similarly, for a Gaussian distributed random variable,

$$p_i(t) \approx \frac{1}{\sqrt{2\pi q_2}} \exp\left(\frac{-(i-q_1)^2}{2q_2}\right), \text{ for } i = 0, 1, 2, \ldots,$$

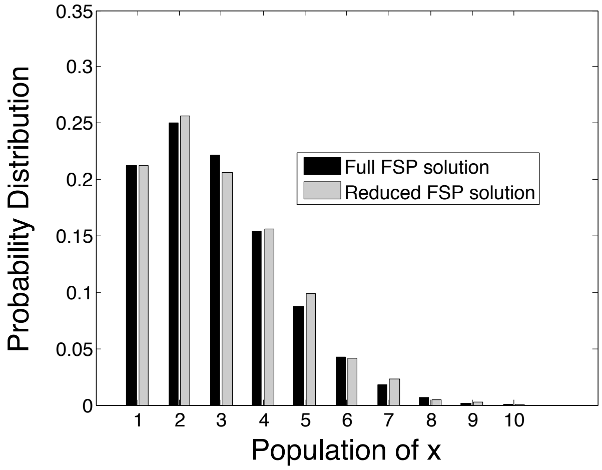


Figure 10.2: The probability distribution for a simple one species chemical reaction as computed using the original FSP solution scheme as well as an interpolation-based reduction of the FSP. The distribution is computed at time $t_f = 1s$.

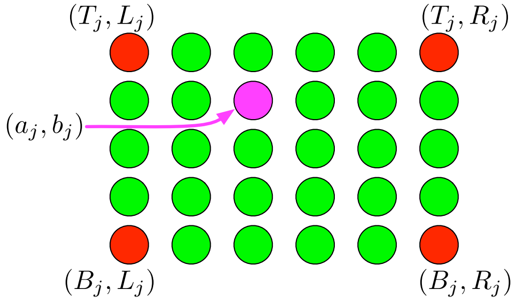


Figure 10.3: Interpolation scheme for a two dimensional Markov lattice. Here the four red corners are the interpolation points which correspond to the approximate probabilities: $q_{BL(j)}(t)$, $q_{BR(j)}(t)$, $q_{TL(j)}(t)$, and $q_{TR(j)}(t)$. The remaining configurations are interpolated from these. For example, the probability of the magenta point (a_j, b_j) is approximated as: $p_j(t) \approx \mathcal{N}(j)\mathbf{q}_j(t) = \frac{3}{20}q_{BL(j)}(t) + \frac{2}{20}q_{BR(j)}(t) + \frac{9}{20}q_{TL(j)}(t) + \frac{6}{20}q_{TR(j)}(t)$.

the variables $q_1(t)$ and $q_2(t)$ are the statistical mean and variance, respectively, of the distribution.

This section seeks to approximate the distribution, $\mathbf{P}(t) \in \mathbb{R}^\infty$ as a function of a vector, $\mathbf{q}(t) \in \mathbb{R}^n$:

$$\mathbf{P}(t) \approx \boldsymbol{\Xi}(\mathbf{q}(t)),$$

where $\boldsymbol{\Xi}(.)$ is nonlinear operator on $\mathbf{q}(t)$ that maps $\mathbb{R}^n \to \mathbb{R}^\infty$. For one dimensional problems, the time derivative of the approximation can be found by applying the chain rule:

$$\begin{aligned}\frac{d\boldsymbol{\Xi}(t)}{dt} &= \frac{d\boldsymbol{\Xi}(\mathbf{q}(t))}{d\mathbf{q}(t)}\frac{d\mathbf{q}(t)}{dt} \\ &= \mathbf{J}(\mathbf{q}(t))\dot{\mathbf{q}}(t),\end{aligned} \tag{10.1.1}$$

where $\mathbf{J}(\mathbf{q}(t))$ is a linear operator that maps $\dot{\mathbf{q}}(t)$ from $\mathbb{R}^n$ to $\mathbb{R}^\infty$.

With this approach, one can rewrite the original master equation: $\dot{\mathbf{P}}(t) = \mathbf{AP}(t)$ with the new lower dimensional non-linear approximate system:

$$\mathbf{J}(\mathbf{q}(t))\dot{\mathbf{q}}(t) \approx \mathbf{A}\boldsymbol{\Xi}(\mathbf{q}(t)).$$

Of course, this approximation cannot be exact as the range of the operator $\mathbf{J}(\mathbf{q})$ is at most dimension n and the range of the operator $\mathbf{A}$ may be infinite. However, one can find an ODE for the evolution of $\mathbf{q}(t)$ that satisfies this approximation in the least squares sense:

$$\dot{\mathbf{q}}(t) = \mathbf{J}^{-L}(\mathbf{q}(t))\mathbf{A}\boldsymbol{\Xi}(\mathbf{q}(t)).$$

We will see in the special case below that when $\mathbf{P}(0)$ and $\mathbf{AP}(0)$ lie within the range of $\boldsymbol{\Xi}(.)$ and $\mathbf{J}(\mathbf{q})$, respectively, then this approximation can be made exact for all times. To see this, we must discuss the error of the approximation.

The error in the approximation can be represented by:

$$\varepsilon(t) = \mathbf{P}(t) - \boldsymbol{\Xi}(\mathbf{q}(t)),$$

and evolves according to the ODE:

$$\begin{aligned}
\frac{d\varepsilon(t)}{dt} &= \frac{d}{dt}\left(\mathbf{P}(t) - \boldsymbol{\Xi}(\mathbf{q}(t))\right) \\
&= \dot{\mathbf{P}}(t) - \dot{\boldsymbol{\Xi}}(\mathbf{q}(t)) \\
&= \mathbf{A}\mathbf{P}(t) - \mathbf{J}(\mathbf{q}(t))\dot{\mathbf{q}}(t) \\
&= \mathbf{A}\mathbf{P}(t) - \mathbf{J}(\mathbf{q}(t))\mathbf{J}^{-L}(\mathbf{q}(t))\mathbf{A}\boldsymbol{\Xi}(\mathbf{q}(t)).
\end{aligned}$$

Consider the case where the initial distribution lies in the space spanned by $\boldsymbol{\Xi}(.)$, i.e. there exists a $\mathbf{q}(0)$ such that $\mathbf{P}(0) = \boldsymbol{\Xi}(\mathbf{q}(0))$. Furthermore, suppose that $\mathbf{A}\mathbf{P}(0)$ and therefore $\mathbf{A}\boldsymbol{\Xi}(\mathbf{q}(0))$ lie in the space spanned by $\mathbf{J}(\mathbf{q}(0))$; i.e. there exists a $\dot{\mathbf{q}}(0)$ such that $\mathbf{J}(\mathbf{q}(0))\dot{\mathbf{q}}(0) = \mathbf{A}\mathbf{P}(0)$. In this case, with the proper choice of $\mathbf{q}(0)$ and $\dot{\mathbf{q}}(t) = \mathbf{J}^{-L}(\mathbf{q}(t))\mathbf{A}\boldsymbol{\Xi}(\mathbf{q}(t))$,

$$\begin{aligned}
\frac{d\varepsilon(0)}{dt} &= \mathbf{A}\boldsymbol{\Xi}(\mathbf{q}(0)) - \mathbf{J}(\mathbf{q}(0))\dot{\mathbf{q}}(0) \\
&= \mathbf{0},
\end{aligned}$$

and the approximation will be exact for all $t \geq 0$. In other words, the true system remains in the space spanned by $\boldsymbol{\Xi}$ for all later times. As examples, the next two subsections consider the special cases in which the Poisson approximation yields the exact solution to the master equation.

10.1.1 Poisson counting process

As a simple example, consider the pure birth (Poisson) process, which is characterized by a single reaction $\emptyset \overset{k}{\to} s_1$. The generator $\mathbf{A}$ for this problem is easy to write and is simply $\mathbf{A} = k(-\mathbf{I}_0 + \mathbf{I}_1)$, where $\mathbf{I}_0$ is an infinite dimensional identity matrix and $\mathbf{I}_1$ is the an infinite dimensional matrix in which the first sub-diagonal is all ones and the remaining elements are all zeros. For the Poisson distribution,

$$\Xi_i(q) = \frac{q^i(t)}{i!} e^{-q(t)},$$

and its derivative with respect to q is easily found to be

$$J_i = \frac{d\boldsymbol{\Xi}}{dq} = \frac{q^i}{i!} e^{-q} \left(\frac{i}{q} - 1 \right).$$

The full vector $\mathbf{J}$ can be written $\mathbf{J} = (\mathbf{I}_0 - \mathbf{I}_1)\boldsymbol{\Xi}(q)$, and its left inverse is simply:

$$\mathbf{J}^{-L} = \boldsymbol{\Xi}^{-L}(q)(-\mathbf{I}_0 + \mathbf{I}_1)^{-1}.$$

Therefore, one can find the simplified ODE for the Poisson process to be:

$$\begin{aligned}
\dot{\mathbf{q}}(t) &= \mathbf{J}^{-L}(\mathbf{q})\mathbf{A}\boldsymbol{\Xi}(\mathbf{q}) \\
&= \boldsymbol{\Xi}^{-L}(\mathbf{q})(-\mathbf{I}_0 + \mathbf{I}_1)^{-1} k(-\mathbf{I}_0 + \mathbf{I}_1)\boldsymbol{\Xi}(\mathbf{q}) \\
&= k,
\end{aligned}$$

which yields the commonly known expression for the evolution of the mean of the standard Poisson process:

$$q(t) = kt + q(0).$$

Note that for a Poisson distributed initial condition, $\mathbf{P}(0) = \mathbf{\Xi}(q(0))$, the range of $\mathbf{AP}(0)$ and that of $\mathbf{J}(q(0))$ are are both $(-\mathbf{I}_0 + \mathbf{I}_1)\mathbf{\Xi}(q(0))$, and the approximation is exact using the argument outlined in the preceding subsection.

10.1.2 Birth-death process

For a slightly more involved example, consider a process with two simple reactions,

$$\emptyset \underset{\gamma}{\overset{k}{\rightleftarrows}} s_1,$$

representing spontaneous $mRNA$ production and linear degradation. In this example, the generator can be written in the form

$$\mathbf{A} = k(-\mathbf{I}_0 + \mathbf{I}_1) + \gamma(-\mathbf{L}_0 + \mathbf{L}_1),$$

where $\mathbf{I}_0$ and $\mathbf{I}_1$ are as above, $\mathbf{L}_0$ is a infinite dimensional diagonal matrix whose entries are $\{0, 1, 2, \ldots\}$ and $\mathbf{L}_1$ is an infinite dimensional matrix whose first super-diagonal is $\{1, 2, 3, \ldots\}$.

As for the previous example, one can chose to use a Poisson distribution as the shape function such that

$$\Xi_i(q) = \frac{q^i(t)}{i!} e^{-q(t)},$$

and

$$\mathbf{J}^{-L} = \mathbf{\Xi}^{-L}(-\mathbf{I}_0 + \mathbf{I}_1)^{-1}.$$

In this case, one can find an explicit expression for $\mathbf{\Xi}^{-L}$ given by:

$$\mathbf{\Xi}^{-L} = \frac{\mathbf{\Xi}^T}{|\mathbf{\Xi}|_2^2}.$$

Thus the ODE for the reduced system can be written as:

$$\begin{aligned}\dot{q} &= \frac{1}{|\mathbf{\Xi}(q)|_2^2}\mathbf{\Xi}^T(q)(-\mathbf{I}_0 + \mathbf{I}_1)^{-1}[k(-\mathbf{I}_0 + \mathbf{I}_1) + \gamma(-\mathbf{L}_0 + \mathbf{L}_1)]\mathbf{\Xi}(q) \\ &= k + \frac{\gamma}{|\mathbf{\Xi}(q)|_2^2}\mathbf{\Xi}^T(q)(-\mathbf{I}_0 + \mathbf{I}_1)^{-1}(-\mathbf{L}_0 + \mathbf{L}_1)\mathbf{\Xi}(q) \\ &= k - \frac{\gamma}{|\mathbf{\Xi}(q)|_2^2}\mathbf{\Xi}^T(q)\mathbf{L}_1\mathbf{\Xi}(q),\end{aligned}$$

where we have used the identity

$$(-\mathbf{I}_0 + \mathbf{I}_1)^{-1}(-\mathbf{L}_0 + \mathbf{L}_1) = -\mathbf{L}_1.$$

Inserting the definition of $\mathbf{\Xi}(q)$ and $\mathbf{L}_1$ allows us to rewrite the ODE as:

$$\begin{aligned}\dot{q} &= k - \gamma\frac{\sum_{i=0}^{\infty}\Xi_i\Xi_{i+1}(i+1)}{\sum_{i=0}^{\infty}\Xi_i\Xi_i} \\ &= k - \gamma\frac{\sum_{i=0}^{\infty}\frac{q^i q^{i+1}(i+1)}{i!(i+1)!}}{\sum_{i=0}^{\infty}\frac{q^i q^i}{i!i!}} \\ &= k - \gamma q\frac{\sum_{i=0}^{\infty}\frac{q^i q^i}{i!i!}}{\sum_{i=0}^{\infty}\frac{q^i q^i}{i!i!}} \\ &= k - \gamma q.\end{aligned}$$

The solution of this system is simply found to be

$$q(t) = \left(q(0) - \frac{k}{\gamma}\right)\exp(-\gamma t) + \frac{k}{\gamma},$$

which has a steady state value of $q_{ss} = \frac{k}{\gamma}$.

Consider the case when the distribution at any $t = 0$ is indeed Poisson with parameter $q(0)$, such as is the case as when the initial condition is specified as $[mRNA]_{t=0} = 0$. In this case, the i^{th} element of the master equation $\mathbf{AP} = k(-\mathbf{I}_0 + \mathbf{I}_1) + \gamma(-\mathbf{L}_0 + \mathbf{L}_1)$ is exactly:

$$[\mathbf{AP}]_i = \left[k\left(\frac{-q^i}{i!} + \frac{q^{i-1}}{(i-1)!}\right) + \gamma\left(\frac{-iq^i}{i!} + \frac{(i+1)q^{i+1}}{(i+1)!}\right)\right]\exp(-q).$$

This simplifies to:

$$[\mathbf{AP}]_i = (k - \gamma q)\frac{-q^i}{i!}\left(\frac{i}{q} - 1\right)\exp(-q),$$

which is simply

$$[\mathbf{AP}]_i = (k - \gamma q)\frac{\partial\left(\frac{-q^i}{i!}\exp(-q)\right)}{\partial q} = \frac{\partial \Xi_i(q)}{\partial q}(k - \gamma q).$$

Therefore, in matrix notation we can write

$$\begin{aligned}[\mathbf{AP}] &= \frac{\partial \Xi(q)}{\partial q}(k - \gamma q) \\ &= \mathbf{J}(q(t))\dot{q}(t).\end{aligned}$$

For any Poisson distributed initial condition $\mathbf{P}(0) = \boldsymbol{\Xi}(q(0))$, this again satisfies the condition that $\mathbf{AP}(0)$ lies within the range of $\mathbf{J}(q(0))$, and the approximation is guaranteed to be exact.

It should be mentioned that since the system studied in this section is an irreducible and recurrent Markov process, it has a unique stationary distribution (see for example [85], Chapter 2.12). Therefore, even if this birth-death process begins with a non-Poisson distribution, it is guaranteed to converge to a Poisson distribution in the limit of long time. This can also be seen by again referring to the error bound between the approximation and the true solution:

$$\begin{aligned}
\varepsilon &= \mathbf{P} - \Xi(q) \\
\dot{\varepsilon} &= \dot{\mathbf{P}} - \dot{\Xi}(q) \\
&= \dot{\mathbf{P}} - \mathbf{J}(q)\dot{q} \\
&= \mathbf{AP} - \mathbf{JJ}^{-L}\mathbf{A}\Xi(q) \\
&= \mathbf{A}(\varepsilon + \Xi(q)) - \mathbf{JJ}^{-L}\mathbf{A}\Xi(q) \\
&= \mathbf{A}\varepsilon + \mathbf{A}\Xi(q) - \mathbf{JJ}^{-L}\mathbf{A}\Xi(q).
\end{aligned}$$

Furthermore, since $\mathbf{A}\Xi(q)$ is in the range of $\mathbf{J}(q)$, then the vector $\mathbf{JJ}^{-L}\mathbf{A}\Xi(q) = \mathbf{A}\Xi(q)$. The ODE for our error reduces to:

$$\dot{\varepsilon} = \mathbf{A}\varepsilon.$$

And since $\mathbf{A}$ is stable, we are guaranteed that the error converges to zero.

Chapter 11

FSP for the Analysis of Stochastic Switches and Trajectories

The FSP approach discussed above systematically collapses the infinite state Markov process into a combination of a truncated finite state process and a single absorbing “error sink”. The resulting system is finite dimensional and solvable. The probabilities of the truncated process give a lower bound approximation to the true CME solution. The probability measure of the error sink gives an exact computation of the error in this approximation. This error can then be decreased to reach any non-zero error tolerance through a systematic expansion of projections known as the FSP algorithm as discussed in Chapters 5.1 and presented in [67, 69]. However, as illustrated in [70] and presented in this chapter, the “error” guarantee of the FSP provides more than a simple distance between the FSP solution and the true solution to the CME. Instead, this important term in the projection provides a wealth of *exact* information about the original Markov process. From it one can determine the statistical distributions of switch rates and escape probabilities and also analyze stochastic pathway bifurcation decisions.

Many recent studies have examined switch rates in the context of stochastic processes operating at their equilibrium or non-equilibrium steady state distributions. As a few representative examples, these methods include Transition Path Sampling [20, 21, 10], Transition Interface Sampling [104, 103], and various approaches of transition path sampling with multiple interfaces [28, 63, 3, 2, 1]. By concentrating on trajectories that eventually result in switches and interrupting the the vast majority trajectories that do not, these approaches are far more efficient than a standard brute force Monte Carlo approach like the SSA. However, as trajectory based analyses, they are limited by the slow convergence of Monte Carlo approaches and cannot provide strict accuracy guarantees. In contrast to these methods, the current study focusses on the *transient evolution of probability distributions* and not on the sampled trajectories of a steady state process. The results sought in this chapter are not histograms of waiting times between switches from one large potential well (or metastable state) to another, but are instead a set of precise upper and lower bounds on the distribution of transition times between specific states and/or arbitrarily chosen state space regions.

This chapter explores the added information contained in the FSP "error" sink and presents some of the types of analyses for which this information provides. Section 11.1 shows how multiple absorbing sinks can be used to effectively analyze pathway bifurcation decisions in stochastic systems. This analysis, in turn, can be used to improve the implementation of the original FSP algorithm from [67]–this result has already appeared briefly in Chapter 5.4 above. Then, Section 11.2 shows how these sinks can be used to determine some statistical quantities for stochastic switches, such as switch waiting and return times, introduces two model reductions to the FSP that can help in the analysis of complex trajectories. Later Chapter 15.1 illustrates how these new approaches can be applied to a stochastic model of the genetic toggle switch [31].

As above, let $\mathcal{M}$ denote a Markov chain on the configuration set $\mathbf{X}$, such as that shown in Fig. 11.1a, whose master equation is $\dot{\mathbf{P}}(t) = \mathbf{AP}(t)$, with initial distribution $\mathbf{P}(0)$. Let $\mathcal{M}_J$ denote a reduced Markov chain, such as that in Fig. 11.1b, comprised of the configurations indexed by J plus a single absorbing state. The master equation of $\mathcal{M}_J$ is given by

$$\begin{bmatrix} \dot{\mathbf{P}}_J^{FSP}(t) \\ \dot{G}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{0} \\ -\mathbf{1}^T\mathbf{A}_J & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ G(t) \end{bmatrix}, \tag{11.0.1}$$

with initial distribution,

$$\begin{bmatrix} \mathbf{P}_J^{FSP}(0) \\ G(0) \end{bmatrix} = \begin{bmatrix} \mathbf{P}_J(0) \\ 1 - \sum \mathbf{P}_J(0) \end{bmatrix}.$$

In previous chapters, the probability lost to the absorbing "error" sink, $G(t)$, is used primarily as in Theorem 5.0.1 as a means to evaluate the FSP projection in terms of its accuracy compared to the true CME solution. As a probability of first transition, however, this "error" term has far more significance than simply the distance between the approximate and exact solutions of the CME. In particular, apart from its use as a measure for the quality of approximation, this error term serves as an exact measure of the rate of first transition from one system region to another. This term may be used to (i) directly determine the statistical distributions for stochastic switch rates, escape times, trajectory periods, and trajectory bifurcations, and (ii) evaluate how likely its is that a system will express certain behaviors during certain intervals of time.

11.1 Pathway Bifurcation analysis with the FSP

There are numerous examples in which biological systems decide between expressing two or more vastly different responses. These decisions occur in developmental pathways in multicellular organisms as heterogeneous cells divide and differentiate, in single cell organisms that radically adapt to survive or compete in changing environments, and even in viruses that must decide to lay dormant or make copies of themselves and ultimately destroy their host [6]. Many of these decisions are stochastic in nature, and models and methods are needed to determine the nature and probability of these decisions. This section shows how the FSP approach can be adapted to answer some of these questions.

In the original FSP approach, a single absorbing state has been used, whose probability coincides with the probability that the system has exited the region $\mathbf{X}_J$. Suppose one wishes to know a little more about *how* the system has exited this region. For example in the process in Fig. 11.1a, one may ask:

Problem 1: What is the probability that the *first* time the system exits $\mathbf{X}_J$ it does so via reaction 1 (rightward horizontal arrow) or via reaction 3 (leftward diagonal arrow)?

Problem 2: What is the probability distribution for the population of species s_2 when the population of s_1 *first* exceeds a specific threshold, s_1^{max}?

These questions can be answered by creating a new Markov process with multiple absorbing states as shown in Fig. 11.1(c,d). Let $\mathcal{M}_J^*$ refer to such a chain where we have included K different absorbing states. The CME for the two prob-

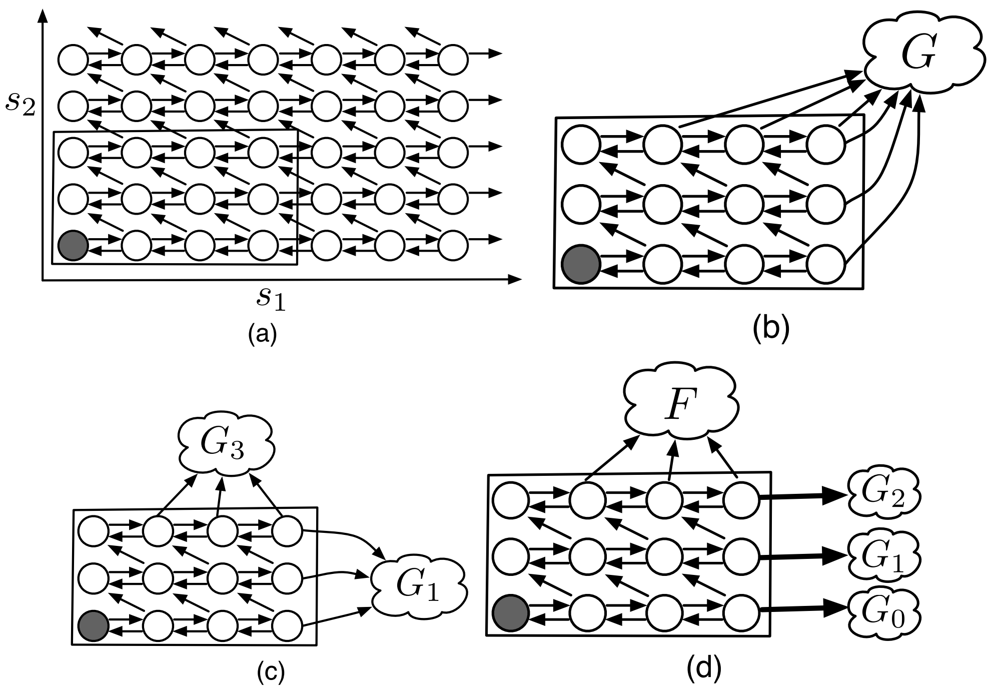


Figure 11.1: (a): A Markov chain for a two species chemically reacting system, $\mathcal{M}$. The process begins in the configuration shaded in grey and undergoes three reactions: The first reaction $\emptyset \to s_1$ results in a net gain of one s_1 molecule and is represented by right arrows. The second reaction $s_1 \to \emptyset$ results in a net loss of one s_1 molecule and is represented by a left arrow. The third reaction $s_1 \to s_2$ results in a loss of one s_1 molecule and a gain of one s_2 molecule. The dimension of the Master equation is equal to the total number of configurations in $\mathcal{M}$, and is too large to solve exactly. (b) In the FSP algorithm a configuration subset, $\mathbf{X}_J$ is chosen and all remaining configurations are projected to a single absorbing point, G. This results in a small dimensional Markov process, $\mathcal{M}_J$. (c,d) Instead of considering only a single absorbing point, transitions out of the finite projection can be sorted as to how they leave the projection space. (c) G_1 and G_3 absorb the probability that has leaked out through reactions 1 or 3, respectively. This information can then be used to analyze the probabilities of certain decisions or to expand the configuration set in later iterations of the FSP algorithm. (d) Each G_i absorbs the probability that s_1 first exceeds a certain threshold, s_1^{max} when $s_2 = i$.

lems above can be written as:

$$\begin{bmatrix} \dot{\mathbf{P}}_J^{FSP}(t) \\ \dot{\mathbf{G}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_J & \mathbf{0} \\ \mathbf{Q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ \mathbf{G}(t) \end{bmatrix}, \tag{11.1.1}$$

where $\mathbf{G} = [G_0, \dots, G_K]^T$ and the matrix $\mathbf{Q}$ is given in Problem 1 by:

$$\mathbf{Q}_{\mu i} = \left\{ \begin{array}{cc} a_\mu(\mathbf{x}_{j_i}) & \text{if } (\mathbf{x}_{j_i} + \nu_\mu) \notin \mathbf{X}_J \\ 0 & \text{Otherwise} \end{array} \right\},$$

and in Problem 2 by:

$$\mathbf{Q}_{ki} = \left\{ \begin{array}{cc} \sum a_\mu(\mathbf{x}_{j_i}) & \text{For all } j_i \text{ s.t. } (\mathbf{x}_{j_i})_2 = k \\ & \text{and } \mu \text{ s.t. } (\mathbf{x}_{j_i} + \nu_\mu)_1 > s_1^{max} \\ 0 & \text{Otherwise} \end{array} \right\}.$$

Note the underlying requirement that each j_i is an element of the index set J. Also recall that $\mathbf{x}_j$ is a population vector–the integer $(\mathbf{x}_j)_n$ is the n^{th} element of that population vector.

For either problem, the solution of (11.1.1) at a time t_f is found by taking the exponential of the matrix in (11.1.1) and has the form

$$\begin{bmatrix} \mathbf{P}_J^{FSP}(t) \\ \mathbf{G}(t) \end{bmatrix} = \begin{bmatrix} \exp(\mathbf{A}_J t_f) & \mathbf{0} \\ \int_0^{t_f} \mathbf{Q} \exp(\mathbf{A}_J \tau) d\tau & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{P}_J^{FSP}(0) \\ \mathbf{G}(0) \end{bmatrix}. \tag{11.1.2}$$

This solution yields all of the same information as previous projections with regards to the accuracy of $\mathbf{P}_J^{FSP}(t)$, but it now provides additional useful knowledge. Specifically, each $G_k(t)$ gives the cumulative probability distribution at time t that the system will have exited from $\mathbf{X}_J$ at least once and that that exit transition will have occurred in the specific manner that was used to define the k^{th} absorbing state.

Section 5.1 above and [67] show a FSP algorithm that relied on increasing the set $\mathbf{X}_J$ until the solution reaches a certain pre-specified accuracy. This expansion was performed using the concept of $N-$step reachability (see Section 5.3), where each set $\{\mathbf{X}_{J_N}\}$ included all configurations that are reachable from $\mathbf{X}_{J_0}$ in N reactions or fewer. The additional knowledge gained from solving Problems 1 or 2 above is easily incorporated into this algorithm. If most of the probability measure left via one particular reaction or from one particular region of $\mathbf{X}_J$, it is reasonable to expand $\mathbf{X}_J$ accordingly. Such an approach is far more efficient that the original FSP algorithm and has been considered in [69] and earlier in Section 5.4.

11.2 Analyzing switch statistics with the FSP

As discussed above, the term $G(t)$ in the equation (11.0.1) for the process $\mathcal{M}_J$ is simply the probability that the system has escaped from $\mathbf{X}_J$ at least once in the time interval $[0, t]$. With such an expression, it is almost trivial to find quantities such as median or p^{th} percentile escape times from the set $\mathbf{X}_J$. One need only find the time t such that $G(t)$ in (11.0.1) is equal to $p\%$. In other words, one finds t such that

$$G(t) = 1 - \left|\exp(\mathbf{A}_J t)\mathbf{P}_J(0)\right|_1 = 0.01p. \tag{11.2.1}$$

This can be solved with a relatively simple line search as will be done in the example of the Gardner switch in Section 15.1. Using a multiple time interval

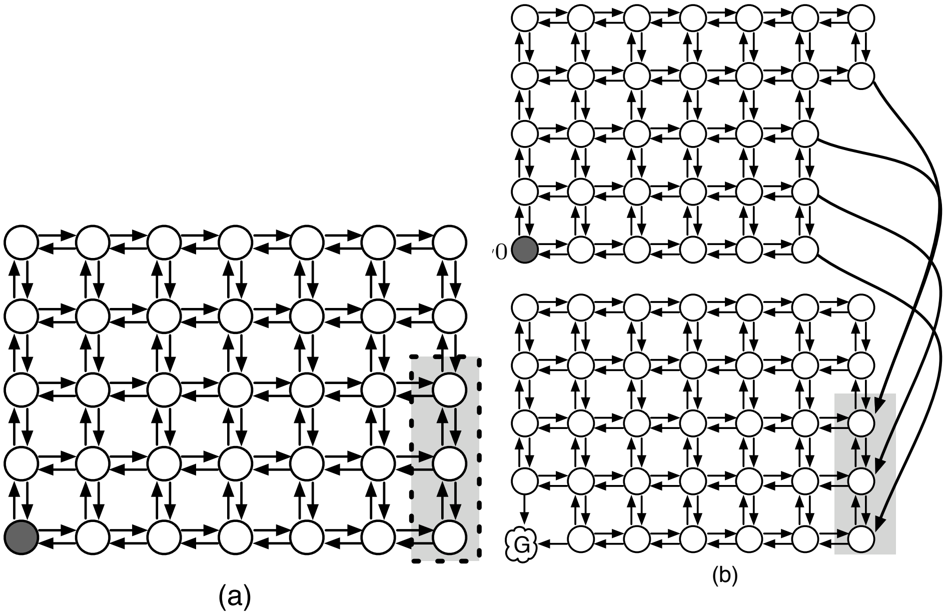


Figure 11.2: Schematic representation for the computation of round trip times for discrete state Markov processes. (a) A Markov chain $\mathcal{M}$ where the system begins in the shaded circle, and we wish to find the distribution for the time at which the system first enters then shaded region and then returns to the initial state. (b) A corresponding Markov process where the top points correspond to states on the journey from the dark circle to the shaded box, and the bottom circles correspond to states along the return trip. In this description, the absorbing point $G(t)$ corresponds to the probability that the system has gone from the initial condition to the grey box and then back again.

FSP approach such as those explored in [69, 14] and Chapter 9 could significantly speed up such a search, but this has not been applied in this study.

Alternatively, one may wish to ask not only for escape times, but for the periods required to complete more complicated trajectories. For example, consider a Markov chain such as that in Fig. 11.2a. The system begins in the state represented by the shaded circle, and one wishes to know the distribution for the time until the system will first visit the region in the grey box and then return to the original state. Biologically this may correspond to the probability that a system will switch from one phenotypical expression to another and then back again. To solve this problem, one can duplicate the lattice as shown in Fig. 11.2b. In this description, the top lattice corresponds to states where the system has never reached the grey box, and the bottom lattice corresponds to states where the system has first passed through that box. The master equation for this system is given by:

$$\begin{bmatrix} \dot{\mathbf{P}}^1_{J_1}(t) \\ \dot{\mathbf{P}}^2_{J_2}(t) \\ \dot{G}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{J_1} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_2\mathbf{C}_1 & \mathbf{A}_{J_2} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_2 & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P}^1_{J_1}(t) \\ \mathbf{P}^2_{J_2}(t) \\ G(t) \end{bmatrix}, \tag{11.2.2}$$

where $\mathbf{X}_{J_1}$ includes every state except those in the grey box, and $\mathbf{X}_{J_2}$ includes every state except the final destination. The matrix $\mathbf{C}_1$ is the output matrix for the first sub-chain and accounts for transitions that exit the $\mathbf{X}_{J_1}$ (via a transition into the grey box):

$$[\mathbf{C}_1]_{ik} = \left\{ \begin{array}{ll} w_\mu(\mathbf{x}) & \text{for } \mathbf{x} = k^{th} \text{ state in } \mathbf{X}_{J_1}, \text{ and } \mathbf{x} + \nu_\mu = i^{th} \text{ state in the grey box} \\ 0 & \text{Otherwise} \end{array} \right\}. \tag{11.2.3}$$

The matrix $\mathbf{B}_2$ is the input matrix that maps the outputs of the first sub-chain to the correct states of the second sub-chain:

$$[\mathbf{B}_2]_{ji} = \left\{ \begin{array}{ll} 1 & \text{for } \mathbf{x} = j^{th} \text{ state in } \mathbf{X}_{J_2}, \text{ and } \mathbf{x} = i^{th} \text{ state in the grey box} \\ 0 & \text{Otherwise} \end{array} \right\}. \tag{11.2.4}$$

The probability of the absorbing point, $G(t)$, in this description is now exactly the probability that the system has completed the return trip in the time interval $[0, t]$. This solution scheme requires a higher dimensional problem than the original problem. However, with the FSP approach from [67], this dimension can be reduced while maintaining a strict measure of the method's accuracy.

11.2.1 Input-output description of connected Markov chains

Each part of the multiple phase trajectories described above has a common form:

$$\begin{aligned}\dot{\mathbf{P}}_i(t) &= \mathbf{A}_i\mathbf{P}_i(t) + \mathbf{B}_i\mathbf{u}_i(t) \\ \mathbf{y}_i(t) &= \mathbf{C}_i\mathbf{P}_i(t),\end{aligned} \qquad (11.2.5)$$

where $\mathbf{u}_i(t)$ and $\mathbf{y}_i(t)$ are the flow of probability into and out of the the i^{th} Markov sub-chain, respectively, and $\mathbf{P}_i(t)$ is the vector of probabilities of the states within the i^{th} Markov sub-chain. In this description the *input matrix* $\mathbf{B}_i$ shows where and how the inputs enter into the i^{th} sub-chain, and the *output matrix* $\mathbf{C}_i$ maps the distribution $\mathbf{P}_i(t)$ to the output $\mathbf{y}_i(t)$. Once each input-output sub-system has been written in the form of the triplet $(\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i)$, one may apply many standard tools to reduce their orders based upon Hankel singular values (see, for example, Chapter 4 of [23]). Many of these tools are available as part of the Robust Control Toolbox in `Matlab`, and for the examples below, we will apply the `Matlab` function `balancmr`. Upon application of these tools, the reduced system is then characterized by a lower order triplet $(\tilde{\mathbf{A}}_i, \tilde{\mathbf{B}}_i, \tilde{\mathbf{C}}_i)$, which can be directly substituted into (11.2.2).

11.2.2 Numerical convolution to compute trajectory times

So far complex trajectories were analyzed by creating a Markov sub-chain for each phase of the trajectory and then creating a new, much larger Markov chain by connecting these sub-chains in series. This can quickly result in a very high dimensional problem, which can require excessive memory and/or be very computationally intensive to solve. As an alternative, one can utilize the linearity of the system to treat each sub-chain separately and then reconnect them with a numerical convolution approach. For example, in Fig. 11.2b, one can first consider the top portion of the chain to find the rate of probability flow into the grey box as a response to beginning at the initial state u_0 at time $t = 0$. This flow is simply the response to the initial distribution:

$$\mathbf{y}(\tau) = \mathbf{C}_1 \exp(\mathbf{A}_{J_1}\tau)\mathbf{P}_{J_1}(0),$$

where each element of the vector $\mathbf{y}(\tau)$ corresponds to the flow into a specific point in the grey box. This probability flow is then the input to the bottom portion of the Markov chain. In practice $\mathbf{y}(\tau)$ is computed using an ODE solver and then stored at N points logarithmically distributed points between $t = 0$ and $t = t_f$. This discrete time signal is then interpolated for use as the forcing term for a second ODE system describing the bottom portion of the chain. Thus, two smaller order ODEs are solved rather than a single much larger order system. One can readily extend this approach to compute the time distributions to complete more complicated trajectories such as hitting multiple way points or completing multiple circuits of the same return trip. The next subsection illustrates how such a convolution based approach can be particularly useful in the computation of probabilities of complex trajectories.

11.2.3 Probabilities of Specific Trajectories

In addition to computing the time a system would take to complete a trajectory, one can also compute the probability that a system will exhibit specific traits at specific instances in time. Define a *partial* probability density vector $\mathbf{P}\{\mathcal{C}_n\}(t_n) = \mathbf{P}\{(\mathbf{X}_{J_0}, t_0); (\mathbf{X}_{J_1}, t_1); \ldots; (\mathbf{X}_{J_{n-1}}, t_{n-1})\}(t_n)$, as the probability that the system satisfies the conditions $\{\mathcal{C}_n\}$:={it begins in the region $\mathbf{X}_{J_0}$ at $t = t_0$; is later is in the region $\mathbf{X}_{J_1}$ at the time $t_1 \geq t_0$; and so on until it is finally in the various states of $\mathbf{X}$ at the time $t_n \geq t_{n-1}$}. Note that the vector $\mathbf{P}\{\mathcal{C}_n\}(t_n)$ has the same dimension as $\mathbf{X}$. The FSP approach provides a simple method to compute $\mathbf{P}\{\mathcal{C}_n\}(t_n)$. This requires the use of an embedding operator $\mathcal{D}_J\{.\}$ as follows: Given any vector $\mathbf{v}$ and its J indexed sub-vector $\mathbf{v}_J$, the vector $\mathcal{D}_J\{\mathbf{v}_J\}$ has the same dimension as $\mathbf{v}$ and its only non-zero entries are the elements of $\mathbf{v}_J$ distributed according to the indexing set J. Furthermore, let $\mathbf{\Phi}(t_2 - t_1) = \exp(\mathbf{A}(t_2 - t_1))$ denote the transition operator that maps distributions at the time t_1 to the corresponding distributions at the later time t_2. Finally, let the vector $\mathbf{P}_{J_n}\{\mathcal{C}_n\}(t_n)$ denote the J-indexed sub-vector of $\mathbf{P}\{\mathcal{C}_n\}(t_n)$.

Proposition 11.2.1. *Using the above notation, the vector* $\mathbf{P}\{\mathcal{C}_n\}(t_n)$ *follows the recursive formula*

$$\mathbf{P}\{\mathcal{C}_{n+1}\}(t_{n+1}) = \mathbf{\Phi}(t_{n+1} - t_n)\mathcal{D}_{J_n}\left\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(t_n)\right\}, \qquad (11.2.6)$$

for all $t_0 \leq t_1 \leq \ldots \leq t_{n+1}$.

Proof. Let $\mathbf{P}(t_n)$ be the full probability distribution at t_n, which can be separated into two parts:

$$\mathbf{P}(t_n) = \mathbf{P}\{\mathcal{C}_n\}(t_n) + \mathbf{P}\{\mathcal{C}'_n\}(t_n),$$

where $\mathbf{P}\{\mathcal{C}_n\}(t_n)$ and $\mathbf{P}\{\mathcal{C}'_n\}(t_n)$ are the partial distributions at t_n that do and do not satisfy the conditions in $\{\mathcal{C}_n\}$, respectively. The full distribution distribution at t_{n+1} is

$$\begin{aligned}\mathbf{P}(t_{n+1}) &= \mathbf{\Phi}(t_{n+1} - t_n)\mathbf{P}(t_n) \\ &= \mathbf{\Phi}(t_{n+1} - t_n)\left(\mathbf{P}\{\mathcal{C}_n\}(t_n) + \mathbf{P}\{\mathcal{C}'_n\}(t_n)\right) \\ &= \mathbf{\Phi}(t_{n+1} - t_n)\left(\mathcal{D}_{J_n}\mathbf{P}_{J_n}\{\mathcal{C}_n\}(t_n) + \mathcal{D}_{J'_n}\mathbf{P}_{J'_n}\{\mathcal{C}_n\}(t_n) + \mathbf{P}\{\mathcal{C}'_n\}(t_n)\right),\end{aligned}$$

where J'_n denotes the complement of J_n. By definition the partial distribution $\mathbf{\Phi}(t_{n+1} - t_n)\mathcal{D}_{J_n}\mathbf{P}_{J_n}\{\mathcal{C}_n\}(t_n)$ satisfies the conditions $\{\mathcal{C}_{n+1}\}$, while the second and third partial distribution terms $\mathbf{\Phi}(t_{n+1} - t_n)\mathcal{D}_{J'_n}\mathbf{P}_{J'_n}\{\mathcal{C}_n\}(t_n)$ and $\mathbf{\Phi}(t_{n+1} - t_n)\mathbf{P}\{\mathcal{C}'_n\}(t_n)$ do not, and we are left with the final result in (11.2.6). □

As a more general form, suppose that the conditions in $\mathcal{C}_n$ are that the system will be in sets $\{\mathbf{X}_{J_i}\}$ not at specific t_i's but at any time during the finite intervals $\{T_i = [a_i, b_i]\}$. In this case, we let the partial probability density vector $\mathbf{P}\{\mathcal{C}_n\}(t_n) = \mathbf{P}\{(\mathbf{X}_{J_0}, T_0); (\mathbf{X}_{J_1}, T_1); \ldots; (\mathbf{X}_{J_n}, T_{n-1})\}(t_n)$ denote the probability density that the system satisfies the conditions $\{\mathcal{C}_n\}$ that it begins in the region $\mathbf{X}_{J_0}$ at some $t_0 \in T_0$, is later in the region $\mathbf{X}_{J_1}$ at some $t_1 \in T_1$, and so on until it is finally in the various states of $\mathbf{X}$ at a time $t_n \geq b_n$. As above $\mathbf{P}\{\mathcal{C}_n\}(t_n)$ satisfies a recursive formula but in a more more general form:

Proposition 11.2.2. *The vector* $\mathbf{P}\{\mathcal{C}_n\}(t_n)$ *follows the recursive formula*

$$\mathbf{P}\{\mathcal{C}_{n+1}\}(t_{n+1}) = \mathbf{\Phi}(t_{n+1} - a_n)\mathcal{D}_{J_n}\left\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(a_n)\right\} + \int_{a_n}^{b_n} \mathbf{\Phi}(t_{n+1} - \tau)\mathbf{y}(\tau)d\tau, \tag{11.2.7}$$

where $\mathbf{y}(\tau)$ *is given by*

$$\mathbf{y}(\tau) = \mathcal{D}_{J_n}\{\mathbf{A}_{J_n,J'_n}\exp(\mathbf{A}_{J'_n}(\tau - a_n))\mathbf{P}_{J'_n}\{\mathcal{C}_n\}(a_n)\}.$$

Proof. Let $\mathbf{P}(t)$ be the full probability distribution at $t \geq a_n$, which can be separated into two parts:

$$\mathbf{P}(t_{a_n}) = \mathbf{P}\{\mathcal{C}_n(t_{a_n})\} + \mathbf{P}\{\mathcal{C}'_n\}(t_{a_n}),$$

where $\mathbf{P}\{\mathcal{C}_n(t_{a_n})\}$ and $\mathbf{P}\{\mathcal{C}'_n\}(t)$ are the portions of the distribution that do and do not satisfy $\{\mathcal{C}_n\}$, respectively. Furthermore, we can separate $\mathbf{P}\{\mathcal{C}_n(t_{a_n})\}$ into two components

$$\mathbf{P}\{\mathcal{C}_n(t_{a_n})\} = \mathbf{w}(t) + \mathcal{D}_{J'_n}\{\mathbf{z}_{J'_n}(t)\},$$

where $\mathbf{w}(t)$ is the partial probability distribution that satisfies $\{\mathcal{C}_n\}$ and the additional condition that the system is in $\mathbf{X}_{J_n}$ at *any* time $\tau \in [a_n, t]$, and $\mathbf{z}_{J_n}(t)$ is the partial distribution where the system satisfies $\{\mathcal{C}_n\}$ and the additional condition that the system remains in the set $\mathbf{X}_{J'_n}$ for *all* times $\tau \in [a_n, t]$. Note that elements of $\mathbf{w}(t)$ refer to each of the states in $\mathbf{X}$ while $\mathbf{z}_{J'_n}(t)$ refers only to states in $\mathbf{X}_{J'_n}$. During the interval $[a_n, b_n]$ the partial distributions $\mathbf{w}(t)$ and $\mathbf{z}_{J'_n}(t)$ evolve according to the linear system:

$$\begin{bmatrix} \dot{\mathbf{w}}_{J_n}(t) \\ \dot{\mathbf{w}}_{J'_n}(t) \\ \dot{\mathbf{z}}_{J'_n}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{J_n} & \mathbf{A}_{J_n,J'_n} & \mathbf{A}_{J_n,J'_n} \\ \mathbf{A}_{J'_n,J_n} & \mathbf{A}_{J'_n} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{J'_n} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{J_n}(t) \\ \mathbf{w}_{J'_n}(t) \\ \mathbf{z}_{J'_n}(t) \end{bmatrix},$$

with initial conditions

$$\begin{bmatrix} \mathbf{w}_{J_n}(a_n) \\ \mathbf{w}_{J'_n}(a_n) \\ \mathbf{z}_{J'_n}(a_n) \end{bmatrix} = \begin{bmatrix} \mathcal{D}_{J_n}\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(a_n)\} \\ \mathbf{0} \\ \mathcal{D}_{J'_n}\{\mathbf{P}_{J'_n}\{\mathcal{C}_n\}(a_n)\} \end{bmatrix}.$$

Solving this system at $t = b_n$ yields:

$$\begin{aligned} \mathbf{w}(b_n) &= \mathbf{\Phi}(b_n - a_n)\mathcal{D}_{J_n}\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(a_n)\} \\ &\qquad + \int_{a_n}^{b_n} \mathbf{\Phi}(b_n - \tau)\mathcal{D}\{\mathbf{A}_{J_n,J'_n}\exp(\mathbf{A}_{J'_n}(\tau - a_n))\mathbf{P}_{J'_n}\{\mathcal{C}_n\}(a_n)\} \\ &= \mathbf{\Phi}(b_n - a_n)\mathcal{D}_{J_n}\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(a_n)\} + \int_{a_n}^{b_n} \mathbf{\Phi}(b_n - \tau)\mathbf{y}(\tau), \text{ and} \\ \mathbf{z}_{J'_n}(b_n) &= \exp(\mathbf{A}_{J'_n}(b_n - a_n))\mathbf{P}_{J'_n}\{\mathcal{C}_n\}(a_n). \end{aligned}$$

The total distribution at time t_{n+1} can be written as:

$$\begin{aligned} \mathbf{P}(t_{n+1}) &= \mathbf{\Phi}(t_{n+1} - b_n)\mathbf{P}\{\mathcal{C}_n(t_{a_n})\} + \mathbf{\Phi}(t_{n+1} - a_n)\mathbf{P}\{\mathcal{C}'_n\}(t_{a_n}) \\ &= \mathbf{\Phi}(t_{n+1} - b_n)\left(\mathbf{w}(b_n) + \mathcal{D}_{J'_n}\{\mathbf{z}_{J'_n}(b_n)\}\right) + \mathbf{\Phi}(t_{n+1} - a_n)\mathbf{P}\{\mathcal{C}'_n\}(t_{a_n}). \end{aligned}$$

By the definitions of $\mathbf{w}(t)$, $\mathbf{z}_{J'_n}(b_n)$ and $\mathbf{P}\{\mathcal{C}'_n\}(t_{a_n})$, only $\mathbf{\Phi}(t_{n+1} - b_n)\mathbf{w}(b_n)$ satisfies the conditions of $\{\mathcal{C}_{n+1}\}$, and

$$\begin{aligned}\mathbf{P}\{\mathcal{C}_{n+1}\}(t_{n+1}) &= \mathbf{\Phi}(t_{n+1}-b_n)\mathbf{w}(b_n)\\ &= \mathbf{\Phi}(t_{n+1}-b_n)\left(\mathbf{\Phi}(b_n-a_n)\mathcal{D}_{J_n}\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(a_n)\}+\int_{a_n}^{b_n}\mathbf{\Phi}(b_n-\tau)\mathbf{y}_n(\tau)\right)\\ &= \mathbf{\Phi}(t_{n+1}-a_n)\mathcal{D}_{J_n}\{\mathbf{P}_{J_n}\{\mathcal{C}_n\}(a_n)\}+\int_{a_n}^{b_n}\mathbf{\Phi}(t_{n+1}-\tau)\mathbf{y}_n(\tau),\end{aligned}$$

thus completing the proof. □

Section 15.1 applies the approach and the methods from above in the switch rate analysis of a stochastic model for Gardner's genetic toggle switch [31].

Chapter 12

Sensitivity Analysis and Identification of Stochastic Models

As discussed above, the rare and discrete cellular nature of chemical components such as genes, RNA molecules, and proteins, can lead to large amounts of intrinsic noise [60, 27, 95, 44, 76, 29, 51]. This intrinsic noise in gene regulatory networks has attracted much recent attention, and it is well established that different systems will exhibit different noise transmission properties. In some systems noise can be focussed [77]; in some noise may cause or enhance resonant fluctuations [57]; some systems may result in stochastic switching [6, 66, 97]; and in some systems noise may be repressed [22].

So far in this work and in in most previous studies, noise in systems biology has often been viewed as a computational obstacle to be overcome. If one does not include it in the model, then one cannot hope to match the behavior of the actual system. However, in many cases, the inclusion of noise in a model results in an explosion of computational complexity. The preceding chapters have discussed many approaches to assist in the modeling of discrete stochastic systems such as kinetic Monte Carlo algorithms and stochastic differential equation approaches in

Chapter 3, the linear noise approximation and other moment matching techniques in Chapter 4, and finite state projection approaches in Chapters 5 through 11. At present, none of these approaches suffices to handle all systems, and there remains much work to be done to improve our computational capabilities. However, as these tools develop, it becomes more possible to overcome the obstacle of intrinsic noise and gain significant benefits in analytical studies. This chapter shows how careful consideration of the transmission of noise can lead to a significant amount of information about the process. By careful sensitivity analysis to determine how system properties affect noise transmission, this information will in turn enable one to better identify properties of the system from experimental data.

The next section provides a brief description of a simple sensitivity analysis approach for stochastic systems. Then Section 12.2.1 presents a simple mathematical description of a stochastic gene regulatory system with transcription and translation. Then Sections 12.2.2 through 12.2.4 show how the parameters of this model can be identified from various pieces of limited information.

12.1 Sensitivity Analyses of Stochastic Processes

The object of a mathematical model is not just to match observed behavior, but to determine how a system will change when modified. As an analytical tool, this ability would allow researchers to predict how a system will react in a wide array of environments. As a design tool, such knowledge is even more useful as it allows researchers to determine how best to alter a system to achieve a desired result. Owing to their immense computational burden, such understanding is as yet out of reach for most discrete stochastic systems. While Monte Carlo type approaches such as those reviewed in Chapter 3 provide great tools with which to visualize a system's dynamics, they require a huge collection of simulations to obtain an accurate statistical solution. This becomes particularly troublesome, when one wishes to compare distributions arising from slightly different parameter sets. Unless sensitivities are very large or the distribution is very precise, changes in the distribution may be hidden by the simulation errors [40]. The advantages of a direct deterministic analysis such as Moment analysis approaches or the FSP is that they are readily repeatable and easily compared.

The sensitivity analysis approach taken in this report is a very simple finite perturbation analysis. First, the master equation is solved with a nominal set of parameters. This results in a full probability distribution at specific points in time. Then one or more of the parameters are changed by a slight amount, and the master equation is solved again for the new parameter set. The sensitivity of the probability distribution is then simply the difference in the solutions of the two master equations divided by the value of the perturbation. As an example, Chapter 15.2 applies such a sensitivity analysis on a stochastic model of a genetic toggle switch. In many cases the sensitivity of the full distribution is more information than is necessary, and instead one may only wish to analyze how certain functions of that distribution change with the parameters. For example, in the identification schemes below, one may define a metric $\rho(\mathbf{P}_{mod}, \mathbf{P}_{exp}) \in \mathbb{R}_{\geq 0}$ which compares model distribution $\mathbf{P}_{mod}$ and a experimental distribution $\mathbf{P}_{exp}$. Then by computing the sensitivity of ρ to the parameters one can determine the best direction to search for for better parameter sets to match the experimental data.

12.2 Identification of Stochastic Processes

In addition to sensitivity analyses, precise computations of system statistics allow can allow one to distinguish between systems with slightly different parameter sets. This enables one to determine which set of parameters is better in the sense that it matches some known information or experimental data. This converging process of proposing and rejecting models is at the heart of scientific inquiry [58]. This section illustrates how such a process can be used in the identification of the parameters of a simple stochastic network of gene transcription and translation. Below, Chapter 15.3 illustrates the identification procedure on a stochastic model of the gene toggle switch.

12.2.1 Moment analysis of a simple gene regulatory network

Consider a simple description of gene transcription and translation. Let x denote the population of mRNA molecules, and let y denote the population of proteins in the system. The system population is assumed to change only through four reactions:

$$\emptyset \to mRNA; \quad mRNA \to \emptyset; \quad mRNA \to mRNA + protein; \quad protein \to \emptyset,$$

for which the propensity functions (or stochastic reaction rates) are

$$w_1(x,y) = k_1 + k_{21}y; \quad w_2(x,y) = \gamma_1 x; \quad w_3(x,y) = k_2 x; \quad w_2(x,y) = \gamma_2 y.$$

Here the term k_{21} corresponds to a feedback effect that the protein is assumed to have on the transcription process. In positive feedback, $k_{21} > 0$, the protein increases transcription; in negative feedback, $k_{21} < 0$, the protein inhibits transcription.

For this system, one can write the master equation [105]:

$$\begin{aligned} \dot{P}_{i,j}(t) = &-(k_1{+}k_{21}j + \gamma_1 i + k_2 i + \gamma_2 j)P_{i,j}(t) \\ &+ (k{+}k_{21}j)P_{i-1,j}(t) + \gamma(i+1)P_{i+1,j}(t) \\ &+ k_2 i P_{i,j-1}(t) + \gamma_2(j+1)P_{i,j+1}(t), \end{aligned} \tag{12.2.1}$$

where $P_{i,j}(t)$ is the probability that $(x, y) = (i, j)$ at the time t, conditioned on some initial probability distribution $\mathbf{P}(t_0)$. In this expression, the first negative term corresponds to the probability of transitions that begin at the state $(x, y) = (i, j)$ and leave to another state, and the remaining positive terms correspond to the reactions that begin at some other state $(x, y) \neq (i, j)$ and transition into the state (i, j).

The mean values of x and y can be written as:

$$\begin{aligned} v_1(t) = E\{x\} &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} i P_{i,j}(t) \\ v_3(t) = E\{y\} &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} j P_{i,j}(t). \end{aligned} \tag{12.2.2}$$

The derivatives of these mean values are found simply by substituting (12.2.1) into (12.2.2):

$$\dot{v}_1(t) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} i \dot{P}_{i,j}(t) = k_1{+}k_{21}v_3 - \gamma_1 v_1,$$

and

$$\dot{v}_3 = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} j\dot{P}_{i,j}(t) = k_2 v_1 - \gamma_2 v_3.$$

Similarly, expressions for the second uncentered moments can be written:

$$\begin{aligned} v_2 = E\{xx\} &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} iiP_{i,j}, \\ v_4 = E\{yy\} &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} jjP_{i,j}, \\ v_5 = E\{xy\} &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} ijP_{i,j}, \end{aligned} \tag{12.2.3}$$

and evolve according to the set of ordinary differential equations:

$$\begin{aligned} \dot{v}_2 &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} i^2\dot{P}_{i,j}(t) = k_1 + (2k_1+\gamma_1)v_1 - 2\gamma_1 v_2 + k_{21}v_3 + 2k_{21}v_5, \\ \dot{v}_4 &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} j^2\dot{P}_{i,j} = k_2 v_1 + \gamma_2 v_3 - 2\gamma_2 v_4, +2k_2 v_5, \\ \dot{v}_5 &= \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} ij\dot{P}_{i,j} = k_2 v_2 + k_1 v_3 + k_{21}Ev_4 - (\gamma_1+\gamma_2)v_5. \end{aligned}$$

Altogether the various components of the first two moments,

$$\mathbf{v}(t) := \left[\begin{array}{ccccc} E\{x\} & E\{x^2\} & E\{y\} & E\{y^2\} & E\{xy\} \end{array}\right]^T,$$

evolve according to the linear time invariant ODE:

$$\begin{aligned}\dot{\mathbf{v}} &= \begin{bmatrix} -\gamma_1 & 0 & k_{21} & 0 & 0 \\ \gamma_1 + 2k_1 & -2\gamma_1 & k_{21} & 0 & 2k_{21} \\ k_2 & 0 & -\gamma_2 & 0 & 0 \\ k_2 & 0 & \gamma_2 & -2\gamma_2 & 2k_2 \\ 0 & k_2 & k_1 & k_{21} & -\gamma_1 - \gamma_2 \end{bmatrix} \mathbf{v} + \begin{bmatrix} k_1 \\ k_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \\ &= \mathbf{A}\mathbf{v} + \mathbf{b}. \end{aligned} \tag{12.2.4}$$

With these expressions for the dynamics of the first two moments, the following subsections will show how these expressions can be used to help identify the various parameters: $[k_1, \gamma_1, k_2, \gamma_2, k_{21}]$ from properly chosen data sets.

12.2.2 Identifying transcription parameters

Begin by considering a simpler birth-death process of mRNA transcripts, whose populations are denoted by x. The moment equation for this system is:

$$\frac{d}{dt}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} -\gamma & 0 \\ \gamma + 2k & -2\gamma \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} + \begin{bmatrix} k \\ k \end{bmatrix}.$$

By applying the nonlinear transformation:

$$\begin{bmatrix} \mu \\ \sigma^2 - \mu \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 - v_1^2 - v_1 \end{bmatrix},$$

where μ and σ^2 refer to the mean and variance of x, respectively, one arrives at the transformed set of equations:

$$\begin{aligned}
\frac{d}{dt}\begin{bmatrix} \mu \\ \sigma^2 - \mu \end{bmatrix} &= \begin{bmatrix} \dot{v_1} \\ \dot{v_2} - 2\bar{x}\dot{\bar{x}} - \dot{v_1} \end{bmatrix} \\
&= \begin{bmatrix} -\gamma_1 v_1 + k \\ (\gamma_1 + 2k)v_1 - 2\gamma v_2 + k - (2v_1 + 1)(-\gamma v_1 + k) \end{bmatrix} \\
&= \begin{bmatrix} -\gamma & 0 \\ 0 & -2\gamma \end{bmatrix}\begin{bmatrix} \mu \\ \sigma^2 - \mu \end{bmatrix} + \begin{bmatrix} k \\ 0 \end{bmatrix}. \qquad (12.2.5)
\end{aligned}$$

Suppose that μ and σ^2 are known at two instances in time, t_0 and $t_1 = t_0 + \tau$, and denote their values at time t_i as μ_i and σ_i^2, respectively. The relationship between (μ_0, σ_0^2) and (μ_1, σ_1^2) is governed by the solution of (12.2.5), which can be written:

$$\begin{bmatrix} \mu_1 \\ \sigma_1^2 - \mu_1 \end{bmatrix} = \begin{bmatrix} \exp(-\gamma\tau)\mu_0 \\ \exp(-2\gamma\tau)(\sigma_0^2 - \mu_0) \end{bmatrix} + \begin{bmatrix} \frac{k}{\gamma}(1 - \exp(-\gamma\tau)) \\ 0 \end{bmatrix}. \qquad (12.2.6)$$

In this expression there are two unknown parameters, γ and k, to be identified from the data $\{\mu_0, \sigma_0^2, \mu_1, \sigma_1^2\}$. If $\mu_0 = \sigma_0^2$, the second equation is trivial, and the solution could be any pair:

$$\left(\gamma,\ k = \gamma\frac{\mu_1 - \exp(-\gamma\tau)\mu_0}{1 - \exp(-\gamma\tau)}\right).$$

If for the first measurement $\mu_0 \neq \sigma_0^2$ and for the second measurement $\mu_1 \neq \sigma_1^2$, then the solution is unique:

$$\gamma = -\frac{1}{2t}\log\left(\frac{\sigma_1^2 - \mu_1}{\sigma_0^2 - \mu_0}\right)$$
$$k = \gamma\frac{\mu_1 - \exp(-\gamma t)\mu_0}{1 - \exp(-\gamma\tau)}.$$

Note that if μ_1 and σ_1^2 are very close, the sensitivity of γ to small errors in this difference becomes very large. From (12.2.6), one can see that as τ becomes very large $(\sigma_1^2 - \mu_1)$ approaches zero, and *steady state measurements do not suffice to uniquely identify both parameters.*

12.2.3 Identifying transcription and translation parameters

The full system in (12.2.4) has the solution:

$$\mathbf{v}_1 = \mathrm{e}^{\mathbf{A}\tau}\mathbf{v}_0 + \int_0^\tau \mathrm{e}^{\mathbf{A}(\tau-s)}\mathbf{b}ds, \qquad (12.2.7)$$

with the notation $\mathbf{v}_i = \mathbf{v}(t_i)$ and $t_{i+1} = t_i + \tau$. Drawing upon the fact that the parameters $\{k_1, \gamma_1, k_2, \gamma_2\}$ are all positive, one can show that the matrix $\mathbf{A}$ is stable and invertible so long as the following condition holds on the feedback term:

$$k_{21} \leq \frac{\gamma_1\gamma_2}{k_2}.$$

Under this condition, (12.2.7) can be written as:

$$\mathbf{v}_1 = \mathrm{e}^{\mathbf{A}\tau}\mathbf{v}_0 - \mathbf{A}^{-1}\left(\mathbf{I} - \mathrm{e}^{\mathbf{A}\tau}\right)\mathbf{b}. \qquad (12.2.8)$$

Suppose that $\mathbf{v}_j$ has been measured at some equally distributed points in time $\{t_0, t_1, \ldots, t_m\}$, and one wishes to identify the parameters $\overline{\lambda} = \{k_1, \gamma_1, k_2, \gamma_2, k_{21}\}$ that satisfy:

$$\mathbf{J}(\overline{\lambda}) := \sum_{j=1}^{m} \left| \mathbf{v}_j - \mathrm{e}^{\mathbf{A}\tau} \mathbf{v}_{j-1} + \mathbf{A}^{-1} \left(\mathbf{I} - \mathrm{e}^{\mathbf{A}\tau} \right) \mathbf{b} \right| = \mathbf{0}.$$

The following subsections provide a few possible approaches to identify these parameters.

Looking at the invariant distribution

If the probability distribution dynamics described in (12.2.4) has an invariant distribution, then the steady state moments,

$$\mathbf{v}_\infty = \lim_{t \to \infty} [v_1, v_2, v_3, v_4, v_5]^T,$$

must satisfy:

$$\mathbf{A}\mathbf{v}_\infty - \mathbf{b} = \mathbf{0}.$$

This equation can be rewritten in terms of the unknown parameters as:

$$\mathbf{\Psi}_\infty \overline{\lambda} = \lim_{t \to \infty} \mathbf{\Psi}(t) \overline{\lambda} = \mathbf{0},$$

where

$$\boldsymbol{\Psi}(t) = \begin{bmatrix} 1 & -v_1 & 0 & 0 & v_3 \\ 1{+}2v_1 & v_1 - 2v_2 & 0 & 0 & v_3{+}2v_5 \\ 0 & 0 & v_1 & -v_3 & 0 \\ 0 & 0 & v_1{+}2v_5 & v_3{-}2v_4 & 0 \\ v_3 & -v_5 & v_2 & -v_5 & v_4 \end{bmatrix}.$$

From this expression, it is obvious that there are two possible cases: (1) the rank of the matrix is full and only the trivial solution is possible: $\overline{\lambda} = \mathbf{0}$, or (2) the matrix has a null-space spanned by $\{\phi_1, \ldots, \phi_p\}$ and there are an infinite number of parameter sets that will result in the same invariant distribution:

$$\overline{\lambda} = \sum_{i=1}^{p} \alpha_i \phi_i, \text{ for any } [\alpha_1, \ldots, \alpha_p] \in \mathbb{R}^p.$$

So long as the parameters enter linearly into the propensity functions $w(\mathbf{x}) = \sum_{\mu=1}^{M} c_\mu f(\mathbf{x})$, then one can extend this argument for any finite number of n moments of the stationary distribution. This tells us that *the steady state distribution cannot provide enough information* to uniquely identify the set of system parameters. Additional information is needed. For example, if the rank of the null space is one, then the knowledge of any one parameter from the set $\overline{\lambda}$ can provide an additional linearly independent equation, and can enable the unique determination of the parameters. If the rank of the null space is p, then at least p additional, linearly independent, pieces of information will be required.

Identifying parameters with full state and derivative information

Suppose that it is possible to measure both the moments and their time derivatives at specific instances in time. In this case, one obtains the same expressions as above but at a finite time where the time derivatives are non-zero:

$$\mathbf{\Psi}(t)\overline{\lambda} = \dot{\mathbf{v}}(t).$$

Depending on the values of $\mathbf{v}(t)$, the matrix $\mathbf{\Psi}(t)$ may or may not have full rank. In particular, if the system is at an invariant distribution as above, then $\mathbf{\Psi}(t)$ will not be invertible. As another example, if the measurements are taken when $y = 0$ then $E\{y\} = E\{y^2\} = E\{xy\} = 0$ and the 4th and 5th columns of $\mathbf{\Psi}(t)$ will be zero, and the rank will be at most 3. In this case, the parameters γ_2 and k_{12} will not be identifiable. If $\mathbf{v}(t)$ can be specified such that $\mathbf{\Psi}(t)$ is invertible, then the parameters can be identified directly from the measurement of $\mathbf{v}(t)$ and its derivative, $\dot{\mathbf{v}}(t)$.

Identification without derivative knowledge

In most cases it is not feasible to measure the time derivative of the moments. More likely, one will only be able to measure the moments at discrete instances in time. In this case one must perform the identification analysis in discrete time according to (12.2.8), which can be rewritten as:

$$\mathbf{v}_j = \mathbf{G}\mathbf{v}_{j-1} + \psi.$$

Here, the matrix $\mathbf{G}$ and the vector ψ are the unknown quantities that we wish to identify. These matrices will be subject to some nonlinear constraints of the form

$$\begin{aligned} \mathbf{G} &= \exp\left(\mathbf{A}\tau\right), \text{ and} \\ \psi &= -\mathbf{A}^{-1}\left(\mathbf{I} - \mathrm{e}^{\mathbf{A}(t_1-t_0)}\right)\mathbf{b}, \end{aligned} \tag{12.2.9}$$

where $\mathbf{A} = \mathbf{A}(\lambda)$ and $\mathbf{b} = \mathbf{b}(\lambda)$ are given as above in (12.2.4).

The relation between $\mathbf{v}_i$ and $\mathbf{v}_{i-1}$ in (12.2.8) can be rearranged as:

$$\mathbf{v}_i = [\mathbf{G}, \psi] \begin{bmatrix} \mathbf{v}_{i-1} \\ 1 \end{bmatrix}.$$

For now, one may ignore the constraints in (12.2.9) and attempt to solve for the 5×6 matrix $[\mathbf{G}, \psi]$. With measurements of $\mathbf{v}_0$ and $\mathbf{v}_1$, one would have only five equations but thirty unknown values (twenty-five in $\mathbf{G}$ and five in ψ). This is not yet enough. However, if one takes measurements at seven equally distributed points in time $\{\mathbf{v}(t_i)\}$, one can write:

$$\begin{aligned} \begin{bmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_6 \end{bmatrix} &= [\mathbf{G}, \psi] \begin{bmatrix} \mathbf{v}_0 & \dots & \mathbf{v}_5 \\ 1 & \dots & 1 \end{bmatrix} \\ \mathbf{V}_f &= \hat{\mathbf{G}} \mathbf{V}_i, \end{aligned} \tag{12.2.10}$$

where $\hat{\mathbf{G}} = [\mathbf{G}, \psi]$ is the matrix of unknown values. Now there are thirty equations with which one can find the thirty unknown values provided that the equations are linearly independent–a fact that can be checked by examining the rank and condition of the matrix $\mathbf{V}_i$. As long as $\mathbf{V}_i$ has full rank, then the solution for $\hat{\mathbf{G}}$ is given by:

$$\hat{\mathbf{G}} = \mathbf{V}_f \mathbf{V}_i^{-1}.$$

In the case of measurement noise it is often advantageous to have more than the minimum number of measurements in (12.2.10). In this case $\hat{\mathbf{G}}$ should be chosen as the argument that minimizes $\mathbf{V}_f - \hat{\mathbf{G}}\mathbf{V}_i$ in the least squares sense:

$$\hat{\mathbf{G}} = \mathbf{V}_f \mathbf{V}_i^{-R}.$$

Once one has extracted $\mathbf{G}$ from $\hat{\mathbf{G}}$, one can diagonalize it:

$$\mathbf{G} = \mathrm{e}^{\mathbf{A}\tau} = \mathbf{S}^{-1}\mathrm{e}^{\mathbf{\Lambda}\tau}\mathbf{S},$$

and solve for the matrix $\mathbf{A}$:

$$\mathbf{A} = \mathbf{S}^{-1}\mathbf{\Lambda}\mathbf{S} = \frac{1}{\tau}\mathbf{S}^{-1}\log(\mathbf{S}\mathbf{G}\mathbf{S}^{-1})\mathbf{S},$$

where $\log(\mathbf{S}\mathbf{G}\mathbf{S}^{-1})$ corresponds to the natural logarithm of the elements of diagonal matrix $\mathbf{S}\mathbf{G}\mathbf{S}^{-1}$. Finally,

$$\psi = -\mathbf{A}^{-1}\left(\mathbf{I} - \mathbf{G}\right)\mathbf{b},$$

gives:

$$\mathbf{b} = -\left(\mathbf{I} - \mathbf{G}\right)^{-1}\mathbf{A}\mathbf{v},$$

and it is relatively easy to solve for the parameters: $\{k_1, \gamma_1, k_2, \gamma_2, k_{21}\}$ from the definition of $\mathbf{A}$ in (12.2.4).

12.2.4 Non-linear optimization based identification

The previous section did not utilize the nonlinear constraints (12.2.9) on the unknown values of $\mathbf{G}$ and ψ. As a result, we were left with thirty unknowns for which we required thirty linearly independent equations. The advantage of such an approach is that the parameters are easily identified from the data by performing a few simple matrix operations. However, to get these equations, one is forced to measure $\mathbf{v}_i$ at seven different points in time. Since $\mathbf{G}$ and ψ are defined by non-linear equations of only five variables, it is reasonable to expect that these parameters should be recoverable with far fewer measurements. However, in this case it is no longer easy to find closed analytical expressions to determine the parameters from the measurements. Instead one must seek to find the argument that minimizes

$$J(\overline{\lambda}) = \left|\left|\mathbf{V}_f - \hat{\mathbf{G}}\mathbf{V}_i\right|\right|_F,$$

where the $||.||_F$ refers to the Frobenious norm (sum of squares of all elements). In the examples below, this minimization is done numerically under the constraints in (12.2.9), and the definitions of $\mathbf{A}$ and $\mathbf{b}$ in (12.2.4).

Identifying parameters with protein distributions only

While it is not currently possible to measure the cell by cell distribution of mRNAs, it is possible to get this information for protein distributions. To do this, one can attach florescent tags, such as green florescent protein (GFP), to the protein of interest and then measure the expression of that protein using flow cytometry or fluorescence activated cell sorting (FACS). Such an approach will yield a histogram of the number of cells containing different levels of the protein. This section presents an identification approach with which this protein distribution information is sufficient to identify rates for transcription and translation.

Supposing that it is only possible to measure the first and second moment of the protein distribution, then these measurements are of the form: $\mathbf{q}_i = \mathbf{C}\mathbf{v}_i$, where

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

In the previous cases, it has been assumed that the initial distribution is known or measurable, but in this case the five initial values of $\mathbf{v}_0$ must now also be estimated in the identification procedure. The identification problem is now to find the set of parameters $\overline{\lambda} = [k_1, \gamma_1, k_2, \gamma_2, k_{21}] \bigcup \hat{\mathbf{v}}_0 \in \mathbb{R}^{10}$, all positive except k_{21} that minimizes

$$J(\overline{\lambda}) = \sum_{i=0}^{m} \left| \mathbf{q}_i - \mathbf{C}\hat{\mathbf{v}}_i \right|_2,$$

where $\mathbf{q}_i$ is the measurement at the i^{th} time point, and $\hat{\mathbf{v}}_i$ is the corresponding estimate of $\mathbf{v}_i$. Substituting the expression (12.2.8) for $\hat{\mathbf{v}}$ yields

$$J(\overline{\lambda}) = \left| \mathbf{q}_0 - \mathbf{C}\hat{\mathbf{v}}_0 \right|_2 + \sum_{i=1}^{m} \left| \mathbf{q}_i - \mathbf{C}\left(\mathbf{G}^i \hat{\mathbf{v}}_0 + \sum_{j=0}^{i-1} \mathbf{G}^j \psi \right) \right|_2,$$

where $\mathbf{G}$ and ψ are functions of $(k_1, \gamma_1, k_2, \gamma_2, k_{21})$ subject to the constraints in (12.2.9), and the definitions of $\mathbf{A}$ and $\mathbf{b}$ in (12.2.4).

In order to fit the ten unknown quantities in $\overline{\lambda}$, one requires at least ten independent equations and ten data points. In the case where the protein first and second moments are measured, this requires measurements at five different time points. With full state measurement, $\mathbf{C} = \mathbf{I}$, as few as two time points will be sufficient, provided that those measurements are rich in all transient dynamics.

12.2.5 Transcription and translation identification examples

In order to examine the utility of the above identification techniques, a set of over 2200 gene regulatory networks have been numerically generated. In each case, the parameters are randomly chosen:

$$\begin{aligned}
k_1 &= \mathrm{U}(0, 0.2), & \gamma_1 &= \mathrm{U}(0, 0.002), \\
k_2 &= \mathrm{U}(0, 0.2), & \gamma_2 &= \mathrm{U}(0, 0.002), \text{ and} \\
k_{21} &= \mathrm{U}(-0.0002, 0),
\end{aligned}$$

where the notation $\mathrm{U}(a, b)$ denotes a uniform random number between a and b. The initial distributions are also chosen randomly according to:[1]

$$\begin{aligned}
v_1^0 &= E\{x(0)\} = \mathrm{U}(0, 10), \\
v_2^0 &= E\{x^2(0)\} = (v_1^0)^2 \mathrm{U}(1, 2), \\
v_3^0 &= E\{y(0)\} = \mathrm{U}(0, 100), \\
v_4^0 &= E\{y^2(0)\} = (v_3^0)^2 \mathrm{U}(1, 2), \text{ and} \\
v_5^0 &= E\{x(0)y(0)\} = v_3^0 v_1^0.
\end{aligned}$$

The goal is to identify these parameters and initial conditions through three approaches.

1. *Full state knowledge without non-linear constraints (FL, Section 12.2.3).*
2. *Full state knowledge with non-linear constraints (FNL, Section 12.2.4).*

[1] The initial distributions are chosen in this manner to guarantee that the variance is non-negative, and the covariance of x and y is zero.

3. Partial state knowledge with non-linear constraints (PNL, Section 12.2.4).

Each identification is conducted under the assumption that there is no measurement noise contained in the identification data. For the non-linear optimization approaches (FNL and PNL), the initial guess for each parameter is randomly chosen to be within one degree of magnitude above or below its true value. All non-linear optimizations use `Matlab`'s standard optimization routine `fminsearch`. In cases when the optimization terminates with a loss function that is greater than ε, the optimization routine makes a new random initial guess and reattempts the optimization. Three cases are possible: (*i*) If the optimization does not converge within twenty attempts, then identification is deemed inconclusive. (*ii*) When the loss function converges to less than ε, and the corresponding parameters, $\hat{\lambda}_i$, satisfy

$$\sum_i \left(\frac{\hat{\lambda}_i - \overline{\lambda}_i}{\overline{\lambda}_i} \right)^2 \leq \delta^2,$$

then that identification is considered to have been successful. (*iii*) Finally, if the optimization routine converges within ε, but the parameters are not satisfactorily close to the true values, the optimization is considered to have yielded a false positive. For this analyses, $\varepsilon = 10^{-7}$ and $\delta = 0.01$.

In every case the FL optimization procedure successfully identified all of the unknown parameters. Also, because this procedure relies only upon a few relatively simple matrix operations and not a numerical optimization, this approach is by far the fastest. However, this identification approach requires a total of thirty-five measurement quantities for each system (five states at seven time points). In practice such experimental results may be prohibitively expensive or otherwise impossible to obtain.

The FNL routine has been applied for measurements of all five states in $\mathbf{v}(t)$,

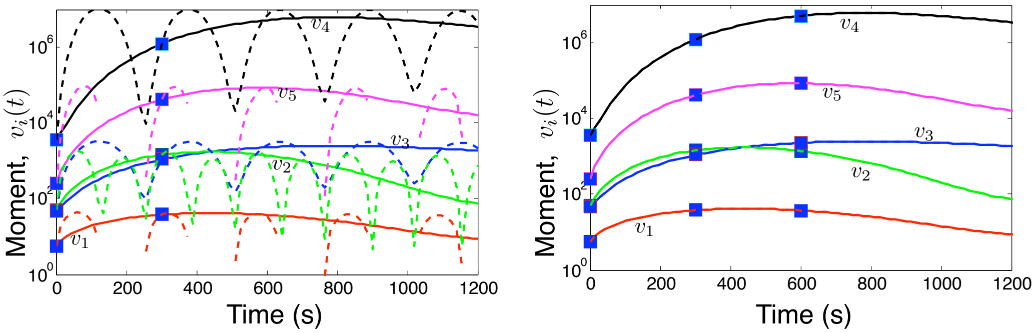


Figure 12.1: Comparison of the dynamics of the true (solid lines) and estimated (dashed lines) system moments for a random set of parameters and initial distribution. Here the FNL estimation uses all five elements of the first two moments (v_1 through v_5). (left) Estimation based upon the measurements at two time points shown in squares. (right) Estimation based upon the measurements at three time points.

but at only two points in time. The numerical optimization converged in every case but two, but falsely identified the system parameters for about 8.5% of the systems. By increasing the number of measurements to three time points (less than half the number of measurements required for the FL method), the success rate of the FNL on the same systems and same initial conditions rose to 100%. Fig. 12.1 illustrates one case in which the FNL identification failed dramatically for a data set of two time points, but succeeded with one additional time point.

For the PNL identification, the parameters are identified using only the protein information at five separate equally distributed points in time. This more computationally intensive approach correctly identified the parameters for about 66.5% of the systems. However this approach failed to converge for 16.1% of the systems and provided false identifications for 17.4% of the systems. Once again, the addition of more time points confers a large advantage (See for example Fig. 12.1). With protein measurements at 6 time points, the false identification rate dropped to less than 0.2%.

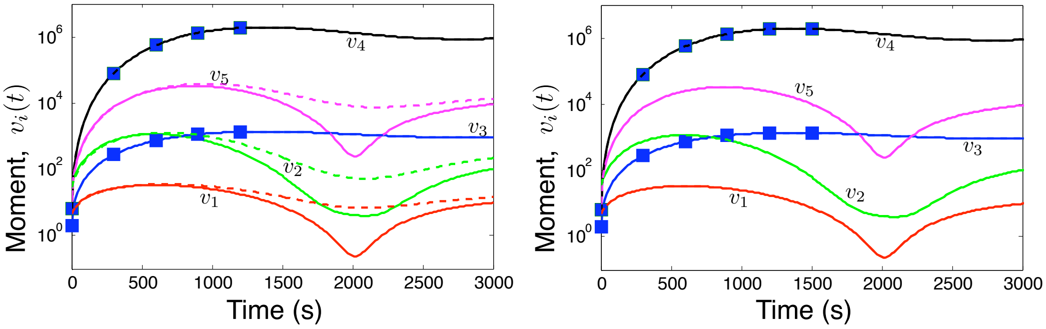


Figure 12.2: Comparison of the dynamics of the true (solid lines) and estimated (dashed lines) system moments for a random set of parameters and initial distribution. Here the PNL estimation uses only data about the protein mean and second moments only (v_3 and v_4). (left) Estimation based upon the measurements at five time points shown in squares. (right) Estimation based upon the measurements at six time points.

12.2.6 Effect of measurement errors on identification

In the real world, it is impossible to obtain perfect measurements, and collected data will always contain some unknown amounts of measurement errors. It is therefore important to characterize how sensitive parameter estimation is to measurement error. This sensitivity depends upon many factors including the identification strategy, the number of measurements, and the periods at which the measurements are taken (See Fig. 12.3).

We have explored the effects of measurement noise on two identification strategies: First, if one can simultaneously measure mRNA and protein levels in individual cells, then one can get all five joint moments $\mathbf{v}(t_i)$ at multiple time points. Second, if one can *only* measure the cellular populations of proteins, then one can only obtain the marginal moments $v_3(t_i) = E\{y\}$ and $v_4(t_i) = E\{y^2\}$. Because the latter strategy then requires the indirect identification of v_1, v_2 and v_5, it is typically less effective than the former strategy (compare dashed and solid lines in Fig. 12.3). For either strategy, more measurements are better as shown in Fig.

12.3a. However, the time of the measurements is also important–periods between measurements must be long enough for transient dynamics to evolve perceptibly, but short enough that the dynamics do not die out altogether (See Fig. 12.3b).

Identification of system parameters also depends upon the robustness of the system response to parametric variations. To see this connection, consider the first order effects that a parameter change[2] of $\Delta\Lambda = [\Delta\lambda_1, \ldots, \Delta\lambda_n]^T$ may have on the a set of measurements $\mathcal{Y} = [\mathcal{Y}_1, \ldots, \mathcal{Y}_{mp}]^T$:

$$\Delta\tilde{\mathcal{Y}} \approx \mathbf{S} \cdot \Delta\tilde{\Lambda},$$

where $\Delta\tilde{\mathcal{Y}} = [\Delta\mathcal{Y}_1/\mathcal{Y}_1, \ldots, \Delta\mathcal{Y}_{mp}/\mathcal{Y}_{mp}]^T$ denotes relative change in each of the p outputs at m time points, and $\Delta\tilde{\Lambda} = [\Delta\lambda_1/\lambda_1, \ldots, \Delta\lambda_n/\lambda_n]^T$ denotes the relative change in the n different parameters. The sensitivity matrix,

$$\mathbf{S} = \begin{bmatrix} \frac{\partial\mathcal{Y}_1/\mathcal{Y}_1}{\partial\lambda_1/\lambda_1} & \cdots & \frac{\partial\mathcal{Y}_1/\mathcal{Y}_1}{\partial\lambda_n/\lambda_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial\mathcal{Y}_{mp}/\mathcal{Y}_{mp}}{\partial\lambda_1/\lambda_1} & \cdots & \frac{\partial\mathcal{Y}_{mp}/\mathcal{Y}_{mp}}{\partial\lambda_n/\lambda_n} \end{bmatrix},$$

is relatively easy to compute using small perturbations about the nominal values of Λ or with a sensitivity function as in [52]. The inverse of this relationship approximates how the parameter estimation would change if one were to have inaccuracies in the measurements:

$$\Delta\tilde{\Lambda} \approx \mathbf{S}^{-L} \cdot \Delta\tilde{\eta},$$

where $\Delta\tilde{\eta} = [\eta_i/\mathcal{Y}_i]$ is the relative amount of noise in each of the measurements

[2] For ease of notation, the unknown initial conditions $\mathbf{v}(0)$ are treated here as elements of the parameter vector Λ.

used for the identification. For each parameter, two situations are possible. If the system is sensitive to a parameter, then small variations in that parameter will lead to large variations in system response. In this case, small measurement errors will have little effect on the estimation. Alternatively, if a system is robust to a certain parameter, small measurement errors will lead to much larger discrepancies in the estimation. However, the model will not require as much precision in the estimation of the more robust parameters. In the modeling endeavor, those parameters which are most sensitive and important are the same that are most easily identified. For the examination of this tradeoff, consider the average absolute sum of the column of $\mathbf{S}$ corresponding to the parameter λ_j:

$$r_j \approx \frac{1}{mp} \left|\mathbf{S}_j\right|_1 .$$

This quantity can be seen as a metric of the average absolute change in the measurements dues to a relative change in the parameter λ_j. Small values of r_j denote that the system is relatively insensitive to changes in the parameter λ_i. Similarly, define the quantity:

$$s_i \approx \frac{1}{mp} \left|\mathbf{R}_{ij}^{-L} \Delta\tilde{\eta}\right|_1 ,$$

to approximate the average relative estimation error in λ_i due to a random relative noise vector $\Delta\tilde{\eta}$ in the measurement data. Small values of s_i show that the estimation of the parameter λ_i is insensitive to measurement errors. A set of 50,000 systems have been randomly generated according to the rules in the previous subsection. The measurements of these systems are subject to random amounts of Gaussian distributed measurement noise with 1% variance. For each system and set of noisy measurements, $\{s_i\}$ and $\{r_i\}$ have been computed, and their median values are listed in Table 12.1. From the table, we see that the transcription/translation process is more robust to changes in γ_1 and γ_2 (r_2 and r_4 are small), but these same parameters are the most sensitive to measurement errors (s_2 and s_4 are relatively large).

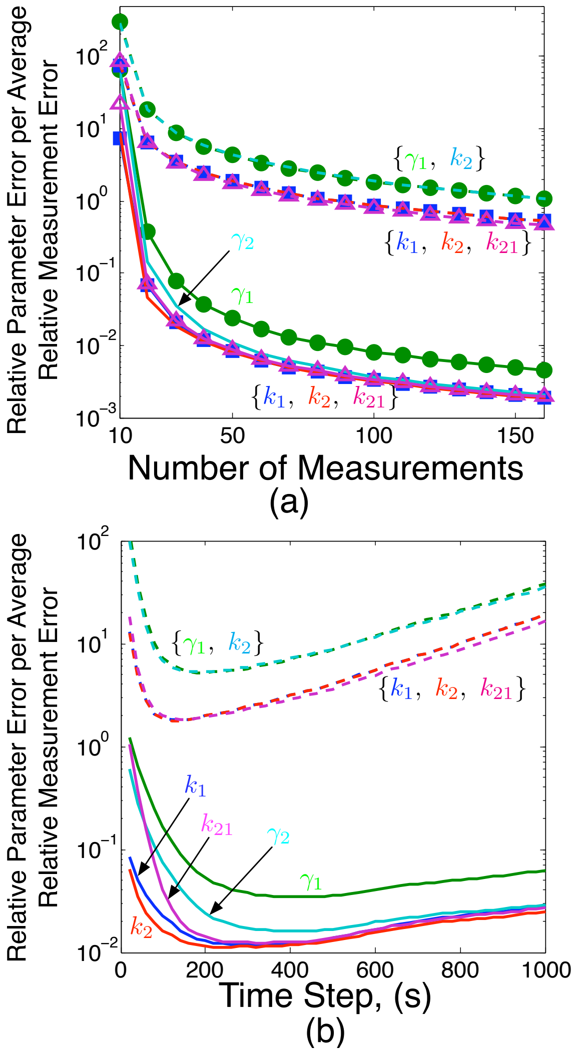


Figure 12.3: Median relative error gain per measurement, $(1/\eta)|\hat{\lambda}_i - \lambda_i|/|\lambda_i|$ in each of the parameters $\{k_1, \gamma_1, k_2, \gamma_2, k_{21}\}$ versus (a) the number of measurements for a time step of $300s$ and (b) the time step for 40 total measurements. Solid lines correspond to the identification with full state measurements, and the dashed lines correspond to the identification with protein mean and second moment only.

Table 12.1: Various strategies for the identification of the five parameters in the gene transcription/translation process with feedback and affine linear propensity functions. The strategies are evaluated by two metrics: ($\{r_i\}$) the median sensitivity of the response to parameter changes and ($\{s_i\}$) the median sensitivity of the parameter identification to measurements errors. See also Fig. 12.3.

Measured Moments	# Time (Data) pts	$\{r_i\}$	$\{s_i\}$
Full State	2 (10)	{0.662, 0.104, 0.376, 0.058, 0.123}	{7.292, 65.67, 8.899, 65.88, 22.16}
Full State	4 (20)	{1.140, 0.267, 0.684, 0.213, 0.649}	{0.067, 0.370, 0.046, 0.148, 0.070}
Full State	8 (40)	{1.368, 0.401, 1.036, 0.566, 1.365}	{0.012, 0.037, 0.011, 0.017, 0.012}
Full State	16 (80)	{1.477, 0.422, 0.941, 0.694, 1.423}	{0.004, 0.011, 0.004, 0.005, 0.004}
Protein Only	5 (10)	{1.100, 0.193, 0.757, 0.215, 0.571}	{76.16, 305.7, 77.26, 290.9, 87.72}
Protein Only	10 (20)	{1.300, 0.215, 0.684, 0.225, 1.076}	{6.549, 18.22, 6.540, 18.07, 6.509}
Protein Only	20 (40)	{1.398, 0.197, 0.508, 0.203, 1.209}	{2.424, 5.659, 2.414, 5.819, 2.272}
Protein Only	40 (80)	{1.448, 0.168, 0.354, 0.170, 1.290}	{1.105, 2.375, 1.099, 2.435, 1.014}

Chapter 13

Applying the FSP to a Toy Model of the Pap Switch

The first example to illustrate the various Finite State Projection approaches is a toy model of the Pap-Pili epigenetic switch. Pili are small hair-like structures that enable bacteria to bind to epithelial cells and thereby significantly increase the bacteria's ability to infect host organisms. However, pili expression comes at cost to the bacteria, as the production of pili requires a large portion of the cellular energy. Whether or not *E. coli* are piliated depends upon the regulation of genes such as the pyelonephritis-associated pili (*pap*) genes. The model presented in this chapter is a very simplified version of the full pap model, which will be considered in great detail in Chapter 16. The majority of this and the extended model is based upon experimental observations made by David Low's group at UCSB [9, 45, 46], and specific results of the more detailed model are presented in [66] and in Chapter 16 below. Figure 13.1 shows a simple illustration of the system consisting of a single operon with two binding sites and a regulatory protein, Leucine-Responsive regulatory Protein (Lrp). Lrp binds reversibly at either or both of the *pap* binding sites such that the operon can exhibit four different configurations (in this simplified model). Each of these configurations is considered as a separate chemical species: g_1 to g_4 as defined in Figure 13.1. When Lrp binds to the upstream site (left) and not to the downstream (right) site, the cell is considered to be in a *production* state - when in this state (circled in Figure 13.1), the cell can produce the proteins necessary to begin production of the pap pili. All other configurations do not produce the necessary proteins.

In addition to the operon and Lrp, this model also considers the local regulatory protein, PapI, which acts to decrease the rate at which Lrp unbinds from the operon. In the real system the change in the population of PapI serves as a positive feedback loop in that larger concentrations of PapI make it more likely for the gene to express the g_2 configuration and continue to produce Pili [45, 46]. In the first example, the population of PapI is assumed to be constant, and the system has exactly four reachable states from the initial condition. In this case the chemical master equation can be solved exactly to find the probability density vector at any future time. In the second example the population of PapI is allowed to change according to translation and degradation events, and the resulting Markov process describing the chemical system has an infinite number of possible states. In each example, the solution scheme is first presented, followed by documentation of the specific parameters and a presentation of computed results. The FSP analyses are then compared to those obtained through use of the SSA and a few of its approximants, and comments are made regarding the comparative efficiency and accuracy of the methods.

13.1 Exact Solution for Finite State Problem

The first example considers the Pap-Pili system shown in Figure 13.1, in which it is assumed that the total concentrations of Lrp and PapI are finite integer

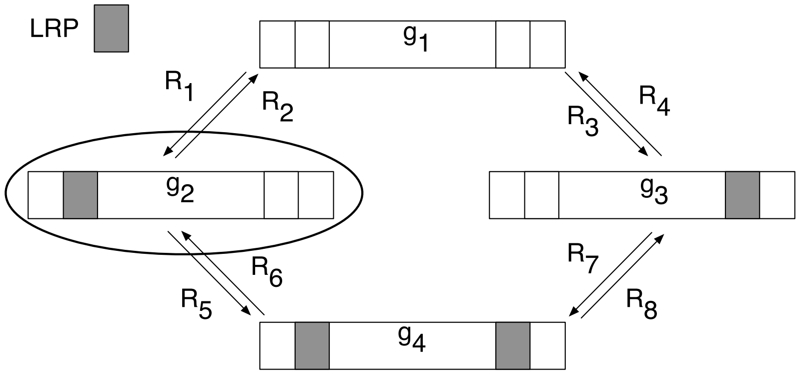


Figure 13.1: Schematic of the four possible DNA-Lrp binding configurations of the Pap Pili operon and the eight possible Lrp binding and unbinding reactions. The circled state corresponds to the *production* state in which transcription of the messenger rna's for pili production and PapI translation is possible.

quantities fixed at u_0 and r_0, respectively. With these assumptions, one can uniquely write out all four possible state descriptions in terms of the populations of each of the important species.

$$\mathbf{X} = \left\{ \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ Lrp \\ PapI \end{bmatrix}_i \right\} = \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ u_0 \\ r_0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ u_0 - 1 \\ r_0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ u_0 - 1 \\ r_0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ u_0 - 2 \\ r_0 \end{bmatrix} \right\}. \tag{13.1.1}$$

The propensity function for each of the eight possible chemical reactions, $a_\mu(\mathbf{X})$ for $\mu = \{1, 2, \ldots, 8\}$, is given by a PapI-dependent reaction rate constant, $c_\mu([PapI])$, multiplied by the product of the concentrations of the reactants. For example, reaction number 1 of the form $R_1 : g_1 + Lrp \rightarrow g_2$ has a propensity function: $a_1 = c_1([PapI])[g_1][Lrp]$, where brackets, [.], around a chemical species denote the population of that chemical species. Since in this case, the populations of PapI and Lrp are assumed to be constant, and g_i is either zero or one, the complete reaction matrix $\mathbf{A}$ can be written as:

$$\mathbf{A} = \begin{bmatrix} -c_1 u_0 - c_3 u_0 & c_2 & c_4 & 0 \\ c_1 u_0 & -c_2 - c_5(u_0 - 1) & 0 & c_6 \\ c_3 u_0 & 0 & -c_4 - c_7(u_0 - 1) & c_8 \\ 0 & c_5(u_0 - 1) & c_7(u_0 - 1) & -c_6 - c_8 \end{bmatrix}. \tag{13.1.2}$$

Suppose that that at time, $t = 0$, the system is in the $\mathbf{x}_1$ state–it has the initial probability density vector,

$$\mathbf{P}(\mathbf{X}; 0) = \begin{bmatrix} P(\mathbf{x}_1; 0) & P(\mathbf{x}_2; 0) & P(\mathbf{x}_3; 0) & P(\mathbf{x}_4; 0) \end{bmatrix}^T = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T.$$

Then one can exactly calculate the solution of the probability density vector at time, t_f, as: $\mathbf{P}(\mathbf{X}; t_f) = \exp(\mathbf{A} t_f)\mathbf{P}(\mathbf{X}; 0)$.

Table 13.1 provides the system parameters and reaction constants for this example. For the state reaction matrix, $\mathbf{A}$ given in Eqn 13.1.2, the state transition matrix, $\exp(\mathbf{A} t_f)$, has been calculated in `Matlab` using the command `expm(.)`. Figure 13.2, black bars, shows the probability density vector of the system at the final time, $t_f = 10s$, as calculated using the FSP. Figure 13.2, also shows the same probability density vectors as averaged using 10^4 simulations of the SSA (dark gray bars) included in the software package `StochKit` [80]. In terms of accuracy, Figure 13.2 shows that the SSA and the FSP produce very similar results. However, even after 10^4 simulations, the pdv acquired with the SSA differs noticeably from the more accurate FSP solution. Suppose one is only interested in the probability that the gene will be in the g_1 configuration. From the FSP computation this state has a probability of 2.433×10^{-3}. Five independent sets of 10^4 SSA simulations predicted this probability to be $\{2.2, 2.9, 2.6, 1.6 \text{ and } 3.7\} \times 10^{-3}$, respectively. Thus the SSA results have relative errors that range from -34 to +52 percent. Depending upon the needs of the researcher, such errors may be unacceptable, and more simulations will be required. As the number of simulations increases, the SSA approaches the accuracy of the FSP; however, even at one million runs the relative errors in the prediction of the g_1 state often exceeds 0.6 percent. On average, each SSA run required the simulation of about 24 events. However, if one were to increase all of the rate constants by a large constant (or equivalently increase the time of simulation), then the number of reactions would increase proportionately. As more reactions occur, the computational effort of the SSA also increases, while the effort required for the FSP method remains unchanged. For a comparison of the time required, the FSP solution took less than 0.3 seconds while the SSA took slightly more time (0.4 seconds) to simulate the system 10^4 times or about 40 seconds to simulate the system one million times.

As stated above, the use of time leaping methods has dramatically improved the computational efficiency of the SSA in many circumstances. However, for this particular example, these methods offer no advantage. At any instant in time, each of the four molecular species, g_1 to g_4, has a population of either zero or one. It is not possible for *any* reaction to occur twice consecutively without resulting in negative populations. Furthermore, every propensity function switches between zero and some positive value within the space of a single reaction. In order to avoid impossible populations, therefore, no τ leap may include more than a single reaction, which is no better than an SSA step. The reader should note that

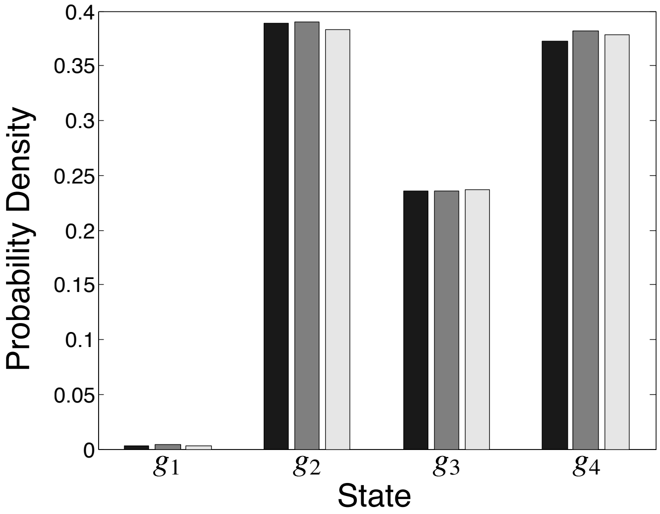


Figure 13.2: Probability density vector for the simple 4-state model at time, $t_f = 10s$, as calculated directly using the exact FSP method (black bars) and as averaged using 10,000 runs of the Stochastic Simulation Algorithm (dark gray bars) and an adaptive τ leaping algorithm (light gray bars). Initial conditions were *pap* operon configuration, g_1, at $t_o = 0s$ (See parameters and initial conditions in Table 13.1).

statement applies to Binomial τ leaping as well as Poisson τ leaping. For example, `StochKit`'s adaptive step size τ leaping code [80] automatically reverts to the SSA and takes about 0.4 seconds for 10^4 realizations. Figure 13.2, light gray bars, illustrates the results using 10^4 τ leaping simulations.

13.2 Approximate Solution Infinite State Problem

In most realistic biological systems, the chemical concentrations of regulatory proteins are constantly changing by discrete values through transcription, translation, degradation, and similar events. This example adds additional reactions to

Reactions				
	Number	Stoichiometry	Rate Constant(c_μ)	Units
	R_1	$\mathbf{X}_1 + Lrp \rightarrow \mathbf{X}_2$	100	s^{-1}
	R_2	$\mathbf{X}_2 \rightarrow \mathbf{X}_1 + Lrp$	$250 - 225\left(1 - \frac{r}{1+r}\right)$	s^{-1}
	R_3	$\mathbf{X}_1 + Lrp \rightarrow \mathbf{X}_3$	100	s^{-1}
	R_4	$\mathbf{X}_3 \rightarrow \mathbf{X}_1 + Lrp$	$120 - 20\left(1 - \frac{r}{1+r}\right)$	s^{-1}
	R_5	$\mathbf{X}_2 + Lrp \rightarrow \mathbf{X}_4$	1	s^{-1}
	R_6	$\mathbf{X}_4 \rightarrow \mathbf{X}_2 + Lrp$	$120 - 20\left(1 - \frac{r}{1+r}\right)$	s^{-1}
	R_7	$\mathbf{X}_3 + Lrp \rightarrow \mathbf{X}_4$	1	s^{-1}
	R_8	$\mathbf{X}_4 \rightarrow \mathbf{X}_3 + Lrp$	$250 - 225\left(1 - \frac{r}{1+r}\right)$	s^{-1}

Parameters and Initial Conditions			
	Parameter	Notation	Value
	Lrp Population	u_o	200
	PapI Population	r_o	5
	Initial Time	t_o	$0s$
	Final Time	t_f	$10s$
	Initial *pdv*	$\mathbf{P}(\mathbf{X};0)$	$[1,0,0,0]^T$

Table 13.1: Reactions and parameters used in the SSA and exact FSP solutions for the four-state, eight-reaction system describing the Pap-Pili Epigenetic Switch.

the above system and allows the population of PapI to change over time. For later convenience, let the variable, r, to denote the concentration of PapI: $r \equiv [PapI]$. Suppose r increases by a stochastic reaction that can occur when the gene is in the g_2 configuration. Also, let r decrease through a stochastic degradation event that is independent of the gene state. The propensity functions for these events can then be given, respectively, as: $a_T = c_T[g_2]$ and $a_D = c_D r$. Because r is allowed to change, the set of all possible states becomes:

$$\mathbf{X}=\left\{\begin{bmatrix} g_1\\ g_2\\ g_3\\ g_4\\ Lrp\\ PapI\end{bmatrix}_i\right\}=\left\{\begin{bmatrix}1\\0\\0\\0\\u_0\\r\end{bmatrix},\begin{bmatrix}0\\1\\0\\0\\u_0-1\\r\end{bmatrix},\begin{bmatrix}0\\0\\1\\0\\u_0-1\\r\end{bmatrix},\begin{bmatrix}0\\0\\0\\1\\u_0-2\\r\end{bmatrix}\right\}, \tag{13.2.1}$$

for $r=\{0,1,2,\ldots\}$. At this point is is useful to establish a unique ordering system for the elements in the configuration space, $\mathbf{X}$. For this particular problem, it is convenient to arrange the states according to the population of PapI:

$$j=\left\{\begin{matrix}(4r+1) & \text{if } [g_1]=1\\ (4r+2) & \text{if } [g_2]=1\\ (4r+3) & \text{if } [g_3]=1\\ (4r+4) & \text{if } [g_4]=1\end{matrix}\right\}=\sum_{i=1}^{4}(r+i[g_i]), \tag{13.2.2}$$

where j is the index of the state $\mathbf{x}_j\in\mathbf{X}$. The system changes from one state to another through three types of reactions: first, the operon configuration can change according to the reactions described above in the first example. The rates for these reactions are now dependent upon the variable concentration of PapI: $\mathbf{A}_r=\mathbf{A}(r)$, where the form of $\mathbf{A}$ is given in Eqn 13.1.2. The second reaction type allows for the translation of PapI only when the *pap* operon is in the g_2 configuration. The third type allows for PapI to degrade. Using the ordering defined in Eqn 13.2.2, all reactions can be combined to form the global infinitesimal generator:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_0-\mathbf{T}_0-\mathbf{D}_0 & \mathbf{D}_1 & 0 & 0 & \cdots \\ \mathbf{T}_0 & \mathbf{A}_1-\mathbf{T}_1-\mathbf{D}_1 & \mathbf{D}_2 & 0 & \cdots \\ 0 & \mathbf{T}_1 & \mathbf{A}_2-\mathbf{T}_2-\mathbf{D}_2 & \mathbf{D}_3 & \ddots \\ \vdots & 0 & \mathbf{T}_2 & \mathbf{A}_3-\mathbf{T}_3-\mathbf{D}_3 & \ddots \\ \vdots & \ddots & 0 & \mathbf{T}_3 & \ddots \\ \vdots & \ddots & \ddots & 0 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}, \tag{13.2.3}$$

where the transcription and the degradation matrices, $\mathbf{T}$ and $\mathbf{D}$, respectively, are given by:

$$\mathbf{T} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & c_T & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \text{ and } \mathbf{D} = \begin{pmatrix} c_D[r] & 0 & 0 & 0 \\ 0 & c_D[r] & 0 & 0 \\ 0 & 0 & c_D[r] & 0 \\ 0 & 0 & 0 & c_D[r] \end{pmatrix}. \tag{13.2.4}$$

The production and degradation of PapI are modeled as stochastic events, such that it is possible (although with zero probability if c_T is finite and c_D is non-zero) that infinitely more PapI-production events will occur than PapI-degradation events in finite time. This suggests that the value of r must be allowed to grow unbounded, and one cannot compute an exact analytical solution as in the previous example. In this case it will be necessary to truncate $\mathbf{A}$ using the FSP algorithm.

Suppose that at time, $t = 0$, it is known that the gene is in the g_1 configuration, and there are exactly r_o molecules of PapI present in the system:

$$\mathbf{P}_{J_o}(0) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \text{ where } J_o = \begin{bmatrix} 4r_o + 1 \\ 4r_o + 2 \\ 4r_o + 3 \\ 4r_o + 4 \end{bmatrix}.$$

Then, using the FSP algorithm, if one can find a principle sub-matrix, $\mathbf{A}_{J_k}$, such that:

$$|\exp(\mathbf{A}_{J_k} t_f)\mathbf{P}_{J_k}(0)|_1 \geq 1 - \varepsilon, \tag{13.2.5}$$

then it is guaranteed that the probability density for every state at time, $t = t_f$, satisfies:

$$\left| \begin{bmatrix} \mathbf{P}_{J_k}(t_f) \\ \mathbf{P}_{J_k'}(t_f) \end{bmatrix} - \begin{bmatrix} \mathbf{P}^{FSP}_{J_k}(t_f) \\ \mathbf{0} \end{bmatrix} \right|_1 = \varepsilon. \tag{13.2.6}$$

For this problem, it is easy to choose a searching algorithm to dictate the expansion of the set J_k until the condition specified by Eqn 5.3.1 is met. The most reasonable search algorithm is to simply continue adding adjacent block structures of the form given in Eqn 13.2.3 - this corresponds to increasing the space of sets that are sequentially reachable from J_o through PapI translation and degradation events.

Tables 13.1 and 13.2 provide the reaction parameters that have have been used for this example. In this example, the total error tolerance is $\varepsilon = 10^{-6}$ for the probability density vector at time t_f. Figure 13.3 shows the lower bound on the probability density vector at the final time as computed with the FSP algorithm. In this figure, the states have been arranged according to their index as specified in Eqn 13.2.2. Recall that although inclusion of states is based upon reachability, the choice of enumeration is arbitrary, such that it is often necessary to reorder and combine states to illustrate more meaningful results. For instance, in this example one may be most interested in the distribution of the different operon states: g_1 through g_4 or the distribution of the population of PapI. Figure 13.4 shows the partial probability density vectors for the population of PapI as separated for each possible operon configuration. From the figure, one can observe that the *production* operon configuration, g_2 (top right), has a different distribution shape than do the the other states. In particular, the median population of PapI is much larger when the operon is in the g_2 configuration. In this *pap* system, the population of PapI can be related to amount of pili expression found on the bacteria, and it might not actually be interesting to know the gene configuration of the system. In this case, it is helpful to consider the distribution in the format of Figure 13.5, which shows the probability density of the total amount of PapI. For these results, the FSP required the inclusion of all values of r from zero to 30 (corresponding to a total of 124 states), and the total sum of the probability density was found to be greater than 0.999999. The results provide us a guarantee that the probability of every state (including those with more than 30 copies of r) are known within a positive error of 10^{-6}. We also have a guarantee that the error in the full probability density vector is non-negative and sums to less than 10^{-6}.

The most biologically interesting results correspond to cells in which there is a large amount of PapI; these are the cells that will actually succeed in turning ON and express Pili. For this model, define an ON cell as a cell that contains at least 20 molecules of PapI. In Figure 13.5, ON cells are all those to the right of the dashed line. From the figure one can immediately see that the probability turning ON is very low; using the FSP, this probability is guaranteed to be within the interval

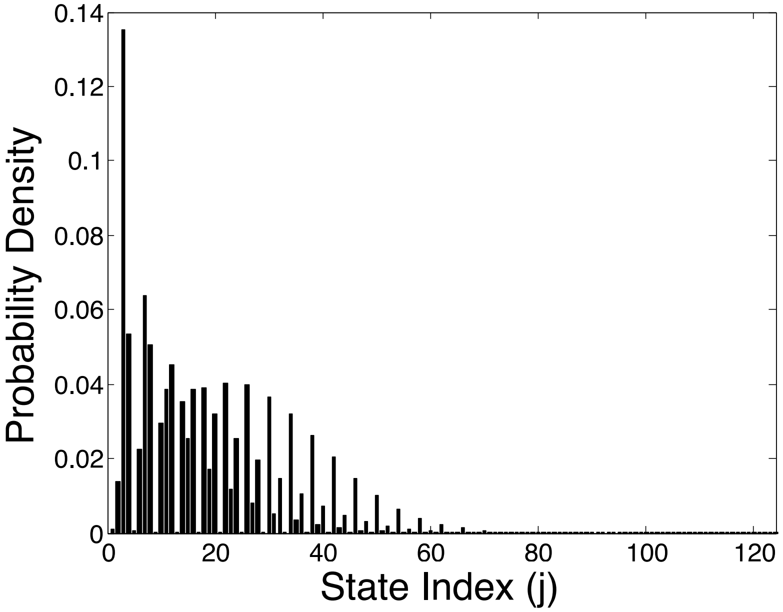


Figure 13.3: Probability density vector solution for the pap-pili model in which PapI is allowed to change through stochastic translation and degradation events. The states are ordered according to Eqn 13.2.2, and the density vector is shown as time $t_f = 10s$ for the initial condition of state $j = 21$ ($[PapI]$=5 molecules and *pap* operon in state g_1) at time $t_o = 0s$ (See also parameters and initial conditions in Tables 13.1 and 13.2).

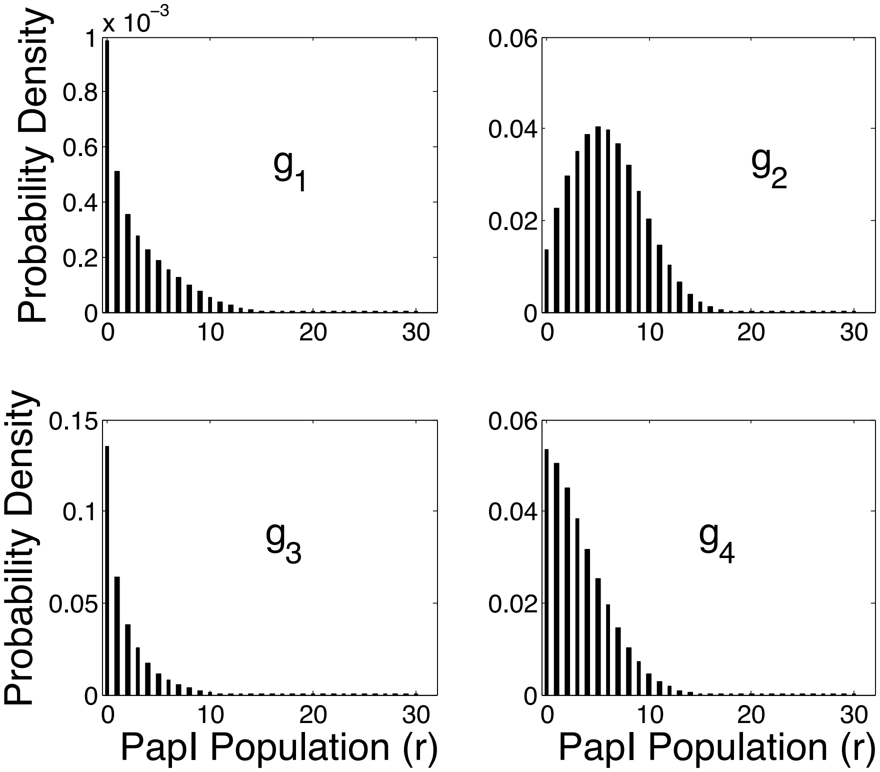


Figure 13.4: Solution for the pap-pili model in which PapI is allowed to change through stochastic translation and degradation events. The probability density vector from Fig. 13.3 is separated into four components according to when the *pap* operon is in (top-left) g_1, (top-right) g_2, (bottom-left) g_3 and (bottom-right) g_4 (See also Fig. 13.3).

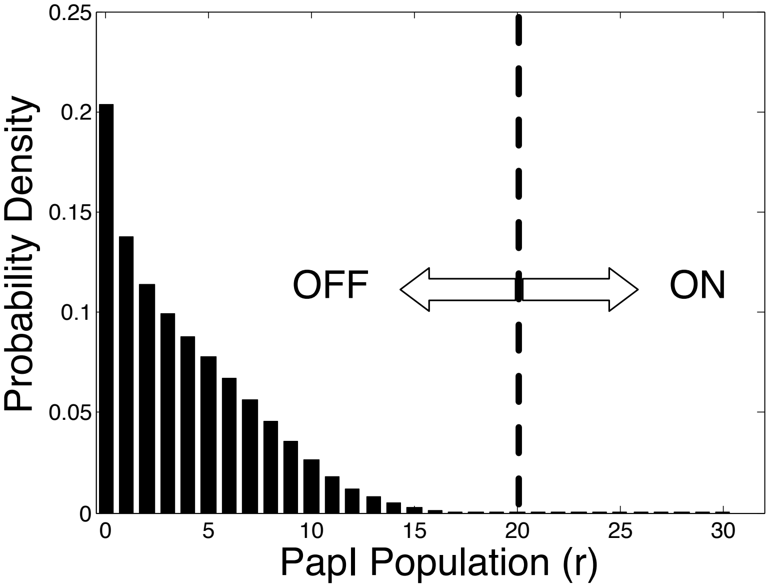


Figure 13.5: The probability density vector of the population of PapI as calculated in Example 2 at final time $t_f = 10s$. All cells that contain more than twenty molecules of PapI are considered to be **ON**. (See also Figs. 13.3 and 13.4).

$[1.376, 1.383] \times 10^{-4}$. For comparison, five sets of 10^5 SSA simulations–each with a different seed for the random number generator compute the the probability of having more than twenty molecules of PapI to be $\{1.9, 2.0, 1.1, 1.4 \text{ and } 1.3\} \times 10^{-4}$. For the five sets of 10^5 SSA simulations, the relative error ranged between -20 to +45 percent. For comparison, the relative error of the FSP is guaranteed to be in the range -0.46 to 0.00 percent (more than three orders of magnitude more precise than 10^5 simulations of the SSA). Figure 13.6 (light line) plots the average number of times the SSA produces the result that there are more than twenty molecules of PapI at time t_f as a function of the number of simulation runs. The horizontal line in the figure shows the probability as calculated using the FSP algorithm, where the thickness of the line exceeds the difference between the computed upper and lower bounds. As in the previous example, more SSA simulations allow for better accuracy at the cost of additional computational expense. For a comparison of the methods' efficiency and accuracy, Table 13.3 provides the computational time and relative error in the prediction of the *pap* OFF to ON switching rate after ten thousand, one hundred thousand, and one million simulations. From the table one can immediately see that the performance of the FSP is far superior to that of the SSA for this example.

As above, the use of time leaping methods can do little to improve the computational efficiency of the SSA for this example. In this case, negative molecular populations will always result if any Lrp binding/unbinding reaction is simulated twice consecutively before a different Lrp binding/unbinding event. In order to avoid impossible populations, therefore, one must use an adaptive step size algorithm, and no τ leap may be allowed to include more than a single reaction from the set R_1 to R_8. In the SSA simulations, more than one quarter of all of the reactions involved operon configuration changes. Therefore, if we make the liberal assumptions that a single τ leap step is as fast as a single SSA step, and that there is exactly one R_1 to R_8 reaction included in each τ leap, then a τ leaping method can boost the speed of the SSA by a maximum factor of less than four. It must be mentioned, however, that PapI production and degradation reactions can also result in excessively large changes in propensity functions, thus further restricting the size of allowable time leaps. In practice τ leap steps may take far longer to compute than individual SSA steps, and one would expect that τ leaping will provide far less benefit over the SSA in this example. As in the previous example, it does not matter what type of τ leaping is chosen (Poisson or Binomial); the leap size will be similarly restricted in each. As an example of the failure of τ leaping to handle this example, we have again utilized `StochKit` [80], and we have set the

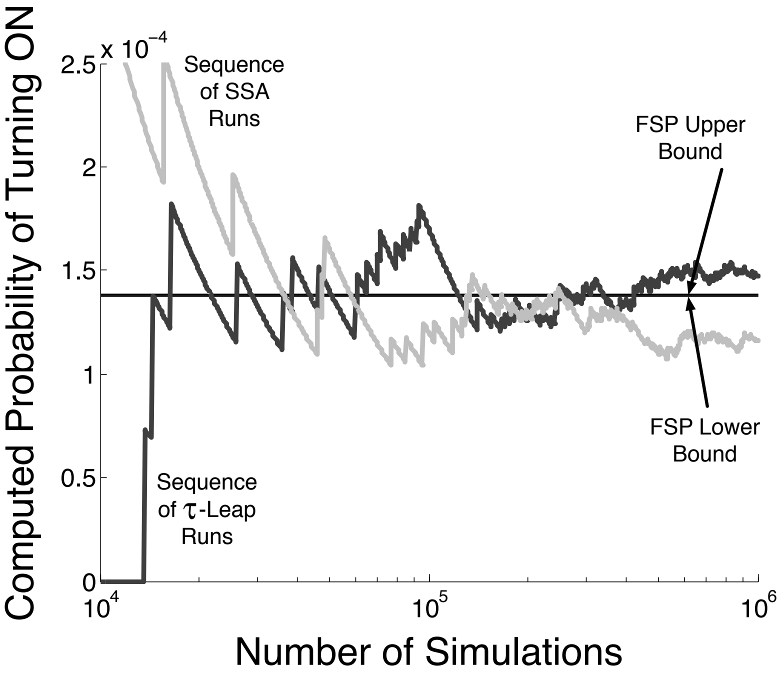


Figure 13.6: Predictions of **OFF** to **ON** switching rate using the SSA (light grey) and an explicit, adaptive step size τ leaping algorithm (dark grey) from StochKit. The bottom axis shows the number of conducted simulations, and the top axis shows the computational time required for that number of simulations. As the number of simulations increases, the computed probability converges toward the more precise FSP solution. For comparison, the thickness of the horizontal line corresponds to the upper and lower bounds on the switch rate as computed in less than four seconds using the FSP algorithm.

program to use an adaptive explicit τ leaping algorithm [17]. For this algorithm, computation took about fourteen seconds for 10^5 runs (the same as the direct step SSA), and the accuracy was similar to that of the SSA. The dark grey line in Figure 13.6 illustrates the convergence of the τ leaping predictions as more and more simulations have been conducted (see also Table 13.3).

Reactions				
	Number	Stoichiometry	Rate Constant($c_{/mu}$)	Units
	R_T	$\mathbf{X}_2 \rightarrow \mathbf{X}_2 + r$	1000	s^{-1}
	R_D	$r \rightarrow \emptyset$	100	s^{-1}

Parameters and Initial Conditions			
	Parameter	Notation	Value
	Initial Catalyst Protein	r_o	5
	Initial *pap* operon State	g_1	–
	Initial State	$j_o = 4r_o + 1$	21
	Initial *pdv*	$\mathbf{P}(\mathbf{X}_{J_o}; 0) = 1$	–
	Allowable Error in *pdv*	ϵ	10^{-3}

Table 13.2: Reactions and parameters used in the SSA and exact FSP solutions for the Pap-Pili epigenetic switch in which the population of the regulatory protein PapI may change according to stochastic translation and degradation events. See also Table 13.1.

Method	# Simulations	Time (s)	Relative Error in switch rate
FSP	Does not apply.	< 4	$< 0.5\%$
SSA	10^4	≈ 1.4	100%
SSA	10^5	≈ 14	23%
SSA	10^6	≈ 140	6.8%
τ **leaping**	10^4	≈ 1.4	118%
τ **leaping**	10^5	≈ 14	13%
τ **leaping**	10^6	≈ 140	16%

Table 13.3: A comparison of the efficiency and accuracy of the FSP, SSA, and adaptive explicit τ leaping methods for the prediction of the *pap* OFF to ON switching rate. Using the FSP, it takes less than four seconds to guarantee that the OFF to ON switch rate is within the interval $[1.376, 1.383] \times 10^{-4}$, a relative error of less than 0.5 percent. The table shows the results of a single set of 10^6 statistically independent simulations for each the SSA and the τ leaping methods. The relative errors have been calculated after 10^4, 10^5, and 10^6 simulations. Simulation sets with different random number generator seed values will produce different results (some are better and some are worse–results not shown). In contrast, every run of the FSP algorithm always produces the exact same result. All codes are run on the same 1.50GHz Intel Pentium 4 processor running a Linux environment. See also Figure 13.6.

Chapter 14

Applying the FSP to the Toy Heat Shock Model

When a cell's environment changes, that cell must either adapt or perish. As a result, biological systems have evolved many intricate mechanisms to deal with the frequent changes that occur in complex environments. One particular such system that has received a lot of recent attention is the cellular heat shock response in *E. coli*. At higher than normal temperatures, cellular proteins often fold incorrectly, and are no longer able to perform their functions. In order to survive, the cell avoids this outcome by producing molecular chaperones and proteases, which refold denatured proteins and degrade irreversibly aggregated proteins. At the heart of the heat shock response mechanism in *E. coli* is the formation of the σ_{32}-RNAP complex [25], shown in Fig. 14.1. Here a simplified model for σ_{32}-RNAP formation illustrates how one can combine the reduction methods in Chapters 7, 9 and 10 to significantly increase the power of the FSP algorithm.

The simple Heat Shock regulatory mechanism is comprised of three reactions,

$$s_1 \rightleftharpoons s_2 \rightarrow s_3, \qquad (14.0.1)$$

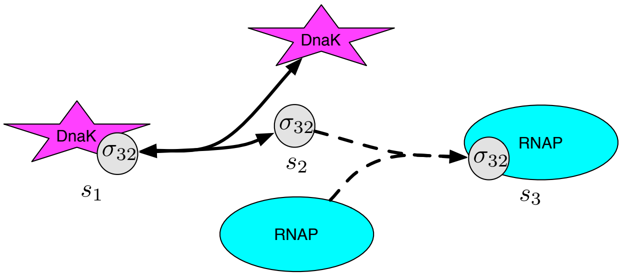


Figure 14.1: Schematic representation of the Toy Heat shock model [25]. Species s_1, s_2 and s_3 represent the σ_{32}-DnaK complex, free σ_{32} and the σ_{32}-RNAP complexes, respectively. In the model the free amounts of DnaK and RNAP are assumed constant and are lumped into the reaction rates. The solid arrows correspond to rapid binding and unbinding of DnaK and free σ_{32}. The dashed lines correspond to infrequent irreversible binding of RNAP and σ_{32}.

where s_1, s_2 and s_3 correspond to the σ_{32}-DnaK complex, the σ_{32} heat shock regulator and the σ_{32}-RNAP complex, respectively. For this model, the propensity functions of the three reactions are given by

$$a_1 = c_1[s_1],\ a_2 = c_2[s_2],\text{ and } a_3 = c_3[s_2], \tag{14.0.2}$$

where the bracket notation [.] refers to the integer population of the enclosed species. This model of the heat shock subsystem has been analyzed before using various computational methods including Monte Carlo implementations [15, 24, 78, 71, 69].

Typically, the relative rates of the reactions are such that the reaction from s_2 to s_1 is by far the fastest ($a_2 \gg a_1$ and $a_2 \gg a_3$), and σ_{32} molecules infrequently escape from DnaK long enough to form the σ_{32}-RNAP complex. The purpose of this mechanism is to strike a balance between fixing the damage produced by heat and saving the cell's resources, as a significant portion of cell energy is consumed when producing heat shock proteins. For this example, the parameters are set as

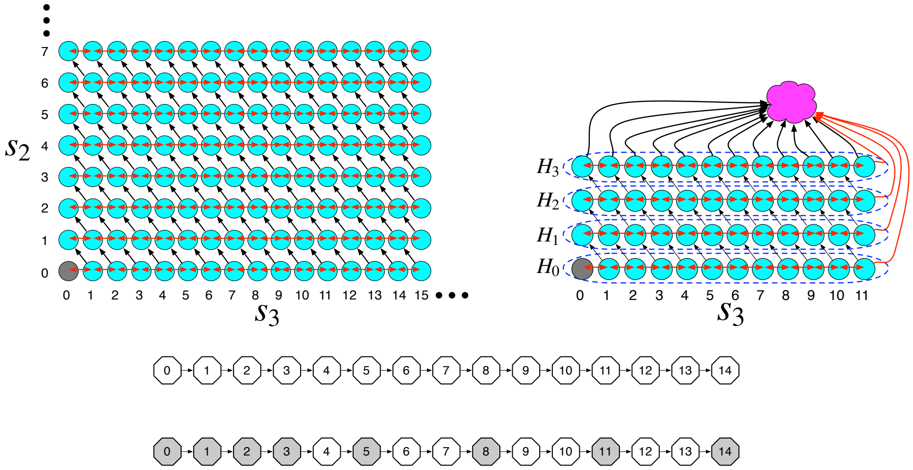


Figure 14.2: (a) Two dimensional integer lattice representing possible configurations of the toy heat shock model. Here s_2 and s_3 are populations of free σ_{32} molecules and σ_{32}-RNAP compounds, respectively, while s_1 is the population of σ_{32}-DnaK compounds. Reactions $s_1 \rightleftharpoons s_2$, are represented by bidirectional horizontal arrows and reactions $s_2 \rightarrow s_3$ is represented with diagonal arrows. The total number of σ_{32} is constant, so the chemical state of the system is uniquely defined by s_2 and s_3 alone. (b) The same lattice after applying the finite state projection. Unlikely states have been aggregated into a single sink state. Each horizontal row of configurations is separated from the rest by the slow reaction 3 and then is used to form the fast block generator $\mathbf{H}_i$. (c) The slow manifold FSP that is found by projecting the dynamics of each fast interconnected set onto its equilibrium distribution. In this new system, each fast interconnected configuration subset is represented by a single node. (d) Applying the interpolation-based projection to the Slow Manifold FSP system. Here the shaded nodes are interpolation points on which the dynamics are now to be approximated.

follows:

$$c_1 = 10, \;\; c_2 = 4 \times 10^4, \;\; c_3 = 2,$$
$$s_1(0) = 2000, \;\; s_2(0) = s_3(0) = 0.$$

With only the reactions above, the total number of σ_{32}–free or in compounds–is constant, so that $s_1 + s_2 + s_3 = K$. With this constraint, the reachable states of this three species problem can be represented on a two dimensional lattice as shown in Fig. 14.2a. The goal in this study is to find the probability distribution of the population of s_3 at time $t_f = 300s$, and seven different methods have been used to achieve this goal: (1) the original finite state projection method (FSP), (2) the FSP with a multiple time step algorithm (MTI-FSP) [69], (3) the FSP with the slow manifold assumption (SM-FSP), (4) the FSP with the interpolation based reduction (I-FSP), (5) the FSP with first the slow manifold assumption and then an interpolation based reduction (SM/I-FSP), (6) the basic stochastic simulation algorithm (SSA), and (7) the SSA with the slow manifold approximation (SM-SSA). Fig. 14.4 shows the distribution as computed with each of these methods, and Table 14.1 summarizes the efficiency and accuracy of each method. First however, it is useful to describe the full CME.

Method 0: **Full CME**. The full master equation for this problem can be arranged using the following enumeration scheme;

$$i(s_1, s_2, s_3) = s_3(K + 1) + s_2 + 1,$$

which has the inverse

$$\mathbf{x}_i = \begin{bmatrix} s_1(i) \\ s_2(i) \\ s_3(i) \end{bmatrix} = \begin{bmatrix} K - \text{mod}(i-1, K+1) - \text{floor}((i-1)/(K+1)) \\ \text{mod}(i-1, K+1) \\ \text{floor}((i-1)/(K+1)) \end{bmatrix},$$

where $\text{mod}(x, y)$ is the remainder after dividing x by y, and $\text{floor}(x)$ rounds x down to the nearest integer. With this enumeration scheme and the propensity functions and stoichiometry from above, one can form the the infinitesimal generator $\mathbf{A}$ as:

$$\mathbf{A}_{ij} = \left\{ \begin{array}{cc} -c_1 s_1(i) - c_2 s_2(i) - c_3 s_2(i) & \text{for } (i = j) \\ c_1 s_1(j) & \text{for } j \text{ s.t. } \mathbf{x}_i = \mathbf{x}_j + [-1, 1, 0]^T \\ c_2 s_2(j) & \text{for } j \text{ s.t. } \mathbf{x}_i = \mathbf{x}_j + [1, -1, 0]^T \\ c_3 s_2(j) & \text{for } j \text{ s.t. } \mathbf{x}_i = \mathbf{x}_j + [0, -1, 1]^T \\ 0 & \text{Otherwise} \end{array} \right\}. \tag{14.0.3}$$

For the initial conditions above, the reachable configuration set is the set of all configurations such that $s_1 + s_2 + s_3 = K$. For $K = 2000$, one can show that the number of points in this set is

$$\sum_{s_3=0}^{K} \sum_{s_2=0}^{K-s_3} 1 = \sum_{s_3=0}^{K} s_3 = 2,001,000,$$

and therefore, the full CME is too large to be solved exactly, and an approximation is necessary.

Method 1: **FSP**. Applying the original Finite State Projection method (see Chapter 5) allows one to significantly reduce the order of the problem and achieve a manageable solution at least for small time intervals ($t \leq 300s$). With a projection such that that $s_2 \leq 12$ and $s_3 \leq 342$, the FSP solutions provides an accuracy guarantee of 3.0×10^{-5}, and takes 750 seconds.

Method 2: **MTI-FSP**. By separating the time interval into 150 equal time intervals as described in Chapter 9 and [69], one can significantly improve the efficiency of the FSP for this system. Rather than compute a 4459^{th} order matrix exponential, one can instead acquire a solution by computing 70 different matrix exponentials each of which is 195^{th} order or smaller. The total required time of this approach was 40.2 seconds, and the maximum error is guaranteed to be less than 1.7×10^{-4}.

While the accuracy of the multiple time interval FSP is guaranteed, the efficiency of the algorithm depends upon the chosen interval size. Fig. 14.3 illustrates some of the subtleties of this tradeoff by plotting the size of the largest exponentiated matrix, the number of matrix exponentials, and the computational time all as functions of the number of time intervals (bottom axis) and the interval length (top axis). As we use more time intervals, the probability distribution has less time to disperse between one interval and the next, and the required matrix exponentials are smaller as shown in Fig. 14.3a. However, because the matrix dimension is a discrete integer quantity, this decrease is stepwise rather than smooth, and a large range of interval lengths may require the same matrix size. If an interval length is at the low end of that range, the matrix exponentials required to get each $\mathbf{E}_i$ are often slightly more precise than is absolutely necessary, and are therefore more likely to provide other $\mathbf{E}_j$'s as well–fewer exponential computations are necessary. Conversely, if an interval length is at the high end of the range for a given matrix size, fewer $\mathbf{E}_j$'s will come from each exponential computation–more exponential computations are necessary. This trend is clear when one compares Fig. 14.3a to 14.3b.

In order to show how these concerns affect the computation, we have broken the total computational cost in Fig. 14.3c into three components. The first cost is that of computing the matrix exponentials; the second cost is the combined cost of storing the vectors $\{\mathbf{E}_i\}$ and then updating the solution from one interval to the next; and the third cost is the cost of initializing the first projection set with a set of 20 SSA runs. For $t_f = 300s$, this tradeoff is optimized for 360 time intervals corresponding to a interval length of $\tau \approx 0.83s$. To obtain the solution with this time interval, the algorithm needed to compute 122 matrix exponentials of size 121×121 or smaller, and the computation takes about 31.4s.

Method 3: **SM-FSP**. In the Heat Shock model, the first two reactions, $s_1 \rightarrow s_2$ and $s_2 \rightarrow s_2$, are much faster than the third. These fast reactions are used to define sets of fast interconnected configurations, as described in Chapter 7. In Fig. 14.2b, these sets are the horizontal rows of configurations. Using the slow manifold projection discussed above and in [78, 71], each fast interconnected configuration set can be collapsed to a single point to form the 1D lattice Markov chain shown in Fig. 14.2c. The reduced problem can now be solved as a system of only 343 ODEs, which takes only 0.94s to solve including reduction time.

Method 4: **I-FSP**. The interpolation based solution to the FSP (see Chapter 10) uses a sparse grid where s_2 is in the set {0,1,2,3,4,5,6,7,8,10,12} and s_3 is in the set {0,1,2,3,5,8,11,14,{14+8n)}}. This enables the reduction of the 4459^{th} order FSP solution to a set of 539 ODEs. The reduced problem takes 6.1s to compute (including reduction time), and provides a solution with a maximum error of 7.7×10^{-4}. The choice of interpolation points may not have provided the best possible reduction; better choices in terms of accuracy and efficiency may exist and are left as a topic for further research.

Method 5: **SM/I-FSP**. For a fifth solution scheme, the interpolation based reduction of method 4 is combined with the slow manifold model of method 3. From the reduced 1D lattice in Fig. 14.2c, this reduced model includes only the grid points where s_3 is in the set {0,1,2,3,5,8,11,14,{14+8n}}. The resulting reduced system contains only 49 ODEs and takes less that 0.04 seconds to solve (after the reduction), but its results are nearly indistinguishable from the full system in that the maximum error is only 8.2×10^{-4}.

Method 6: **SSA**. The sixth method to generate the probability distribution is Gillespie's stochastic simulation algorithm [35]. A single run of the SSA takes about 20 seconds to complete. 10^4 simulations would take over 50 hours and have not been computed for this report.

Method 7: **SM-SSA**. As discussed above, the toy heat shock model exhibits two significantly different time scales. Therefore, in addition to being an excellent candidate for the analytical FSP-SM method, the heat model is also amenable to Monte Carlo algorithms that utilize the same time scale separation reduction. One such approach is to reduce the system to its slow manifold as in method 3, but then use the SSA. This method, for which we use the acronym SSA-SM is very similar to the methods in [82, 43, 15]. This SSA-SM takes only 0.1s per run and is 170 times faster than the original SSA, but it still requires many realizations before the solution to the CME will sufficiently converge. A set of 10^3 runs take 84 seconds to compute and yields a maximum error of about 0.012. By increasing the number of runs by a factor of one hundred, this implementation takes 100 times longer and yields an error less than ten times better as summarized in Table 14.1.

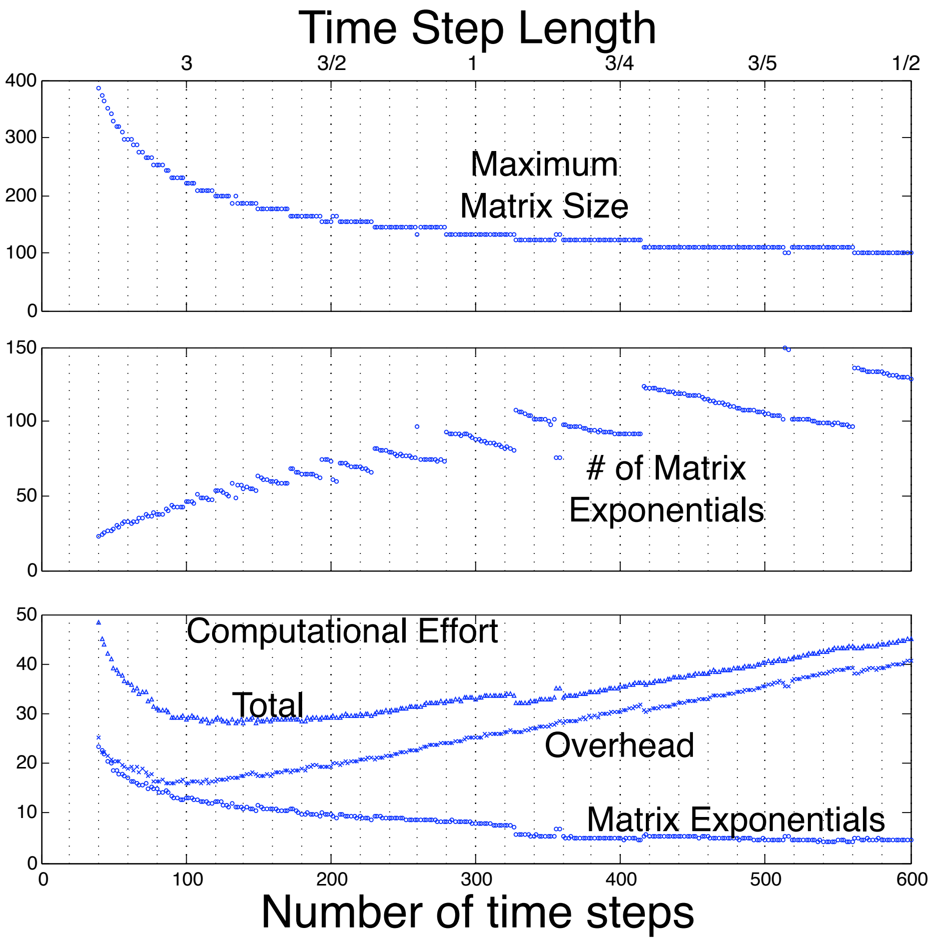


Figure 14.3: Trade off between more and fewer time intervals in the Multiple Time Interval FSP (MTI-FSP) algorithm solution for the toy heat shock model at a final time of $t_f = 300s$. The following are plotted as function of the number of intervals: (top) the size of the largest required matrix exponential computation, (middle) the number of matrix exponential computations performed, (bottom) the computational time required for the MTI-FSP algorithm split into three components: the smallest is the cost of using 20 SSA to initialize the projection for the first time interval, the next smallest is total cost of computing matrix exponentials, and the largest is the remaining overhead costs (primarily data storage and retrieval). All computations have been performed in Matlab 7.2 on a Dual 2 Ghz PowerPC G5.

Longer Time Intervals. If we were to consider longer time intervals for the toy heat shock model, the size of the projection would also need to increase. For an interval of 1200s, one needs to include every configuration such that $s_2 \leq 12$ and $s_3 \leq 1022$. This includes over 13000 configurations with one ODE for each. While computing a system of that size is often possible using Krylov based solutions such as Roger Sidje's `expokit` [90], it is beyond the capabilities of the chosen software (`Matlab`'s `expm(.)` routine), especially when there is significant numerical stiffness in the ODE's. In this case the reduced solutions are not only beneficial, they are necessary. Fig. 14.4(b) shows the distribution of the number of s_3 molecules as computed with the various FSP reduction schemes. In the interpolation-based FSP reductions, a slightly coarser mesh is applied, which includes all configurations where s_3 is in the set {0,1,2,3,5,8,11,14,{14+12n}}. Once again, all FSP based methods provide results that are virtually indistinguishable from the true solution, but they reach these results in far less time. In particular, the reduced model formed by projecting the system onto its slow manifold and then performing the interpolation-based projection results in a model of only 92 ODEs which takes less than one tenth of a second to solve.

For final time $t_f = 300s$				
Method	Matrix Size	J_{solve}	J_{total}	∞-norm Error
FSP	4459	750s	750s	$< 3.0 \times 10^{-5}$
MTI-FSP	195	-	40.2s	$< 1.68 \times 10^{-4}$
SM-FSP	343	0.25s	0.94s	$\approx 5.1 \times 10^{-4}$
I-FSP	539	5.1s	6.1s	$\approx 7.7 \times 10^{-4}$
SM/I-FSP	49	0.04s	0.78s	$\approx 8.2 \times 10^{-4}$
10^4 SSA	Results would take more than 55 hours.			
10^3 SM-SSA	-	-	84.1s	≈ 0.0116
10^4 SM-SSA	-	-	925s	$\approx 3.4 \times 10^{-3}$
10^5 SM-SSA	-	-	9360s	$\approx 1.6 \times 10^{-3}$

For final time $t_f = 1200s$				
Method	Matrix Size	J_{ODE}	J_{total}	∞-norm Error
FSP	13274	Exceeds machine capabilities		
MTI-FSP	325	-	253s	$< 1.2 \times 10^{-4}$
SM-FSP	1023	4.66s	10.66s	$\approx 1.2 \times 10^{-4}$
I-FSP	1012	40.5s	44.6s	$\approx 6.1 \times 10^{-4}$
SM/I-FSP	92	0.09s	6.19s	$\approx 5.7 \times 10^{-4}$
10^4 SSA	Results would take more than 180 hours.			
10^3 SM-SSA	-	-	272s	$\approx 9.9 \times 10^{-3}$
10^4 SM-SSA	-	-	3000s	$\approx 3.5 \times 10^{-3}$
10^5 SM-SSA	-	-	2.99×10^4s	$\approx 1.2 \times 10^{-3}$

Table 14.1: Comparison of the computational efficiency and accuracy of various solutions of the chemical master equation for the the Toy Heat Shock model.

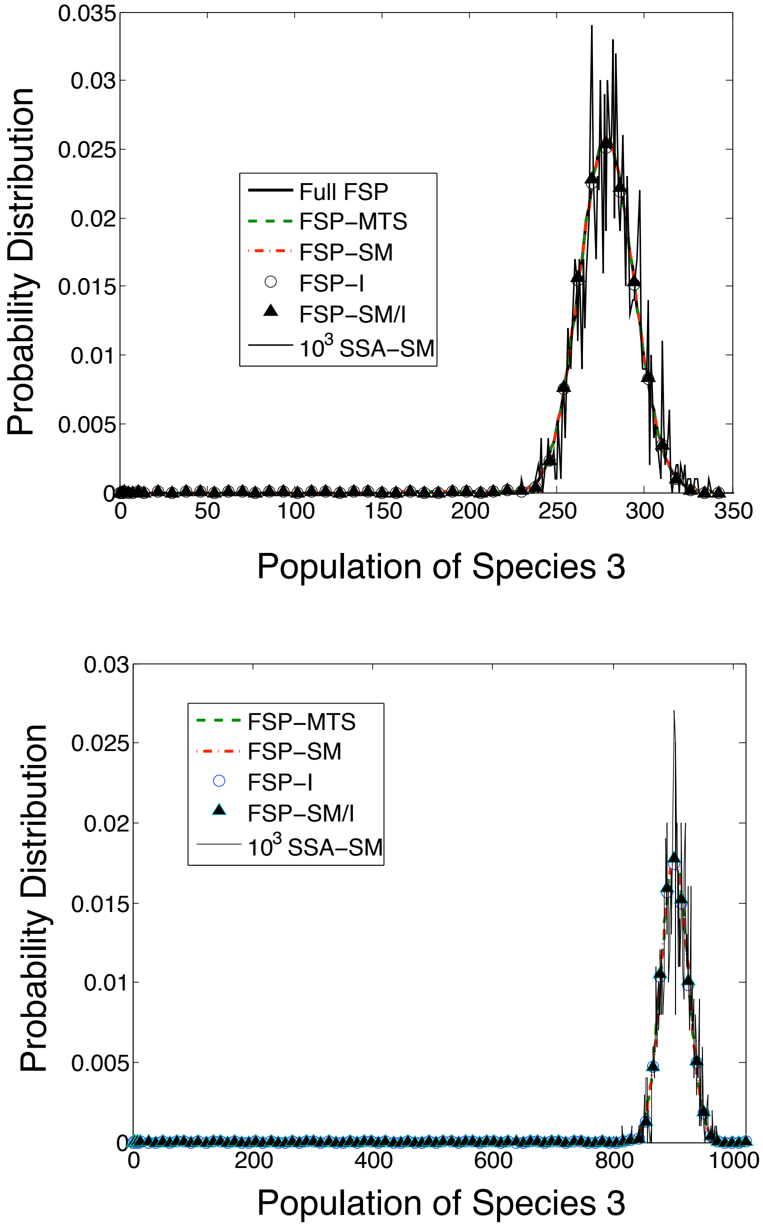


Figure 14.4: The probability distribution of the amount of the σ-RNAP complex formed at (a) $t_f = 300s$ and (b) $t_f = 1200s$ as computed using the toy heat shock model. Different analytical reductions have been applied to the chemical master equation, and each provides results that are virtually indistinguishable from the full FSP solution. See also Table 14.1.

Chapter 15

Applying the FSP to the Genetic Toggle Switch

One of the most important genetic regulatory problems is that of stochastic switching. Two identical cells within the same environment and with the same initial conditions may express wildly different phenotypes; a few such examples include the *pap* (pili) regulatory switch in *E. coli* [66] as well as cell fate decisions in developing organisms. Alternatively, a cell may switch from one state to another as has been shown in previous stochastic models of the lysis-lysogeny decision of phage lambda [6]. This chapter considers a simple stochastic version of the genetic toggle system constructed and presented by Gardner, Cantor and Collins [31]. Fig. 15.1 illustrates this genetic regulatory system, which is comprised of two promoters each of whose products inhibits the other promoter. The signals of the network are the populations of the two repressors, s_1 and s_2. These repressors react according to the simple production and degradation reactions:

$$\emptyset \rightleftharpoons s_1 \qquad \emptyset \rightleftharpoons s_2,$$

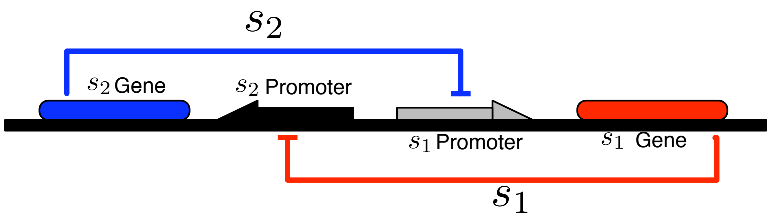


Figure 15.1: Schematic of the toggle model comprised of two inhibitors: s_1 inhibits the production of s_2 and vice-versa.

where the degradation rates (left arrows) of s_1 and s_2 are δ_1 and δ_2, respectively, and the synthesis rates (right arrows) of s_1 and s_2 depend upon the populations $[s_2]$ and $[s_1]$, respectively, and are given by:

$$\alpha([s_2]) = \frac{\alpha_1}{1+[s_2]^\beta}, \text{ and } \beta([s_1]) = \frac{\alpha_2}{1+[s_1]^\gamma},$$

respectively.

For the first analysis of the toggle switch, consider the following set of parameters:

$$\delta_1 = \delta_2 = \gamma = 1, \alpha_1 = 25, \alpha_2 = 30, \beta = 1, \tag{15.0.1}$$

and the initial condition of zero for both species s_1 and s_2. Three methods have been used to find the probability distribution at the time $t_f = 10^4 s$: (1) the finite state projection method (FSP), (2) the interpolation-reduced FSP (I-FSP), and (3) the stochastic simulation algorithm (SSA). Other methods have been considered, but an initial examination of the system presents no clear separation between time scales, so the slow manifold based reductions (either for the FSP or for KMC algorithms) have not be considered. For each method, the accuracy is measured by the maximum error in the computed distribution, and the efficiency is measured by two costs: J_{ODE}: the time required to compute the solution after the system reduction, and J_{total} the total time required to find and solve the reduced system. Each of these costs can be important in different situations: J_{total} is the stand-alone cost of solving this problem only once, and J_{ODE} represents the repetitive cost of solving the system as part of larger more complex problem. With these metrics accuracy and efficiency of these different methods is discussed below and summarized in Table 15.1.

Method 1: **FSP**. In order to use the original FSP method, one must first choose a configuration subset on which to perform the projection. Fig. 15.2 illustrates one such set chosen to include all configurations such that $[s_1] \leq 64$, $[s_2] \leq 88$, and $[s_1][s_2] \leq 220$. For this configuration subset, the finite state projection of the CME is comprised of 1014 configurations, which took about 8 seconds to solve. Fig. 15.3(a) provides a contour map of the distribution for the full FSP approach for this first parameter set; Fig. 15.4(a,b) show the same probability distributions for the populations of s_1 and s_2, respectively. In terms of accuracy, the full FSP implementation yielded a maximum error less than 5.3×10^{-5}.

Method 2: **I-FSP**. Using the methodology in Chapter 10, one can project the finite state system from the previous method onto a grid defined by integers distributed as follows. The first 8 are separated by one point: $\{0, 1, \ldots, 7\}$; the next 8 are separated by two points: $\{8, 12, \ldots, 22\}$; the next 8 by four points: $\{24, 28, \ldots, 52\}$; and the remaining points are separated by eight points until the maximum value is reached. Each of these grid points is illustrated in Fig. 15.2 by a single dot. Figures 15.3(b) and 15.4(a,b) show the distribution contours as computed using the interpolation-based model reduction approach. From these figures, one can see that there is very little observable difference between the full FSP results and the interpolation-reduced FSP results. However, the

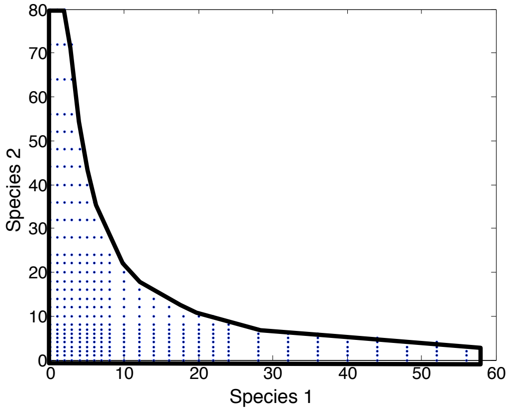


Figure 15.2: Projection used for the genetic toggle model for parameter set (15.0.1). Here the finite state projection is first applied to remove the unlikely configurations and reduce the system to a set of 1014 ODEs, then the distribution of the remaining configurations is projected to a lower dimensional space for a set of 353 ODEs. The probability density is approximated assuming linear dynamics for the distribution at each point, and the full distribution is approximated via interpolation.

interpolation-based approach required solving about a third as many ODEs and took less than a quarter of the time to complete, including the model reduction. The interpolation-based reduction had a maximum error of about 2.9×10^{-4}.

Method 3: **SSA**. For comparison with a typical Monte Carlo algorithm, the SSA [35] has also been run. After 10^3 simulations of the SSA, the total computational time was almost two hours, and the maximum error was about 30 times greater than that of the other methods. While approximations to the SSA, such as τ leaping, may significantly speed up the computational time, they can do little to improve the accuracy of solution.

Different Parameter Sets. To extend the comparison of the three methods, two additional parameter sets have been considered:

$$\delta_1 = \delta_2 = \gamma = 1, \alpha_1 = 80, \alpha_2 = 100, \beta = 1, \qquad (15.0.2)$$

and

$$\delta_1 = \delta_2 = \gamma = 1, \alpha_1 = 100, \alpha_2 = 25, \beta = 2.5, \qquad (15.0.3)$$

which are more computationally difficult to solve (for all methods) because more reactions occur, and the system tends to reach a larger portion of the configuration set. For parameter set (15.0.2), the chosen FSP solution includes all configurations such that $[s_1] \leq 120$, $[s_2] \leq 200$ and $[s_1][s_2] \leq 700$. For the original FSP algorithm this requires solving an 3340^{th} order ODE and takes 288s to compute, but with the grid described for the previous set of parameters, the system is reduced to 665 ODEs and took only eight seconds to compute. Comparable improvements were also found for the third set of parameters for which the FSP solution includes every configuration such that $[s_1] \leq 176$, $[s_2] \leq 96$ and $[s_1][s_2] \leq 500$. Fig. 15.5 shows the probability distribution of species s_2 for at the time $10^4 s$ for both parameter sets (15.0.2) and (15.0.3) as computed with the FSP and the interpolation reduced FSP methods, and Table 15.1 summarizes the accuracy and efficiency of the same implementations. Once again there is very good agreement between the two solutions. For parameter sets (15.0.2) and (15.0.3), single runs of the SSA took 20.8 and 17.4 seconds, respectively. At these rates, 10^4 simulations take two days for each set yet and yield ten times worse convergence than any of the other methods.

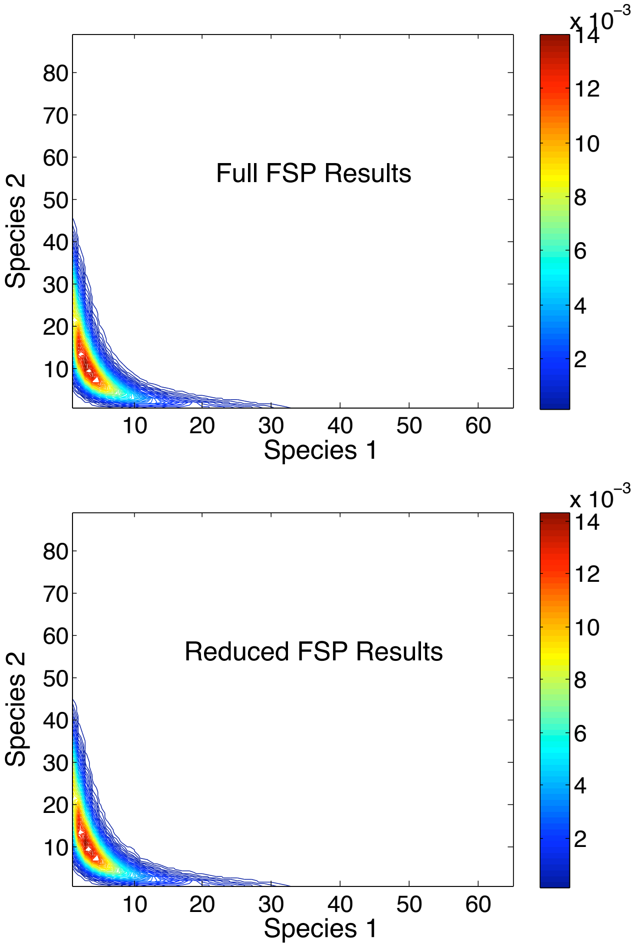


Figure 15.3: Contour plots of the probability distributions of the reachable configurations of the Genetic toggle regulatory network with parameters in (15.0.1). We begin with a known initial condition of $s_1 = s_2 = 0$ and compute the distributions at $t_f = 10^4 s$. Computations have been made using two analytical solution techniques: (a) The full, original FSP implementation. (b) The interpolation-based reduction of the FSP.

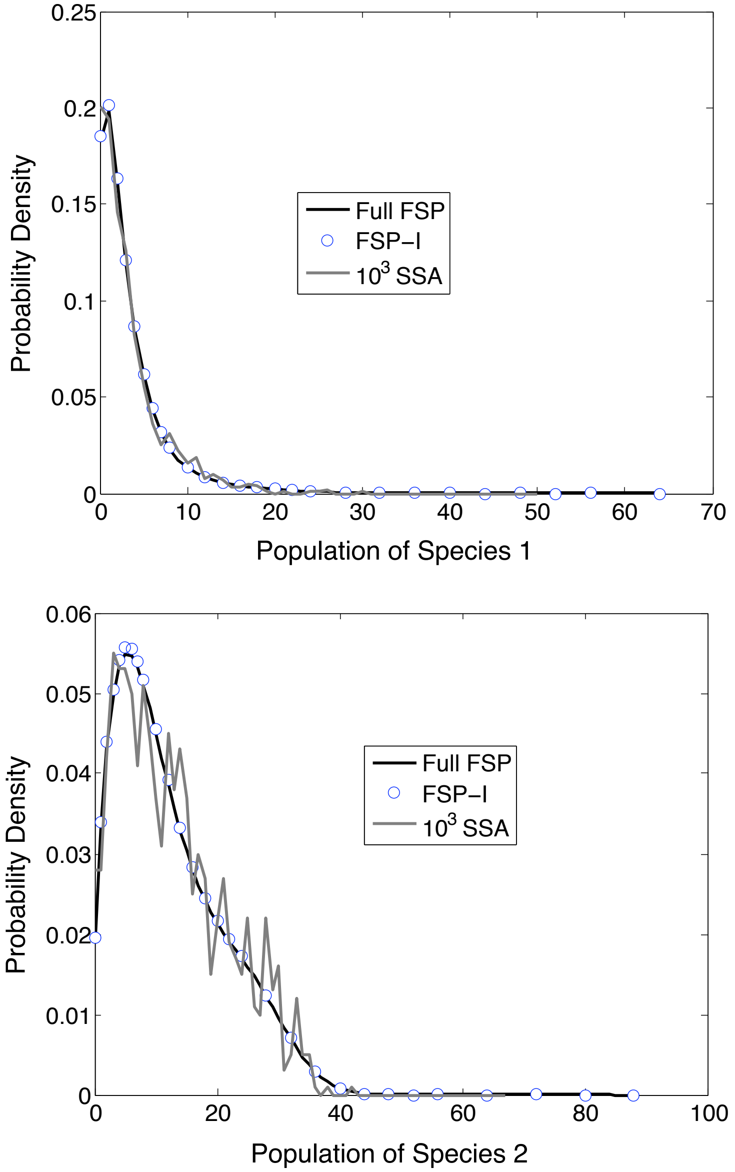


Figure 15.4: An alternate representation of the probability distributions of the configurations reachable in the genetic toggle model with parameters in (15.0.1) (see also Fig. 15.3). (a) The probability density for species 1, (b) The probability density of species 2. The density as computed with the full FSP equations is represented by the smooth line, and the density as computed with the interpolation based reduction approach is represented with circles, and the density as computed with 10^4 runs of the SSA is given by the jagged line.

For Parameters (15.0.1) and final time $t_f = 10^4 s$				
Method	Matrix Size	J_{solve}	J_{total}	∞-norm Error
FSP	1014	7.27s	7.41s	$\leq 5.3 \times 10^{-5}$
FSP-I	353	0.89s	1.40s	$\approx 2.9 \times 10^{-4}$
SSA (10^3)	-	-	$6920s$	$\approx 8.9 \times 10^{-3}$
SSA (10^4)	-	-	$7.1 \times 10^4 s$	$\approx 3.4 \times 10^{-3}$

For Parameters (15.0.2) and final time $t_f = 10^4 s$				
Method	Matrix Size	J_{solve}	J_{total}	∞-norm Error
FSP	3340	287s	288s	$< 6.4 \times 10^{-5}$
FSP-I	665	7.56s	8.75s	$\approx 8.1 \times 10^{-5}$
SSA (10^4)	-	-	$2.2 \times 10^5 s$	$\approx 2.9 \times 10^{-3}$

For Parameters (15.0.3) and final time $t_f = 10^4 s$				
Method	Matrix Size	J_{solve}	J_{total}	∞-norm Error
FSP	2404	93.2s	93.8s	$< 6.5 \times 10^{-6}$
FSP-I	556	4.09s	5.02s	8.5×10^{-4}
SSA (10^4)	-	-	$1.7 \times 10^5 s$	5.6×10^{-3}

Table 15.1: Comparison of the computational efficiency and accuracy of three different solutions of the chemical master equation for the stochastic genetic toggle model. For our analysis, we begin at a known initial condition of $s_1 = s_2 = 0$ and compute the distributions at $t_f = 10^4 s$. Two computational costs are given: J_{ODE}–the time required to solve the reduced system of ODEs and J_{total}–the total time required to reduce and solve the system. See also Figs. 15.3-15.5.

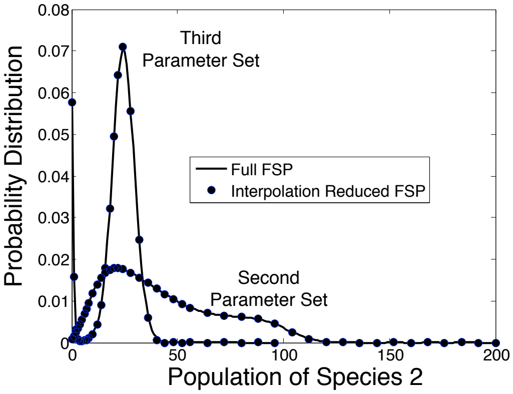


Figure 15.5: The probability distribution of the population of s_2 molecules in the genetic toggle model for parameter sets (15.0.2). The densities as computed with the full FSP equations are represented by the solid lines, and the densities computed with the interpolation based reduction approach are represented with circles.

15.1 FSP Switch Rate Analysis

This section examines the usefulness of the FSP absorbing sink in the analysis of switching behavior of Gardner's gene toggle model [31] with the parameter set:

$$\delta_1 = \delta_2 = \gamma = 1, \alpha_1 = 16, \alpha_2 = 30, \beta = 2.5. \tag{15.1.1}$$

For these parameters, the system exhibits two distinct phenotypes: Define the cell to be OFF when the population of s_1 exceeds 5 molecules and s_2 is less than 15 molecules, and ON when the population of s_2 exceeds 15 molecules and s_1 is less than 5 molecules. Each of these phenotypes is relatively stable–once the system reaches the ON or OFF state, it tends to stay there for some time. For this study, the system begins with a population $s_1 = s_2 = 0$, and the goal is to analyze the subsequent switching behavior.

Q1. *After the process starts, the system will move within its configuration space until eventually the cell turns OFF or the cell turns ON. What percentage will choose to turn ON first (s_2 exceeds 15* before *s_1 exceeds 5)?*

One can use the methodology outlined in Section 11.1 to analyze this initial switch decision. Let $\mathbf{X}_J$ include all states such that $s_1 \leq 5$ and $s_2 \leq 15$. There are only two means through which the system may exit this region: If $s_1 = 5$ and R_1 occurs (making $s_1 = 6$), then the system is absorbed into a sink state G_{OFF}. If $s_2 = 15$ and R_3 occurs, then the system is absorbed into a sink state G_{ON}. The master equation for this Markov chain has the form of that in (11.1.1) and contains 98 states including the two absorbing sinks. By solving this equation for the given initial condition, one can show that the probability of turning ON first is 78.1978%. Thus, nearly four-fifths of the cells will turn ON before they turn OFF. The asymptotes of the dashed lines in Fig. 15.6b correspond to the probabilities of that the system will first turn ON and OFF, respectively.

Q2. *Find the times t_{50} and t_{99} at which 50% and 99% of all cells will have made their initial decision to turn ON or OFF?*

To solve this question, one can use the same Markov chain as in Q1, and search for the times, t_{50} and t_{99}, at which $G_{OFF}(t_{50}) + G_{ON}(t_{50}) = 0.5$ and $G_{OFF}(t_{99}) + G_{ON}(t_{99}) = 0.99$, respectively. This has been done using a simple line search, which found that $t_{50} = 0.5305s$ and $t_{99} = 5.0595s$. In Fig. 15.6b these times correspond to the points in time where the dashed line labeled “First Switch” crosses 0.5 and 0.99, respectively.

Q3. *What is the time at which 99% of all cells will have turned ON at least* *once?*

Because one must include the possibility that the cell will first turn OFF and then turn ON, the solution for this question requires a different projection. Let $\mathbf{X}$ be the set of states such that $s_1 \leq 50$, $s_2 \leq 105$, and $s_1 s_2 \leq 300$. Furthermore, let the projection, $\mathbf{X}_{ON'}$ include all states in $\mathbf{X}$ that are *not* ON ($s_1 < 15$ *or* $s_2 > 5$). As time passes, probability measure will leave this region in two manners: either it exits in to the aggregated ON sink (G_{ON}) or it exits out of $\mathbf{X}$ altogether in to a second absorbing sink G_{err}, which results in a loss of precision. This error comes into play as follows: If t_1 is defined as the time at which $G_{ON}(t_1)+G_{err}(t_1) = 0.99$, and t_2 is defined as the time at which $G_{ON}(t_2) = 0.99$, then the time, t_{99}, at which 99% turn ON is bounded by $t_1 \leq t_{99} \leq t_2$. For the chosen projection, this bound is very tight yielding a guarantee that $t_{99} \in [1733.3153, 1733.3157]s$. For comparison, 10^4 runs of the SSA give a much less accurate estimate of $t_{99} \approx 1735.7$. Similarly, one can use a projection $\mathbf{X}_{OFF'}$, which includes all points in $\mathbf{X}$ that are not OFF, to find that it will take between 800.495 and 800.487 seconds until 99% of cells will turn OFF (compared to $t_{99} \approx 827s$ found with 10^4 SSA runs). In addition, median times, t_{50} have been computed and are listed in Table 15.2.

Note that the times for Q3 are very large in comparison to those in Q2; this results from the fact that the ON and OFF regions are relatively stable. This trait is evident in Fig. 15.6, where the dashed lines correspond to the time of the first ON (or OFF) decision *provided that the system has not previously turned OFF (or ON)*. Since about 78% percent turn ON before they turn OFF, this dashed ON curve asymptotes at about 0.78 (see Q1 and Q2). On the other hand, the solid lines corresponds to the times for the first ON (or OFF) decision *whether or not the system has previously turned OFF (or ON)*. The kinks in these distributions, where the solid and dashed curves separate, result from the stability of the two regions region. In particular, the solid ON curve exhibits a more severe kink due to the fact that the OFF region is more stable than the ON region (compare solid lines).

The projections $\mathbf{X}_{ON'}$ and $\mathbf{X}_{OFF'}$ used here included 715 and 782 states respectively. While systems of this size are still relatively inexpensive to analyze, the computational cost will build significantly should we desire to add more complexity. Using balanced truncation, each of these systems can be reduced to 10^{th} order with very little loss in accuracy (compare solid lines and circle markers in Fig. 15.6, and see Table 15.2).

Q4. *What is the distribution for the round trip time until a cell will first turn ON and then turn OFF?*

In order to answer this question one may use the round-trip methodology from the latter half of Section 11.2. Intuitively, this approach is very similar to that depicted in Fig. 11.2b, except that the top and bottom portions of the Markov chain are not identical and the final destination is a region of the chain as opposed to a single point. Also, since the Markov process under examination is infinite dimensional, one must first apply a finite state projection to reduce this system to the finite set $\mathbf{X}$ described in Q3. For the system's outbound journey into the ON region, we use the projection $\mathbf{X}_{ON'}$ from Q3. After the system turns ON, it begins the second leg of its trip to the OFF region through a different projection $\mathbf{X}_{OFF'}$. When the system reaches the OFF region on the second leg, it is absorbed into a sink $G(t)$. The full master equation for this process can be written as:

$$\begin{bmatrix} \dot{\mathbf{P}}^1_{ON'}(t) \\ \dot{\mathbf{P}}^2_{OFF'}(t) \\ \dot{G}(t) \\ \varepsilon(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{ON'} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_2\mathbf{C}_1 & \mathbf{A}_{OFF'} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_3\mathbf{C}_2 & 0 & 0 \\ \mathbf{B}_\varepsilon\mathbf{C}_1 & \mathbf{B}_\varepsilon\mathbf{C}_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P}^1_{ON'}(t) \\ \mathbf{P}^2_{OFF'}(t) \\ G(t) \\ \varepsilon(t) \end{bmatrix}, \qquad (15.1.2)$$

where $\mathbf{A}_{ON'}$ and $\mathbf{A}_{OFF'}$ are the corresponding principle sub-matrices of the standard infinitesimal generator defined in (2.0.5). The matrices $\mathbf{C}_1$ and $\mathbf{B}_2$ are defined as in (11.2.3) and (11.2.4) above and account for the transitions from the states in $\mathbf{X}_{ON'}$ to the corresponding states in $\mathbf{X}_{OFF'}$. The vector $\mathbf{B}_3\mathbf{C}_2$ corresponds to the transitions that exit $\mathbf{X}_{OFF'}$ and turn OFF (completing the full trip). The last two vectors $\mathbf{B}_\varepsilon\mathbf{C}_1$ and $\mathbf{B}_\varepsilon\mathbf{C}_2$ correspond to the rare transitions that leave the projected space, $\mathbf{X}$, and therefore contribute to a computable error, $\varepsilon(t)$ in the analysis.

The solution of this system for the scalar $G(t)$ then gives us the joint probability that (*i*) the system remains in the set $\mathbf{X}_{ON'}$ until it enters the ON region at some time $\tau_1 \in [0, t)$, and (*ii*) it then remains in the set $\mathbf{X}_{OFF'}$ until it enters the OFF region at some time $\tau_2 \in (\tau_1, t]$. This distribution is plotted with the dotted lines in Fig. 15.6. Once again we can see the effect that the asymmetry of the switch plays on the times of these trajectories; the ON region is reached first more often and the ON region is less stable, thus the ON then OFF trajectory will occur significantly faster than the OFF then ON trajectory (compare dotted lines in Fig. 15.6, and see Table 15.2).

In Fig. 15.6, the distributions have been computed in two different manners, which yield nearly indistinguishable results (Compare lines and circles in Fig.

15.6). First, the lines correspond to solutions where (15.1.2) has been solved as a single large system of 1496 ODEs. In the second approach, the system has been analyzed as two separate sub-systems defined by the triplets $SYS_1 = (\mathbf{A}_{ON'}, \mathbf{P}_{ON'}, \mathbf{C}_1)$ and $SYS_2 = (\mathbf{A}_{OFF'}, \mathbf{B}_2, \mathbf{B}_3\mathbf{C}_2)$. Each of these systems has been reduced to 10^{th} order using balanced truncation. Once reduced, the systems were again reconnected resulting in a 22^{nd} order approximation, consisting of the two 10^{th} order reduced systems plus $G(t)$ and $\varepsilon(t)$. Table 15.2 gives the predicted median time t_{50} and the associated computational costs for these methods as well as for 10^4 runs of the stochastic simulation algorithm (SSA). Both FSP methods are far faster and more accurate than the corresponding SSA approach. Comparing the full and reduced FSP approaches, note that the reduced systems retain a high degree of the full systems' accuracy, but the reduction itself is very expensive. In these numerical experiments, we have used `Matlab`'s balanced truncation code `balancmr`, which does not take advantage of the extreme sparsity of the FSP formulation. With parallel algorithms for the balanced truncation of sparse systems, such as those in [7], much of this computational cost may be recovered.

Q5. *What is the probability that the system will be (*i*) ON at some point* $t_1 \in [a_1, b_1] = [100s, 110s]$*, then (*ii*) OFF at some point* $t_2 \in [a_2, b_2] = [200s, 210s]$ *and finally (*iii*) ON at* $t_3 = 300s$*?*

To answer this question we again use the projections, $\mathbf{X}$, $\mathbf{X}_{ON'}$ and $\mathbf{X}_{OFF'}$ from above. In terms of the notation used in Section 11.2.3, we are seeking to compute $\mathbf{P}_{ON}\{\mathcal{C}_3\}(t_3)$, where

$$\{\mathcal{C}_3\} = \{(\mathbf{x}_0, 0); (\mathbf{X}_{ON}, [100, 110]); (\mathbf{X}_{OFF}, [200, 210])\}.$$

This computation is done recursively as follows:

$$\begin{aligned}
\mathbf{P}\{\mathcal{C}_1\}(a_1) &= \exp(\mathbf{A}a_1)\mathbf{P}(0)\\
\mathbf{P}\{\mathcal{C}_2\}(a_2) &= \exp(\mathbf{A}(a_2-a_1))\mathcal{D}_{ON}\{\mathbf{P}_{ON}\{\mathcal{C}_1\}(a_1)\} + \int_{a_1}^{b_1}\exp(\mathbf{A}(a_2-\tau))\mathbf{y}_1(\tau)d\tau\\
&\qquad \mathbf{y}_1(\tau) = \mathcal{D}_{ON}\{\mathbf{A}_{ON,ON'}\exp(\mathbf{A}_{ON'}(\tau-a_1))\mathbf{P}_{ON'}\{\mathcal{C}_1\}(a_1)\}\\
\mathbf{P}\{\mathcal{C}_3\}(a_3) &= \exp(\mathbf{A}(t_3-a_2))\mathcal{D}_{OFF}\{\mathbf{P}_{OFF}\{\mathcal{C}_2\}(a_2)\} + \int_{a_2}^{b_2}\exp(\mathbf{A}(t_3-\tau))\mathbf{y}_2(\tau)d\tau\\
&\qquad \mathbf{y}_2(\tau) = \mathcal{D}_{OFF}\{\mathbf{A}_{OFF,OFF'}\exp(\mathbf{A}_{OFF'}(\tau-a_2))\mathbf{P}_{OFF'}\{\mathcal{C}_2\}(a_2)\}.
\end{aligned}$$

Using this approach, one can compute the probability of the first measurement:

$$|\mathbf{P}_{ON}\{\mathcal{C}_1\}(100)|_1 = 0.543,$$

$$|\mathbf{P}_{OFF}\{\mathcal{C}_2\}(200)|_1 = 0.174, \text{ and}$$

$$|\mathbf{P}_{ON}\{\mathcal{C}_3\}(300)|_1 = 0.0266.$$

Also, by keeping track of the amount of the probability measure that exits $\mathbf{X}$ through each stage, one can obtain a guarantee that these computations are accurate to within relative errors of 9.1×10^{-6}, 4.9×10^{-5}, and 3.3×10^{-4} percent, respectively. The total computational effort is 63.2s. For comparison 10^4 SSA runs take 2020s to complete this same study, and provide an estimate for $|\mathbf{P}_{ON}\{\mathcal{C}_3\}(300)|_1$ of 0.0270, which is a relative error of 1.63%.

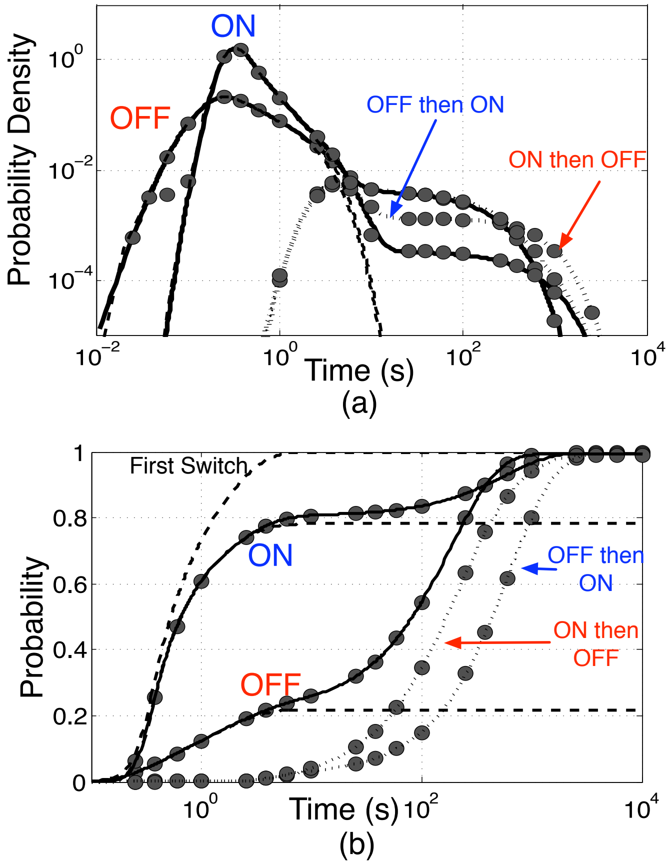


Figure 15.6: Probability densities (a) and cumulative distributions (b) of the times of switch decisions for a stochastic model of Gardner's gene toggle switch [31]. The dashed lines correspond to the probabilities that the first switch decision will be to enter the ON or OFF region. Note that the system will turn ON first for about 78% of trajectories (Q1); the rest will turn OFF first–see asymptotes of dashed lines in (b). A third dashed line in (b) corresponds to the cumulative distribution until the time of the first switch decision (Q2). The solid lines correspond to the probabilities for the first time the system will reach the ON (or OFF) region (Q3). The dotted lines correspond to the times until the system completes a trajectory in which it begins at $s_1 = s_2 = 0$, it turns ON (or OFF), and finally turns OFF (or ON) (Q4). Two methods have been used in these analyses: the lines correspond to the original FSP solution, and circle markers denote the reduced order model solutions (See also Table 15.2).

Single Stage Trajectories

First Switch to OFF					
Method	J_{red}	J_{solve}	J_{total}	t_{50}	% Error
FSP	-	31.0s	31.0s	81.952s	$< 2 \times 10^{-5}$
FSP-RED	111.8	1.85s	113.7s	81.952s	$< 4 \times 10^{-5}$
10^4 SSA	-	2068s	2068s	78.375s	≈ 4.3

First Switch to ON					
	J_{red}	J_{solve}	J_{total}	t_{50}	% Error
FSP	-	25.7s	25.7s	0.65655s	$< 1 \times 10^{-7}$
FSP-RED	133.5s	1.85s	135.3s	0.65656s	$< 8 \times 10^{-4}$
10^4 SSA	-	404.4s	404.4	0.65802s	≈ 0.22

Two Stage Trajectories

First Completion of OFF then ON trajectory					
	J_{red}	J_{solve}	J_{total}	t_{50}	% Error
FSP	-	46.9s	46.9s	434.969s	$< 3.5 \times 10^{-5}$
FSP-RED	222.0s	1.95s	224.0s	434.968s	$< 4.5 \times 10^{-3}$
10^4 SSA	-	3728s	3728s	441.394	≈ 1.5

First Completion of ON then OFF trajectory					
	J_{red}	J_{solve}	J_{total}	t_{50}	% Error
FSP	-	51.0s	51.0s	167.530s	$< 6 \times 10^{-7}$
FSP-RED	241.4s	1.98s	243.4s	167.939	≈ 0.24
10^4 SSA	-	3073s	3073	166.860	≈ 0.40

Table 15.2: Comparison of the computational efficiency of computing switch rates of a Stochastic Gene Toggle Switch using three techniques to solve the chemical master equation: the original Finite State Projection approach (FSP), the FSP approach with balance truncation reduction (FSP-RED), and 10^4 runs of the SSA.

15.2 Sensitivity Analysis of the Toggle Switch

To illustrate the use of the FSP for sensitivity analysis on the gene toggle model, a nominal set of parameters has been chosen to be: $\alpha_1 = 50$, $\alpha_2 = 16$, $\delta_1 = \delta_2 = \gamma = 1$ and $\beta = 2.5$. Fig. 15.7a shows the probability distribution as computed using these parameters. By perturbing α_1, α_2 and β each by a small amount (0.01 percent), one can compute the sensitivities of the distribution; see Fig. 15.7(b-d). In this figure one can observe that an increase in α_1 shifts the s_1 dominant peak to a higher level, and decreases the probability at the s_2 dominant peak (see Fig. 15.7b). Similarly, an increase in α_2 shifts the s_2 dominant peak to a higher level, and decreases the s_1 peak (see Fig. 15.7c).

15.3 Identifying Gene Toggle Parameters

The toggle switch has been engineered in such a manner that it can be used as a sensor of environmental influences such as radiation or external chemical signals [54]. Under certain environmental conditions the system will exhibit a bias toward one phenotype; in others it is biased toward another phenotype. In [54] two toggle mechanisms were constructed; one in which the SOS signaling pathway detects DNA damage resulting from UV radiation or mitomycin C (MMC), and the other which detects quorum sensing molecules. Here we concentrate on the former mechanism, in which the two proteins species refer to λCI and LacI, respectively. When UV or MMC are introduced into the system, the SOS pathway results in RecA coproteases, which increase the degradation rate of λ CI. As a result, for different amounts of UV or MMC, the trade-off between λCI and LacI molecules will change. The output of the mechanism is GFP, which is assumed

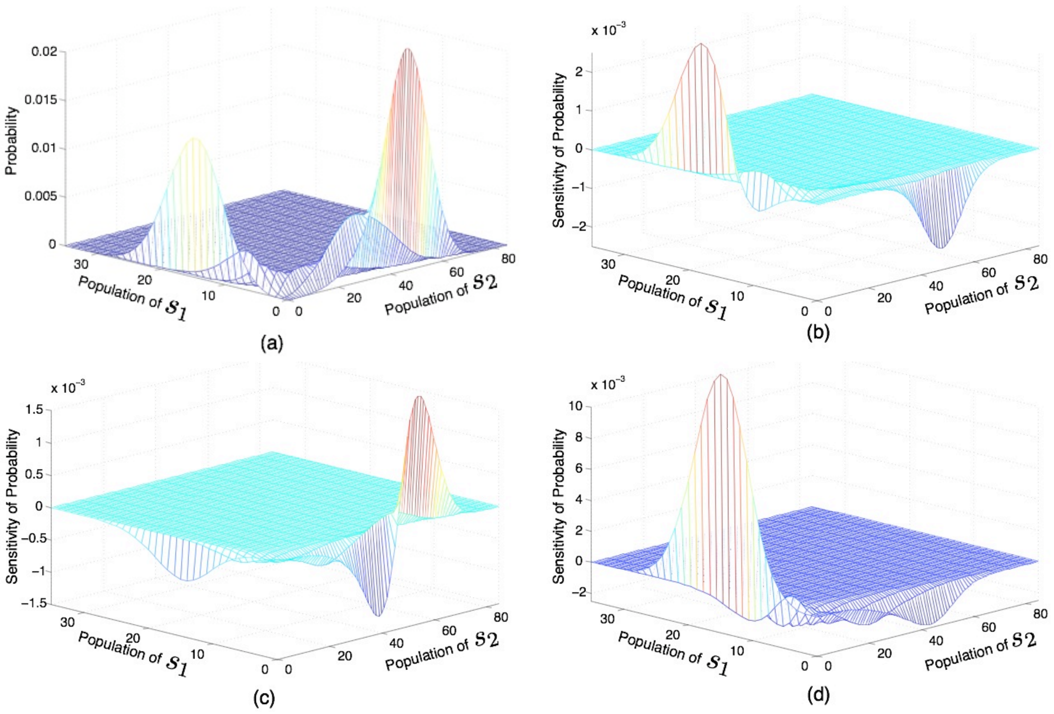


Figure 15.7: Analysis of probability distributions for the parameter set: $\alpha_1 = 50$, $\alpha_2 = 16$, $\delta_1 = \delta_2 = \gamma = 1$ and $\beta = 2.5$. Results plotted at a time $t = 5s$. (a) Joint probability density; (b) Relative sensitivity $\Delta\mathbf{P}(t)/\Delta\alpha_1$, (c) Relative sensitivity $\Delta\mathbf{P}(t)/\Delta\alpha_2$, (d) Relative sensitivity $\Delta\mathbf{P}(t)/\Delta\beta$.

to be expressed at the same level as LacI. The solid black lines in Fig. 15.8(a-c) show the distribution of this output at three different levels of UV radiation (0, 6, and 12 J/m^2) taken from [54]; these three data sets are to be used as the training set to identify parameters for the proposed model–see Table 15.3. In the model below, the *only* difference between these three figures is in the λCI degradation rate, which increases as a function of the applied UV.

The model of this toggle system consists of four basic reactions:

$$\begin{array}{ccccccc} R_1 & ; & R_2 & ; & R_3 & ; & R_4 \\ \emptyset \rightarrow u & ; & u \rightarrow \emptyset & ; & \emptyset \rightarrow v & ; & v \rightarrow \emptyset, \end{array}$$

and the rates of these reactions, $\mathbf{w}(u, v, \bar{\lambda}) = [w_1(u, v, \bar{\lambda}), \ldots, w_4(u, v, \bar{\lambda})]$ depend upon the populations of the proteins u and v[1] as well as a set of unknown parameters, $\bar{\lambda} = [\lambda_1, \ldots, \lambda_p]$. The number and meaning of the parameters vary with the model. The goal of this study is to determine how well one can identify $\bar{\lambda}$ from the experimental data presented in [54]. We will show that one cannot uniquely identify all parameters from this data but we will propose a few additional experiments with which one can complete this identification task.

For the identification study, we consider the following stochastic model in which there are assumed to be four reactions, $\{R_i\}$, with non-linear propensity

[1] For convenience within this section, I have replaced s_1 and s_2 with u and v, respectively.

functions, $\{w_i(u,v)\}$:

$$\begin{aligned}
&R_1 : \emptyset \rightarrow u; w_1(u,v) = k_{11} + \frac{k_{12}}{1+(k_{13}v)^3} \\
&R_2 : u \rightarrow \emptyset; w_2(u,v) = \delta_u u \\
&R_3 : \emptyset \rightarrow v; w_3(u,v) = k_{21} + \frac{k_{22}}{1+(k_{23}u)^3} \\
&R_4 : v \rightarrow \emptyset; w_4(u,v) = \delta_v v. \qquad (15.3.1)
\end{aligned}$$

In order to run the identification, one must first choose a cost function with which to compare the numerical predictions with the experimental results of [54]. There are many possibilities for this cost function, and each may provide slightly different optimums. In this study, it is most important to capture the bimodal behavior of the distribution. Let $\mathbf{P}^\star$ denote the experimental distribution and $\mathbf{P}^{FSP}(\tilde{\lambda})$ represent the model distribution with parameter set $\tilde{\lambda}$. The comparison metric we have chosen is:

$$J(\tilde{\lambda}) := \sum_i w_i \left| \mathbf{P}_i^\star - \mathbf{P}_i^{FSP}(\tilde{\lambda}) \right|,$$

where w_i are the weights placed on each element of the distribution. These values have been chosen as:

$$w_i = \left\{ \begin{array}{ll} 2 & \text{for } i \in \{0,1,\ldots,20\} \\ 1 & \text{for } i > 20 \end{array} \right\}.$$

The rationale behind this choice of weight is to place more importance on the model's ability to capture the leftmost peak.

With this objective function, the identification has been conducted with many randomly assigned initial guesses for the parameters. Two methods have been considered for the optimization: `Matlab`'s `fminsearch` and a generic simulated annealing algorithm. Fig. 15.8(a-c) shows that the model can indeed capture the experimentally observed behavior of the toggle system. The parameters found in this identification are listed as Parameter Set 1 in Table 15.3. To further test this model, all of the parameters with the exception of δ_u are fixed at the values of Parameter Set 1, and we explore how the degradation of u changes with the level of MMC. This is done by setting the target objective function to each of the experimental distributions in Fig. 15.8(d-f) and searching for the best $\delta_u(MMC)$ to match that distribution. Once again, we find the the parameters provide a reasonable fit to the experimental data.

In the model, the degradation of u, which depends upon the level of UV radiation or MMC level, is the only parameter that changes between Figs. 15.8a through 15.8f. Fig. 15.9 shows the identified value of δ_u as function of the UV or MMC levels. In [54] the effect of UV on the degradation rate is given by a function

$$\delta_u = \alpha_1 + \frac{\alpha_2 r^n}{\alpha_3 + r^n},$$

where α_i are positive constants, r is the level of radiation in J/m^2 and n is a hill coefficient. Using this expression, and assuming a hill coefficient of $b = 2$, the three values for δ_u can be used to find the set of $\{\alpha_i\}$:

$$\{\alpha_1, \alpha_2, \alpha_3\} = \{0.533, 8.43, 94.8\},$$

which is plotted as the solid green line in Fig. 15.9(left). It should be noted that

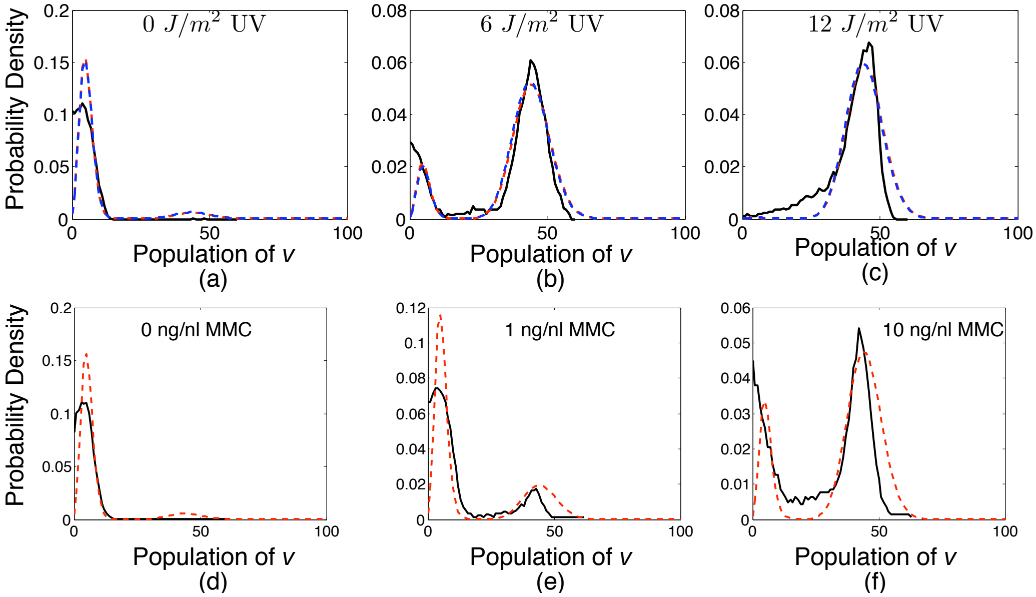


Figure 15.8: Toggle switch GFP distributions in various conditions of DNA damage. The solid black lines correspond to experimentally measured data [54], and the blue and red dashed lines correspond to two fits with two different parameter sets. Figs. (a-c) correspond to the distributions of GFP under three different levels of UV radiation. These data sets were used as the training data to obtain the parameter listed in Table 15.3. These data were later used to compare to the test data in Figs. (d-f). The degradation parameter δ_u is the only parameter that changes between the six figures.

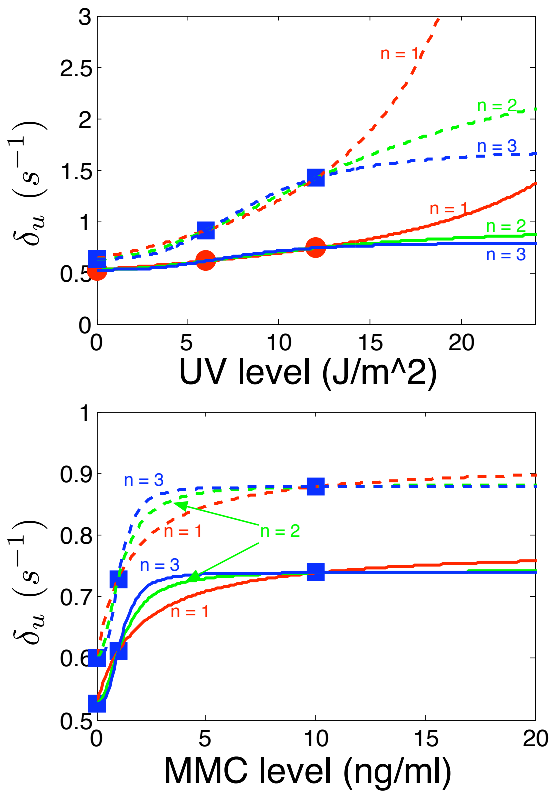


Figure 15.9: Effect of varying levels of UV (left) and MMC (right) on the degradation of the protein u. The data points are the identified values for the degradation parameter at each level of UV radiation or concentration of MMC. The lines correspond to hill functions fit to these data points with different exponents n=1, 2 and 3. The solid lines correspond to Parameter Set 1 and the dashed lines correspond to Parameter Set 2.

for a hill coefficient of 1, the corresponding fit yields non-physical negative results:

$$\{\alpha_1, \alpha_2, \alpha_3\} = \{0.533, -0.811, -26.8\}.$$

With information on how the system changes with varying conditions, one can in principle use this toggle switch to indirectly measure the UV or MMC levels (as is the general objective of [54]).

The available data for this identification is severely limited in that it only gives the distribution of a single protein, v, whereas the model computes the simultaneous joint distributions for both proteins u and v. It has been found that two very different parameter sets can match the same data set for the distributions of v, and therefore this identification is not unique (see Parameter Sets 1 and 2 in Table 15.3). Even though the parameter sets are not unique, comparing parameter sets to each other can reveal some interesting information. In particular, the comparison reveals that the v production and degradation values $(k_{21}, k_{22}, \delta_v)$ in parameter set 2 differ from those in set 1 by a fixed ratio (See Table 15.3). Thus, this identification has at least revealed that the ration between these three parameters as:

$$[k_{21} : k_{22} : \delta_v] = [5.19 : 38.8 : 1].$$

At this point, however, it is not clear if any additional information is forthcoming, and more information is needed to distinguish between these parameter sets. Close examination finds that although parameter sets 1 and 2 predict the same distribution for v, these parameters result in very different behavior for the distribution of u as can be seen in Fig. 15.10. Therefore these parameters set produce distinct results, and with more information, the unique identification of the parameters may become achievable.

Identification with full Distribution

In the previous identification attempt, we tried to identify the parameters from the distribution of a single protein at a single point in time. We found that there are many possible parameter sets that will succeed in matching the distribution of v but which have very different distributions for u. As a numerical experiment, we have used parameters set 1 to generate the full joint distribution of u and v,

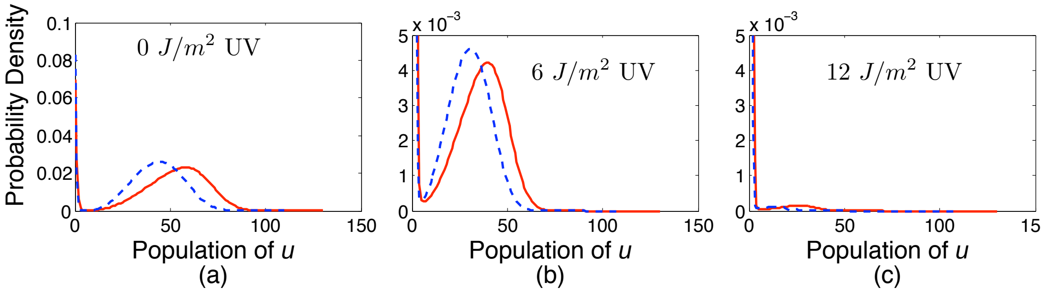


Figure 15.10: Toggle switch distributions for u for parameters sets 1 (red) and 2 (blue) from Table 15.3. Note that the distributions for u here are different while the distributions for v in Fig. 15.8 were identical.

which we then use as the target distribution in the identification procedure.[2]

Using the full distribution at a time of 1000s allowed for a reasonably close identification of the ten parameters (see Parameter set 3 in Table 15.3), but the identification is still not unique. In this case, closer examination reveals that there is much stronger correlation between the parameters. In particular, the values for all of the production and degradation parameters $(k_{12}, k_{21}, k_{22}, \delta_v, \delta_u(0), \delta_u(6), \delta_u(12))$ are all a constant factor of 1.19 removed from the correct parameters. Thus, we have uniquely established the ratio between all of these parameters but not their exact numbers. This suggests that the identification is very close, and may be complete with a little additional information.

Closer examination of the model with Parameter Set 1 reveals that at least some of the transient modes have died out on a time scale less than the chosen 1000s. This can be seen readily by comparing the distributions computed with Parameter Sets 1 and 3 at different times. In Fig. 15.11 the marginal distribution of u at different times as computed from Set 1 (solid blue line) and Set 3 (dashed red line). For very short times of 1 or 10s (top two rows), these distributions

[2] The cost function for this and all remaining identification procedures has been set to a simple un-weighted 1-norm difference between the estimated and target distributions.

are distinguishable from one another. However, after a short transient time of 100s, the two distributions are indistinguishable (compare bottom three rows). In essence, conducting the identification at 1000s is effectively the same as identifying the system after it has already reached some lower dimensional manifold. As was the case in Chapter 12 above, we discover that it is impossible to uniquely identify all parameters from insufficiently rich dynamics. However, if instead we attempt the identification at a shorter time step of 50s before these dynamics have fully decayed, then the identification becomes possible, and we are able to uniquely identify every parameter each within an error of 0.5% (See parameter set 4 in Table 15.3).

Identification with a single protein at multiple time points

Chapter 12 showed that by taking data many time points, one can identify transcription and translation parameters from protein data alone. Although it is not possible to uniquely identify all ten parameters of model with the quasi-steady state distribution of v, it may be possible to identify these parameters from the distribution of v at multiple transient points in time. With data taken at five time points, many of the parameters can indeed be identified as documented in Set 5 of Table 15.3. However, a few parameters remain unidentified. These include the repressed production rates for u as well as the effects of u and v levels on the production rate of v and u respectively. It is interesting to note that the degradation rates of u are correctly identified. A possible reason for this is as follows: In the absence of v, the production of u is very fast, and the level of u quickly reaches a relatively high level. This is observed in Fig. 15.11 Once v in which u reaches its highest levels on the order of 10s. The second protein v

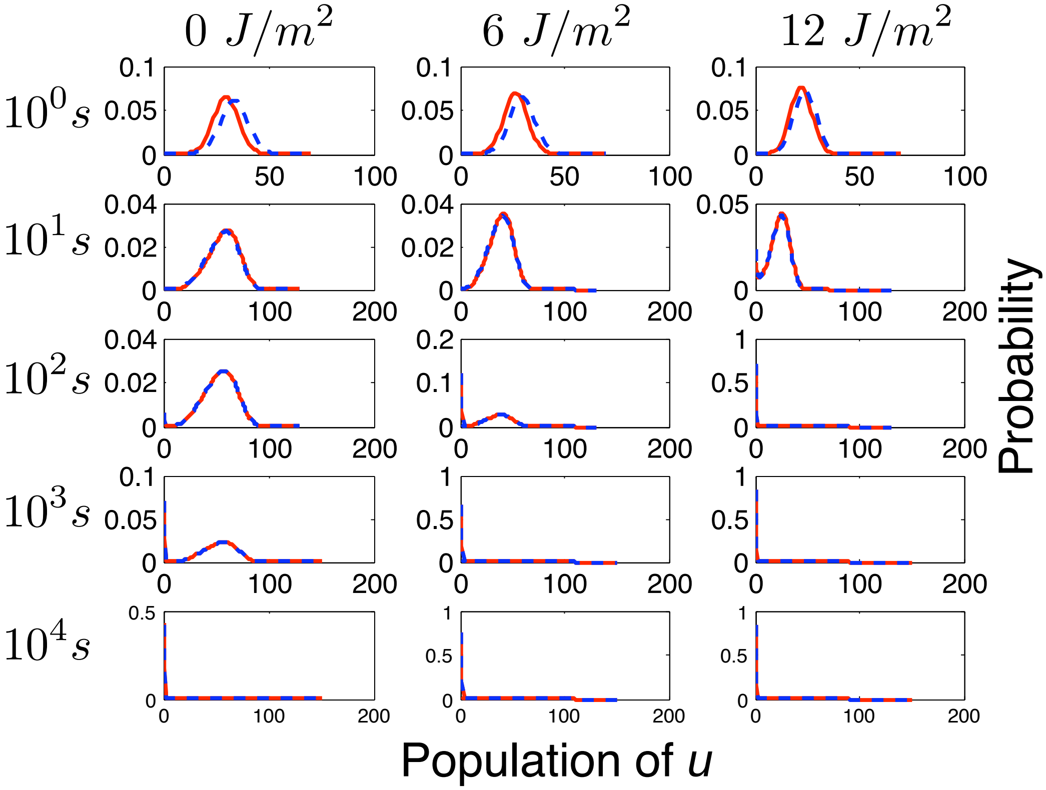


Figure 15.11: Comparison of the marginal distribution, f_u at different UV levels of $\{0, 6, 12\}$ J/m^2 at different points in time $\{1, 10, 100, 10^3, 10^4\}s$ for two different sets of parameters. Parameter Set 1 corresponds to the solid blue lines, and Set 3 corresponds to the dashed red line.

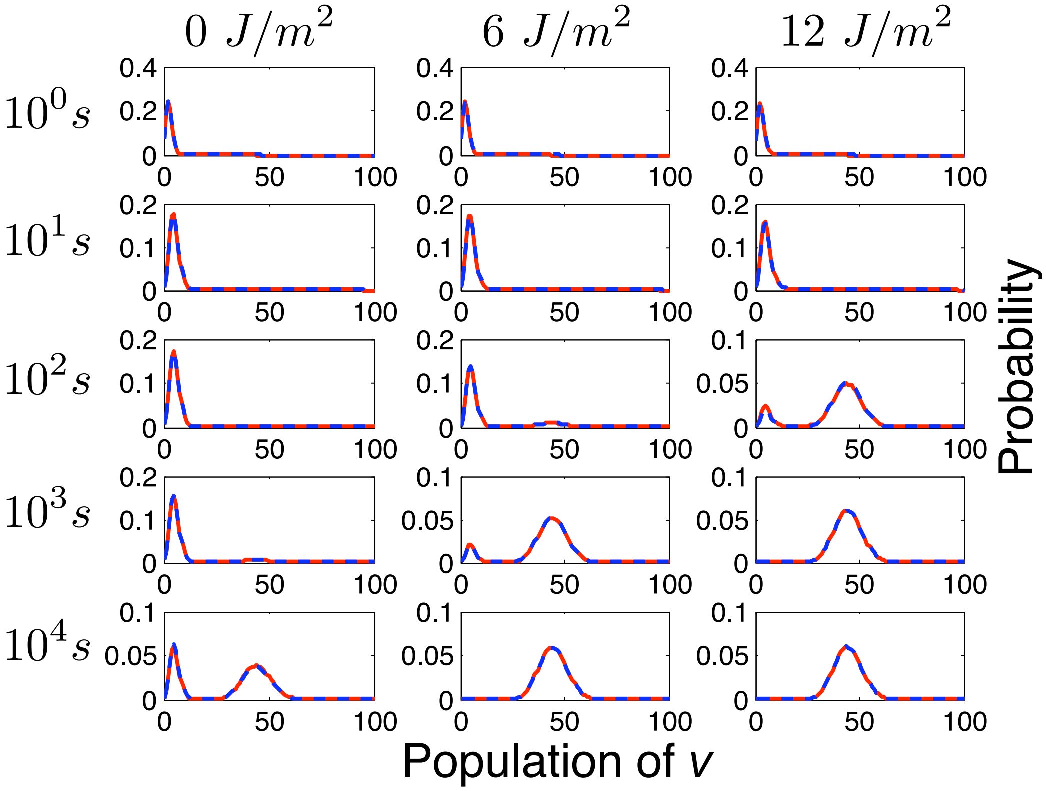


Figure 15.12: Comparison of the marginal distribution, f_v at different UV levels of $\{0, 6, 12\}$ J/m^2 at different points in time $\{1, 10, 100, 10^3, 10^4\}s$ for two different sets of parameters.

also reaches a moderate level on the same time scale and effectively shuts off the production of u. By the time 20s has passed u can have relatively high levels, but it is no longer being produced. As more time passes the degradation of u can be observed in increasing levels of v production. In order to capture the rates of u production, however, it is envisioned that a much shorter time scale may be necessary.

In each of the above optimization procedures, very fine precision is required in the solution of the distribution. In some cases moderate changes in the parameters result in relatively small changes in the distribution. If these changes are on the same order of the error in the master equation solution, then it is impossible to judge whether one parameter set is better than another. This issue is particularly critical in gradient based searches, where a small perturbation approach is used to estimate the gradients. In the procedures above the FSP tolerance, ε, has been set to 10^{-6}. Additional numerical experiments with tolerances of 10^{-4} have been considered, but were found to be far less reliable in the estimation. This observation has two consequences: (*i*) it would be very difficult to conduct such an estimation with Monte Karlo type solutions, such as the SSA, because these would require on the order of 10^{12} runs per parameter set. (*ii*) Small amounts of measurement error will have a strong adverse effect on the identifiability of any given system. These two consequences may be somewhat ameliorated by taking measurements at a much large number of time points and effectively averaging out the measurement noise.

Parameter	Set 1	Set 2: f_v at 10^3s	Set 3: $f_{u,v}$ at t=10^3s	Set 4: $f_{u,v}$ at t=50s	Set 5: f_v at t={20,40,...,100}s
k_{11}	0.0099	0.0089 (0.90)	0.011 (1.13)	0.0098 (0.996)	2.1×10^{-9} (0)
k_{12}	40.3	27.5 (0.68)	48.1 (1.19)	40.2 (0.998)	33.89 (0.84)
k_{21}	1.35	2.49 (1.84)	1.62 (1.19)	1.35 (0.998)	1.35 (0.996)
k_{22}	10.1	18.6 (1.84)	12.1 (1.19)	10.1 (0.998)	10.1 (0.996)
δ_v	0.26	0.47 (1.84)	0.31 (1.19)	0.26 (0.998)	0.26 (0.998)
$\delta_u(0)$	0.53	0.40 (0.75)	0.64 (1.19)	0.53 (0.998)	0.53 (0.998)
$\delta_u(6)$	0.77	0.57 (0.74)	0.91 (1.19)	0.77 (0.998)	0.77 (1.00)
$\delta_u(12)$	1.19	1.46 (1.23)	1.42 (1.19)	1.19 (0.998)	1.20 (1.01)
k_{13}	0.0025	0.0040 (1.61)	0.0025 (1.0)	0.0025 (1.00)	0.0024 (0.973)
k_{23}	0.0084	0.025 (2.93)	0.009 (1.10)	0.0084 (0.998)	0.0149 (1.77)

Table 15.3: Four parameter sets that yield the same distribution for v for the stochastic toggle model. Parameter set 1 has been identified directly from the experimental data using a time of $t = 1000s$. Parameter set 2 is identified from the distribution of v at $t = 1000s$. Parameter set 3 is identified from the full joint u and v distribution at $t = 1000s$. Parameter set 4 is identified from the full distribution at $t = 50s$. Parameter set 5 is identified from the distribution of v only but at five time points $t = \{20, 40, 60, 80, 100\}s$. For parameter sets 2-5, the values in the parenthesis denote the ratios of the identified parameters to the "true" parameters of Set 1.

Chapter 16

Case Study: The Pap Pili Epigenetic Switch in *E. Coli*

Each year urinary tract infections result in about 8.3 million doctor visits in United States alone[1]. Over 90% of the *Escherichia Coli* bacteria isolated from these infections are covered with small hair-like structures known as Pyelonephritis-Associated Pili, or Pap [75]. From the perspective of *E. coli* survival within a host organism, pili expression is both beneficial and detrimental–Pili enable *E. coli* to bind to host epithelial cells, establish colonies and feed off host organisms. Without the binding capabilities of pili, *E. coli* colonies would be more easily flushed from the host (i.e. during urination). Conversely, pili production consumes a significant portion of the cellular energy, thus weakening individual bacteria [46]. Further, pili to host attachment may irritate the host and trigger an immune response. Thus, it is beneficial for any population descending from a single ancestor cell to have different pili expression phenotypes. This variation in expression comes as a result of an epigenetic switch–two cells with the exact same DNA can have vastly different expression: one expresses pili (phase ON) and one does not

[1] *Ambulatory Care Visits to Physician Offices, Hospital Outpatient Departments, and Emergency Departments: United States, 1999–2000.* Vital and Health Statistics. Series 13, No. 157. Hyattsville, MD: National Center for Health Statistics, Centers for Disease Control and Prevention, U.S. Dept. of Health and Human Services; September 2004.

(phase OFF).

Previous experimental research conducted by David Low's group at UCSB has produced a vast amount of understanding regarding the Pap system switching mechanism [9, 12, 74, 13, 73, 100, 49, 45, 46]. The key element of the system is the *pap* gene (see Figs 16.1 and 16.4) which controls the transcription of the PapBA sequence of messenger RNAs necessary for pili expression. There are two areas to which regulators bind and alter the output of the PapBA promoter. These are the proximal area (sites 1, 2 and 3) and the distal area (4, 5 and 6). The two most influential global regulators are: leucine-responsive regulatory protein (Lrp), which binds to sites 1-6, and DNA adenine methylase (DAM), which methylates the four GATC sequences found at the top and bottom strands at sites 2 and 5 [45]. In addition to the global regulators, the Pap-encoded local regulator protein (PapI) is produced within and is specifically linked to the Pap network [46]. Depending upon how the regulators alter the epigenetic structure of the *pap* operon, the PapBA promoter may be active or inactive. The key-ingredients for the active cell (see Figure 16.4a) is DAM methylation of the top and bottom GATC sequences in site 2 and Lrp bound to distal sites 4, 5, and 6 [46]. PapI is produced when the gene is in this active configuration and subsequently increases the affinity of Lrp for the distal sites, and thus acts as a positive feedback regulator [46].

Building upon this vast understanding of the pap system, at least three independent computational research groups have been studying the effects of DAM methylation, Lrp binding, growth rate, initial state dependence, and molecular noise on the Pap switching behavior. Liao's group in UCLA published two such models [48, 111] in which the authors utilized the Monte Carlo based Stochastic

Simulation Algorithm [35] to model the pap system dynamics. At UCSB, Shoemaker and Doyle considered a hybrid Boolean/stochastic model of the pap pili switch to demonstrate that molecular noise may lead to disparities between cell genotype and phenotype [89]. We recently proposed a third concurrent model [66], to analyze the core regulatory region of the *pap* gene and its stochastic interactions with the key regulatory components: Lrp, PapI and DAM. This model is particularly exciting in that research on the Pap system has directly led to the development of the FSP methods discussed above. This Chapter expands upon the model in [66] to make specific predictions regarding the dependence of the pap system upon the concentration of its various regulatory chemicals: Lrp, DAM, PapB and PapI. First, the next section begins with a detailed description of the Pap system. Then, Section 16.2 provides a detailed analysis of the Pap switch.

16.1 The Core Mechanics of the Pap Switch

The expression of pyelonephritis-associated pili (Pap) in *E. Coli* is stochastic in nature in that two cells of identical ancestry and identical environment may develop vastly different traits: ON (pilliated) or OFF (bald). This section discusses the most important mechanisms that control the Pap switch and provide the assumptions upon which the current model is based.

The first subsection describes the important genetic features of the *pap* operon: the genetic sequence which provides the basic structure through which Pap expression is activated and regulated. The following subsections then describe how various regulators interact with the operon to effect Pap switching behavior. The final subsections then describes the necessary genetic configurations for *pap* transcription.

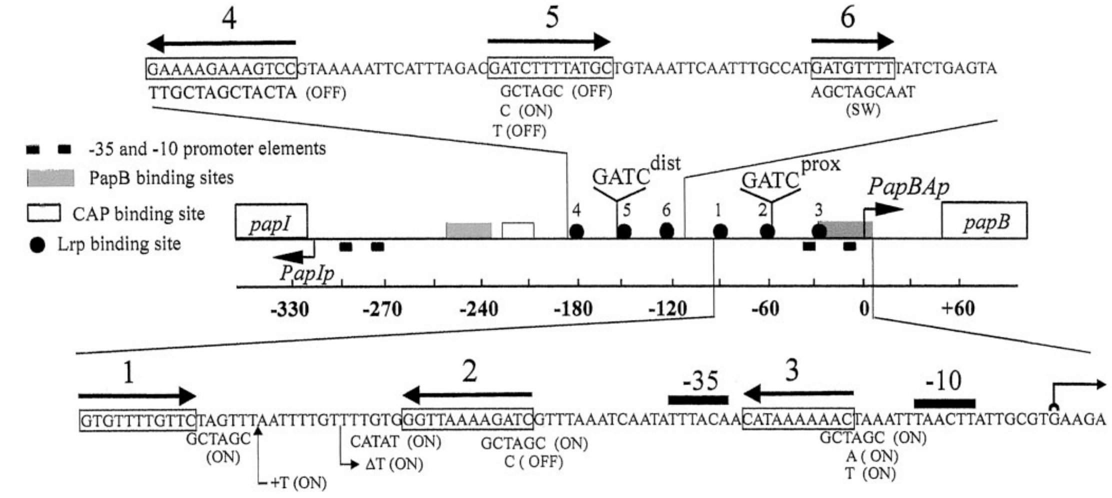


Figure 16.1: Organization of the pap operon region between the divergently transcribed papBA and papI promoters (from [45]). The two GATC sites subject to methylation by DAM are $GATC^{prox}$ and $GATC^{dist}$, are located within Lrp binding sites 2 and 5, respectively. The Lrp sites are shown as filled circles and as boxed regions on the expanded DNA sequence. The orientation of the Lrp sites (using a consensus sequence 5'-Gnn(n)TTTt-3') is indicated with arrows above the sequence. The distance between sites 2 and 5 is 102 bp and the distance between sites 1 and 6 is 32 bp, measured between conserved base-pairs within the Lrp binding sites. The PapB binding site is shown as a hatched box. A few mutations are shown below the wild-type sequence; switch phenotypes are for these indicated in parentheses.

16.1.1 The *pap* operon

The *pap* operon provides the basic structure of the switch and defines the rules of all regulatory actions. The *pap* regulatory region encompasses the divergently transcribed *papI* and *papB* genes together with the 416 bp intergenic region (Fig. 16.1) [45]. In wild-type *E. Coli*, PapB is the first of many convergently transcribed proteins that result in the eventual production of Pili. In addition to being produced by the *pap* operon, both PapB and PapI are local regulatory proteins [8, 100, 110, 42, 49, 46]; see below.

In the intergenic region between the *papI* and *papB* promoters, there are six *pap* DNA Lrp binding sites spaced three helical turns apart; each Lrp binding site contains the sequence GxxxxTT [73]. These sites are designated by numbers 1 to 6 in Fig. 16.1, where 1-2-3 correspond to the sites proximal to the *papB* promoter, and sites 4-5-6 are those distal to the *papB* promoter [73].

The regulatory region also contains four DNA sites with the genetic sequence GATC. Two of these occur at the top and bottom strands at Lrp binding site 2, designated $GATC_{prox}$. The other two occur at Lrp binding site 5, designated $GATC_{dist}$. DNA GATC sites are target sites for DNA adenine methylase (DAM), which places a methyl group on the adenine of each GATC sequence [13].

The Pap switch is controlled by the three primary regulatory factors that interact with the regulatory sites 1-6. These factors are Leucine-responsive Regulatory Protein (Lrp) [74, 12, 13, 73, 102, 101], DNA Adenine Mathylase (DAM) [12, 74, 13, 101, 42, 46] and the local pap encoded regulatory protein PapI [74, 49]. The next subsections provide a brief description of these regulatory factors and lists the related assumptions for the proposed model. For the readers' convenience, the major assumptions are listed in bullets below.

16.1.2 Leucine-Responsive regulatory Protein (Lrp)

Leucine-responsive regulatory protein is a global regulator that affects many of the genetic processes of the cell including Pap. The number of Lrp molecules found in the cell is on the order of 3000 copies [109], but most of these are bound to other non-pap-specific regions of the DNA. Furthermore, the remaining Lrp molecules aggregate into large groups, such that the number of *free* Lrp molecular groups may be as few as 50 to 150 per cell [88]. The current model assumes that each such group behaves as a single reactant Lrp molecule.

Model Assumption 1: Based upon this data, the model assumes that the number of free Lrp molecules is held constant at 100 per cell.

In vitro DNA footprint analyses indicate that Lrp binds with highest affinity to Lrp sites 1-2-3, and with lower affinity to sites 4-5-6 [73, 74, 13]. Furthermore, this binding is highly cooperative.

Model Assumption 2: Lrp binds simultaneously at all three proximal sites (1-2-3) and/or at all three distal sites (4-5-6). Thus there are four possible Lrp binding patterns as illustrated in Figure 16.2. These Lrp bindings are reversible in that Lrp can bind and unbind from the DNA.

Mutational analyses show that disruption of Lrp binding sites 2 or 3 results in increased *papB* activation. In contrast, disrupting Lrp sites 4 or 5 results in decreased *papB* activity [73]. These results suggest that binding of Lrp proximal to the *papB* promoter inhibits transcription whereas binding of Lrp at the distal site activates transcription, and supports the following assumptions:

Model Assumption 3: RNAP binds at full strength to the *pap* operon when Lrp is bound to distal sites 4-5-6. RNAP does not bind when Lrp is bound to proximal sites 1-2-3. RNAP binds at one tenth its full strength when Lrp is bound at neither 1-2-3 or 4-5-6.

Based upon the results reported in [74, 73, 13, 46], Table 16.1, top section, provides the ratios of dissociation/association rates of Lrp for the proximal and distal locations for two levels of PapI. For the distal Lrp binding region, the affinities are given for the four possible methylation patterns: Hemi-0: no methylation, Hemi-T: methylation of top strand only, Hemi-B: methylation of bottom strand only, and Hemi-2: full methylation. For the proximal location, it is assumed that all methylation patterns have the same Lrp binding affinities.

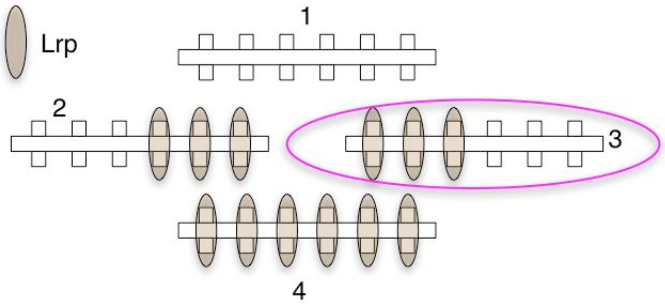


Figure 16.2: Schematic of the Possible *pap*-Lrp binding configurations.

From unpublished results produced by Aaron Hernday in Low's group, Table 16.1, second section, presents the half life's for Lrp dissociation at the proximal and distal locations for two different quantities of PapI. From these half-life data, one may derive the dissociation rates of Lrp. Table 16.1, third section, presents these derived dissociation rates.

In [45], Hernday *et al* showed that Lrp binding at the proximal sites 1-2-3 reduces the affinity of Lrp for the distal sites 4-5-6 by a factor of 10-fold. This mutual exclusion effect acts in the reverse as well in that Lrp binding at 4-5-6 reduces the affinity of Lrp at 1-2-3. This supports the assumption:

Model Assumption 4: Lrp bound at sites 4-5-6 decreases the affinity at 1-2-3 by a factor of 10. Lrp bound at sites 1-2-3 decreases the affinity at 4-5-6 by a factor of 10.

16.1.3 DNA Adenine Methylase (DAM)

DNA adenine methylase is another global regulator for *E. coli.* This regulator occurs in low copy numbers on the order of about 130 molecules per cell [11]. DAM targets GATC sequences throughout the genome and places methyl groups

1. LRP off/on Binding Ratios ($k_D = k_{off}/k_{on}$ Measured [46])					
	Locations	Methylation	PapI amount	Value	Units
	dist	hemi-0	0nM	2.2	nM
	dist	hemi-T	0nM	9.0	nM
	dist	hemi-B	0nM	15.5	nM
	dist	hemi-2	0nM	20.0	nM
	dist	hemi-0	Saturation	0.2	nM
	dist	hemi-T	Saturation	2.0	nM
	dist	hemi-B	Saturation	6.5	nM
	dist	hemi-2	Saturation	20.0	nM
	prox	all	0nM	1.2	nM
	prox	all	Saturation	0.4	nM
2. LRP Dissociation Half Life's ($\tau_{1/2}$Measured–Unpublished)					
	Locations	Methylation	PapI amount	Value	Units
	prox	all	0nM	3600	s
	prox	all	Saturation	10200	s
	dist	all	0nM	90	s
	dist	all	Saturation	1080	s
3. LRP Dissociation Rates (k_{off} Derived)					
	Locations	Methylation	PapI amount	Value	Units
	prox	all	0nM	1.92×10^{-4}	$N^{-1}s^{-1}$
	prox	all	Saturation	6.80×10^{-5}	$N^{-1}s^{-1}$
	dist	all	0nM	7.70×10^{-3}	$N^{-1}s^{-1}$
	dist	all	Saturation	6.42×10^{-4}	$N^{-1}s^{-1}$
4. Lrp Prox/Dist Mutual Exclusion Effect: 10x (Measured [46])					
5. PapI effect exponential constant (α Fitted)					
	Locations	Methylation	Value	Units	
	dist	all	5	N^{-1}	
	prox	all	80	N^{-1}	

Table 16.1: Reaction rate parameters for the Lrp association and dissociation events.

(CH_3) upon the adenine element of these GATC sequences. For most GATC sequences, it appears that DAM methylates processivly following DNA replication [99]. By moving along the one dimensional DNA strand and methylating each sequential site, this allows DAM to remethylate the GATC sites much faster and with far fewer DAM molecules than would be required by a random distributed methylation process. However, it has been shown that the specific GATC targets at in the *pap* regulatory region (at sites 2 and 5) are methylated much slower than other GATC sites, and in much more of distributive, point-wise manner [79]. The related modeling assumption is:

Model Assumption 5: DAM applies methyl groups to the *pap* GATC sequences in a distributed stochastic process with an exponentially distributed holding time. DAM can individually methylate any of four *pap* GATC sequences: top-dist, top-prox, bottom-dist, bottom-prox, and there are a total of $2^4 = 16$ possible methylation patterns as illustrated in Fig. 16.3.

Urig *et al.* observed that the remethylation of GATC following replication occurs very quickly with a half life of about 4 seconds [99]. If one assumes that this remethylation is distributed and that there are approximately 130 molecules of DAM in the cell, then one can derive the DAM methylation rate as:

$$k_{meth} = -log(0.5)/(130 * 4s) = 0.00133N^{-1}s^{-1}.$$

Of course, this rate corresponds to the half life for processive methylation. The rate for the actual methylation of the GATC sites in the *pap* operon should be much less. For the results presented in this study, the methylation rate is assumed to be:

$$k_{meth} = 0.00025N^{-1}s^{-1}.$$

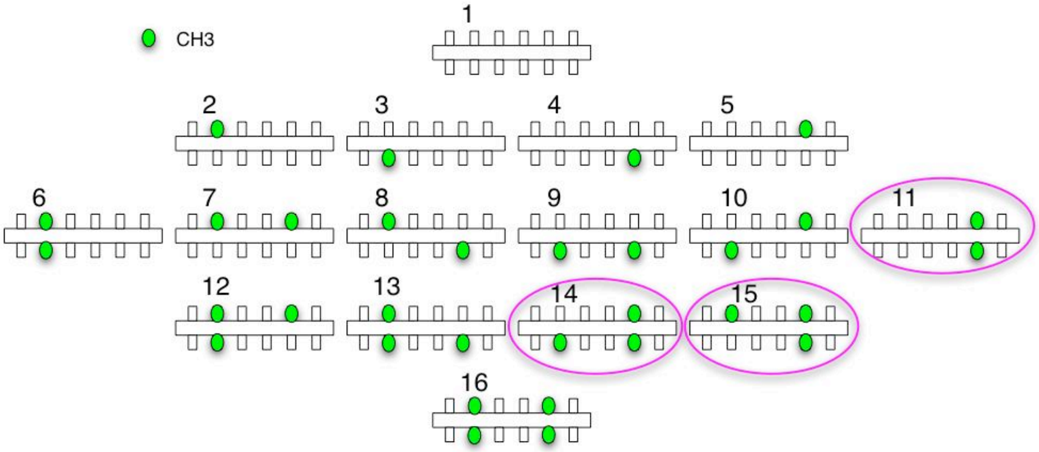
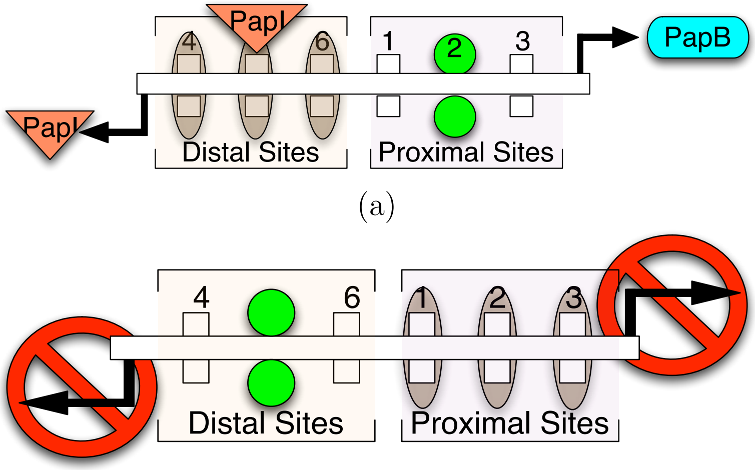


Figure 16.3: Schematic of the 16 possible *pap* methylation configurations.

Examination of the pap DNA methylation patterns showed that in phase OFF cells $GATC_{prox}$ is nonmethylated and $GATC_{dist}$ is methylated whereas the converse pattern exists in phase ON cells ($GATC_{dist}$ nonmethylated, $GATC_{prox}$ methylated) [13]. This information supports the following assumptions regarding the disassociation rates of RNAP:

Model Assumption 6: RNAP disassociates at its basal rate only when $GATC_{prox}$ is fully methylated and $GATC_{dist}$ is not fully methylated. RNAP disassociates at its 400× its basal rate when neither $GATC_{dist}$ nor $GATC_{prox}$ are fully methylated. RNAP disassociates at its $400^2\times$ its basal rate when $GATC_{dist}$ is fully methylated.

The ability of DAM to methylate the GATC sequences depends upon Lrp [12, 74, 101]. In particular, addition of Lrp to pap DNA in vitro blocks methylation of the *pap* regulatory GATC sequences [101]. These data indicate that in phase OFF cells, Lrp is bound at sites 1-2-3 and blocks methylation of $GATC_{prox}$ within site 2. In contrast, Lrp bound to sites 4-5-6 in phase ON cells blocks methylation



(a)

(b)

Figure 16.4: (a) Full OFF transcription configuration of the *pap* operon. $GATC_{dist}$ is fully methylated (green circles at site 5), Lrp (grey ovals) is bound at proximal sites (1-3), *pap* transcription is shut off. (b) Full ON transcription configuration of the *pap* operon. $GATC_{prox}$ is fully methylated (green circles at site 2), Lrp (grey ovals) is bound at distal sites (1-3), *pap* transcription is at full strength. *PapI* binds to, and stabilizes the Lrp-DNA complex at site 5.

of $GATC_{dist}$ within site 5 (see Fig. 16.4). This information supports the following model assumptions:

Model Assumption 7: Lrp bound at sites 4-5-6 blocks DAM methylation at site 5. Lrp bound at sites 1-2-3 blocks DAM methylation at site 2.

There is no known mechanism through which once applied a methyl group may be removed from a GATC site. Therefore, unlike Lrp binding events, DAM methylation events are modeled as irreversible reactions. Combining the four different Lrp binding configurations in Figure 16.2 and the sixteen methylation patterns in Fig. 16.3, there are a total of 64 possible *pap* operon configurations as shown in Figure 16.5. Only three of these 64 configurations satisfy the conditions of assumptions 3 and 6 and lead to full *pap* transcription; these are circled in Figure 16.5. The 64 operon configurations are linked by 192 different reactions:

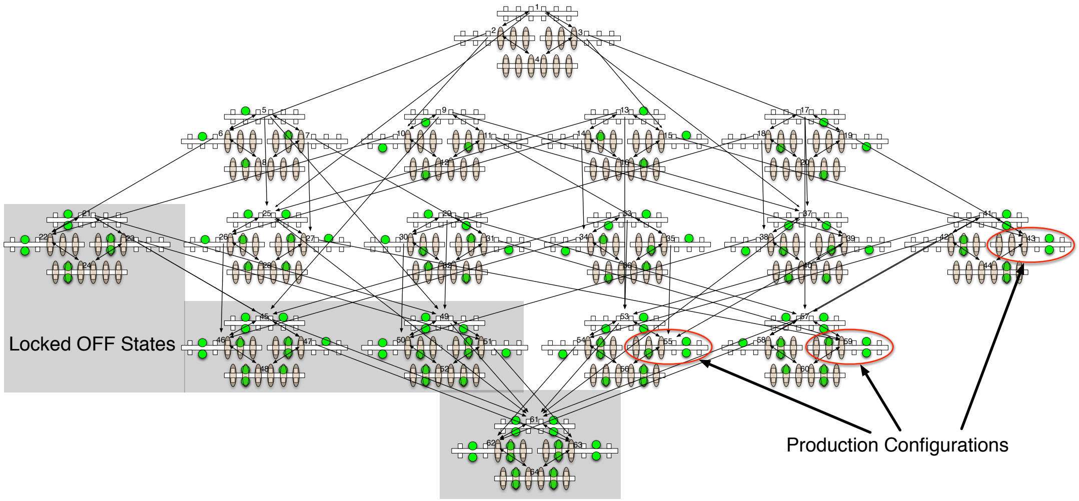


Figure 16.5: Schematic of the 64 Possible *pap* operon configurations, and the 192 reactions between the different configurations. The three configurations that allow transcription of the *pap* operon are circled. The sixteen of the configurations in the shaded region are locked OFF; they will not result in much *pap* transcription.

64 Lrp binding reactions, 64 Lrp unbinding reactions, and 64 DAM methylation events. These reactions are illustrated by the arrows in Figure 16.5. Because DAM methylation is irreversible, there are certain configurations, which will result in very little *pap* transcription; these configurations are shaded in gray. Once the operon reaches a "locked OFF" configuration, DNA replication is necessary before the *pap* can again be transcribed at a sizable level.

16.1.4 The PapI and PapB local regulatory proteins

The Pap switch is also highly dependent upon the effects of local *pap* regulator proteins, PapI and PapB. PapB and PapI are divergently transcribed, and are the key feedback signals in the Pap system. The mechanisms by which PapB and PapI affect the pap switch are well characterized as follows.

PapB is a 12 kDa regulatory protein that binds to DNA targets containing the sequence GACACAAAC [110] and plays an essential role in *pap* activation [8]. When the *pap* operon is in a production configuration (Lrp is bound to 4-5-6 but not to 1-2-3 and DAM has fully methylated site 2, but not site 5) then the *PapB* gene is transcribed. PapB, in turn, binds with high affinity near the *papI* promoter, and is essential for the initiation of *papI* transcription [110]. The production of PapB is subject to auto-regulation; at high levels of PapB, *papB* transcription is inhibited [30]. This auto-regulation appears to be due to the presence of a low affinity PapB binding site located overlapping the -10 hexamer RNA polymerase binding site in the *papB* promoter, although this has not been directly shown.

PapI is a small 8 kDa regulatory protein that interacts with low affinity to both Lrp [49] and specific DNA sequences within sites 2 and 5 [46]. When PapI is present in the system, it has been experimentally observed that the dissociation rate of Lrp is significantly decreased (Krabbe and Low, unpublished data). Possibly, this mechanism is achieved by PapI binding with *high* affinity to Lrp in complex with *pap* sites 2 and 5 [49, 46], thereby creating a stable PapI-Lrp-DNA complex at site 2 or 5. Without specific information regarding the exact mechanism of PapI's effect, it is reasonable to make the simplifying assumption:

Model Assumption 8: Lrp dissociation rate and Lrp binding affinities are functions of the level of PapI present in the system. Table 16.1 lists these dissociation rates and affinities for two levels of PapI population: low and at saturation.

The quantitative effect of a specific level of PapI on Lrp affinity and dissociation rates varies from site 2 to site 5 and also depends upon the methylation pattern of the GATC sequences (Hemi-0, Hemi-T, Hemi-B or Hemi-2). In particular, low levels of PapI have a much greater influence on the distal sites 4-5-6, especially when $GATC_{dist}$ is not fully methylated [46]. See Table 16.1. Hernday et al. showed that the PapI effect on a methylated site 5 reaches saturation at a very low level of about 5nM, while the effect of PapI on the proximal site 2 reaches saturation at a much higher level (greater than 600nM–see Ref [46], Figures 4 and 6). PapI appears to have no effect on the binding affinity at site 5, when the contained GATC sequence is fully methylated. Based upon the PapI response given in [46], the following assumption is made:

Model Assumption 9: The effect of PapI on the Lrp affinity is assumed to vary as:

$$k_D(PapI) = k_{D_{sat}} + (k_{D_0} - k_{D_{sat}})\exp(-\alpha[papI]),$$

where all k_D's are dependent upon the location (prox or dist) and the methylation as shown in Table 16.1, section 1. The parameter α is fit to match the results in [46] and is given in Table 16.1, section 5.

At present, there is a lack of quantitative information regarding the production, degradation and interactions of the local regulatory proteins PapB and PapI. Faced with this lack of information, it is natural to seek a simple consistent model, which can build upon as more information become available. For this reason, the two regulatory proteins are treated as one.

Model Assumption 10: Unless otherwise specified, PapI and PapB are assumed to occur in equal populations: $Pap := PapI = PapB$. In effect the two proteins are considered as a single generic Pap protein. This assumption will be relaxed in the study of the individual PapI and PapB proteins.

Before the protein Pap can be produced it must first go through the complicated process of transcription and translation. This process has been simplified to two steps–first RNAP binds to and unbinds from the *pap* operon in a stochastic event, and then Pap proteins are created in a stochastic event.

Model Assumption 11: RNA polymerase (RNAP) attach to the operon in a stochastic event, thereby initiating a production configuration. Production capability is ended when RNAP detaches from the operon. The binding and unbinding rates of RNAP depend upon the configuration of the *pap* operon as described in Assumptions 3 and 6.

Model Assumption 12: It is assumed that the generic Pap gene is transcribed and translated simultaneously and instantaneously during a single exponentially distributed stochastic event. Thus transcription and translation are combined into a single reaction: $DNA - RNAP \rightarrow DNA - RNAP + Pap$; the rate of this reaction is assumed to be $4Ns^{-1}$.

Model Assumption 13: The generic Pap protein is assumed to degrade as a non-linear stochastic event: $PapI \rightarrow \emptyset$, with rate $w = 10^{-4}[Pap]+10^{-6}[Pap][Pap-1]$.

For the simplified model, the negative feedback auto-regulation mechanism of PapB is included as follows:

Model Assumption 14: If RNAP is attached to the operon, it can be detached in a stochastic event with rate proportional to the square of the population of Pap. This simple mechanism auto-regulates the amount of Pap in the system.

16.1.5 Modeling of multiple generations

As discussed above, DAM methylation is an irreversible process. If left for very long periods of time, every GATC sequence would eventually become methylated (Methlylation pattern 16 in Figure 16.3). This combined with assumption 6 would suggest that every cell would eventually turn OFF and remain OFF. This, of course, is not the biological case. In a single generation, once the operon has reached one of the configurations in the shaded region of Figure 16.5, that cell will not produce more Pap until the subsequent generation. Thus, in order to model OFF to ON switching, it is necessary to consider multiple generations. This section describes the current model's treatment of multiple generations.

Model Assumption 15: Replication is modeled as a exponentially distributed stochastic event with fixed rate constant, c_{split}.

At time of replication, Lrp presumably dissociates from the DNA as a result of the DNA polymerase III replication machinery, and the gene forks into two hemi-methylated daughter strands [46]. Depending upon the methylation pattern of the DNA at time of replication, the daughter strands will have different initial configurations for the next generation. For example, methylation pattern 12 will generate one daughter with methylation pattern 7 and one with pattern 3 (see Fig. 16.3). After replication DAM again begins its competition with Lrp to remethylate the DNA. In order to track the evolution of a population arising from single cell, one may make the following model assumption:

Model Assumption 16a: For tracking of populations, it is assumed that in each replication event the mother cell simultaneously gives rise to two hemi-methylated daughters. Table 16.2, section 1, provides the stoichiometry replication events under this assumption.

Using this assumption, one can simulate the evolution of a population beginning from a single cell for many generations and obtain a very detailed description of how one colony might develop over a handful of generations. Although the information obtained through this method is very easily interpreted and closely resembles patterns observed under the microscope, the actual method requires tracking of an exponentially increasing number of cells. For more general results and faster predictions, the proportion of cells in each state can be found more easily by utilizing the following assumption and tracking only a single cell at a time:

Model Assumption 16b: For tracking of a single cell, it is assumed that in each replication event the mother cell gives rise to one of two possible hemi-methylated daughters; each daughter has equal probability. Table 16.2, section 2, provides the stoichiometry replication events under this assumption.

In addition to the splitting of the *pap* DNA, the replication event also results in a redistribution of the generic Pap protein-some goes to each daughter cell. This distribution can be modeled in many different manners, but the following assumption has been made.

Model Assumption 17: Each daughter cell has half the amount of Pap protein as did the mother.

With these assumptions, the pap model can be constructed and is thoroughly analyzed in the following section.

16.2 Analysis of the Pap Switch

Under the assumptions in the previous section there are 128 configurations in which the *pap* operon may be found (4 Lrp binding patterns × 16 methylation patterns × 2 RNAP binding patterns). Furthermore, the population of the generic Pap protein can be any non-negative integer number such that there are an infinite number of possible states in which the system may be found. The number of reactions channels linking one configuration to another is also quite high. There are 64 Lrp binding events, 64 Lrp unbinding events, 64 methylation events, 64 RNAP binding events, 64 RNAP unbinding events, 64 Pap transcription events,

1. Replication stoichimetries under assumption 16a			
	Mother Meth.	Daughter Meth.	Stoichiometry
	M_1	$\{M_1, M_1\}$	$\{M_1 \rightarrow M_1 + M_1\}$
	M_2	$\{M_2, M_1\}$	$\{M_2 \rightarrow M_2 + M_1\}$
	M_3	$\{M_1, M_3\}$	$\{M_3 \rightarrow M_1 + M_3\}$
	M_4	$\{M_1, M_4\}$	$\{M_4 \rightarrow M_1 + M_4\}$
	M_5	$\{M_5, M_1\}$	$\{M_5 \rightarrow M_5 + M_1\}$
	M_6	$\{M_2, M_3\}$	$\{M_6 \rightarrow M_2 + M_3\}$
	M_7	$\{M_7, M_1\}$	$\{M_7 \rightarrow M_7 + M_1\}$
	M_8	$\{M_2, M_4\}$	$\{M_8 \rightarrow M_2 + M_4\}$
	M_9	$\{M_1, M_9\}$	$\{M_9 \rightarrow M_1 + M_9\}$
	M_10	$\{M_3, M_5\}$	$\{M_{10} \rightarrow M_3 + M_5\}$
	M_11	$\{M_5, M_4\}$	$\{M_{11} \rightarrow M_5 + M_4\}$
	M_12	$\{M_7, M_3\}$	$\{M_{12} \rightarrow M_7 + M_3\}$
	M_13	$\{M_2, M_9\}$	$\{M_{13} \rightarrow M_2 + M_9\}$
	M_14	$\{M_5, M_9\}$	$\{M_{14} \rightarrow M_5 + M_9\}$
	M_15	$\{M_7, M_4\}$	$\{M_{15} \rightarrow M_7 + M_4\}$
	M_16	$\{M_7, M_9\}$	$\{M_{16} \rightarrow M_7 + M_9\}$
2. Replication stoichimetries under assumption 16b			
	Mother Meth.	Daughter Meth.	Stoichiometry
	M_1	$\{M_1, M_1\}$	$\{M_1 \rightarrow M_1, M_1 \rightarrow M_1\}$
	M_2	$\{M_2, M_1\}$	$\{M_2 \rightarrow M_2, M_2 \rightarrow M_1\}$
	M_3	$\{M_1, M_3\}$	$\{M_3 \rightarrow M_1, M_3 \rightarrow M_3\}$
	M_4	$\{M_1, M_4\}$	$\{M_4 \rightarrow M_1, M_4 \rightarrow M_4\}$
	M_5	$\{M_5, M_1\}$	$\{M_5 \rightarrow M_5, M_5 \rightarrow M_1\}$
	M_6	$\{M_2, M_3\}$	$\{M_6 \rightarrow M_2, M_6 \rightarrow M_3\}$
	M_7	$\{M_7, M_1\}$	$\{M_7 \rightarrow M_7, M_7 \rightarrow M_1\}$
	M_8	$\{M_2, M_4\}$	$\{M_8 \rightarrow M_2, M_8 \rightarrow M_4\}$
	M_9	$\{M_1, M_9\}$	$\{M_9 \rightarrow M_1, M_9 \rightarrow M_9\}$
	M_10	$\{M_3, M_5\}$	$\{M_{10} \rightarrow M_3, M_{10} \rightarrow M_5\}$
	M_11	$\{M_5, M_4\}$	$\{M_{11} \rightarrow M_5, M_{11} \rightarrow M_4\}$
	M_12	$\{M_7, M_3\}$	$\{M_{12} \rightarrow M_7, M_{12} \rightarrow M_3\}$
	M_13	$\{M_2, M_9\}$	$\{M_{13} \rightarrow M_2, M_{13} \rightarrow M_9\}$
	M_14	$\{M_5, M_9\}$	$\{M_{14} \rightarrow M_5, M_{14} \rightarrow M_9\}$
	M_15	$\{M_7, M_4\}$	$\{M_{15} \rightarrow M_7, M_{15} \rightarrow M_4\}$
	M_16	$\{M_7, M_9\}$	$\{M_{16} \rightarrow M_7, M_{16} \rightarrow M_9\}$

Table 16.2: Pap replication stoichiometries for various assumptions.

and one Pap degradation event. In all this totals over 350 different reaction types, each with their own distinct stoichiometries and state dependent reaction rates. With the inclusion of multiple generations, this number is much larger. About half of these reactions and configurations are illustrated in Fig.
16.5, which does not include RNAP or replication events.

The following sections use the FSP tools described above in order to analyze the effects that various of the chemical players have on the Pap switch.

16.2.1 Wild-type Pap analysis

This section begins with an analysis of the wild-type Pap behavior in terms of both single generations as well as over multiple generations. The single generation analyses consider three different initial configurations for the cell. For each initial cell, Fig.
16.6 shows the probability distribution for the population of Pap protein at the end of thirty minutes (a typical generation length). From the figure, one can observe that this distribution has a bimodal form. Low Pap expression levels (left peak) correspond to OFF cells and high expression levels (right peak) correspond to ON cells. For convenience, let any cell with more than ten Pap molecules be considered ON and all others be considered OFF.

The prototypical OFF cell has methylation pattern 6 (see Fig.
16.3) and contains no *Pap* protein. Immediately after replication, this mother cell gives rise to two daughter cells: one Hemi-T and the other Hemi-B, which correspond to methylation patterns 2 and 3 in Fig. 16.3, respectively. Because Lrp binding affinity has a non-symmetric dependence on the top or bottom methylation of the $GATC_{dist}$ site (See Table 16.1 and [46]), these two daughter cells

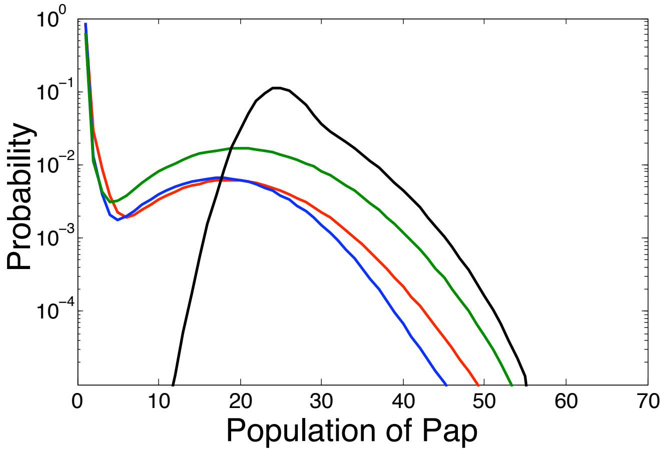


Figure 16.6: Probability distribution of Pap molecules in wild-type *E. coli* at the completion of a single cell cycle beginning at various initial gene configurations. The red and blue curves correspond to initial conditions with no Pap molecules and methylation patterns 2 and 3, respectively. The green and black curves correspond to initial pattern 4 with 0 or 30 molecules of Pap, respectively.

exhibit different behavior from one another (compare red and blue curves in Fig. 16.6).

A prototypical ON cell has methylation pattern 11 in Fig. 16.3 and a significant amount of *Pap*. Upon replication, this cell gives rise to two hemi-methylated daughter cells of methylation patterns 4 and 5. With the absence of experimentally measured affinities of Lrp binding to the proximal sites (1-2-3), we have assumed that these affinities do not depend upon the methylation pattern, and the mode shows no difference between the top and bottom methylated $GATC_{prox}$ initial configurations. In Fig. 16.6, the green curve represents the probability distribution of Pap molecules after one generation for cells beginning with methylation pattern 4 or 5 and with no Pap present in the system. From the figure it is clear that methylation at the proximal site (green line) results in far more Pap that methylation at the distal site (red and blue curves. One can also consider the initial condition where the cell begins at methylation pattern 4 or 5 and where

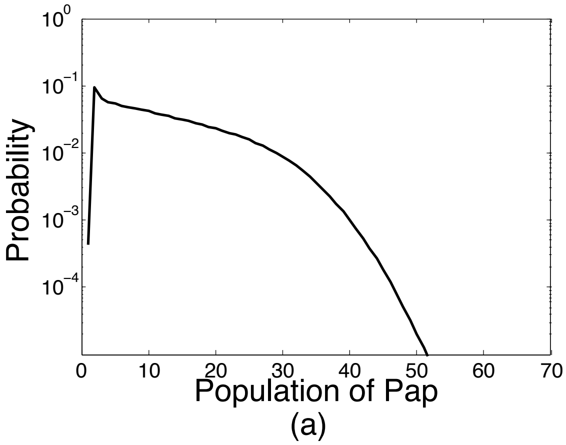


Figure 16.7: Probability distribution of Pap molecules in wild-type *E. coli* at the completion of ten hours or approximately twenty generations.

the cell begins with thirty molecules of Pap from the previous generation (black line in Fig. 16.6). In this case, the positive feedbacks mechanism of Pap make it far more likely to have significant Pap than in any of the previous cases.

After ten hours spanning multiple generations, the model predicts that the distribution of Pap molecules reaches the stationary distribution shown in Fig. 16.7. In addition to looking at the population of Pap molecules, it is of interest to examine the methylation and Lrp binding patterns as shown in Fig. 16.8 and 16.9. In the subsequent sections, we will see how these patterns are effected by the populations of various chemical players in the Pap switch.

16.2.2 Effect of DNA Adenine Methylase

DNA Adenine Methylase has a number of competing effects on the Pap system. Methyation of the $GATC_{prox}$ sites helps protect the *pap* operon from Lrp binding near the Pap promoter. Without this methylation, the cell will not produce significant quantities of Pap. Conversely, methylation of the $GATC_{dist}$ sites blocks the Lrp binding in that location that is required fro Pap transcription.

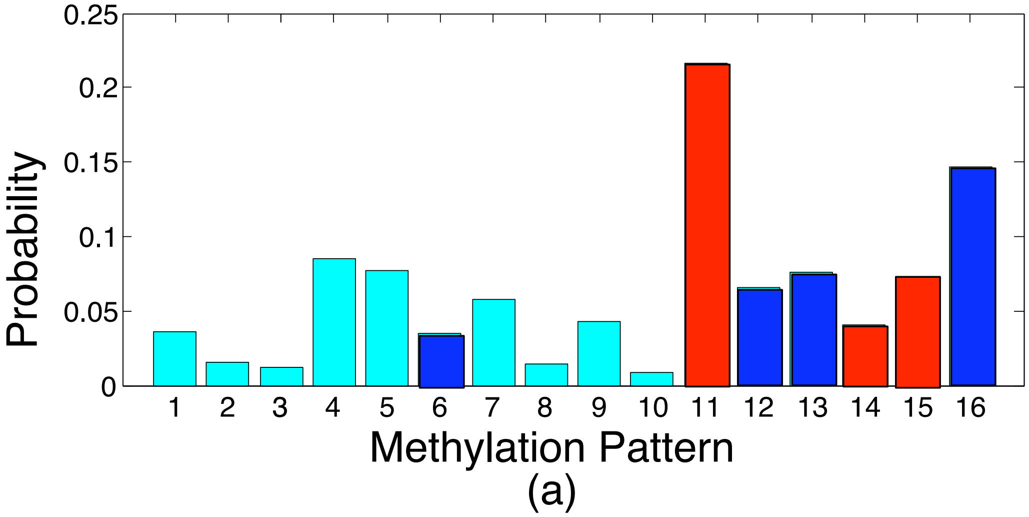


Figure 16.8: Probability distribution of methylation patterns in wild-type *E. coli* at the completion of ten hours or approximately twenty generations. The methylation patterns are grouped into three categories: (*blue*) Methylation patterns 6, 12, 13, and 16 are fully methyated at $GATC_{dist}$–these are *Over Methylated*. (*cyan*) Methylation patterns 1, 2, 3, 4, 5, 7, 8, 9 and 10 are not fully methyated at $GATC_{prox}$–these are *Under Methylated*. (*red*) the remaining methylation patterns 11, 14, and 15 are the fully *Productive* patterns.

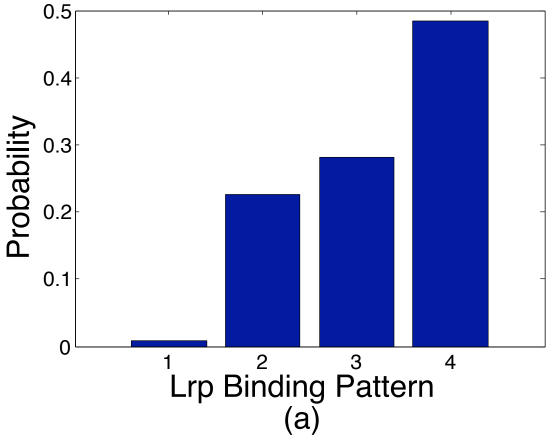


Figure 16.9: Probability distribution of Lrp binding patterns in wild-type *E. coli* at the completion of ten hours or approximately twenty generations. Pattern 1–no Lrp is bound. Pattern 2–Lrp bound to Distal only. Pattern 3–Lrp bound to proximal only. Pattern 4–Lrp bound at both distal and proximal.

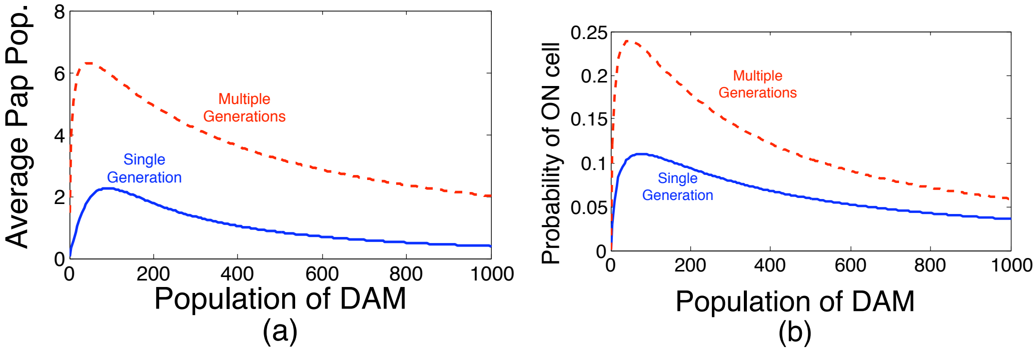


Figure 16.10: (a) Mean number of Pap molecules versus the population of DAM at the completion of a single cell cycle. (b) Probability of turning ON (expressing more than ten molecules of Pap) in a single generation versus the number of DAM molecules in the system. For both of these plots, the initial condition was methylation pattern 2, with no bound Lrp an no Pap. Solid lines correspond to a the Pap levels after a single generation beginning with a cell in methylation pattern 2, and dashed lines correspond to a population after ten hours.

In order to explore how the model captures this tradeoff, the model simulates the switching response for a large number of different DAM populations. Fig. 16.10a plots the average number of Pap molecules per cell versus the population of DAM, and Fig. 16.10b plots the probability of an ON cell versus the population of DAM, where an ON cell is defined as as a cell that contains more than ten molecules of Pap. From either figure, one can see that the model predicts that DAM is required to initiate Pap transcription (there is no Pap at low DAM levels), and that DAM shuts down Pap transcription at high levels.

To further examine the effect of DAM on the Pap switch, one should examine how different levels of DAM effect the methylation and Lrp binding patterns. In terms of Pap expression, the sixteen methylation patterns can be categorized into three important groups: (*i*) Methylation patterns 6, 12, 13, and 16 are fully methyated at $GATC_{dist}$–these are *Over Methylated*. (*ii*) Methylation patterns 1, 2, 3, 4, 5, 7, 8, 9 and 10 are not fully methyated at $GATC_{prox}$–these are *Under*

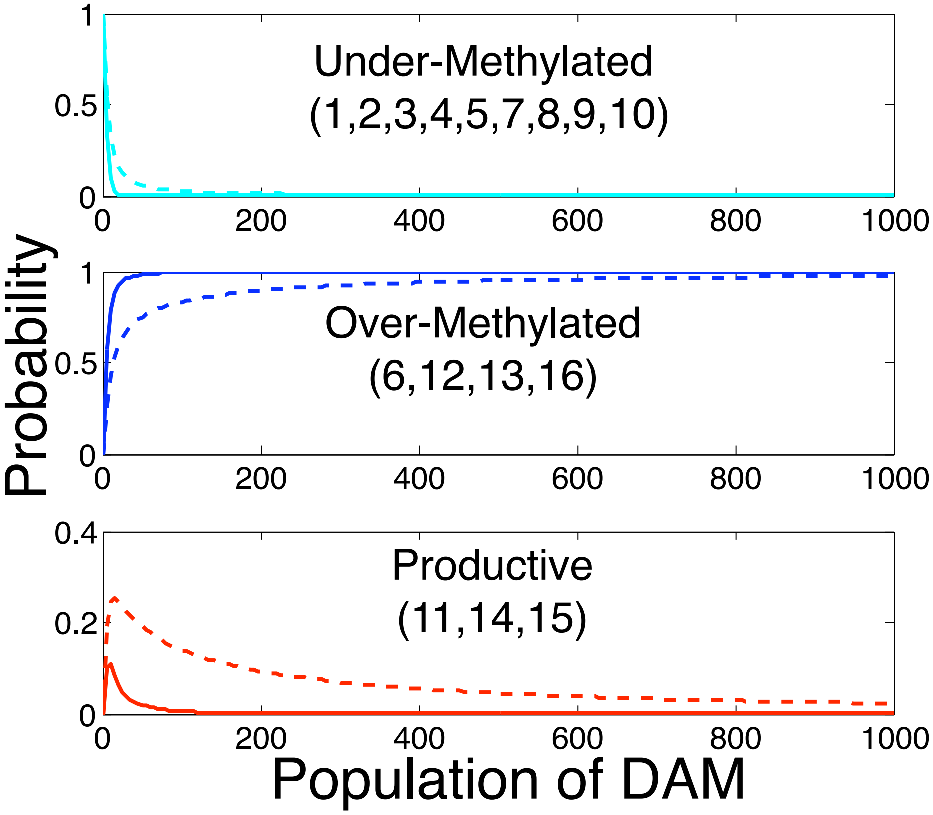


Figure 16.11: Effect of DAM population on *pap* Methylation Patterns. Solid lines correspond to a the Pap levels after a single generation beginning with a cell in methylation pattern 2, and dashed lines correspond to a population after ten hours.

Methylated. Finally, (*iii*) the remaining methylation patterns 11, 14, and 15 are the fully *Productive* patterns. Fig. 16.11 shows how the probabilities of these three categories change as the level of DAM increases. At low levels of DAM the majority of the cells are under-methylated. At high levels, the majority are over-methylated. Only at the moderate DAM population levels are there a significant number of cells expressing one of the productive methylation patterns.

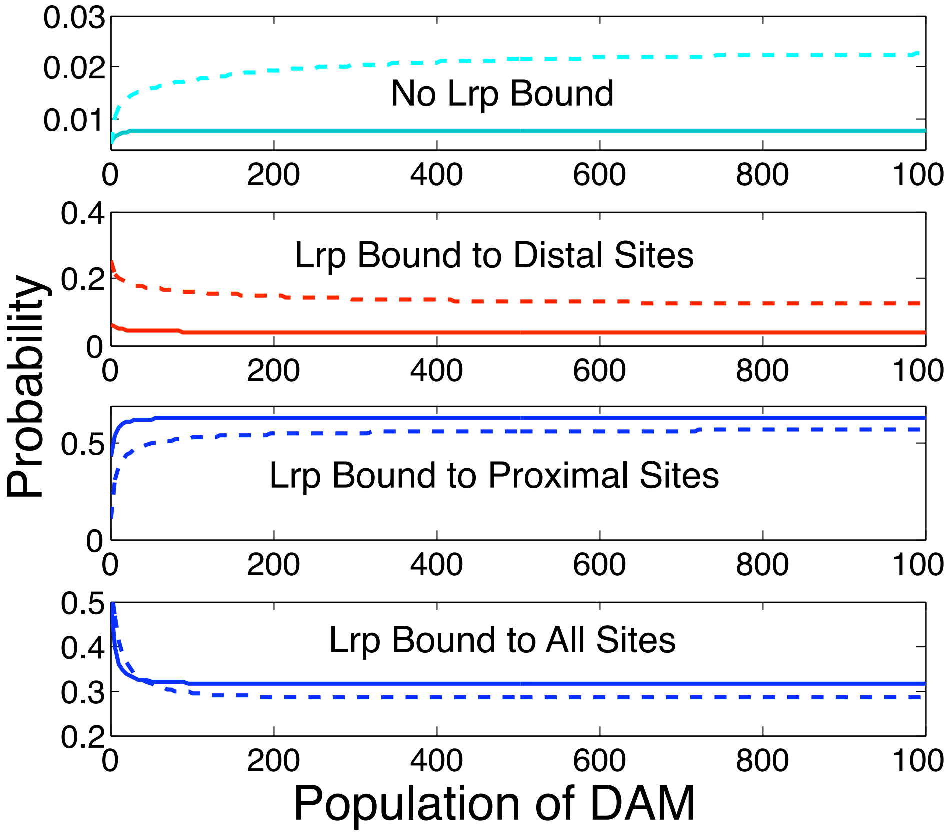


Figure 16.12: Effect of DAM population on *pap* Lrp Binding Patterns. (cyan) The *under-bound* phase where Lrp is not bound to any site. (red) The *production* phase where Lrp is bound to the distal site. (blue) The *over-bound* phases where Lrp is bound to the proximal sites. Solid lines correspond to a the Pap levels after a single generation beginning with a cell in methylation pattern 2, and dashed lines correspond to a population after ten hours.

16.2.3 Comparison with experimental results

In order to validate the model, Fig. 16.10 also provides experimental observations from a similar DAM titration study performed in David Low's group at UCSB.[2] In this study, the gene coding for DAM was placed under the control of an externally applied inducer isopropyl β-D-1-thiogalactopyranoside (IPTG). In the experiment, it was not possible to directly control or measure the populations of DAM, but the populations of DAM was reasonably expected to vary similarly to the applied concentration of IPTG. By inserting green florescent protein (GFP)just down stream of the gene for PapB the experimentalists were able to approximately measure the average levels of Pap protein in a cellular population. In Fig. 16.10 the top axes correspond to the levels of IPTG in the system, and the right axes to the measured level of Pap.

Although a direct comparison between the model predictions and the experimental observation is not possible, their qualitative behavior is very similar and suggests that the current model does indeed capture the effect of Dam on the systems operation.

16.2.4 Effect of Leucine Responsive regulatory Protein

Like DAM, Lrp can also have a diverse range of effects on the Pap system. If Lrp binds in the proximal location, it will block Pap transcription. If it fails to bind at the distal location, transcription will not be fully activated. Fig. 16.14 shows the predicted effect of changing Lrp population levels for a wild-type level of DAM (130 molecules). To further illustrate the tradeoff between DAM and Lrp, Fig. 16.15 shows the contour plots of the average Pap population as a function of

[2] Unpublished data.

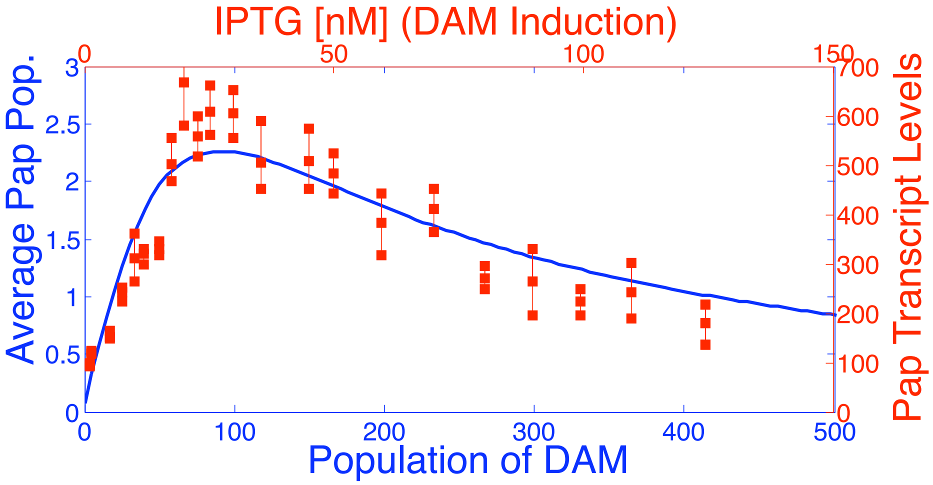


Figure 16.13: Model predictions and experimental measurements of Pap pili OFF to ON switching behavior in response to varying levels of DAM expression for wild-type *pap*. (blue) Predicted OFF to ON switch rate (left axis) for different levels of DAM population (bottom axis). (red) Experimentally measured Pap transcript levels (right axis) under different DAM expression levels (top axis). The absolute DAM levels have not yet been determined, but are expected to be linearly related to IPTG concentration over the concentration range shown. These data were obtained with *E. coli pap-lac* containing *dam* under control of *plac* [106].

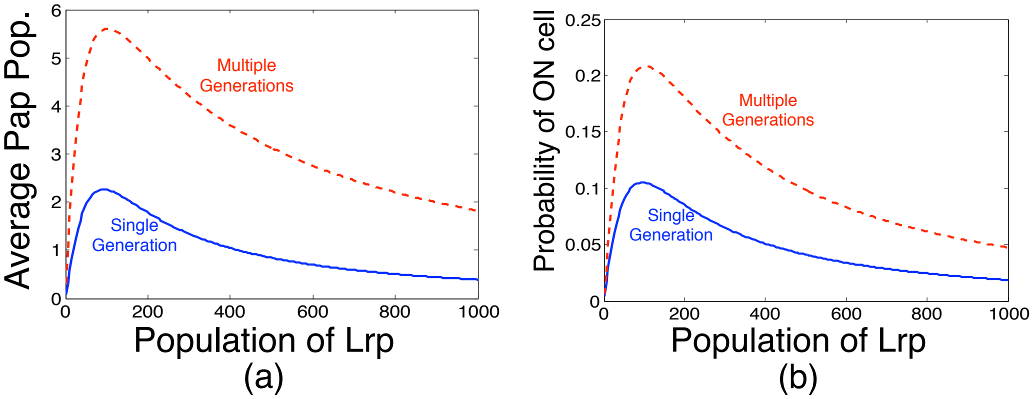


Figure 16.14: (a) Mean number of Pap molecules versus the population of Lrp at the completion of a single cell cycle. (b) Probability of turning ON (expressing more than ten molecules of Pap) in a single generation versus the number of Lrp molecules in the system. For both of these plots, the initial condition was methylation pattern 2, with no bound Lrp an no Pap. Solid lines correspond to a the Pap levels after a single generation beginning with a cell in methylation pattern 2, and dashed lines correspond to a population after ten hours.

both DAM and Lrp. From the figure, it is obvious that as DAM increases, more Lrp is needed to successfully compete and induce maximal Pap production.

Because Lrp competes with DAM, it also has a large effect on the methylation patterns and Lrp binding patterns of the *pap* operon, as is shown in Figures 16.16 and 16.17. At low levels of Lrp, the distal site will remain free from Lrp and Pap transcription will not be fully initiated. At high levels, Lrp will overcome its mutual exclusion and both sites will be bound with Lrp, thus shutting off Pap transcription. Lrp also has an indirect effect through its influence on the methylation of the *pap* operon. At low Lrp levels, DAM has the edge in the competition and more of the cells will reach the over-methylated stage, and fewer will be able to produce Pap. Conversely, at high levels, lrp will block DAM from accessing the operon, and few cells will reach one of the productive methylation patterns.

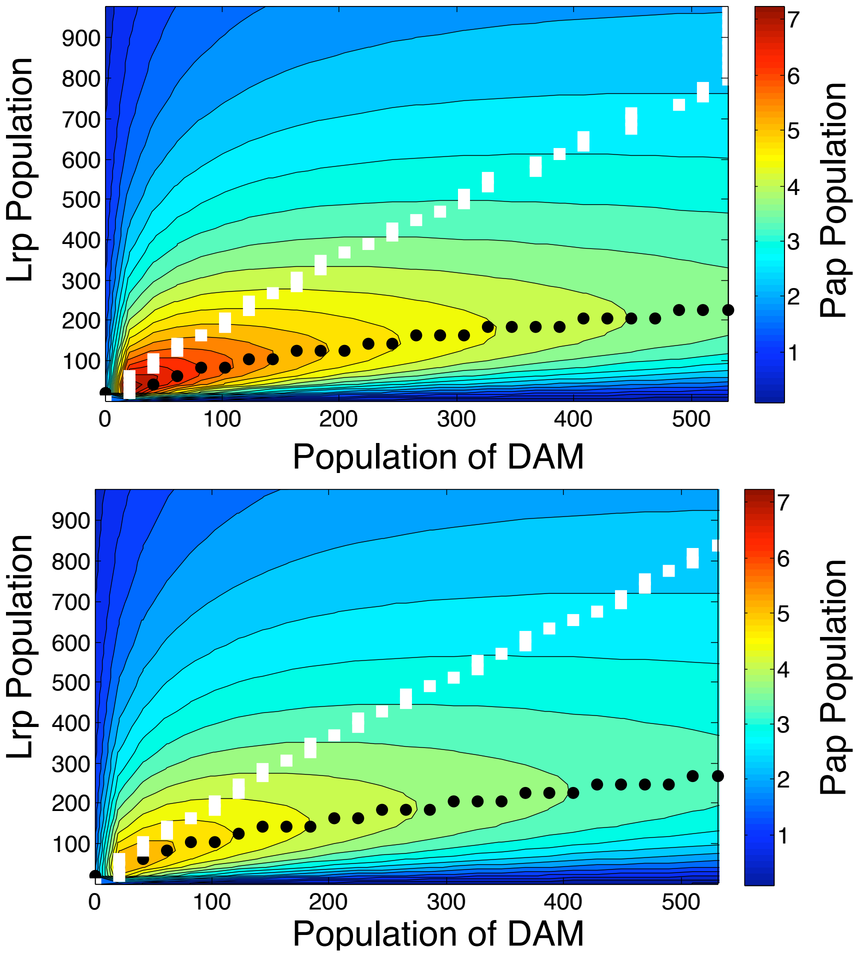


Figure 16.15: Contour plot showing the levels of Pap proteins versus levels of DAM and Lrp populations after multiple generations. The white squares correspond to the level of DAM that produces the most Pap for each level of Lrp. The black circles correspond to the level of Lrp that produces the most Pap for each level of DAM. (top) With the PapI feedback mechanism. (bottom) Without the PapI feedback mechanism.

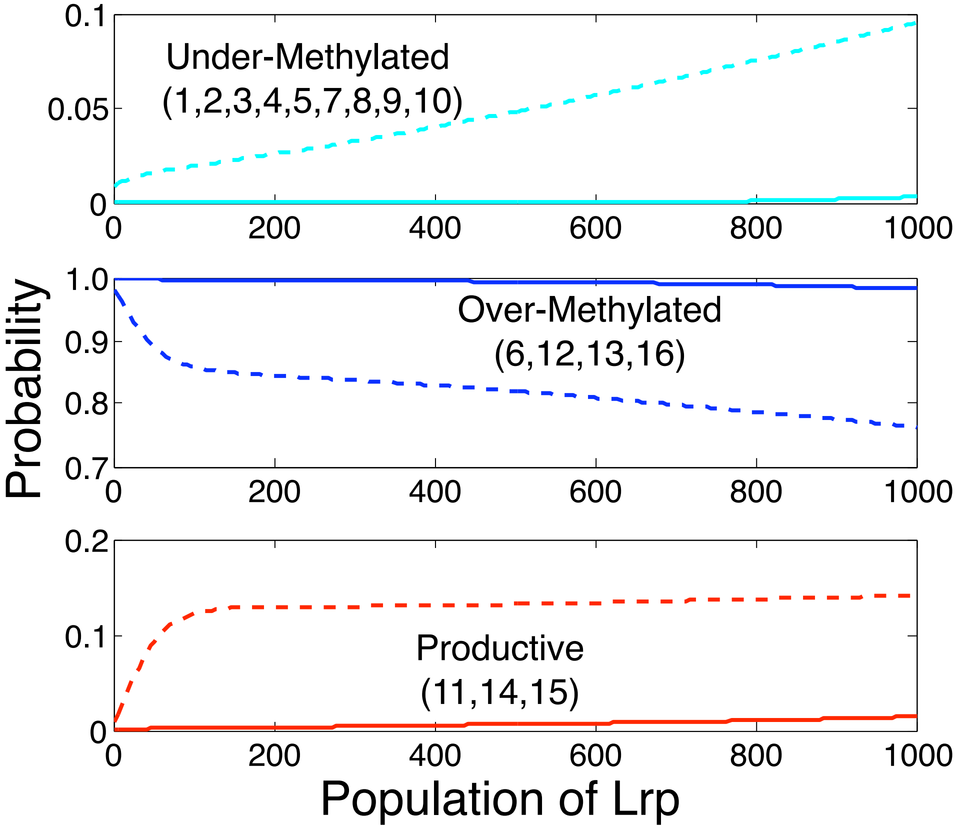


Figure 16.16: Effect of Lrp population on *pap* Methylation Patterns. Solid lines correspond to a the Pap levels after a single generation beginning with a cell in methylation pattern 2, and dashed lines correspond to a population after ten hours.

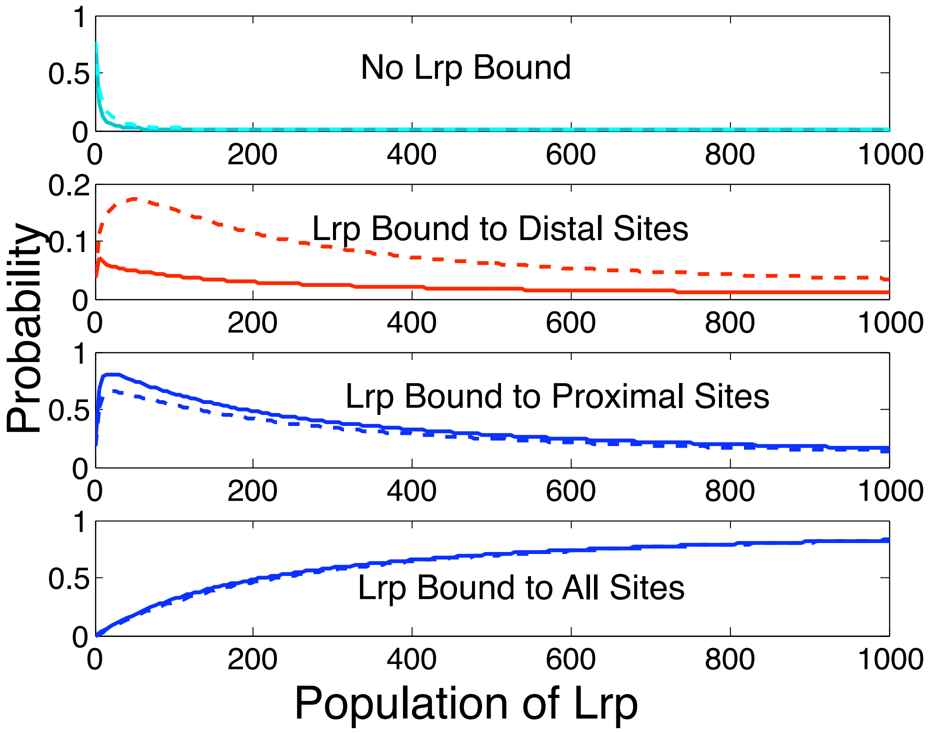


Figure 16.17: Effect of Lrp population on *pap* Lrp Binding Patterns. (cyan) The *under-bound* phase where Lrp is not bound to any site. (red) The *production* phase where Lrp is bound to the distal site. (blue) The *over-bound* phases where Lrp is bound to the proximal sites. Solid lines correspond to a the Pap levels after a single generation beginning with a cell in methylation pattern 2, and dashed lines correspond to a population after ten hours.

16.2.5 Effect of PapI

In this model, the feedback effect of PapI results in an increase in the Lrp binding affinity to the proximal and distal sites. At low levels of PapI, this effect is far stronger at the distal sites than at the proximal sites. The result is that Lrp is more likely to bind at the distal site and the system will remain in a production pattern for a longer portion of the cell's lifetime.

In order to test the importance of this feedback mechanism in this model, the model has also been used to predict the Pap levels in which the PapI feedback mechanism has been turned off–that is the Lrp binding rates remain at the same levels as when there is no Pap in the system (PapI minus mutant. Fig. 16.18 shows the ON portion of a population after ten hours versus DAM and Lrp with and without the PapI feedback mechanism, and Fig. 16.15(bottom) shows the contour levels of the expression versus DAM and Lrp. The model has also been used to analyze the system in which the Lrp binding rates are set at the values corresponding to 5nM of PapI independent of the actual Pap levels (PapI plus mutant). From the plots, one can see that the current model indeed captures the fact that PapI helps the system to retain the ON state (compare dotted, dashed and solid lines in Fig. 16.18 and the top and bottom plots of Fig. 16.15). Specifically the PapI plus mutant (dotted line) is ON far more often than the wild type *E. coli* and the PapI minus mutant (dashed line) is OFF more than the wild type. However, experimental results show that cells without PapI remain in a locked OFF state, and the current model does not appear to capture the full scale of PapI importance.

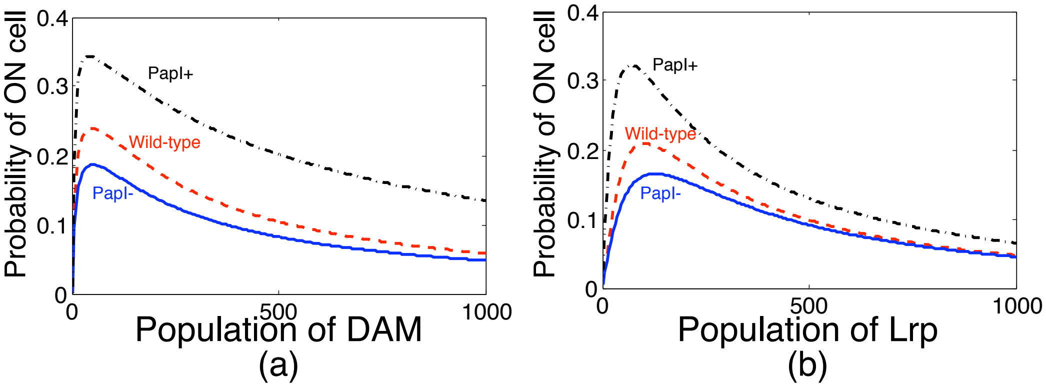


Figure 16.18: Effect of PapI Feedback on the Pap Switch. Red dashed lines–With wild-type PapI enhanced Lrp binding. Solid blue lines–Without no PapI enhanced Lrp binding. Black dotted lines–With PapI feedback and elevated Pap levels. (a) ON rate versus population of DAM. (b) ON rate versus population of Lrp.

16.2.6 Effect of various *pap* mutations

The above model has also been used to predict the behavior of four experimentally constructed mutants that have been considered in earlier studies by Lows group [13, 73, 45]. All simulations begin with a single cell in which there is no previous methylation of GATC sites 2 or 5 (methylation pattern number 1 in Fig. 16.3), there is no Lrp bound to either the proximal or distal sites, and the initial population of PapI is set to zero. The simulations were aimed at predicting the pap switching behavior over multiple generations as functions of the specific mutations and the concentrations of DAM and Lrp.

In mutants 1 and 2, the adenine in the GATC as site 2 or 5 are replaced with cytosine, respectively. This blocks DAM methylation at the proximal or distal locations, respectively. In mutant 1, the inability of DAM to methylate the proximal site allows Lrp to bind more readily, and blocks Pap transcription (see blue line in Fig. 16.19). In mutant 2, DAM cannot methylate the distal site, which encourages Lrp to bind there and initiate transcription (see black line

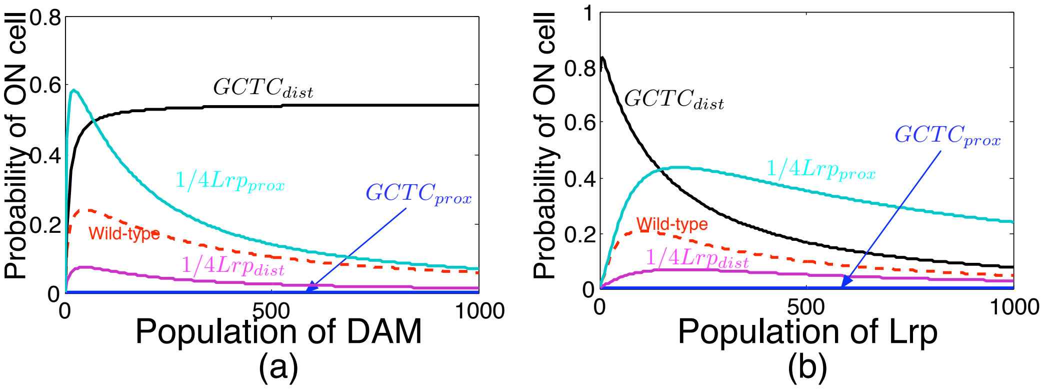


Figure 16.19: Effect of various mutations on the Pap Switch. Red dashed lines–Wild-type Pap. Blue–Proximal site 2 cannot be methylated. Black–Distal site 5 cannot be methylated. Cyan–Lrp has 1/4 wild-type affinity at proximal sites 1-2-3. Magenta–Lrp has 1/4 wild-type affinity at distal sites. (a) ON rate versus population of DAM. (b) ON rate versus population of Lrp.

in Fig. 16.19). In this mutation, more DAM in the system always increases Pap transcription. Mutations 3 and 4 decrease Lrp binding affinities proximal or distal locations, respectively. In mutant 3 the decreased affinity to the distal site causes more of the Lrp to bind at the proximal site and shuts down Pap transcription (see magenta line in Fig. 16.19). Conversely, in mutant 2 the decreased affinity for the proximal site helps to initiate transcription (see cyan line in Fig. 16.19). For all four mutants, the numerical predictions match experimental observations from [73, 13, 45]. For ease of comparison, Table 16.3 summarizes the experimental results for these mutations. In all cases the predicted behavior matches the qualitative behavior of the observed experiments.

Gene Alteration	Low DAM	Wild type	High DAM
Wild-type	OFF	switching	OFF
$GCTC_{prox}$	OFF	OFF	OFF
$GCTC_{dist}$	OFF	ON	very ON
$1/4Lrp_{prox}$	ON	ON	ON
$1/4Lrp_{dist}$	OFF	OFF	OFF

Table 16.3: Experimentally observed Pap switching behavior for wild-type *E. coli* and four constructed mutations.

Chapter 17

Conclusions and Future Work

Many important biochemical processes, especially those involving gene regulatory networks, occur on a very small scale, where mass action kinetics are not valid and the system is dominated by fluctuations. As the size of the system shrinks to the point where only a few copies exist of certain important chemical species, these species must be described not by concentrations but by integer populations numbers. In this regime continuous variable deterministic models are unrealistic, and discrete stochastic models are necessary. Here the system can no longer be usefully described by a single trajectory of the system through the state space; that trajectory may be only one of many wildly different possibilities. Instead, the system must be described by probabilities that the system will have certain traits at certain times. For discrete population chemically reacting systems, the evolution of this probability distribution is well understood to evolve according to a system of equations known variously as the chemical master equation (CME), master equation, or forward Kolmogorov equation.

This dissertation focuses on the Finite State Projection (FSP) method, the FSP algorithm for the approximate solution of the CME, and various reductions to improve the efficiency of the FSP approach. Unlike previous Monte Carlo analyses, the FSP directly computes the system's probability density vector at a given time and does not require the computation of large numbers of process realizations. In the case of any Markov process containing only a finite number of states, the FSP method provides an exact analytical solution. When the number of possible states is infinite or extremely large, the approximate solution on the projected space guarantees upper and lower bounds on the solution of the true system. The FSP algorithm provides a systematic means of increasing the size of the finite state projection until these bounds are within any pre-specified error tolerance.

Although the original finite state projection method can significantly reduce the order of the chemical master equation for many problems, this initial reduction is not sufficient for all systems. Fortunately, the FSP approach is amenable to numerous modifications, which can considerably improve upon the method's range and potency. This dissertation considers many of these modifications. Chapters 6 and 11.2.1 present two methods which allow one to obtain a minimal realization for relevant portions of the master equation. The first and simplest method simply determines the configurations that are both observable from the output as well as controllable from the initial condition and removes the remaining extraneous states. The second approach in Chapter 11.2.1 uses balanced truncation to reduce the system. Chapter 7 develops the Slow Manifold FSP approach, which relies upon projecting the dynamics of the full FSP onto its lower dimensional slow manifold. Chapters 8 and 9 present the Multiple Time Interval FSP algorithm, which is essentially an incremental approach to solving the original FSP using different projections at different periods of time. Chapter 10 presents an interpolation based FSP approach, in which one chooses a small subset of configuration points and assumes (*i*) that the probability distribution varies linearly between these points and (*ii*) that the resulting model has linear dynamics. All of the reduction approaches presented here can easily be used in conjunction with one another such that the greatest reductions can often be achieved by sequentially applying two or more methods.

While the practical limits of the finite projection based approach are yet unknown, future implementations will greatly expand the class of problems for which the FSP is an efficient and versatile tool for stochastic analysis. A few of the planned improvements for the FSP include the following: (1) Variable time step FSP implementations similar to that in Chapter 9 could allow for the use of long time steps in the state space regions where the distribution spreads slowly, and short time steps in the regions where the distribution spreads more quickly. (2) One could use higher order shape functions or wavelets for the interpolation reduced FSP approach. These shape functions could be allowed to adapt over time as the distribution evolves. (3) Many of the current model reduction approaches utilize algorithms originally developed for densely connected systems. The ultra-sparsity of the master equation could be more fully exploited. (4) Many of the FSP implementation and reductions could in principle be modified to be run on multiple processors. This is particularly evident in the case of the multiple time interval FSP and the computation of the generator in the slow manifold FSP method.

Even with the improvements outlined above, the FSP approach will likely never fully replace all other available stochastic methods such as stochastic simulation algorithms, stochastic differential equations or moment closure approaches. Each of these has its own particular advantages and disadvantages. The FSP method is very fast and precise for systems in which the number of possible configurations is small (or if the important dynamics of the master equation are sufficiently low in dimension). However, for large systems with many interacting chemical species, the FSP approach suffers greatly from the curse of dimensionality and cannot be applied. Furthermore, although the majority of the approaches explored here can be fully automated, the actual coding is more complicated than that of the stochastic simulation algorithm. For widespread accessibility beyond the engineering community, a user-friendly FSP software package remains to be developed. Conversely, stochastic simulations and SDEs are typically far easier for the lay-person to implement and require only the simplest computational tools. Furthermore, such Monte Carlo approaches can be applied to far more complex problems. For many systems, trajectories may take hours, days, or longer to generate, but if researchers seek only to explore overriding qualitative trends, a few realizations may suffice. However, Monte Carlo approaches have very poor convergence for the solution of the master equation, and are therefore very inefficient in the analysis of rare events. Systems with relatively simple distribution shapes can be adequately captured with a few low order moments evolving according to low dimensional nonlinear systems, that can be far faster to solve than the much higher dimensional linear ODEs of the FSP. However, more complicated, especially multi-modal, distributions will be very poorly captured with such low order approximations. It is envisioned that all of the FSP reductions, moment closure techniques, and Monte Carlo algorithms can be linked together into for their mutual benefit. Where one method fails, some combination of others may succeed. This hybridization of methods is a large open area for ongoing research.

In cases where the FSP approach succeeds, it can provide a wealth of information about a stochastic model. In the original FSP, the projection is done in such a way as to obtain an accuracy guarantee on the solution of the full master equation. This error can be made very small and can enable the comparison of slightly different systems. In turn this precision enables sensitivity analysis and system identification as is discussed in Chapter 12. In a different direction of analysis, one can change how this projection is made to gather different types of information. For example, Chapter 11 demonstrates how this term may be used to (i) directly determine the statistical distributions for stochastic switch rates, escape times, trajectory periods, and trajectory bifurcations, and (ii) evaluate how likely it is that a system will express certain behaviors during certain intervals of time.

The FSP methods were effectively demonstrated on many real biological examples: a toy model of the heat shock mechanism in *E. coli*, a genetic toggle model, and a detailed model of the *pap-Pili* epigenetic switch in *E. coli.* In each case, the FSP method generates the probability density vector to describe the process at specific points in time. Chapters 13 through 15 have compared the accuracy and efficiency of the FSP and popular Monte Carlo methods such as the SSA, τ leaping algorithms and SSA approaches with time scale separation based reductions. In many of these examples, the FSP algorithm outperforms Monte Carlo methods, especially when computing the probability of unlikely events, such as *pap* OFF to ON switching. These examples suggest that the finite state projection and its various reduction schemes provide a very promising toolbox–especially in the field of system's biology, where very small chemical populations are common and in which unlikely events may be of critical importance.

Bibliography

[1] R. Allen and P. Frenkel, D. Rein ten Wolde. Forward flux sampling-type schemes for simulating rare events: Efficiency analysis. *J. Chem. Phys.*, 124(194111), May 2006.

[2] R. Allen and P. Frenkel, D. Rein ten Wolde. Simulating rare events in equilibrium or nonequilibrium stochastic systems. *J. Chem. Phys.*, 124(024102), Jan. 2006.

[3] R. Allen, P. Warren, and P. Rein ten Wolde. Sampling rare switching events in biochemical networks. *Phys. Rev. Lett.*, 94(018104), Jan. 2005.

[4] A. Antoulis. *Approximation of Large-Scale Dynamical Systems*. SIAM, 2005.

[5] J. Aparicio and H. Solari. Population dynamics: Poisson approximation and its relation to the langevin process. *Physical Review Letters*, 86(18):4183–4186, April 2001.

[6] A. Arkin, J. Ross, and McAdams H. Stochastic kinetic analysis of developmental pathway bifurcation in phage λ-infected escherichia coli cells. *Genetics*, 149:1633–1648, 1998.

[7] J. Badfa, P. Benner, R. Mayo, and E. Quintana-Orti. Parallel algorithms

for balanced truncation model reduction of sparse systems. *Applied Parallel Computing*, 3732:267–275, 2006.

[8] M. Baga, M. Goransson, S. Normark, and B. Uhlin. Transcriptional activation of a pap pilus virulence operon from uropathogenic escherichia coli. *Embo J.*, 4(13B):3887–3893, 1985.

[9] L. B. Blyn, B. A. Braaten, C. A. White-Ziegler, D. H. Rolfson, and D. A. Low. Phase-variation of pyelonephritis-associated pili in escherichia coli: Evidence for transcriptional regulation. *EMBO J.*, 8:613–620, 1989.

[10] P. Bolhuis, D. Chandler, C. Dellago, and P. Geissler. Transition path sampling: Throwing ropes over rough mountain passes, in the dark. *Annu. Rev. Phys. Chem.*, 53:291–318, 2002.

[11] E. Boye, M. Marinus, and A. Lobner-Olesen. Quantification of dam methyltransferase in escherichia coli. *J. Bacteriol.*, 174(5):1682–1685, Mar. 1992.

[12] B. A. Braaten, L. B. Blyn, B. S. Skinner, and D. A. Low. Evidence for a methylation-blocking factor (*mbf*) locus involved in *pap* pilus expression and phase variation in *escherichia coli*. *J. Bacteriology*, 173:1789–1800, 1991.

[13] B. A. Braaten, X. Nou, L. S. Kaltenbach, and D. A. Low. Methylation patterns in *pap* regulatory dna control pyelonephritis-associated pili phase variation. *Cell*, 76:577–588, 1994.

[14] K. Burrage, M. Hegland, S. Macnamara, and R. Sidje. A krylov-based finite state projection algorithm for solving the chemical master equation arising in the discrete modelling of biological systems. *Proc. of The A.A.Markov 150th Anniversary Meeting*, pages 21–37, 2006.

[15] Y. Cao, D. Gillespie, and L. Petzold. The slow-scale stochastic simulation algorithm. *J. Chem. Phys.*, 122(014116), Jan. 2005.

[16] Y. Cao, D. T. Gillespie, and L. R. Petzold. Accelerated stochastic simulation of the stiff enzyme-substrate reaction. *J. Chem. Phys.*, 123(144917), 2005.

[17] Y. Cao, D. T. Gillespie, and L. R. Petzold. Avoiding negative populations in explicit poisson tau-leaping. *J. Chem. Phys.*, 123(054104), 2005.

[18] Y. Cao, H. Li, and L. Petzold. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *JCP*, 121(9):4059–4067, Sep. 2004.

[19] A. Chatterjee, D. Vlachos, and M. Katsoulakis. Binomial distribution based tau-leap acelerated stochastic simulation. *JCP*, 122(024112), 2005.

[20] C. Dellago, P. Bolhuis, F. Csajka, and D. Chandler. Transition path sampling and the calculation of rate constants. *J. Chem. Phys*, 108(5):1964–1977, 1998.

[21] C. Dellago, P. Bolhuis, and P. Geissler. Transition path sampling. *Adv. Chem. Phys.*, 123:1–78, 2001.

[22] Y. Dublanche, K. Michalodimitrakis, N. Kummerer, M. Foglierini, and L. Serrano. Noise in transcription negative feedback loops: simulation and experimental analysis. *Molecular Systems Biology*, 2(41), 2006.

[23] G. Dullerud and F. Paganini. *A Course in Robust Control Theory: a Convex Approach.* Springer, 1 edition, 2000.

[24] H El Samad, M. Khammash, L. Petzold, and D. Gillespie. Stochastic modeling of gene regulatory networks. *Int. J. Robust Nonlin.*, 15:691–711, 2005.

[25] H. El Samad, H. Kurata, J. Doyle, C. Gross, and Khammash M. Surviving heat shock: Control strategies for robustness and performance. *PNAS*, 102(8):27362741, 2005.

[26] J. Elf and M. Ehrenberg. Fast evaluations of fluctuations in biochemical networks with the linear noise approximation. *Genome Research*, 13:2475–2484, 2003.

[27] M Elowitz, A. Levine, E. Siggia, and P. Swain. Stochastic gene expression in a single cell. *Science*, 297:1183–1186, 2002.

[28] A. Faradjian and R. Elber. Computing time scales from reaction coordinates by milestoning. *J. Chem. Phys.*, 120(23):10880–10889, 2004.

[29] N. Federoff and W. Fontana. Small numbers of big molecules. *Science*, 297(5584):1129–1131, 2002.

[30] K. Forsman, M. Goransson, and B. E. Uhlin. Autoregulation and multiple dna interactions by a transcriptional regulatory protein in *e. coli* pili biogenesis. *EMBO Journal*, 8, 1989.

[31] T. Gardner, C. Cantor, and J. Collins. Construction of a genetic toggle switch in escherichia coli. *Nature*, 403:339–242, 2000.

[32] G. George Yin and Q. Zhang. *Continuous-Time Markov Chains and Applications: A Singular Perturbation Approach.* Springer, 1998.

[33] M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem.*, 104:1876–1889, 2000.

[34] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comp. Phys.*, 22:403–434, 1976.

[35] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25):2340–2360, May 1977.

[36] D. T. Gillespie. A rigorous derivation of the chemical master equation. *Physica A*, 188:404–425, 1992.

[37] D. T. Gillespie. The chemical langevin equation. *J. Chem. Phys.*, 113(1):297–306, Jul. 2000.

[38] D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, Jul. 2001.

[39] D. T. Gillespie. The chemical langevin and fokker-plank equations for the reversible isomerization reaction. *J. Phys. Chem.*, 106:5063–5071, 2002.

[40] R. Gunawan, Y. Cao, L. R. Petzold, and F.J. Doyle III. Sensitivity analysis of discrete stochastic systems. *Biophysical Journal*, 88:2530–2540, Apr. 2005.

[41] C. Gmez-Uribe and G. Verghese. Mass fluctuation kinetics: Capturing stochastic effects in systems of chemical reactions through coupled mean-variance computations. *JCP*, 126(024109), Jan. 2007.

[42] W. B. Hale, M. van der Woude, and D. Low. Regulation of uropathogenic escherichia coli adhesin expression by dna methylation. *Molecular Genetics and Metabolism*, 65(3):191–196, Nov. 1998.

[43] E. Haseltine and J. Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *J. Chem. Phys.*, 117(15):6959–6969, Jul. 2002.

[44] J. Hasty, J. Pradines, M. Dolnik, and J.J. Collins. Noise-based switches and amplifiers for gene expression. *PNAS*, 97:2075–2080, 2000.

[45] A. Hernday, M. Krabbe, B. Braaten, and D. Low. Self-perpetuating epigenetic pili switches in bacteria. *PNAS*, 99(4):16470–16476, December 2002.

[46] A. D. Hernday, B. A. Braaten, and D. A. Low. The mechanism by which dna adenine methylase and papi activate the pap epigenetic switch. *Mol. Cell*, 12:947–957, October 2003.

[47] J. Hespanha and A. Singh. Stochastic models for chemically reacting systems using polynomial stochastic hybrid systems. *Int. J. on Robust Control, Special Issue on Control at Small Scales: Issue 1*, 15(15):669–689, 2005.

[48] L. Jarboe, D. Beckwith, and J. Liao. Stochastic modeling of the phase-variable pap operon regulation in uropathogenic escherichia coli. *Biotechnology and Bioengineering*, 88(2):189–204, 2004.

[49] L. Kaltenbach, B. A. Braaten, and D. A. Low. Specific binding of papi to lrp-pap dna complexes. *J. Bacteriology*, 177:5785–5797, 1995.

[50] M. Keeling. Multiplicative moments and measures of persistence in ecology. *J. theor. Biol.*, 205:269–281, 2000.

[51] T. Kepler and T. Elston. Stochasticity in transcriptional regulation: origins, consequences, and mathematical representations. *Biophys. J.*, 81:3116–3136, 2001.

[52] H.K. Khalil. *Nonlinear Systems.* Prentice Hall, 3 edition, 2002.

[53] M. Khammash and H. El Samad. Stochastic modeling and analysis of genetic networks. *Proc. 44th IEEE CDC*, Dec. 2005.

[54] H. Kobayashi, M. Kaern, M. Araki, K. Chung, T. Gardner, C. Cantor, and J. Collins. Programmable cells: Interfacing natural and engineered gene networks. *PNAS*, 101(22):8414–8419, June 2004.

[55] P. Kokotovich, H. Khalil, and J. O'Reilly. *Singular Perturbation Methods in Control: Analysis and Design.* Academic Press, 1986.

[56] I. Lestas, Paulsson J., N. Ross, and G. Vinnicombe. Noise in gene regulatory networks. *To appear in IEEE Trans. on Automatic Control*, 2008.

[57] H. Li, Z. Hou, and H. Xin. Internal noise stochastic resonance for intracellular calcium oscillations in a cell system. *Phys. Rev. E*, 71(061916), 2005.

[58] L. Ljung. *System Identification, Theory for the User.* Prentice Hall PTR, 2 edition, 1999.

[59] M. McAdams and A. Arkin. Stochastic mechanisms in gene expression. *PNAS*, 94:814–1819, 1997.

[60] M. McAdams and A. Arkin. Its a noisy business! *Tren. Gen.*, 15(2):65–69, 1999.

[61] C. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix. *SIAM Review*, 20(4):801–836, 1978.

[62] C. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Review*, 45(1):3–49, Feb. 2003.

[63] D. Moroni, P. Bolhuis, and T. van Erp. Rate constants for diffusive processes by partial path sampling. *J. Chem. Phys.*, 120(9):4055–4065, 2004.

[64] C. Morton-Firth and D. Bray. Predicting temporal fluctuations in an intracellular signaling pathway. *J. Theoretical Biology*, 192(1):117–128, May 1998.

[65] C. Morton-Firth, T. Shimizu, and D. Bray. A free-energy-based stochastic simulation of the tar receptor complex. *J. Molecular Biology*, 286(4):1059–1074, March 1999.

[66] B. Munsky, A. Hernday, D. Low, and M. Khammash. Stochastic modeling of the pap-pili epigenetic switch. *Proc. FOSBE*, pages 145–148, August 2005.

[67] B. Munsky and M. Khammash. The finite state projection algorithm for the solution of the chemical master equation. *J. Chem. Phys.*, 124(044104), 2006.

[68] B. Munsky and M. Khammash. A reduced model solution for the chemical master equation arising in stochastic analyses of biological networks. *Proc. 45th IEEE Conference on Decision and Control*, pages 25–30, December 2006.

[69] B. Munsky and M. Khammash. A multiple time interval finite state projection algorithm for the solution to the chemical master equation. *J. Comp. Phys.*, 226(1):818–835, 2007.

[70] B. Munsky and M. Khammash. Precise transient analysis of switches and trajectories in stochastic gene regulatory networks. *Submitted to IET Systems Biology*, 2008.

[71] B. Munsky, S. Peles, and M. Khammash. Stochastic analysis of gene regulatory networks using finite state projections and singular perturbation. *Proc. 26th American Control Conference (ACC)*, pages 1323–1328, July 2007.

[72] I. Nasell. An extension of the moment closure method. *Theoretical Population Biology*, 64:233–239, 2003.

[73] X. Nou, B. A. Braaten, L. Kaltenbach, and D. A. Low. Differential binding of lrp to two sets of *pap* dna binding sites mediated by pap i regulates pap phase variation in *escherichia coli*. *EMBO Journal*, 14:5785–5797, 1995.

[74] X. Nou, B. S. Skinner, B. A. Braaten, L. B. Blyn, D. Hirsch, and D. A. Low. Regulation of pyelonephritis-associated pili phase variation in *escherichia coli*: binding of the papi and lrp regulatory proteins is controlled by dna methylation. *Molecular Microbiology*, 7:545–553, 1993.

[75] P. O'Hanley, D. Low, I. Romero, D. Lark, K. Vosti, S. Falkow, and G. Schoolnik. Gal-gal binding and hemolysin phenotypes and genotypes associated with uropathogenic escherichia coli. *N. Engl. J. Med.*, 313(7):414–420, Aug. 1985.

[76] E. Ozbudak, M. Thattai, I. Kurtser, A. Grossman, and A. van Oudenaarden. Regulation of noise in the expression of a single gene. *Nature Genetics*, 31:69–73, 2002.

[77] J. Paulsson, O. Berg, and M. Ehrenberg. Stochastic focusing: Fluctuation-enhanced sensitivity of intracellular regulation. *PNAS*, 97(13):7148–7153, 2000.

[78] S. Peles, B. Munsky, and M. Khammash. Reduction and solution of the chemical master equation using time-scale separation and finite state projection. *J. Chem. Phys.*, 125(204104), Nov. 2006.

[79] S. Peterson and N. Reich. Gatc flanking sequences regulate dam activity: Evidence for how dam specificity may influence pap expression. *J. Molecular Biology*, 355(3):459–472, Jan. 2006.

[80] L. Petzold and coworkers. *StochKit Beta Version*, Nov. 2004. Downloaded from the Internet at http://www.engineering.ucsb.edu/ cse/StochKit/index.html.

[81] J. Puchalka and A. Kierzek. Bridging the gap between stochastic and deterministic regimes in the kinetic simulations of the biochemical reaction networks. *Biophysical Journal*, 86:1357–1372, 2004.

[82] C. V. Rao and A. P. Arkin. Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the gillespie algorithm. *J. Chem. Phys.*, 118(11):4999–5010, Mar. 2003.

[83] M. Rathinam and H. El Samad. Reversible-equivalent-monomolecular tau:

A leaping method for “small number and stiff” stochastic chemical systems. *J. Comp. Phys.*, 224(2):897–923, June 2007.

[84] 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.*, 119(24):12784–12794, Dec. 2003.

[85] S. I Resnick. *Adventures in Stochastic Processes.* Birkhauser, 1 edition, 2002.

[86] H. Salis and Y. Kaznessis. Accurate hybrid stochastic simulation of a system of coupled chemical or biological reactions. *J. Chem. Phys.*, 112(054103), 2005.

[87] A. Samant and D. Vlachos. Overcoming stiffness in stochastic simulation stemminf from partial equilibrium: A multiscale monte carlo algorithm. *JCP*, 123(144114), Oct. 2005.

[88] Eva Bienieka Shaolin Chena, Zhiqi Haoa and Joseph M. Calvo. Modulation of lrp action in escherichia coli by leucine: effects on non-specific binding of lrp to dna. *J. Mol. Biology*, 314(5):1067–1075, Dec. 2001.

[89] J. Shoemaker and F. Doyle. A hybrid stochastic/boolean approach to understanding the pap epigenetic switch. *Proc. FOSBE*, pages 99–102, 2005.

[90] Roger B. Sidje. EXPOKIT: Software package for computing matrix exponentials. *ACM Transactions on Mathematical Software*, 24(1):130–156, March 1998.

[91] A. Singh and J. Hespanha. Lognormal moment closures for biochemical

reactions. *Proc. of the 45th IEEE Conference on Decision and Control*, pages 2063–2068, Dec 2006.

[92] A. Singh and J. Hespanha. Moment closure techniques for stochastic models in population biology. *Proc. of the 2006 ACC*, pages 4730–4735, June 2006.

[93] A. Singh and J. Hespanha. A derivative matching approach to moment closure for the stochastic logistic model. *Bulletin of Mathematical Biology*, 69:1909–1925, 2007.

[94] K. Takahashi, K. Kaizu, B. Hu, and M. Timita. A multi-algorithm, multi-timescale method for cell simulation. *Bioinformatics*, 20(4):538–546, Jan. 2004.

[95] M. Thattai and A. van Oudenaarden. Intrinsic noise in gene regulatory networks. *Proc. Natl. Acad. Sci.*, 98:8614–8619, 2001.

[96] T. Tian and K. Burrage. Binomial leap methods for simulating stochastic chemical kinetics. *J. Chem. Phys.*, 121(21):10356–10364, Dec. 2004.

[97] T. Tian and K. Burrage. Stochastic models for regulatory networks of the genetic toggle switch. *PNAS*, 103(22):8372–8377, May 2006.

[98] R. Tomioka, H. Kimura, T. Kobayashi, and K. Aihara. Multivariate analysis of noise in genetic regulatory networks. *J. Theoretical Biology*, 229(4):501–521, 2004.

[99] S. Urig, H. Gowher, A. Hermann, C. Beck, M. Fatemi, A. Humeny, and A. Jeltsch. The *escherichia coli* dam dna methyltransferase modifies dna in a highly processive reaction. *J. Molecular Biology*, 319:1085–1096, 2002.

[100] M. van der Woude, B. Braaten, and D. A. Low. Epigenetic phase variation of the pap operon in escherichia coli. *Trends Mircobiol.*, 4(1):5–9, Jan 1996.

[101] M. van der Woude, W. B. Hale, and D. A. Low. Formation of dna methylation patterns: nonmethylated gatc sequences in *gut* and *pap* operons. *J. Bacteriology*, 180:5913–5920, 1998.

[102] M. van der Woude, L. Kaltenback, and D. Low. Leucine-responsive regulatory protein plays dual roles as both an activator and a repressor of the escherichia coli pap fimbrial operon. *Mol. Microbiol.*, 17(2):303–312, Jul. 1995.

[103] T. van Erp. and P. Bolhuis. Elaborating transition interface sampling methods. *J. Comp. Phys.*, 205:157–181, 2005.

[104] T. van Erp., D. Moroni, and P. Bolhuis. A novel path sampling method for the calculation of rate constants. *J. Chem. Phys.*, 118(17):7762–7774, 2003.

[105] N.G. van Kampen. *Stochastic Processes in Physics and Chemistry.* Elsevier, 3 edition, 2001.

[106] J. Warren, J. Walker, J. Roth, and E. Altman. Construction and characterization of a highly regulable expression vector, plac11, and its multipurpose derivatives, plac22 and plac33. *Plasmid*, 44(2):138–151, Sept. 2000.

[107] E. Weinan, D. Liu, and E. Vanden-Eijnden. Nested stochastic simulation algorithm for chemical kinetic systems with disparate rates. *JCP*, 123(194107), Nov. 2005.

[108] P. Whittle. On the use of the normal approximation in the treatment of stochastic processes. *J. Roy. Statist. Soc., Ser. B*, 19:268–281, 1957.

[109] D. Willins, C. Ryan, J. Platko, and J. Calvo. Characterization of lrp, and escherichia coli regulatory protein that mediates a global response to leucine. *J. Biol. Chem.*, 266(17):10768–10774, June 1991.

[110] Y. Xia, K. Forsman, J. Jass, and B. E. Uhlin. Oligomeric interaction of the papb transcriptional regulator with the upstream activating region of pili adhesin gene promoters in *escherichia coli*. *Molecular Microbiology*, 30:513–523, 1998.

[111] B. Zhou, D. Beckwith, L. Jarboe, and J. Liao. Markov chain modeling of pyelonephritis-associated pili expression in uropathogenic escherichia coli. *Biophysical Journal*, 88:2541–2553, 2005.