

Diss. ETH No. 20105

Partial-propensity simulation algorithms for stochastic chemical kinetics and the role of fluctuations in mesoscopic reaction systems

*A dissertation submitted to
ETH Zürich*

*for the degree of
Doctor of Sciences*

*presented by
Rajesh Ramaswamy*

B.Eng., University of Madras, India
M.Sc., Max Planck Research School, University of Tübingen, Germany

*born 10 april 1983
citizen of India*

accepted on the recommendation of
Prof. Dr. Ivo F. Sbalzarini, ETH Zürich, Switzerland, examiner
Prof. Dr. Petros Koumoutsakos, ETH Zürich, Switzerland, co-examiner
Prof. Dr. Johan Paulsson, Harvard University, USA, co-examiner

2011

the beginning:



the never ending:



the quest in between:

“To see a world in a grain of sand,
And a heaven in a wild flower,
Hold infinity in the palm of your hand,
And eternity in an hour.”
– *Auguries of Innocence*, William Blake

Abstract

Chemical reactions are fundamental processes in nature. They are conventionally modeled using reaction rate equations (RRE), which are ordinary differential equations describing the rate of change of the concentrations of species in a well-mixed system. RRE, however, are a macroscopic description that is only valid in the order of Avogadro's number of molecules of all reactants. It is well known that when the population of reactant species is small, fluctuations due to intrinsic noise can lead to non-trivial chemical kinetics. This intrinsic noise is accounted for by the chemical master master (CME). Using newly developed stochastic simulation algorithms (SSA) we describe novel qualitative and quantitative effects of intrinsic noise on chemical kinetics in small populations.

We propose a quantity called *partial propensity* and use it to construct a family of novel SSA formulations. Introducing a topological property of reaction networks, called the *degree of coupling*, we classify networks into weakly coupled and strongly coupled ones. We demonstrate that the computational cost of SSA depends on the coupling class of the network simulated. For strongly coupled networks, the computational cost of partial-propensity SSA is at most proportional to the number of species in the chemical reaction network. For weakly coupled networks, the cost can be independent of network size. We benchmark the partial-propensity SSA formulations on prototypical test cases and derive an upper bound for the scaling of their computational cost with increasing network size. We further show that all partial-propensity formulations can be composed from a set of common algorithmic building blocks, which we implemented in a modular software package.

We apply partial-propensity SSA to study the effects of intrinsic noise on the steady-state mean concentrations in nonequilibrium monostable reaction networks. Assuming that the molecular population is proportional to the reactor volume, intrinsic noise due to apparent molecular discreteness increases with decreasing reactor volume. We show that there exists a critical reactor volume (or, equivalently, population size) below which the CME predicts a different ordering of the steady-state mean concentrations than the RRE. We present a general theory that explains this *discreteness-induced inversion effect* and predicts the critical volume in good agreement with exact SSA simulations. Subsequently, we study the fingerprints of intrinsic and extrinsic noise on the relaxation kinetics of fluctuations in mesoscopic reaction networks. We show that intrinsic and extrinsic noise affect the derivatives of the time-autocorrelation function of the fluctuations in opposite ways, allowing us to separate these two noise sources. In oscillatory mesoscopic reaction systems, we find that intrinsic noise systematically alters the frequency spectrum of the oscillations.

For spatially inhomogeneous reaction–diffusion systems we use the idea of partial propensities to develop a new algorithm for spatiotemporal stochastic reaction-diffusion simulations. Spatial heterogeneity is sustained when the diffusion of species is slower than the reactions

between them. In such cases, the CME is not valid. Conceptually dividing the reactor into well-mixed subvolumes with jump reactions between them modeling diffusion, stochastic reaction-diffusion is modeled using the on-lattice reaction-diffusion master equation (RDME). The computational cost of the presented simulation algorithm is at most logarithmic in the number of subvolumes and at most linear in the number of chemical species.

As a possible application of the presented algorithms and results we consider the problem of identifying the unknown parameters of a stochastic reaction network from a single, noisy population trajectory. This constitutes an inverse problem for which we use the fingerprints of noise-induced effects as additional regularization. We propose a method that combines partial-propensity formulations for sampling trajectories from the exact solution of the CME with an adaptive Monte Carlo sampling technique — called Gaussian adaptation — to search the parameter space. We benchmark the method on synthetic test cases using simulated trajectories.

Zusammenfassung

Chemische Reaktionen sind fundamentale Prozesse in der Natur. Sie werden meistens mittels Reaktionsratengleichungen (RRG) beschrieben, gewöhnliche Differentialgleichungen für die Stoffkonzentrationen der beteiligten Substanzen in homogenen Reaktoren. RRG sind jedoch nur in makroskopischen Systemen gültig in denen die Teilchenzahlen aller Reaktandenmoleküle in der Größenordnung der Avogadrozahl sind. Für kleine Reaktandenpopulationen ist es bekannt dass intrinsische Fluktuationen zu nicht-trivialer Kinetik führen können. Diese intrinsischen Fluktuationen werden von der chemischen Mastergleichung (CMG) beschrieben. Wir entwickelten neue stochastische Simulationsalgorithmen (SSA) und benutzen diese zur Beschreibung neuer qualitativer und quantitativer Effekte von Fluktuationen auf die chemische Kinetik in kleinen Molekülpopulationen.

Wir führen dazu eine neue, *partial propensity* genannte Größe ein und benutzen diese um eine neue Familie von stochastischen Simulationsalgorithmen (SSA) zu konstruieren. Basierend auf dem *Kopplungsgrad* als topologische Größe unterscheiden wir stark gekoppelte und schwach gekoppelte Reaktionsnetzwerke. Wir zeigen dass der Rechenaufwand von SSA vom Kopplungsgrad des simulierten Netzwerks abhängt. Für stark gekoppelte Netzwerke ist der Rechenaufwand von partial-propensity SSA höchstens linear proportional zur Anzahl der chemischen Verbindungen (Reaktanden und Produkte) im Reaktionsnetzwerk. Für schwach gekoppelte Netzwerke kann der Rechenaufwand gar unabhängig von der Netzwerkgröße sein. Wir messen die Rechenzeiten von partial-propensity SSA anhand prototypischer Testfälle und leiten eine obere Schranke für den Rechenaufwand in Funktion der Netzwerkgröße her. Wir zeigen auch dass alle partial-propensity SSA aus einem gemeinsamen Satz algorithmischer Bausteine aufgebaut werden können, welche wir in einer modularen Software implementiert haben.

Wir verwenden partial-propensity SSA um den Einfluss intrinsischer Fluktuationen auf die stationären Ungleichgewichtskonzentrationsmittelwerte in monostabilen Reaktionsnetzwerken zu studieren. Wenn die Molekülpopulation proportional zum Reaktorvolumen ist, dann nehmen die intrinsischen Fluktuationen mit abnehmendem Reaktorvolumen zu. Wir zeigen die Existenz eines kritischen Reaktorvolumens (oder äquivalent einer kritischen Molekülpopulation) unterhalb dessen die CMG eine andere Rangfolge der stationären Konzentrationsmittelwerte vorhersagt als die RRG. Wir präsentieren eine allgemeine Theorie welche diesen *fluktuationsinduzierten Inversionseffekt* erklärt und kritische Volumina in guter Übereinstimmung mit exakten SSA-Simulationen vorhersagen kann. Dann studieren wir wie sich die intrinsischen Fluktuationen auf die Relaxationskinetik mesoskopischer Reaktionsnetzwerke auswirken. Wir zeigen dass intrinsische und extrinsische Fluktuationen die Ableitungen der Zeitautokorrelationsfunktion der Molekülpopulation in entgegengesetzter Weise beeinflussen. Dies erlaubt es uns, die beiden Fluktuationsquellen zu unterscheiden. Zudem verändern die

intrinsichen Fluktuationen systematisch das Frequenzspektrum oszillierender mesoskopischer Reaktionssysteme.

Auch für räumlich heterogene Reaktions-Diffusions-Systeme verwenden wir die Idee der *partial propensities* um einen neuen Simulationsalgorithmus herzuleiten. Räumliche Heterogenität tritt auf wenn die Diffusion der Moleküle langsamer ist als die Reaktionen zwischen ihnen. In diesen Fällen ist die CMG ungültig. Der Reaktor wird dann gedanklich in homogene Teilvolumina unterteilt und Diffusion von Molekülen als Sprungreaktionen in benachbarte Teilvolumina modelliert, was zur gitterbasierten Reaktions-Diffusions-Mastergleichung (RDMG) führt. Der Rechenaufwand des vorgestellten Simulationsverfahrens ist höchstens logarithmisch in der Anzahl der Teilvolumina und höchstens linear in der Anzahl der chemischen Verbindungen.

Als mögliche Anwendung der vorgestellten Algorithmen und Resultate betrachten wir das Problem der Schätzung der unbekannten Parameter eines stochastischen Reaktionsnetzwerks aus einer einzigen, verrauschten Populationstrajektorie. Dies ist ein inverses Problem für das wir die Fingerabdrücke der Fluktuationseffekte als zusätzliche Regularisierung verwenden. Wir schlagen eine Methode vor welche *partial-propensity SSA* zur Simulation von Populationstrajektorien von der CMG mit einem adaptiven Monte Carlo Sampler — genannt “Gaussian Adaptation” — zum Absuchen des Parameterraums verbindet. Wir testen die neue Methode auf synthetischen Testfällen mit simulierten Populationstrajektorien.

Acknowledgements

“We are nothing without the work of others our predecessors, others our teachers, others our contemporaries. Even when, in the measure of our adequacy and our fullness, new insight and new order are created, we are still nothing without others. Yet we are more.”

– *Reith Lectures on The Sciences and Man’s Humanity, Robert Oppenheimer*

First and foremost, I thank Prof. Dr. Ivo F. Sbalzarini for his guidance during my PhD studies. I owe all the positive things about this thesis to Ivo and reserve the negative things to myself. Ivo’s optimism and encouragement was just amazing to say the least. I had a great time interacting with him on various different subjects and thoroughly enjoyed it each and every time. I’ve never heard Ivo say “this is not possible” or “this is too much”.

Another person from whom I learnt a lot, is Nelido. During the first two definitive years of my PhD studies, Nelido was an ever patient and an eager teacher. He taught me the definition of rigor and its importance. I had numerous discussions with him about science, politics, economics and life. It was great to know his rigorous point of view on every subject of our conversation.

It has been an absolute privilege and joy to have the opportunity to do my PhD studies at the MOSAIC group. Since the beginning of my studies, the group has always been populated by a diverse group of people who are creative, hard working, hard “playing”, playful, childish, helpful, cheerful and forthright. I’ve learnt a lot and draw inspiration from Jo’s unconventional way of thinking, Christian’s thoroughness, Janick’s calmness (also known as Bernese-ness) and perfectionism, Omar’s knowledge of computer systems, Sylvain’s mathematical prowess, and Greg’s broad range of knowledge along with his mathematical prowess. Each one of them was always available for discussions and to give their forthright opinion upon request. Omar requires special mention since he helped me out on very many occasions that he articulated as “raj against the machine”. I also thank Oleksandr for his efforts in implementing the pSSA software. Without his efforts, some pages in this thesis would be blank.

Over the last four years, I also befriended an amazing group of people: smole (Zlatko Smole), ivo, Mr. fa (Fabio Serventi), zdravkica (Zdravka Misic), georgie (Georg Ofenbeck), Mr. albi (Albino Troilo) and marco (Marco Perella). Smole, fa and georgie deserve special mention for the very many incredibly amazing scientific and non-scientific discussions that tended towards insanity on almost all occasions. I thank zdravkica, smole, ivo, fa and georgie for their kindness and the most amazing time we’ve had together.

I thank Dr. Ramon Grima for introducing me to van Kampen expansion and for the collaborative effort on discreteness-induced inversion. I learnt a lot about noise in chemical kinetics from Ramon.

I thank Prof. Venkateswaran (“Waran”, Director of Waran research foundation) for introducing me to the world of scientific research. Without him, I wouldn’t be writing these acknowledgements.

My awesome parents (indira and ramaswamy) have been a great source of inspiration. They have been extremely supportive in all my endeavors. They were the first ones who taught me the importance of knowledge and made lots of sacrifices for me and my sister to go in pursuit of our interests. I will never be able to thank them enough. I thank my lovely sister nishu, for being such an awesome sister that she is. Nishu, thanks for the car that isn’t mine.

I was honored to be in ETH, let alone the opportunity to pursue my PhD studies. It is a fantastic university with amazing people and resources. Everything in ETH just works perfectly. The people running the show in ETH make the life of students so much easier. This helped me focus on my PhD studies without worrying about anything else. I thank this institution for the opportunity it has given me to broaden my knowledge.

I thank Switzerland and in particular Zürich for being such a nice host. There are many beautifully weird people in Zürich and I had a great time in this city among such people.

During my PhD studies, I got many opportunities to visit MedILS in Split, Croatia. My first visit to MedILS ignited my fascination for Croatia and its people. I had a great time in MedILS - Omar, I apologize; Janick and Ivo, it was amazing. I also met Dr. Anton Polyansky during my first visit to MedILS. It was wonderful to get to know Anton and I had an amazing time with him. This meeting finally resulted with me contributing my part to Anton’s work on anti-microbial peptides. Soon after I met Anton I ended up visiting Russia, Anton’s country of origin, where I met some interesting people and had some of the most interesting experiences. Due to MedILS, I got the opportunity to get to know Prof. Miroslav Radman and his family. I learnt a lot from the time I spent with Miro and his family. During my numerous visits to Croatia, I met some cool people in Split and Zagreb, and I thank them for the great time.

Zürich, November 2011
Rajesh

Contents

Abstract	v
Zusammenfassung	vii
Acknowledgements	ix
Table of Contents	xi
Introduction	xvii
1 Stochastic chemical kinetics	1
1.1 The chemical master equation (CME)	1
1.2 Approximating the CME	5
1.2.1 The chemical Kramer-Moyal equation and the nonlinear chemical Fokker-Planck equation	6
1.2.2 The macroscopic reaction rate equation (RRE)	8
1.3 Kinetic Monte Carlo: The stochastic simulation algorithm (SSA)	9
1.4 The effect of intrinsic noise on chemical kinetics	11
1.4.1 Monostable, linear reaction networks	12
1.4.2 Monostable, nonlinear reaction networks	14
1.4.3 Brusselator: an oscillatory reaction network	15
1.4.4 Schlogl model: a bistable reaction network	20
1.5 Summary	23
2 Formulations of the stochastic simulation algorithm for chemical reaction networks	25
2.1 Topology of chemical reaction networks	26
2.1.1 Representation of chemical reaction networks	27
2.1.2 Degree of coupling of a reaction network	27
2.1.3 Classification of reaction networks	28
2.1.4 Examples	28
2.1.4.1 Cyclic chain model	28
2.1.4.2 Colloidal aggregation model	30
2.2 SSA formulations	33
2.3 Exact formulations of the SSA and their computational cost	34
2.3.1 The first reaction method (FRM)	34
2.3.2 The direct method (DM)	35

2.3.3	Next reaction method (NRM)	36
2.3.4	Optimized direct method (ODM)	37
2.3.5	Sorting direct method (SDM)	37
2.3.6	Logarithmic direct method (LDM)	38
2.3.7	Composition-rejection method (SSA-CR)	39
2.3.8	Summary of the computational costs and memory requirements of these exact SSA formulations	41
2.4	Approximate SSA formulations	42
2.5	Summary	44
3	Partial-propensity formulations of the stochastic simulation algorithm	45
3.1	Partial propensity	46
3.2	The concept of partial-propensity SSA formulations	46
3.3	The partial-propensity direct method (PDM)	48
3.3.1	Detailed description	48
3.3.2	The sorting partial-propensity direct method (SPDM)	52
3.3.3	Computational cost and memory requirements	52
3.3.3.1	Computational cost	52
3.3.3.2	Memory requirements	53
3.3.4	Benchmarks	53
3.3.4.1	Strongly coupled reaction network: colloidal aggregation model .	57
3.3.4.2	Strongly coupled network of bimolecular reactions	58
3.3.4.3	Weakly coupled reaction network: linear chain model	58
3.3.4.4	Multi-scale biological network: heat-shock response in <i>Escherichia coli</i>	59
3.3.5	Conclusions	63
3.4	The partial-propensity SSA with composition-rejection sampling (PSSA-CR) .	64
3.4.1	Detailed description	65
3.4.2	Computational cost	68
3.4.3	Benchmarks	69
3.4.3.1	A weakly coupled reaction network: Cyclic chain model	70
3.4.3.2	A strongly coupled reaction network: Colloidal aggregation model	70
3.4.4	Conclusions	72
3.5	The delay partial-propensity direct method (dPDM)	72
3.5.1	The delay stochastic simulation algorithm (dSSA)	73
3.5.1.1	The delay direct method (dDM)	74
3.5.2	Detailed description	77
3.5.3	Computational cost	82
3.5.4	Benchmarks	82
3.5.4.1	A strongly coupled reaction network: Colloidal aggregation model	83
3.5.4.2	A weakly coupled reaction network: Cyclic chain model	83
3.5.5	Conclusions	85

3.6	The family of partial-propensity methods	85
3.6.1	Modules of partial-propensity algorithms	86
3.7	Summary	89
4	Discreteness-induced concentration inversion in mesoscopic chemical reaction systems	91
4.1	General theory	95
4.2	Model	97
4.3	Simulation details	100
4.4	Results	101
4.5	Conclusions and Summary	107
5	Effect of noise on the relaxation kinetics of mesoscopic chemical systems	109
5.1	Steady-state relaxation kinetics of mesoscopic non-oscillatory chemical systems	110
5.1.1	Model	112
5.1.2	Low confinement: the linear-noise approximation (LNA)	113
5.1.3	Beyond the LNA: the full CME	117
5.1.3.1	Lifetime	117
5.1.3.2	Derivatives of the time-autocorrelation function	121
5.1.4	Conclusions	124
5.2	Steady-state relaxation kinetics of mesoscopic oscillatory chemical reaction systems	126
5.2.1	Model	127
5.2.2	Results	131
5.2.3	Conclusions	133
5.3	Summary	134
6	Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods	137
6.1	On-lattice stochastic reaction-diffusion	140
6.1.1	General concept	140
6.1.2	Discretization-corrected propensities	142
6.1.3	The Next Subvolume Method (NSM) for on-lattice stochastic reaction-diffusion simulations	144
6.2	The partial-propensity stochastic reaction-diffusion method (PSRD)	147
6.2.1	General concept of PSRD	147
6.2.1.1	Composition-rejection sampling to select the subvolume	147
6.2.1.2	Partial propensities to sample the next reaction within a subvolume	147
6.2.2	Detailed description of the PSRD algorithm	148
6.2.2.1	Data structures	148
6.2.2.2	Algorithms	149
6.2.3	Computational cost	155
6.2.4	Benchmarks	155
6.2.4.1	Colloidal aggregation model	156

6.2.4.2	Linear chain model	159
6.2.5	Two- and three-dimensional SRD simulations using PSRD	161
6.3	Conclusions and Summary	164
7	Global parameter identification for stochastic reaction networks from single trajectories	167
7.1	Problem statement	170
7.2	Gaussian Adaptation (GaA)	170
7.2.1	GaA for global black-box optimization	172
7.2.2	GaA for approximate Bayesian computation and viable volume estimation	172
7.3	Evaluation of the forward model	173
7.4	Objective function	173
7.5	Results	174
7.5.1	Weakly coupled reaction network: cyclic chain	175
7.5.2	Strongly coupled reaction network: colloidal aggregation	179
7.5.2.1	At steady state	179
7.5.2.2	In the transient phase	180
7.6	Conclusions and Summary	181
8	Conclusions, limitations and future work	183
Appendix		189
A1	Method of generating functions to compute moments from the CME	189
A2	Breakdown of a non-elementary reaction into elementary reactions	192
A3	Equivalence of PDM's next reaction sampling to that of DM	194
A4	The partial-propensity SSA (pSSA) software package	195
A4.1	Partial-propensity methods implemented in pSSA	195
A4.2	Platforms	195
A4.3	Dependencies	195
A4.4	Functionality	196
A4.5	Command line interface (CLI) of pSSA	196
A4.6	Calling pSSA from MATLAB	197
A4.7	Simulation of the mitogen-activated protein kinase (MAPK) phosphorylation-dephosphorylation system using pSSA	199
A5	The van Kampen system-size expansion	200
A5.1	The linear-noise approximation	203
A5.2	Beyond the linear-noise approximation: the effective mesoscopic reaction rate equation (EMRE)	205
A5.2.1	Example 1: Linear reaction network	206
A5.2.2	Example 2: Nonlinear reaction network	207
A5.3	Limitations of the van Kampen expansion and the EMRE	209
A6	Discreteness-induced concentration inversion in an <i>in-silico</i> genetic network	211
A6.1	Comparison of critical volumes from theory and simulation	212
A7	Effect of volume, burst and the mass-balance condition on the concentration variance	214

Contents

A8	CME, RRE and the stability of the fixed point of the trimerization system . . .	216
A9	Stability of the fixed point and Hopf bifurcation of the Brusselator	218
A10	Normalized power spectral density of a cascade of Brusselators at larges volume	220
Bibliography		223
Index		243
Publications		245
Curriculum Vitae		247

Introduction

“I do not know anything, but I do know that everything is interesting if you go into it deeply enough.”

— Richard P. Feynman

Chemical reactions are processes where atoms constituting molecules are rearranged to form molecules with different properties. The substances that are consumed in a chemical reaction are called *reactants* and those that are produced are called *products*. A chemical reaction can be spontaneous, leading to the conversion of reactants to products without any input of energy. In other cases, the reactants need to be supplied with some form of energy in order to get converted to products. Reactions can be conceptualized as reactants in a certain energy state being converted to products in a different energy state. If there is an energy barrier between the reactant and the products, energy input is required. Hence, reactions convert mass from one form to another by consuming or releasing energy.

We ourselves perform chemical reactions routinely. The whitening of egg yolk upon heating is the result of denaturation of a protein called albumin. Also, chemical reactions are used extensively in pharmaceutical, chemical and food processing industries. Examples include synthesis of the antibiotic azithromycin from erythromycin, synthesis of fertilizers from ammonia and the fermentation of lactose to produce yoghurt. Plants use photosynthesis to convert carbon-dioxide and water to sugar and oxygen in the presence of light. A series of chemical reactions in our body converts carbohydrates to glucose and subsequently to adenosine triphosphate (ATP), the primary energy source of a biological cell. Using ATP, the cells host a phenomenal variety of chemical reactions that are essential for life. For instance, the process of transcription and translation to produce proteins from deoxyribonucleic acid (DNA).

Chemical reactions have been a key driver in the development of nonequilibrium thermodynamics, starting with the work of Belousov and Zhabotinsky and inspired by Belousov’s experimental observation of an oscillating chemical reaction (Belousov, 1959; Zhabotinsky, 1964; Belousov, 1985; Zhabotinsky, 1991). Alan Turing proposed chemical reactions involving diffusing morphogens as the basis for morphogenesis (Turing, 1952) after discovering their potential of yielding concentration profiles that appeared similar to animal coat patterns. Chemical reactions are thus so fundamental that they can be considered a key cog in the wheel of the universe and of life.

Being so fundamental, chemical reactions and the famous law of mass action—due to the works of Guldberg, Waage, van’t Hoff and Berthollet—are taught in every school. Using the law of mass action, chemical kinetics, which describes the rate of change of the concentrations

Introduction

of molecules involved in a reaction, is formulated as a system of coupled ordinary differential equations, the reaction rate equation (RRE). This description has been experimentally shown to be a good model for chemical kinetics. RREs, however, are a mean-field description where the concentration of molecules is a continuous quantity and reaction rates are quantified using macroscopic rate constants. Therefore, RREs do not always provide an accurate description. This is for instance the case when the number of molecules of a chemical species (also called the “population”) is much smaller than Avogadro’s number (McQuarrie, 1967; Kurtz, 1972; Gardiner et al., 1976; Gillespie, 1976). At low population, the molecular discreteness of the reactant molecules becomes apparent, leading to fluctuations in the concentrations. This *intrinsic noise* stems from the discrete change in the number of molecules when a reaction occurs and its random character is imparted by the contact of the system with a heat bath (Gillespie, 1992, 2007). At low population, the number of molecules is not large enough for averaging out the fluctuations due to intrinsic noise. These fluctuations then play an important role in leading to non-trivial chemical kinetics (McAdams and Arkin, 1997; Arkin et al., 1998; Elowitz and Leibler, 2000; Barkai and Leibler, 2000; Berg et al., 2000; McClintock, 1999; Qian et al., 2002; Elf and Ehrenberg, 2004). This is for example the case in biological cells, where biochemical reactions proceed in sub-micron compartments, the total number of molecules is in the range of one to several thousand per compartment (Ghaemmaghami et al., 2003; Ishihama et al., 2008) and intrinsic noise is expected to play an important functional role in biochemical circuits (McAdams and Arkin, 1997; Arkin et al., 1998; McClintock, 1999; Elowitz and Leibler, 2000; Barkai and Leibler, 2000; Berg et al., 2000; Elf and Ehrenberg, 2004; Eldar and Elowitz, 2010). Fluctuations due to low copy numbers or molecular discreteness have been implicated as an essential player in several observations. Intrinsic noise has been used to explain the fraction of cells choosing the lysis or lysogenic pathway in bacterio-phage λ infection (Arkin et al., 1998), the emergence of oscillations in biochemical clocks and also in their loss of synchrony (Barkai and Leibler, 2000; Elowitz and Leibler, 2000), reduction of cell signal precision (Berg et al., 2000), spontaneous separation of biomolecules into spatial domains in the mitogen activated protein kinase (MAPK) phosphorylation-dephosphorylation system (Elf and Ehrenberg, 2004), and modulation of the response of the MAPK pathway (Takahashi et al., 2010). In all of these examples and many others, correlation in the fluctuations lead to observations (Springer and Paulsson, 2006) that cannot be explained by the RREs.

Intrinsic noise can be accounted for by stochastic kinetic models. An approach that has become canonical is the chemical master equation (CME) (Gillespie, 1976, 1977, 1992), a Markov-chain model obeying the fundamental Chapman-Kolmogorov equation (Gillespie, 1991). The CME is an exact mesoscopic description for any well-stirred and thermally equilibrated gas-phase chemical system (Gillespie, 1992), and for chemical reactions in well-stirred dilute solutions (Gillespie, 2009). Its high dimensionality, however, renders analytical approaches intractable. The analytical intractability is overcome by using numerical methods to sample population trajectories of the chemical species from the CME using a kinetic Monte Carlo approach (Doob, 1942, 1945; Kendall, 1949; Barlett, 1953; Bortz et al., 1975; Gillespie, 1976, 1977, 1992). The canonical kinetic Monte Carlo approach for sampling population trajectories from the exact solution of the CME is Gillespie’s stochastic simulation algorithm (SSA) (Gillespie, 1976, 1977, 1992). The first SSA formulations proposed by Gillespie himself are the first reaction method (FRM) and the direct method (DM) (Gillespie, 1976, 1977). The popularity of these methods coincides with an increasing availability of experimental observations that

could not be explained by RREs.

In this thesis, we review the computational cost of exact SSA formulations and propose a new class of potentially more efficient exact SSAs based on the novel concept of *partial propensities*. In addition, the thesis aims to further understand the role of noise in mesoscopic chemical systems, where low populations at small reactor volumes may deviate stochastic kinetics from its deterministic counterpart. We provide fingerprints of these deviations and conditions for when they are significant. Finally, we apply the presented algorithms and results to stochastic simulations of spatiotemporal reaction-diffusion processes and to global parameter identification in stochastic reaction networks. The thesis is structured as follows:

Chapter 1: Stochastic chemical kinetics

We provide the theoretical background and the necessary definitions for this thesis by recapitulating the derivation of the chemical master equation (CME) from the fundamental assumption of any Markov process as described by the Chapman-Kolmogorov equation. We then present the derivation of Gillespie's stochastic simulation algorithm (SSA) that samples population trajectories from the exact solution of the CME. We present the systematic approximations of the CME that lead to the chemical Kramer-Moyal equation and show that truncating the Kramer-Moyal equation leads to the nonlinear Fokker-Planck equation, whose equation of motion is the chemical Langevin equation. At very large populations, we show the emergence of the classical reaction rate equation (RRE). Finally, making use of Gillespie's SSA, we demonstrate the effect of intrinsic noise due to low population at small reactor volume on different types of chemical reaction networks.

Chapter 2: Formulations of the stochastic simulation algorithm for chemical reaction networks

We define the state of the art by reviewing existing exact SSAs and analyzing their computational costs. The computational cost of SSAs depends on the properties of the simulated reaction networks and of the algorithms used in the SSA formulation. We present the cost-determining topological properties of chemical reaction networks and use them to classify reaction networks into *strongly coupled* and *weakly coupled* networks. We review existing SSAs and their computational costs as a function of the coupling class of the reaction network. Quantifying computational cost as the CPU time per reaction event, we show that the computational costs of exact SSAs are functions of the number of chemical reactions in the network. We further show that the computational cost of existing SSAs has been reduced to constant time for weakly coupled networks. For strongly coupled networks, however, we show that the computational cost of exact SSAs is at best linear in the number of reactions. We also briefly review the general concept of approximate SSAs and their potential in offering superior computational performance for large reactor volumes (or large populations, assuming that population increases linearly with reactor volume). Parts of this chapter have been presented in (Ramaswamy et al., 2009).

Chapter 3: Partial-propensity formulations of the stochastic simulation algorithm.

As a first key contribution of this thesis, we present a novel class of exact SSAs whose computational cost scales at most linearly with the number of *species* in the reaction network. This is based on the novel concept of *partial propensities*. We introduce the partial propensity of a reaction with respect to a reactant as its propensity per molecule of that reactant. Limiting ourselves to elementary reactions, we propose the class of partial-propensity SSAs whose computational cost is at most linear in the number of chemical species, rather than the number of chemical reactions. Since the number of chemical species in a reaction network is expected to be much smaller than the number of chemical reactions, partial propensity SSAs can potentially offer superior computational performance, especially for strongly coupled and large reaction networks. For weakly coupled reaction networks, we further reduce computational cost of partial-propensity formulations to constant time. We also use partial propensities to formulate SSAs for chemical reaction networks with delays. Finally, we show that partial-propensity SSAs constitute a family of exact SSAs with algorithmic building blocks that can be modified to obtain different formulations. Parts of this chapter have been presented in (Ramaswamy et al., 2009; Ramaswamy and Sbalzarini, 2010b,a, 2011c). The concept of partial propensities has later also been proposed by Indurkhy and Beal (Indurkhy and Beal, 2010). Here, however, we present an entire family of partial-propensity SSAs and provide a comprehensive analysis of the computational cost as a function of the coupling class of the reaction network.

Chapter 4: Discreteness-induced concentration inversion in mesoscopic chemical reaction systems

We apply the presented partial-propensity methods to study the effects of noise in mesoscopic reaction networks. The results constitute the second key contribution of this thesis. We start by studying static properties at a nonequilibrium steady state, which leads to the discovery of a novel inversion effect. Assuming that a decrease in reactor volume decreases the population of all species, apparent molecular discreteness increases at small reactor volumes. In addition, many reactors are exposed to extrinsic noise, originating from outside the reactor. Modulating the magnitude of intrinsic noise by the reactor volume and that of extrinsic noise by burst input of a reactant species, we demonstrate the effect of molecular discreteness on the nonequilibrium steady-state mean concentrations in mesoscopic chemical reaction systems. We show that due to noise, the steady-state mean concentration prediction of the RRE and the linear noise approximation (LNA) of the CME are qualitatively correct only for systems above a newly identified critical reactor volume. For sub-critical volumes the ordering of steady-state mean concentrations of species changes. This concentration inversion is not captured by the RRE and the LNA. We hence call this change in ordering of steady-state mean concentrations *discreteness-induced concentration inversion*. We present a predictive theory based on van Kampen's system-size expansion to explain this novel effect and to predict the values of the critical volumes for a given reaction network. Parts of this chapter have been presented

in (Ramaswamy et al., 2012).

Chapter 5: Effect of noise on the relaxation kinetics of mesoscopic chemical systems

Next, we study noise-induced modulations of the dynamic properties of monostable reaction networks around a nonequilibrium steady state. We therefore quantify the relaxation kinetics of the fluctuations around a nonequilibrium steady state of mesoscopic chemical systems. We investigate the effect of an increasing magnitude of intrinsic noise due to decreasing reactor volume, and of an increase in the magnitude of extrinsic noise due to increasing burst size of a reactant fed to the reactor. We show that the lifetimes of fluctuations of species, which we define as the normalized power-spectral density of the concentration fluctuations at zero frequency, increase or decrease with increasing burst size. The lifetimes of all species involved as a reactant in a nonlinear reaction increase with decreasing reactor volume. We also show that the two sources of noise affect the derivative of the time-autocorrelation function in opposite ways, allowing us to differentiate between them. In addition, we show that for mesoscopic oscillatory systems, the peak frequency of the oscillation decreases and the bandwidth increases with decreasing reactor volume. This systematic change in the peak oscillation frequency and the bandwidth corroborates with the increase in lifetime observed in the non-oscillating mesoscopic reaction networks. We believe that understanding the change in the fluctuation characteristics at a nonequilibrium steady state might help estimate reaction rates by accounting for the effects of noise when using fluctuation trajectories from fluorescence correlation spectroscopy. In addition, it might help understand the effects of noise on stochastic chemical kinetics. Parts of this chapter have been presented in (Ramaswamy et al., 2011; Ramaswamy and Sbalzarini, 2011b).

Chapter 6: Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

We use the algorithms and results presented so far in two applications. The first one considers spatiotemporal stochastic simulations of reaction-diffusion processes. The CME assumes that the chemical system is well stirred and hence spatially homogeneous. The implicit assumption behind spatial homogeneity is that diffusion is much faster than reactions. Frequently, however, this condition is not met and hence the chemical systems are spatial heterogeneous. Spatial heterogeneity can be accounted for by the on-lattice reaction diffusion master equation (RDME). The RDME is analogous to the CME and models spatially heterogeneity by dividing the reactor into subvolumes, in each of which the reactions are assumed to be well mixed. Diffusion between subvolumes is modeled as jump reactions. Sampling trajectories from the RDME is computationally expensive. We propose a novel efficient algorithm based on partial propensities to sample trajectories from the exact solution of the RDME. The algorithm combines composition-rejection sampling to find the index of the subvolume with partial propensities to sample the index of the next reaction inside the subvolume. The computational cost of the algorithm is at most logarithmic in the total number of subvolumes and linear in

Introduction

the number of chemical species. Parts of this chapter have been presented in (Ramaswamy and Sbalzarini, 2011a).

Chapter 7: Global parameter identification for stochastic reaction networks from single trajectories

The second application considers the problem of identifying the unknown parameters of a stochastic reaction network from a single, noisy measurement. This constitutes a classical inverse problem. Inverse problems are commonplace in systems biology (Engl et al., 2009). They arise when constructing biochemical networks and when estimating the parameters of a biochemical network from experimental time series of species populations. The former is related to *systems identification*, while the latter amounts to *parameter identification*. We focus on parameter identification assuming that the underlying biochemical network is known. In biological systems, where many reactions are influenced by intrinsic and extrinsic noise, even an ideal experimental measurement of a population trajectory will be noisy due to the inherent fluctuations associated with stochastic chemical kinetics. Assuming that single-cell experimental trajectories from a known biochemical network are available, we propose a novel methodology to identify the parameters of the reaction network, namely the reaction rates and the volume of the reactor. The methodology combines partial-propensity methods for simulating the CME with an adaptive Monte Carlo sampling technique, called Gaussian adaptation (Kjellström and Taxen, 1992; Müller and Sbalzarini, 2010b,a; Müller, 2010), to search the parameter space. Gaussian adaptation iteratively proposes candidate parameters for which partial-propensity methods are then used to sample a trajectory from the CME. Subsequently, we quantify the distance between the simulated and the experimental trajectory using an objective function that includes the fingerprints of noise-induced modulations of the network dynamics as presented in previous chapters. This distance is then used by Gaussian adaptation to propose new candidate parameters. In addition to estimating reaction rates and the reactor volume, the present method also provides an ellipsoidal volume estimate of the viable parameter space, thus quantifying uncertainty of the produced solutions. Parts of this chapter have been presented in (Müller et al., 2012).

1

Stochastic chemical kinetics

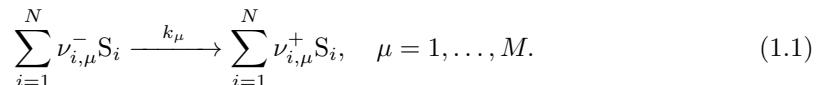
“The actual science of logic is conversant at present only with things either certain, impossible, or entirely doubtful, none of which (fortunately) we have to reason on. Therefore the true logic for this world is the calculus of Probabilities, which takes account of the magnitude of the probability which is, or ought to be, in a reasonable man’s mind.”

— James C. Maxwell

We recapitulate the derivation of the chemical master equation (CME) in Sec. 1.1 and the systematic approximations leading to the chemical Kramer-Moyal equation and the nonlinear chemical Fokker-Planck equation in Sec. 1.2. Further, we show the emergence of the deterministic reaction rate equation (RRE) at very large population in Sec. 1.2.2. In Sec. 1.3, we will present Gillespie’s stochastic simulation algorithm (SSA). We then demonstrate the effect of intrinsic noise on simple model systems in Sec. 1.4, using analytical and numerical approaches.

1.1 The chemical master equation (CME)

In general, a system of chemical reactions can be considered to comprise N species and M reactions, such that



Here, $\boldsymbol{\nu}^- = [\nu_{i,\mu}^-]$ and $\boldsymbol{\nu}^+ = [\nu_{i,\mu}^+]$ are the stoichiometry matrices of the reactants and products, respectively. Both of these matrices are of size $N \times M$. The overall stoichiometry matrix $\boldsymbol{\nu}$ of

1 Stochastic chemical kinetics

the reaction network is given by $\nu = \nu^+ - \nu^-$. The elements of ν^- and ν^+ are non-negative integers, while those of ν can be positive, negative or zero. We denote by ν_μ the μ^{th} column of ν . S_i is the i -th species in the reaction network and $n_i(t)$ is its population (molecular copy number) at time t . The population vector $\mathbf{n}(t) = [n_1, \dots, n_N]^T(t)$ is the state of the system. The reactions occur in a reactor of volume Ω , and the macroscopic reaction rate of reaction μ is k_μ . See Fig. 1.1 for an illustration.

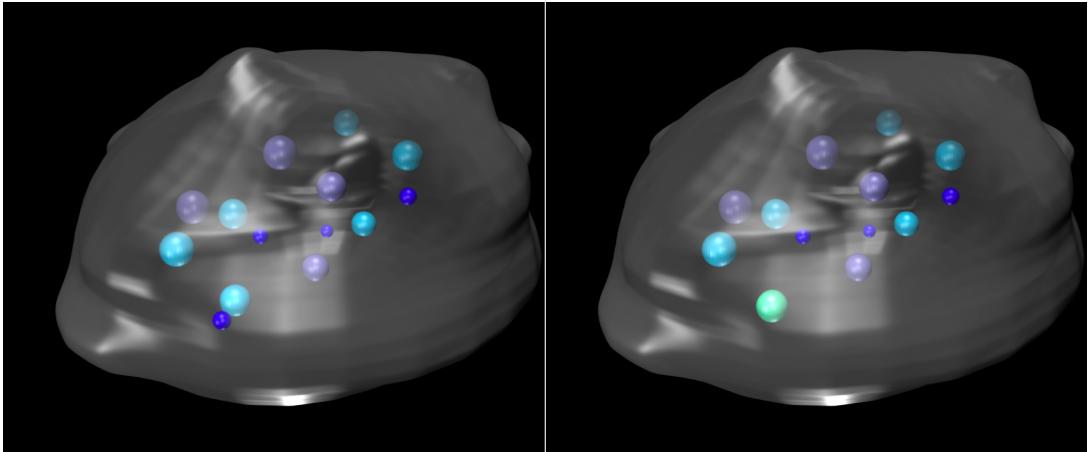


Figure 1.1: Illustration of molecules in a reactor. The reactor boundary is represented by the gray surface enclosing the molecules. The color of the molecules is used to encode the species. The molecules in the reactor possess kinetic energy and move around. Upon collision they may react to form molecules of a different species. Single molecules can also get transformed into a new species, or decay. The left panel shows a reactor with two molecules of different species in contact in the lower left corner. The right panel shows the formation of a molecule of a new species appearing in the place of the two molecules that are in contact in the left panel. This illustrates the occurrence of a reaction event.

It is custom to define a variable c_μ , called the specific probability rate of reaction μ (Gillespie, 1976, 1992), such that

$$c_\mu dt = \text{Probability of reaction } \mu \text{ firing in the next infinitesimal time interval} \\ [t, t + dt) \text{ with } \nu_{i,\mu}^- \text{ molecules of } S_i \quad i = 1, \dots, N. \quad (1.2)$$

This makes c_μ the smallest non-zero probability rate of reaction μ . The specific probability rate c_μ is related to the macroscopic reaction rate k_μ (Gillespie, 1976) as

$$c_\mu = \frac{k_\mu \left(\prod_{i=1}^N \nu_{i,\mu}^-! \right)}{\Omega^{(\sum_{i=1}^N \nu_{i,\mu}^-)-1}}. \quad (1.3)$$

The reaction degeneracy h_μ of reaction μ is defined as (Gillespie, 1976, 1992):

$$h_\mu(\mathbf{n}) = \text{Number of distinct combinations by which the reactants of reaction } \mu \text{ can} \\ \text{react to form products.} \quad (1.4)$$

1.1 The chemical master equation (CME)

Therefore

$$h_\mu = \prod_{i=1}^N \binom{n_i}{\nu_{i,\mu}^-}, \quad (1.5)$$

where

$$\binom{n_i}{\nu_{i,\mu}^-} = \frac{n_i!}{\nu_{i,\mu}^-!(n_i - \nu_{i,\mu}^-)!} = \frac{(n_i - \nu_{i,\mu}^- + 1)(n_i - \nu_{i,\mu}^- + 2) \dots (n_i - 1)n_i}{\nu_{i,\mu}^-!}. \quad (1.6)$$

From Eqs. 1.2 and 1.4 the probability rate or the propensity a_μ of reaction μ becomes

$$a_\mu(\mathbf{n})dt = c_\mu h_\mu(\mathbf{n})dt = \text{Probability of reaction } \mu \text{ firing in the next infinitesimal time interval } [t, t + dt) \text{ with } n_i \text{ molecules of } S_i, \quad i = 1, \dots, N. \quad (1.7)$$

Note that $a_\mu = c_\mu$ when $\mathbf{n} = \boldsymbol{\nu}_\mu^-$.

The state probability distribution function is defined as

$$P(\mathbf{n}, t | \mathbf{n}_0, t_0) = \text{Probability that the population of species is } \mathbf{n} \text{ at time } t, \text{ given a population } \mathbf{n}_0 \text{ at time } t_0. \quad (1.8)$$

Using the Chapman-Kolmogorov equation (Chapman, 1928; Kolmogorov, 1931; Feller, 1940, 1957) describing the time evolution of P for a Markov process

$$P(\mathbf{n}, t + dt | \mathbf{n}_0, t_0) = \sum_{\mathbf{n}_1} P(\mathbf{n}, t + dt | \mathbf{n}_1, t) P(\mathbf{n}_1, t | \mathbf{n}_0, t_0), \quad (1.9)$$

$$t_0 \leq t \leq t + dt. \quad (1.10)$$

For the chemical reaction defined in Eq. 1.1 the above equation can be equivalently written as

$$\begin{aligned} P(\mathbf{n}, t + dt | \mathbf{n}_0, t_0) &= \sum_{\mu=1}^M P(\mathbf{n}, t + dt | \mathbf{n} - \boldsymbol{\nu}_\mu, t) P(\mathbf{n} - \boldsymbol{\nu}_\mu, t | \mathbf{n}_0, t_0) \\ &\quad + P(\mathbf{n}, t + dt | \mathbf{n}, t) P(\mathbf{n}, t | \mathbf{n}_0, t_0), \end{aligned} \quad (1.11)$$

where $\boldsymbol{\nu}_\mu$ is the vector of stoichiometries of reaction μ , and $P(\mathbf{n}, t + dt | \mathbf{n} - \boldsymbol{\nu}_\mu, t)$ is the probability distribution $\Pi(\boldsymbol{\nu}_\mu | dt; \mathbf{n}, t)$ of the Markov propagator such that the equation of motion for the states (population) can be written as

$$\mathbf{n}(t + dt) = \mathbf{n}(t) + \boldsymbol{\Xi}(\boldsymbol{\nu}_\mu | dt; \mathbf{n}, t), \quad (1.12)$$

where $\boldsymbol{\Xi}(\boldsymbol{\nu}_\mu | dt; \mathbf{n}, t)$ is a random variable, called the Markov propagator, distributed such that $\boldsymbol{\Xi}(\boldsymbol{\nu}_\mu | dt; \mathbf{n}, t) \sim \Pi(\boldsymbol{\nu}_\mu | dt; \mathbf{n}, t)$.

Simplifying the notation, Eq. 1.11 reads

$$\begin{aligned} &P(\mathbf{n}, t + dt) \\ &= \sum_{\mu=1}^M P(\mathbf{n} - \boldsymbol{\nu}_\mu, t) \text{ Prob}\{\text{Reaction } \mu \text{ fires once in } [t, t + dt) \text{ given } \mathbf{n} - \boldsymbol{\nu}_\mu \text{ at } t\} \quad (1.13) \end{aligned}$$

$$+ P(\mathbf{n}, t) \text{ Prob}\{\text{No reaction in } [t, t + dt) \text{ given } \mathbf{n} \text{ at } t\}. \quad (1.14)$$

1 Stochastic chemical kinetics

The reasoning behind this equation is as follows: The first summand on the right-hand side is the probability that the chemical reaction system moves from a population state $\mathbf{n} - \boldsymbol{\nu}_\mu$ at time t to the population state \mathbf{n} at time $t + dt$. The second term is the probability that the system is at state \mathbf{n} at time t and stays at the same state at time $t + dt$. The sum of these two terms is therefore the probability of the system being in state \mathbf{n} at time $t + dt$.

The probability of reaction μ firing once in the time interval $[t, t + dt)$ is the product of the probability that one combination of its reactant molecules reacts to form products, and the probability that all other combinations do not react. Therefore,

$$\begin{aligned}\text{Prob}\{\text{Reaction } \mu \text{ fires once in } [t, t + dt) \text{ given } \mathbf{n} \text{ at } t\} &= P(\mathbf{n}, t + dt \mid \mathbf{n} - \boldsymbol{\nu}_\mu, t) \\ &= \Pi(\boldsymbol{\nu}_\mu \mid dt; \mathbf{n}, t) \\ &= c_\mu dt(1 - c_\mu dt)^{h_\nu(\mathbf{n})-1} \\ &= c_\mu h_\mu(\mathbf{n})dt + O(dt^2) \\ &= a_\mu(\mathbf{n})dt + O(dt^2),\end{aligned}\quad (1.15)$$

where $O(dt^2)$ is the Bachmann-Landau big- O symbol* such that $\lim_{dt \rightarrow 0} \frac{O(dt^\alpha)}{dt} = 0$ for $\alpha > 1$. It hence follows that the probability that no reactions occur in $[t, t + dt)$ is given by

$$\begin{aligned}\text{Prob}\{\text{No reaction in } [t, t + dt) \text{ given } \mathbf{n} \text{ at } t\} &= P(\mathbf{n}, t + dt \mid \mathbf{n}, t) \\ &= 1 - \sum_1^M a_\mu(\mathbf{n})dt + O(dt^2) \\ &= 1 - a(\mathbf{n})dt + O(dt^2),\end{aligned}\quad (1.16)$$

where the total propensity $a(\mathbf{n}) = \sum_{\mu=1}^M a_\mu(\mathbf{n})$. Equation 1.14 can therefore be written as

$$\begin{aligned}P(\mathbf{n}, t + dt) &= \sum_{\mu=1}^M P(\mathbf{n} - \boldsymbol{\nu}_\mu, t) [a_\mu(\mathbf{n} - \boldsymbol{\nu}_\mu)dt + O(dt^2)] \\ &\quad + P(\mathbf{n}, t) \left[1 - \sum_1^M a_\mu(\mathbf{n})dt + O(dt^2) \right],\end{aligned}\quad (1.17)$$

i.e.,

$$\begin{aligned}\frac{P(\mathbf{n}, t + dt) - P(\mathbf{n}, t)}{dt} &= \sum_{\mu=1}^M \left[a_\mu(\mathbf{n} - \boldsymbol{\nu}_\mu) + \frac{O(dt^2)}{dt} \right] P(\mathbf{n} - \boldsymbol{\nu}_\mu, t) \\ &\quad - P(\mathbf{n}, t) \left[\sum_{\mu=1}^M a_\mu(\mathbf{n})dt + \frac{O(dt^2)}{dt} \right].\end{aligned}\quad (1.18)$$

Taking the limit $\lim_{dt \rightarrow 0}$ we obtain the *chemical master equation* (CME) (McQuarrie, 1967;

* $f(x) \in O(g(x))$, or $f(x)$ is $O(g(x))$, implies that $f(x)$ is asymptotically bounded from above by $g(x)$, i.e., $\exists \epsilon > 0, x_0 : \forall x > x_0, |f(x)| \leq \epsilon g(x)|$

Gillespie, 1992) as

$$\frac{\partial P(\mathbf{n}, t)}{\partial t} = \sum_{\mu=1}^M a_\mu(\mathbf{n} - \boldsymbol{\nu}_\mu) P(\mathbf{n} - \boldsymbol{\nu}_\mu, t) - P(\mathbf{n}, t) a(\mathbf{n}). \quad (1.19)$$

The CME is the forward equation of a discrete-state jump Markov process obtained as a direct consequence of the Chapman-Kolmogorov equation (Chapman, 1928; Kolmogorov, 1931; Feller, 1940, 1957). The CME can also be written as

$$\frac{\partial P(\mathbf{n}, t)}{\partial t} = \sum_{\mu=1}^M (\mathbb{E}^{-\boldsymbol{\nu}_\mu} - 1) a_\mu(\mathbf{n}) P(\mathbf{n}, t), \quad (1.20)$$

where \mathbb{E} is the step operator such that $\mathbb{E}^{\mathbf{a}} f(\mathbf{n}) = f(\mathbf{n} + \mathbf{a})$. The solution of the CME can therefore be written as

$$P(\mathbf{n}, t) = e^{t[\sum_{\mu=1}^M (\mathbb{E}^{-\boldsymbol{\nu}_\mu} - 1) a_\mu(\mathbf{n})]} P(\mathbf{n}, 0). \quad (1.21)$$

The initial condition is given by

$$P(\mathbf{n}, t=0) = \delta(\mathbf{n} - \mathbf{n}_0), \quad (1.22)$$

where \mathbf{n}_0 is the population at time $t = 0$ and $\delta(\cdot)$ is the Kronecker delta or the unit impulse function.

1.2 Approximating the CME

We define the concentration vector of species in a volume Ω as

$$\phi = \Omega^{-1} \mathbf{n}. \quad (1.23)$$

In terms of the concentration ϕ , the propensity a_μ can be written as

$$\begin{aligned} a_\mu(\phi) &= a_\mu(\mathbf{n}) \\ &= \left(\prod_{i=1}^N \frac{(n_i - \nu_{i,\mu}^- + 1)(n_i - \nu_{i,\mu}^- + 2) \dots (n_i - 1)n_i}{\nu_{i,\mu}^-!} \right) \left(\frac{k_\mu \prod_{i=1}^N \nu_{i,\mu}^-!}{\Omega^{(\sum_{i=1}^N \nu_{i,\mu}^-)-1}} \right) \\ &= \left(\prod_{i=1}^N \frac{(\phi_i - \frac{\nu_{i,\mu}^- - 1}{\Omega})(\phi_i - \frac{\nu_{i,\mu}^- - 2}{\Omega}) \dots (\phi_i - \frac{1}{\Omega})\phi_i}{\nu_{i,\mu}^-!} \right) \left(k_\mu \prod_{i=1}^N \nu_{i,\mu}^-! \right) \Omega \\ &= T_\mu(\phi) \Omega. \end{aligned} \quad (1.24)$$

Rewriting Eq. 1.19 in terms of the concentration ϕ , we get

$$\frac{\partial P(\phi, t)}{\partial t} = \Omega \left[\sum_{\mu=1}^M T_\mu(\phi - \Omega^{-1} \boldsymbol{\nu}_\mu) P(\phi - \Omega^{-1} \boldsymbol{\nu}_\mu, t) - P(\phi, t) \sum_{\mu=1}^M T_\mu(\phi) \right]. \quad (1.25)$$

1.2.1 The chemical Kramer-Moyal equation and the nonlinear chemical Fokker-Planck equation

Assuming that the population \mathbf{n} increases proportionally with reactor volume Ω , such that the concentration ϕ is constant, we can treat ϕ as a continuous random variable for a sufficiently large Ω . Performing a Taylor series expansion of the right-hand side of Eq. 1.25, we get

$$\frac{\partial P(\phi, t)}{\partial t} = \left\{ \sum_{m=1}^{\infty} (-1)^m \Omega^{-(m-1)} \sum_{\substack{i_1, \dots, i_N \\ i_1 + \dots + i_N = m}} \frac{1}{i_1! \dots i_N!} \frac{\partial^m}{\partial^{i_1} \phi_1 \dots \partial^{i_N} \phi_N} \left[\sum_{\mu=1}^M \prod_{j=1}^N \nu_{j,\mu}^{i_j} T_\mu(\phi) \right] \right\} P(\phi, t) \quad (1.26)$$

$$= \left\{ \sum_{m=1}^{\infty} \frac{(-1)^m}{\Omega^{(m-1)}} \sum_{\substack{i_1, \dots, i_N \\ i_1 + \dots + i_N = m}} \frac{1}{i_1! \dots i_N!} \frac{\partial^m}{\partial^{i_1} \phi_1 \dots \partial^{i_N} \phi_N} b_{k; i_1, \dots, i_N} \right\} P(\phi, t), \quad (1.27)$$

where $b_{k; i_1, \dots, i_N}$ is given by

$$b_{k; i_1, \dots, i_N} = \left[\sum_{\mu=1}^M \prod_{j=1}^N \nu_{j,\mu}^{i_j} T_\mu(\phi) \right] \quad i_1 + \dots + i_N = k. \quad (1.28)$$

Eq. 1.27 is the *chemical Kramer-Moyal equation* (Kramers, 1940; Feller, 1940; Moyal, 1949; Feller, 1957; Gillespie, 2000) describing the time evolution of the state probability function of a continuous-state jump Markov process. Its solution is given by

$$P(\phi, t) = e^{\left\{ t \left[\sum_{m=1}^{\infty} \frac{(-1)^m}{\Omega^{(m-1)}} \sum_{\substack{i_1, \dots, i_N \\ i_1 + \dots + i_N = m}} \frac{1}{i_1! \dots i_N!} \frac{\partial^m}{\partial^{i_1} \phi_1 \dots \partial^{i_N} \phi_N} b_{k; i_1, \dots, i_N} \right] \right\}} P(\phi, 0), \quad (1.29)$$

1.2 Approximating the CME

where $P(\phi, 0)$ is the initial condition. The $b_{k; i_1, \dots, i_N}$ are related to the k^{th} jump moments $B_{k; i_1, \dots, i_N}$ of the Markov propagator probability distribution $\Pi(\nu_\mu | dt; \mathbf{n}, t)$ as

$$B_{k; i_1, \dots, i_N} = \lim_{dt \rightarrow 0} \frac{1}{dt} \sum_{\mu=1}^M \prod_{j=1}^N \nu_{j,\mu}^{i_j} \Pi(\nu_\mu | dt; \mathbf{n}, t) \quad (1.30)$$

$$\begin{aligned} &= \lim_{dt \rightarrow 0} \frac{1}{dt} \sum_{\mu=1}^M \prod_{j=1}^N \nu_{j,\mu}^{i_j} [a_\mu(\mathbf{n}) dt + O(dt^2)] \\ &= \sum_{\mu=1}^M \prod_{j=1}^N \nu_{j,\mu}^{i_j} a_\mu(\mathbf{n}) \\ &= \Omega \sum_{\mu=1}^M \prod_{j=1}^N \nu_{j,\mu}^{i_j} T_\mu(\phi) \\ &= \Omega b_{k; i_1, \dots, i_N}. \end{aligned} \quad (1.31)$$

Ignoring all terms with $m > 2$ in Eq. 1.27 yields the *nonlinear chemical Fokker-Planck equation*, also known as the *generalized diffusion equation* (Chandrasekhar, 1943; Haken, 1975; Bedeaux, 1977; Gillespie, 1996a,b, 2000; van Kampen, 2001). This equation appropriately describes stochastic chemical kinetics at large-enough Ω (i.e., at large population \mathbf{n}), where the jump moments $B_{k; i_1, \dots, i_N} = 0$ for $k > 2$. The Markov propagator probability distribution then becomes Gaussian. This truncation transforms the continuous-state jump Markov process describing chemical kinetics at large Ω into a continuous Markov process at even larger Ω . We will show later in Chapter 2 under Section 2.4 that the *nonlinear chemical Fokker-Planck equation* is not just a result of arbitrary truncation of the chemical Kramer-Moyal equation, but is substantiated by physical reasoning (Horsthemke and Brenig, 1977; Gillespie, 2000, 2001). The nonlinear chemical Fokker-Planck equation is an anisotropic inhomogeneous diffusion equation with drift for the probabilities in concentration space. It can be written as

$$\frac{\partial P(\phi, t)}{\partial t} = \nabla \cdot [(2\Omega)^{-1} \mathbf{D}(\phi) \nabla P(\phi, t) - \mathbf{F}(\phi) P(\phi, t)] \quad (1.32)$$

where $\nabla = \left[\frac{\partial}{\partial \phi_1}, \dots, \frac{\partial}{\partial \phi_N} \right]^T$. The drift \mathbf{F} is the vector of first moments $b_{1; i_1, \dots, i_N}$

$$\mathbf{F}(\phi) = \boldsymbol{\nu} \mathbf{T}(\phi) \quad (1.33)$$

where $\mathbf{T}(\phi) = [T_1(\phi), \dots, T_M(\phi)]^T$. The diffusion tensor \mathbf{D} is the matrix of second moments $b_{2; i_1, \dots, i_N}$ (see Eqs. 1.31 and 1.30) (Elf and Ehrenberg, 2003)

$$\mathbf{D}(\phi) = \boldsymbol{\nu} \text{ diag}(\mathbf{T}(\phi)) \boldsymbol{\nu}^T, \quad (1.34)$$

where

$$\text{diag}(\mathbf{T}(\phi)) = \begin{bmatrix} T_1(\phi) & 0 & \dots & 0 \\ 0 & T_2(\phi) & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & T_M(\phi) \end{bmatrix}. \quad (1.35)$$

1 Stochastic chemical kinetics

Under this approximation, the equation of motion of the Markov process can be written as

$$\phi(t + dt) = \phi(t) + \Xi(dt; \phi, t), \quad (1.36)$$

where Ξ is a random variable distributed as $\Xi(dt; \phi, t) \sim \Pi(\Delta\phi | dt; \phi, t)$ with $\Pi(\Delta\phi | dt; \phi, t)$ the Gaussian distribution $\mathcal{N}(\mathbf{F}(\phi)dt, \Omega^{-1}\mathbf{D}(\phi)dt)$ (Gillespie, 1996a,b). Here $\mathcal{N}(\mathbf{m}, \boldsymbol{\Sigma})$ denotes a multivariate Gaussian distribution with mean vector \mathbf{m} and covariance matrix $\boldsymbol{\Sigma}$. Eq. 1.36 is called the *chemical Langevin equation* (CLE) (Gillespie, 2000). According to the CLE, ϕ can be considered the coordinate of an overdamped particle moving in the N -dimensional concentration space under the influence of a force $\mathbf{F}(\phi)$ (drift) and an anisotropic zero-mean Gaussian perturbation with covariance $\Omega^{-1}\mathbf{D}(\phi)$ (diffusion) (Qian et al., 2002; Qian, 2006).

1.2.2 The macroscopic reaction rate equation (RRE)

The function $T_\mu(\phi)$ (Eq. 1.24) explicitly depends on ϕ and Ω . These dependences can be separated by expanding $T_\mu(\phi)$

$$T_\mu(\phi) = \sum_{i=0}^{\infty} \Omega^{-i} T_{\mu,i}(\phi), \quad (1.37)$$

where the $T_{\mu,i}$'s only depend on ϕ and not explicitly on Ω . The moments $b_{k;i_1, \dots, i_N}$ (Eq. 1.31), the drift vector $\mathbf{F}(\phi)$ and the diffusion $\mathbf{D}(\phi)$ in the nonlinear Fokker-Planck equation (Eqs. 1.32, 1.33 and 1.34) can therefore be written as:

$$b_{k;i_1, \dots, i_N} = \sum_{i=0}^{\infty} b_{k;i_1, \dots, i_N}^{(i)} \Omega^{-i}, \quad (1.38)$$

$$\mathbf{F}(\phi) = \sum_{i=0}^{\infty} \mathbf{F}^{(i)}(\phi) \Omega^{-i} \quad (1.39)$$

and

$$\mathbf{D}(\phi) = \sum_{i=0}^{\infty} \mathbf{D}^{(i)}(\phi) \Omega^{-i}. \quad (1.40)$$

The first few $T_{\mu,i}$'s are:

$$T_{\mu,0}(\phi) = k_\mu \prod_{i=1}^N \phi_i^{\nu_{i,\mu}^-}, \quad (1.41)$$

$$T_{\mu,1}(\phi) = -k_\mu \left(\prod_{i=1}^N \phi_i^{\nu_{i,\mu}^-} \right) \left[\sum_{j=1}^N \frac{\nu_{j,\mu}^-(\nu_{j,\mu}^- - 1)}{2\phi_j} \right] = -\sum_{i=1}^N \frac{\phi_i}{2} \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_i^2} \quad (1.42)$$

1.3 Kinetic Monte Carlo: The stochastic simulation algorithm (SSA)

and

$$\begin{aligned} T_{\mu,2}(\phi) &= k_\mu \left(\prod_{i=1}^N \phi_i^{\nu_{i,\mu}^-} \right) \left[\sum_{p=1}^{N-1} \sum_{q=p+1}^N \frac{\nu_{p,\mu}^- \nu_{q,\mu}^- (\nu_{p,\mu}^- - 1)(\nu_{q,\mu}^- - 1)}{2\phi_p \phi_q} \right] \\ &+ k_\mu \left(\prod_{i=1}^N \phi_i^{\nu_{i,\mu}^-} \right) \left[\sum_{j=1}^N \frac{\nu_{j,\mu}^- (\nu_{j,\mu}^- - 1)(\nu_{j,\mu}^- - 2)(3\nu_{j,\mu}^- - 1)}{12\phi_j^2} \right]. \end{aligned} \quad (1.43)$$

Substituting Eq. 1.37 in Eq. 1.26 and grouping the terms according to powers of Ω , we get

$$\begin{aligned} \frac{\partial P(\phi, t)}{\partial t} &= \Omega^0 [-(\boldsymbol{\nu}_\mu \cdot \boldsymbol{\nabla}) T_{\mu,0}(\phi)] \\ &+ \frac{\Omega^{-1}}{2} [(\boldsymbol{\nu}_\mu \cdot \boldsymbol{\nabla})^2 T_{\mu,0}(\phi) - 2(\boldsymbol{\nu}_\mu \cdot \boldsymbol{\nabla}) T_{\mu,1}(\phi)] P(\phi, t) \\ &+ O(\Omega^{-2}) \end{aligned} \quad (1.44)$$

Taking the limit $\lim_{\Omega \rightarrow \infty}$ we get

$$\frac{\partial P(\phi, t)}{\partial t} = -[(\boldsymbol{\nu}_\mu \cdot \boldsymbol{\nabla}) T_{\mu,0}(\phi)]. \quad (1.45)$$

This equation further reduces the continuous-state Markov process described by the chemical nonlinear Fokker-Planck equation to a *Liouville* process. Its solution for $P(\phi', t)$ is the Kronecker delta function $\delta(\phi' - \phi)$ where the concentration ϕ with unit probability is given by the solution of the ordinary differential equation

$$\frac{d\phi}{dt} = \boldsymbol{\nu} \mathbf{T}_0(\phi). \quad (1.46)$$

Here $\mathbf{T}_0(\phi) = [T_{1,0}(\phi), \dots, T_{M,0}(\phi)]^T$. Eq. 1.46 is the classical *reaction rate equation* (RRE), which can be independently derived from statistical mechanics. The initial condition for the RRE is given by

$$\phi(t=0) = \phi_0 = \Omega^{-1} \mathbf{n}_0, \quad (1.47)$$

where \mathbf{n}_0 is the initial population (see Eq. 1.22). In some parts of the literature $T_{\mu,0}$ is referred to as the rate function or the flux of reaction μ since it describes the flux of converting the reactants of reaction μ into the corresponding products.

1.3 Kinetic Monte Carlo: The stochastic simulation algorithm (SSA)

The CME provides an exact description of the kinetics of mesoscopic chemical reaction systems that are well stirred and thermally equilibrated (Gillespie, 1992). The CME incorporates the effects of fluctuations due to low copy numbers of species by describing the effect of discrete nature of molecules involved in chemical reactions. At larger Ω , when the population \mathbf{n}

1 Stochastic chemical kinetics

increases, the effect of intrinsic noise becomes progressively smaller. As seen in Sec. 1.2.2, for very large reactor volumes Ω (or large population \mathbf{n} under the assumption that \mathbf{n} increases proportionally with Ω), the CME reduces to the classical RRE (see Eq. 1.46). For intermediate values of Ω , the CME leads to the chemical Kramer-Moyal equation and to the nonlinear Fokker-Planck equation (See Sec. 1.2.1).

Solving the CME, however, is riddled with problems. Analytically, the CME is generally intractable, except for networks of unimolecular reactions (i.e., when $\sum_{i=1}^N \nu_{i,\mu}^- \leq 1 \forall \mu$ in Eq. 1.1). Such networks are called *linear reaction networks* since the propensities are linear in the population. For networks with even a single bimolecular or higher order reaction (i.e., if $\sum_{i=1}^N \nu_{i,\mu}^- > 1$ for at least one reaction in Eq. 1.1), the solution of the CME is analytically not accessible, except in special cases. Such networks are called *nonlinear reaction networks* since at least one of the propensities is nonlinear in the population.

Numerically simulating the CME using, for example, finite differences is also infeasible for large networks due to the high dimensionality of the domain of the probability distribution $P(\mathbf{n}, t)$, which leads to an exponential increase in computational and memory cost with network size. These problems, however, can be circumvented using Gillespie's stochastic simulation algorithm (SSA), a kinetic Monte Carlo scheme (Doob, 1942, 1945; Kendall, 1949; Barlett, 1953; Bortz et al., 1975; Gillespie, 1976, 1977, 1992). In SSA, the probability $P(\mathbf{n}, t)$ whose time evolution is given by the CME is replaced by the joint probability of a reaction event $p(\tau, \mu | \mathbf{n}(t))$, defined as

$$p(\tau, \mu | \mathbf{n}(t))d\tau = \begin{aligned} &\text{Probability that the next reaction is } \mu \text{ and it fires in} \\ &[t + \tau, t + \tau + d\tau] \text{ given } \mathbf{n} \text{ at time } t. \end{aligned} \quad (1.48)$$

This probability p is derived as follows: Consider that the time interval $[t, t + \tau]$ is divided into k equal intervals of length $\frac{\tau}{k}$ plus a last interval ($k + 1$) of length $d\tau$ (Fig. 1.2). The definition

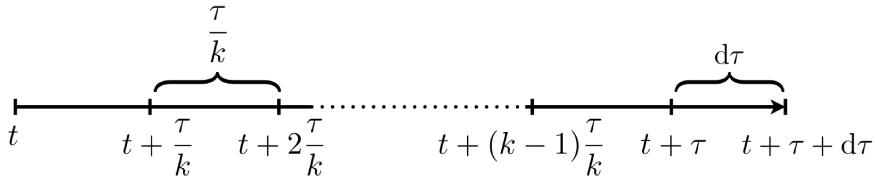


Figure 1.2: Division of the time interval $[t, t + \tau + d\tau]$ into $k + 1$ intervals. Here, t represents the current time. The only reaction firing is reaction μ in the $(k + 1)^{\text{th}}$ infinitesimally small time interval $[t + \tau, t + \tau + d\tau]$.

(Eq. 1.48) of $p(\tau, \mu | \mathbf{n}(t))$ dictates that no reactions occur in each of the first k intervals and that reaction μ fires once in the last interval. Therefore, using Eqs. 1.15 and 1.16

$$p(\tau, \mu | \mathbf{n}(t))d\tau = \left[1 - a(\mathbf{n}) \frac{\tau}{k} + O\left(\frac{\tau^2}{k^2}\right) \right]^k [a_\mu(\mathbf{n})d\tau + O(d\tau^2)].$$

Dividing both sides of the equation by $d\tau$ and taking the limit $\lim_{d\tau \rightarrow 0}$ we obtain

$$p(\tau, \mu | \mathbf{n}(t)) = \left[1 - a(\mathbf{n}) \frac{\tau}{k} + O\left(\frac{\tau^2}{k^2}\right) \right]^k a_\mu(\mathbf{n}).$$

Taking the limit $\lim_{k \rightarrow \infty}$, and noting that $\lim_{k \rightarrow \infty} O\left(\frac{\tau^2}{k^2}\right) = 0$, we get

$$p(\tau, \mu \mid \mathbf{n}(t)) = e^{-a(\mathbf{n})\tau} a_\mu(\mathbf{n}). \quad (1.49)$$

Summing Eq. 1.49 over all reactions (summing over μ) we get the marginal probability density function of τ as

$$\begin{aligned} p(\tau \mid \mathbf{n}(t)) &= \sum_{\mu=1}^M a_\mu(\mathbf{n}) e^{-a(\mathbf{n})\tau} \\ &= a(\mathbf{n}) e^{-a(\mathbf{n})\tau}. \end{aligned} \quad (1.50)$$

Similarly, integrating Eq. 1.49 over τ 's we get the marginal probability distribution function of μ as

$$\begin{aligned} p(\mu \mid \mathbf{n}(t)) &= \int_0^\infty a_\mu(\mathbf{n}) e^{-a(\mathbf{n})\tau} d\tau \\ &= \frac{a_\mu(\mathbf{n})}{a(\mathbf{n})}. \end{aligned} \quad (1.51)$$

From Eqs. 1.49, 1.50 and 1.51 we observe that

$$p(\tau, \mu \mid \mathbf{n}(t)) = p(\tau \mid \mathbf{n}(t)) p(\mu \mid \mathbf{n}(t)) \quad (1.52)$$

thus inferring that μ and τ are statistically independent random variables.

Sampling from Eq. 1.49 constitutes a stochastic simulation of the underlying chemical reaction network. By sampling a reaction event and propagating the simulation in time according to Eq. 1.49, we obtain exact, time resolved trajectories of the population \mathbf{n} as governed by the CME. The SSA, however, is a Monte Carlo scheme and hence several independent runs need to be performed in order to obtain a stable representation of the probability function $P(\mathbf{n}, t)$. In Chapter 2 we describe the different implementations or formulations of the SSA used to simulate chemical kinetics, and we review their computational costs. Subsequently, in Chapter 3 we will present a more efficient class of SSA formulations called partial-propensity methods that use a novel quantity called *partial propensity* instead of the conventional reaction propensities.

1.4 The effect of intrinsic noise on chemical kinetics

When the population \mathbf{n} increases proportionally with Ω such that the total mass density of species is constant, decreasing Ω leads to an increase in intrinsic noise. Intrinsic noise does not only play the role of adding uncertainty to the various statistical estimates of \mathbf{n} , but it also has been shown to alter chemical kinetics in a non-trivial way (Kendall, 1949; van Kampen, 1963; McQuarrie, 1967; Gillespie, 1992; McAdams and Arkin, 1997; Arkin et al., 1998; McClintock, 1999; Elowitz and Leibler, 2000; Barkai and Leibler, 2000; Berg et al., 2000; van Kampen, 2001; Qian et al., 2002; Elf and Ehrenberg, 2003; Eldar and Elowitz, 2010). As a consequence, intrinsic noise may lead to a probability function $P(\phi, t)$ whose mean is not the same as the concentration ϕ predicted by the RRE, but they can be qualitatively different. We illustrate the effect of intrinsic noise on different types of nonequilibrium reaction networks:

1 Stochastic chemical kinetics

1. Monostable reaction networks: linear and nonlinear,
2. An oscillatory reaction network (exhibiting limit-cycle oscillations),
3. A multistable reaction network.

These three different types of reaction networks should be sufficient to understand the qualitative effect of intrinsic noise since they span a large spectrum of dynamics exhibited by chemical reaction networks.

1.4.1 Monostable, linear reaction networks

Linear reaction networks are defined as those in which the sum of the stoichiometries of all reactants is less than 2 in every reaction. Consider the following reaction network as an example of a linear reaction network occurring in a reactor of volume Ω :



The RRE for this reaction network is

$$\frac{d\phi_1}{dt} = k_1 - k_2\phi_1, \quad (1.54)$$

where ϕ_1 is the concentration of species S_1 . The fixed point of the RRE is $\phi_{1,ss} = \frac{k_1}{k_2}$. This sole fixed point is stable irrespective of the values of k_1 and k_2 , the system is monostable.

The CME of this reaction network is

$$\frac{\partial P(n_1, t)}{\partial t} = k_1\Omega P(n_1 - 1, t) + (n_1 + 1)k_2P(n_1 + 1, t) - (k_1\Omega + n_1k_2)P(n_1, t), \quad (1.55)$$

where n_1 is the population of species S_1 . Describing a linear chemical reaction system, the CME can be solved analytically. The steady-state or stationary probability function is

$$P_{ss}(n_1) = \frac{e^{-(k_1\Omega/k_2)}(k_1\Omega/k_2)^{n_1}}{n_1!}. \quad (1.56)$$

Denoting stochastic variables with an asterisk, the steady-state mean concentration becomes

$$\langle\phi_1^*\rangle_{ss} = \frac{\langle n_1 \rangle_{ss}}{\Omega} = \frac{k_1}{k_2}. \quad (1.57)$$

The mean steady-state concentration hence is the same as the steady-state concentration predicted by the deterministic RRE, for all Ω . This is generally true for linear reaction networks.

The steady-state variance of the concentration is

$$(\langle\phi_1^{*2}\rangle - \langle\phi_1^*\rangle^2)_{ss} = \frac{k_1}{k_2\Omega}. \quad (1.58)$$

1.4 The effect of intrinsic noise on chemical kinetics

This shows that the steady-state concentration variance decrease as Ω^{-1} with increase in Ω . This is also generally true for all linear reaction systems.

We set the initial concentration of S_1 to 10, the rates $k_1 = 5$ and $k_2 = 10$. Using these parameters we report trajectories and the steady-state probability function obtained using SSA. Figure 1.3A shows the time evolution of the concentration of species S_1 from a single SSA run for different Ω 's, and from the RRE. It can be seen that the fluctuations increase with decreasing Ω . This is also apparent in Fig. 1.3B showing $P_{ss}(\phi_1^*)$, which is a Kronecker delta for the deterministic RRE and becomes increasing broader for smaller Ω 's. The variance of $P_{ss}(\phi_1^*)$ is shown in Fig. 1.10 for different Ω . As expected from the analytical result (Eq. 1.58), the variance scales with Ω^{-1} .

In summary, for any monostable linear reaction network, the mean of the concentration probability function $P(\phi, t)$ is equal to the deterministic concentration $\phi(t)$ from RRE. The intrinsic noise in the CME provides zero-mean fluctuations about the deterministic concentrations. Further, for monostable linear reaction networks the variance of concentration decrease as Ω^{-1} with increasing Ω .

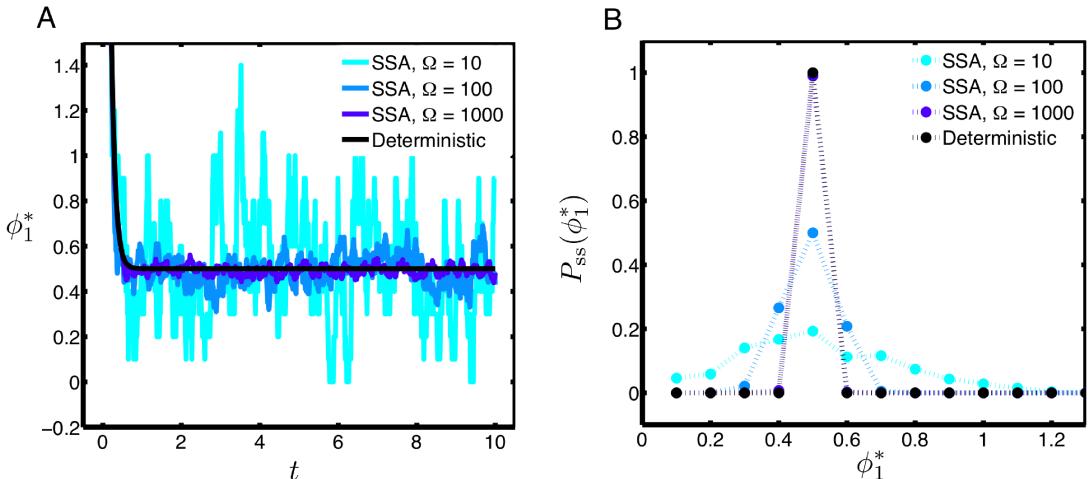


Figure 1.3: (A) Time evolution of the concentration and (B) steady-state concentration probability distribution of species S_1 in the monostable, linear reaction network of Eq. 1.53. The results are obtained using the stochastic simulation algorithm (SSA) for different reactor volumes Ω , and using the deterministic reaction rate equation (RRE). The rates used for the simulation are: $k_1 = 5$ and $k_2 = 10$.

1.4.2 Monostable, nonlinear reaction networks

Any reaction network having at least one reaction in which the sum of the stoichiometries of the reactants is greater than 1 is a nonlinear reaction network. As an example consider:



The corresponding RRE is

$$\frac{d\phi_1}{dt} = k_1 - 2k_1\phi_1^2 \quad (1.60)$$

where ϕ_1 is the concentration of species S_1 . The fixed point of the RRE is $\phi_{1,ss} = \sqrt{\frac{k_1}{2k_2}}$. This fixed point is stable and hence the system is monostable.

The CME of this reaction network is

$$\begin{aligned} \frac{\partial P(n_1, t)}{\partial t} = & k_1 \Omega P(n_1 - 1, t) + (n_1 + 2)(n_1 + 1) \frac{k_2}{\Omega} P(n_1 + 2, t) \\ & - \left[k_1 \Omega + n_1(n_1 - 1) \frac{k_2}{\Omega} \right] P(n_1, t), \end{aligned} \quad (1.61)$$

where n_1 is the population of species S_1 . This is one of the very few nonlinear reaction networks for which the steady-state mean concentration can be obtained analytically using moment generating functions (Engblom, 2006) (see Appendix A1 for the derivation). It is given by

$$\langle \phi_1^* \rangle_{ss} = \frac{1}{4\Omega} + \phi_{1,ss} \frac{I'_1(4\phi_{1,ss}\Omega)}{I_1(4\phi_{1,ss}\Omega)}, \quad (1.62)$$

where $I_a(b)$ is the modified Bessel function of the first kind and $I'_a(b) = \frac{dI_a(x)}{dx}|_{x=b}$. We see that $\langle \phi_1^* \rangle_{ss}$ tends to $\phi_{1,ss}$ as Ω tends to infinity (see Eq. A1.11 and A1.12). For finite Ω , there is a non-zero difference between the steady-state mean concentration obtained from the CME and the steady-state concentration from the RRE. In general, this difference always persists for nonlinear reaction networks at finite Ω .

Using $k_1 = 5$, $k_2 = 10$ and the initial concentration of S_1 set to 10, we see that the steady-state concentration variance scales with Ω^{-1} as shown in Fig. 1.10. Figure 1.4A shows the time evolution of the concentration of species S_1 obtained from a single SSA run for different Ω 's, and from the RRE. It can be seen that the fluctuations increase with decreasing Ω , evident from the broadening of $P_{ss}(\phi_1^*)$ in Fig. 1.4B in agreement with the scaling of the variance observed in Fig. 1.10. The steady-state mean concentrations are 0.5130, 0.5012 and 0.5001 for $\Omega = 10, 100$ and 1000, respectively. The steady-state concentration $\phi_{1,ss}$ from the RRE is 1/2. This shows that even at a small volume of $\Omega = 10$, the difference between $\phi_{1,ss}$ and $\langle \phi_1^* \rangle_{ss}$ is just about 2.6%. This small difference is specific to the particular system and in general the difference need not be so small for other monostable nonlinear reaction networks.

In summary, for any monostable nonlinear reaction network, the mean of the concentration probability function $P(\phi, t)$ is *not* equal to the deterministic concentration $\phi(t)$ obtained

from the RRE. The intrinsic noise in the CME deviates the mean concentration from the deterministic concentration. The variance of the concentrations, however, decrease as Ω^{-1} with increasing Ω . This results in the mean concentration of $P(\phi, t)$ approaching the deterministic concentration with the difference becoming increasingly smaller at larger Ω . In Chapter 4 we will show that for monostable nonlinear reaction networks with multiple species intrinsic noise in the CME can even lead to qualitative difference in the steady-state of the system. Specifically, we will show that intrinsic noise leads to an ordering in steady-state that is different from the RRE prediction, and we will present a theory that is in good agreement with the CME prediction.

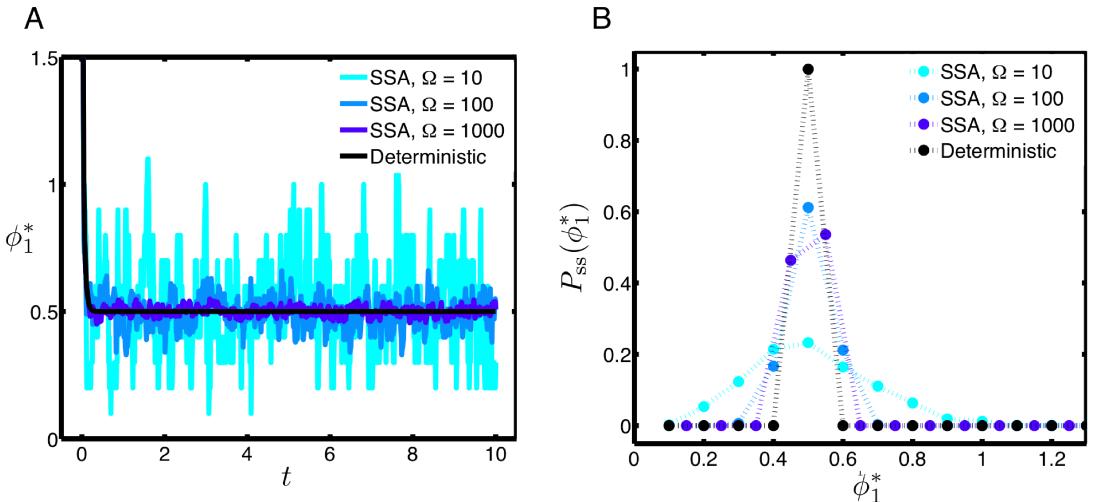


Figure 1.4: (A) Time evolution of the concentration and (B) steady-state concentration probability distribution of species S_1 in the monostable, nonlinear reaction network of Eq. 1.59. The results are obtained using the stochastic simulation algorithm (SSA) for different reactor volumes Ω , and using the deterministic reaction rate equation (RRE). The rates used for the simulation are: $k_1 = 5$ and $k_2 = 10$.

1.4.3 Brusselator: an oscillatory reaction network

The Brusselator (Nicolis and Prigogine, 1977; Prigogine, 1980) is a widely used model system for studying oscillatory reaction networks. It is a model system for autocatalytic reactions. Examples of autocatalytic reaction systems include the Belousov-Zhabotinsky reaction and autophosphorylation reactions ubiquitous in biology. Oscillatory reaction networks are also used to model biological rhythms, such as the circadian clock (Barkai and Leibler, 2000; Tomita et al., 2005; Nakajima et al., 2005; van Zon et al., 2007; Zwicker et al., 2010) and the glycolytic cycle (Prigogine, 1980; Chandra et al., 2011). The Brusselator reaction network involves two

1 Stochastic chemical kinetics

species and is given by



The corresponding RRE is

$$\begin{aligned}
 \frac{d\phi_1}{dt} &= k_1 - k_2\phi_1 - k_3\phi_1 + k_4\phi_1^2\phi_2 \\
 \frac{d\phi_2}{dt} &= k_3\phi_1 - k_4\phi_1^2\phi_2,
 \end{aligned} \tag{1.64}$$

where ϕ_1 and ϕ_2 are the concentrations of S_1 and S_2 , respectively. For simplicity, we set $k_2 = 1$ and $k_4 = 1$. Under these conditions, the fixed point of the deterministic RRE (Eq. 1.64) is given by $\phi_{1,ss} = k_1$ and $\phi_{2,ss} = k_3/k_1$. This fixed point is stable if $k_3 < k_1^2 + 1$, undergoes a Hopf bifurcation at $k_3 = k_1^2 + 1$ and is unstable if $k_3 \geq k_1^2 + 1$. In the latter case, $\phi_1(t)$ and $\phi_2(t)$ exhibit limit-cycle oscillations.

The CME of the reaction network is

$$\begin{aligned}
 \frac{\partial P(n_1, n_2, t)}{\partial t} &= k_1\Omega P(n_1 - 1, n_2, t) + (n_1 + 1)k_2P(n_1 + 1, n_2, t) \\
 &\quad + (n_1 + 1)k_3P(n_1 + 1, n_2 - 1, t) \\
 &\quad + (n_1 - 1)(n_1 - 2)(n_2 + 1)\frac{k_4}{\Omega^2}P(n_1 - 1, n_2 + 1, t) \\
 &\quad - \left[k_1\Omega + n_1k_2 + n_1k_3 + n_1(n_1 - 1)n_2\frac{k_4}{\Omega^2} \right] P(n_1, n_2, t),
 \end{aligned} \tag{1.65}$$

where n_1 and n_2 are the populations of species S_1 and S_2 , respectively. This CME cannot be solved analytically and hence we rely on SSA simulations.

For the simulation, we set $k_1 = 1$ and $k_3 = 1$. For these parameters the fixed point of the RRE is stable and the deterministic solution does not exhibit limit-cycle oscillations. We set the initial concentration of the system to its fixed point, i.e., the initial concentrations of both S_1 and S_2 are set to 1. As expected Fig. 1.5, shows that the deterministic RRE trajectory stays at the fixed point. As Ω is decreased, the SSA simulations display increasing fluctuations. Already at a volume of $\Omega = 1000$ the SSA trajectory shows oscillations, even though the deterministic RRE doesn't show any oscillatory behavior. This is evident from the closed-loop phase-space ($\phi_2^*(t)$ versus $\phi_1^*(t)$) trajectory shown in Fig. 1.6.

Figure 1.7 shows the bivariate steady-state probability function $P_{ss}(\phi_1^*, \phi_2^*)$ for $\Omega = 10$, $\Omega = 100$ and $\Omega = 1000$. As expected, the probability function broadens with decreasing Ω . The steady-state concentration variance of species S_1 scales with Ω^{-1} (see Fig. 1.10). When the fixed point is unstable ($k_1 = 1$, $k_3 = 4$), however, this scaling doesn't hold (See Fig. 1.10) and the variance decreases only slowly with increasing Ω (variance $\sim \Omega^{-\alpha}$ where $\alpha \ll 1$. $\alpha \approx 0.08$ in the present case).

The emergence of oscillations due to intrinsic noise can be understood as an interplay between the drift term and the diffusion term in the CLE (Eq. 1.36), which can make the particle undergo a circular motion analogous to that of the limit cycle in the deterministic case. As the magnitude of intrinsic noise decreases, the noisy diffusion term becomes weaker until, at very large Ω , the motion is dominated by the deterministic drift (Liouville process, see Eq. 1.45). This interplay between drift and noise have been studied in chemical reaction systems and it has been found to be the reason for oscillations in the stochastic trajectory (Qian et al., 2002; Baxendale and Greenwood, 2011).

In summary, intrinsic noise can lead to oscillations even when the corresponding RRE does not exhibit any. In Chapter 5, we will show that intrinsic noise can not only be a reason for the emergence of oscillations, but that it can also lead to a systematic modulation of the frequency spectrum of the oscillations. The scaling of the variance of the concentration with Ω , however, depends on the stability of the fixed point of the RRE. If the fixed point is stable, such that the RRE does not exhibit oscillations, the variance decrease as Ω^{-1} with increasing Ω . If the fixed point is unstable, such that the RRE exhibits limit-cycle oscillations, the variance decreases much slower with increasing Ω .

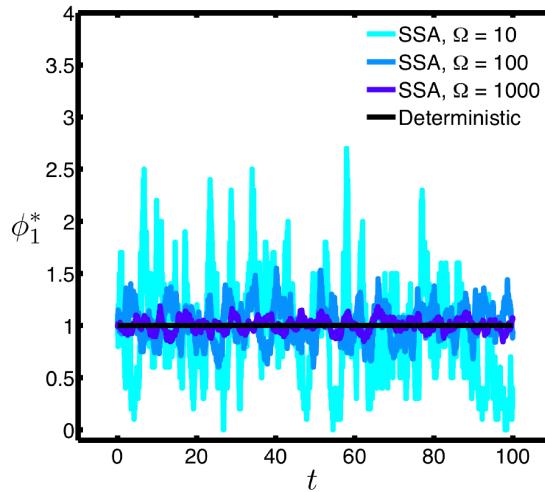


Figure 1.5: Time evolution of the concentration of species S_1 in the Brusselator reaction network of Eq. 1.63. The results are obtained using the stochastic simulation algorithm (SSA) for different reactor volumes Ω , and using the deterministic reaction rate equation (RRE). The rates used for the simulation are: $k_1 = 1$, $k_2 = 1$, $k_3 = 1$ and $k_4 = 1$. The initial concentrations of species S_1 and S_2 are set to 1, the fixed point of the RRE.

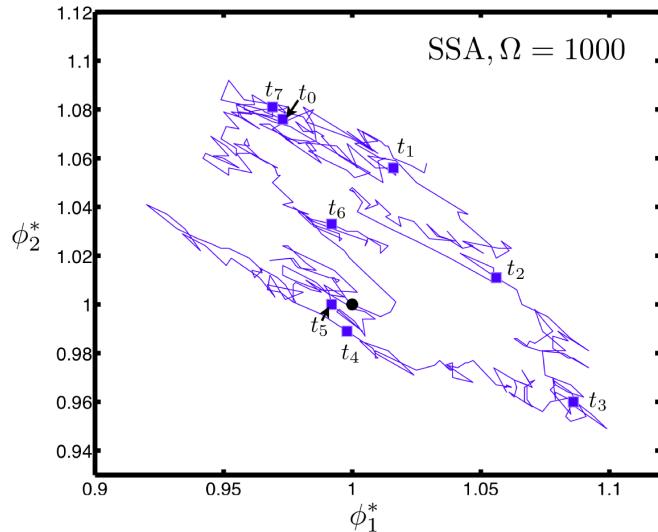


Figure 1.6: Steady-state concentration trajectory in the phase space of the Brusselator reaction network of Eq. 1.63, obtained using the stochastic simulation algorithm (SSA) with a reactor volume of $\Omega = 1000$. The rates used for the simulation are: $k_1 = 1$, $k_2 = 1$, $k_3 = 1$ and $k_4 = 1$. The initial concentrations of species S_1 and S_2 are set to 1, the fixed point of the RRE (circle). The squares mark the implicit time of the stochastic trajectory in phase space such that $t_{i+1} > t_i$. The time stamps help visualize a clockwise oscillatory behavior.

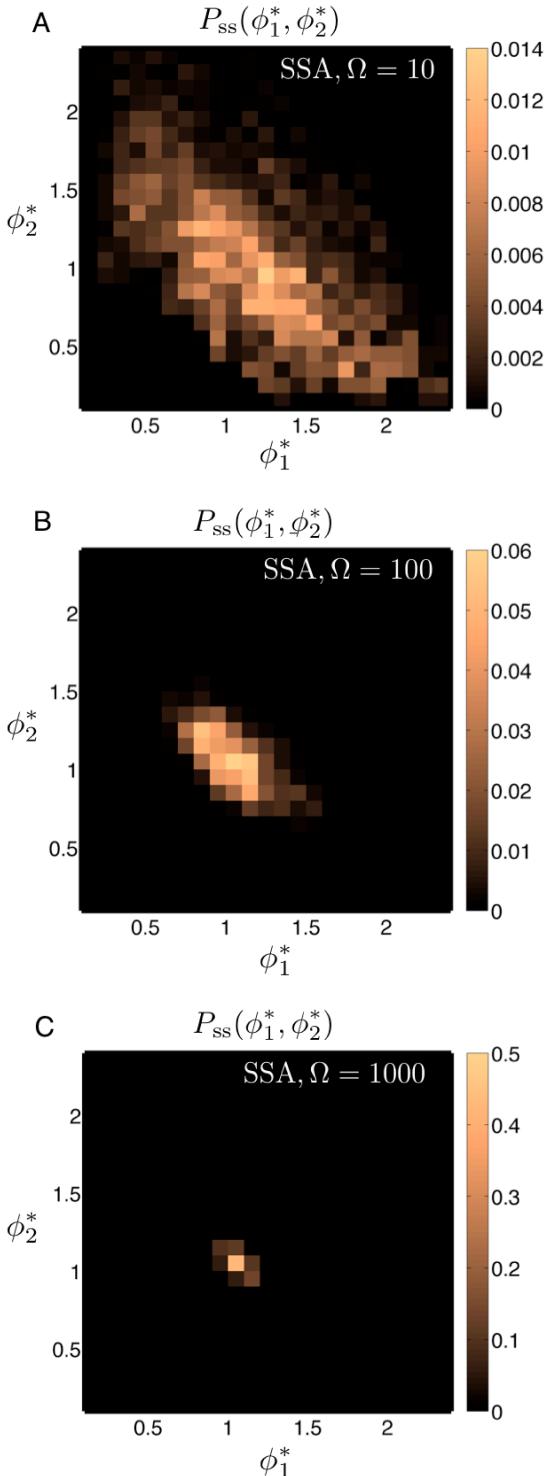


Figure 1.7: Steady-state bivariate probability function of the concentrations of species S_1 and S_2 in the Brusselator reaction network of Eq. 1.63. The results are obtained using the stochastic simulation algorithm (SSA) for different reactor volumes Ω . The rates used for the simulation are: $k_1 = 1$, $k_2 = 1$, $k_3 = 1$ and $k_4 = 1$. The initial concentration of species S_1 and S_2 is set to 1, the fixed point of the RRE.

1.4.4 Schlogl model: a bistable reaction network

The Schlogl model (Schlogl, 1972) is a popular model system for multistable reaction networks. Multistability has also been observed in several biological systems, including the bacterial phenotypic expression (Henderson et al., 1999), lactose utilization network (Ozbudak et al., 2004), the cell cycle regulation network (Pomerening et al., 2003) and in synthetic genetic switches (Gardner et al., 2000). The Schlogl model is given by



and it has the RRE

$$\frac{d\phi_1}{dt} = k_1\phi_1^2 - k_2\phi_1^3 + k_3 - k_4\phi_1, \quad (1.67)$$

where ϕ_1 is the concentration of species S_1 . The rates are set to $k_1 = 18$, $k_1 = 2.5$, $k_3 = 22$ and $k_4 = 37.5$. For this parameter set, the RRE has 3 fixed points $\phi_{1,ss} = 1$, $\phi_{1,ss} = 4$ and $\phi_{1,ss} = 2.2$. The first two are stable, the third is unstable.

The CME of this system is

$$\begin{aligned} \frac{\partial P(n_1, t)}{\partial t} = & (n_1 - 1)(n_1 - 2) \frac{k_1}{\Omega} \Omega P(n_1 - 1, t) + (n_1 + 1)n_1(n_1 - 1) \frac{k_2}{\Omega^2} P(n_1 + 1, t) \\ & + k_3 \Omega P(n_1 - 1, t) + (n_1 + 1)k_4 P(n_1 + 1, t) \\ & - \left[n_1(n_1 - 1) \frac{k_1}{\Omega} \right. \\ & \left. + n_1(n_1 - 1)(n_1 - 2) \frac{k_2}{\Omega^2} + k_3 \Omega + n_1 k_4 \right] P(n_1, t) \end{aligned} \quad (1.68)$$

and analytically intractable. We set the initial concentration of species S_1 to 1, which is one of the stable fixed points of the RRE.

Figure 1.8 shows that in the deterministic case the concentration of species S_1 remains unchanged from the initial concentration at the stable fixed point of the system. Using SSA, we see that for $\Omega = 1000$ the stochastic trajectory shows small fluctuations around the deterministic RRE (Fig. 1.8). This can also be seen from the steady-state probability function $P_{ss}(\phi_1^*)$ (Fig. 1.9C) which is a symmetric, narrow and unimodal distribution with its mean very close to the stable fixed point $\phi_{1,ss} = 1$ of the RRE. For $\Omega = 100$ the stochastic trajectories switch (jump) back and forth between the neighborhoods of the two stable fixed points (Fig. 1.8). This can also be seen in the steady-state probability function $P_{ss}(\phi_1^*)$ (Fig. 1.9B) becoming bimodal with maxima close to the two stable fixed points ($\phi_{1,ss} = 1$, $\phi_{1,ss} = 4$) of the RRE, and a minimum close to the unstable fixed point $\phi_{1,ss} = 2.2$ of the RRE. At $\Omega = 10$, the fluctuations of the stochastic trajectory become even larger (Fig. 1.8). The corresponding steady-state probability function $P_{ss}(\phi_1^*)$ (Fig. 1.9A) is again unimodal, but asymmetric with a long tail. Because of these changes in the qualitative properties of the steady-state probability function,

the scaling of the steady-state concentration variance falls in two regimes (Fig. 1.10). At low intrinsic noise or large Ω when $P_{ss}(\phi_1^*)$ is symmetric, the variance scales with Ω^{-1} . At large intrinsic noise or small Ω when $P_{ss}(\phi_1^*)$ is asymmetric or bimodal, the variance scales with $\Omega^{-\alpha}$ where $\alpha \ll 1$. In the present case, we estimated this exponent α to be approximately 0.12. In addition, the large Ω regime is $\Omega \geq 500$ and small Ω regime is $\Omega \leq 400$ in the present case. The sudden jump observed in the scaling of the variance from the large Ω regime to the small Ω regime is brought about by the change in $P_{ss}(\phi_1^*)$ from being a sharp unimodal distribution at $\Omega = 500$ to a bimodal distribution at $\Omega = 400$. This change in $P_{ss}(\phi_1^*)$ is effected by the fluctuations becoming large enough that the concentration reaches the neighborhood of the unstable fixed point, from where the concentration is then probabilistically shuttled to the neighborhood of the other stable fixed point.

In summary, intrinsic noise in a multistable reaction network can lead to switching behavior between the neighborhoods of the multiple stable fixed points. Due to the associated qualitative change in the steady-state concentration probability function, the variance scales as $\Omega^{-\alpha}$ with $\alpha \ll 1$ for small Ω . For large enough Ω the variance scales with Ω^{-1} .

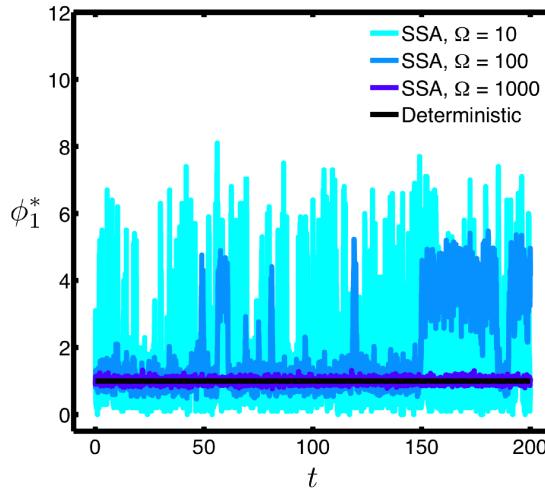


Figure 1.8: Time evolution of the concentration of species S_1 in the Schlogl model of Eq. 1.66. The results are obtained using the stochastic simulation algorithm (SSA) for different reactor volumes Ω , and using the deterministic reaction rate equation (RRE). The rates used for the simulation are: $k_1 = 18$, $k_2 = 2.5$, $k_3 = 22$ and $k_4 = 37.5$. The initial concentration of species S_1 is set to 1, which is one of the stable fixed points of the RRE.

1 Stochastic chemical kinetics

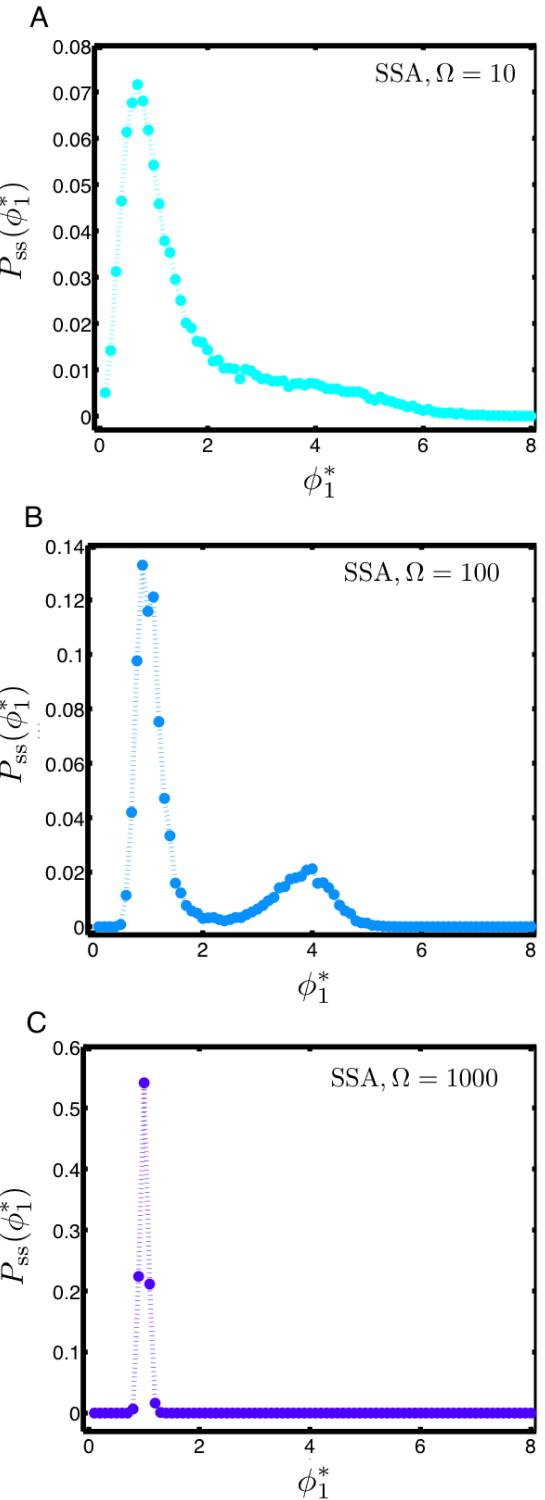


Figure 1.9: Steady-state probability function for the concentration of species S_1 in the Schlogl model of Eq. 1.66. The results are obtained using the stochastic simulation algorithm (SSA) for different reactor volumes Ω , and using the deterministic reaction rate equation (RRE). The rates used for the simulation are: $k_1 = 18$, $k_2 = 2.5$, $k_3 = 22$ and $k_4 = 37.5$. The initial concentration of species S_1 is set to 1, one of the stable fixed points of the RRE.

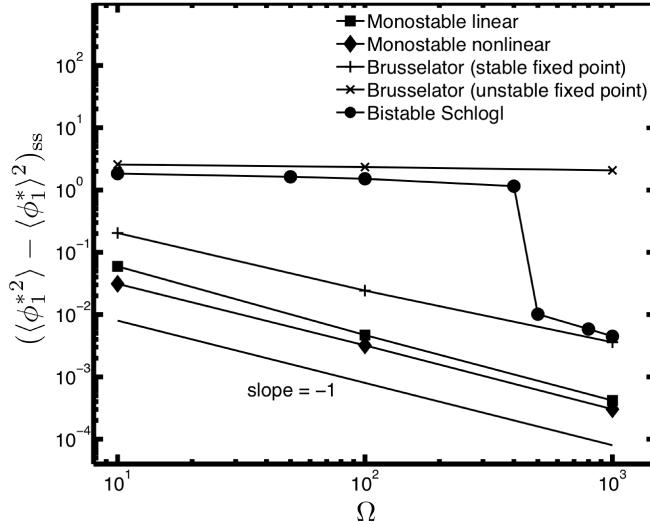


Figure 1.10: Scaling of the steady-state concentration variance of species S_1 with the reactor volume Ω for four different types of reaction networks: monostable linear (Sec. 1.4.1), monostable nonlinear (Sec. 1.4.2), non-oscillatory Brusselator, oscillatory Brusselator (Sec. 1.4.3) and the Bistable Schlogl (Sec. 1.4.4) models. All results are obtained using the stochastic simulation algorithm (SSA). The parameters used for the simulations are given in the corresponding sections that introduce these reaction networks.

1.5 Summary

We revisited the chemical master equation (CME) that describes stochastic chemical kinetics as a discrete-state jump Markov process taking into account intrinsic noise due to the discreteness of the reactant molecules. Assuming that the population of a species increases linearly with the reactor volume Ω we showed that when the concentration vector ϕ can be considered as a continuous random variable, the discrete-state jump Markov process becomes a continuous-state jump Markov process described by the chemical Kramer-Moyal equation. At even larger Ω , the continuous state jump Markov process becomes a continuous-state Markov process yielding the nonlinear chemical Fokker-Planck equation, whose equation of motion is given by the chemical Langevin equation (CLE). Subsequently, at very large Ω the continuous-state Markov process becomes a Liouville process, leading to the classical reaction rate equation (RRE).

We illustrated the effect on intrinsic noise on different types of chemical reaction networks and showed the potential emergence of non-trivial chemical kinetics.

2

Formulations of the stochastic simulation algorithm for chemical reaction networks

“God made the integers; all else is the work of man.”
– Leopold Kronecker

Gillespie’s stochastic simulation algorithm (SSA) presented in Chapter 1 (see Sec. 1.3) defines a kinetic Monte Carlo scheme to simulate stochastic chemical kinetics by sampling population trajectories from the exact solution of the chemical master equation (CME). There are several implementations or formulations of SSA with varying computational cost. This variation comes from the topological properties of the chemical reaction networks and the algorithms used in the SSA formulations. In this chapter, we will present the topological properties of chemical reaction networks that dictate the computational cost of SSA formulations in Sec. 2.1. In Sec. 2.1.3 we classify chemical reaction networks into two classes—*strongly coupled* and *weakly coupled*—based on topological properties derived from a dependency graph representation (Gibson and Bruck, 2000) of the reaction network. In Sec. 2.3 we present different exact SSA formulations and analyze their computational cost with regard to the coupling class of the reaction network. In Sec. 2.4 we review the general concept of approximate SSAs that improve on the computational performance of exact SSAs by sampling from an approximated probability function.

2.1 Topology of chemical reaction networks

Consider the general chemical reaction network described in Chapter 1 with N species and M reactions:

$$\sum_{i=1}^N \nu_{i,\mu}^- S_i \longrightarrow \sum_{i=1}^N \nu_{i,\mu}^+ S_i, \quad \mu = 1, \dots, M. \quad (2.1)$$

The reactant stoichiometry matrix is $\boldsymbol{\nu}^-$ and the product stoichiometry matrix is $\boldsymbol{\nu}^+$. The overall stoichiometry matrix is

$$\boldsymbol{\nu} = \boldsymbol{\nu}^+ - \boldsymbol{\nu}^-. \quad (2.2)$$

All the stoichiometry matrices are of size $N \times M$. All elements of $\boldsymbol{\nu}^+$ and $\boldsymbol{\nu}^-$ are non-negative whereas those of $\boldsymbol{\nu}$ can be negative, zero or positive. Chemical kinetics in the deterministic approximation (see Eq. 1.46) is described by the reaction rate equation (RRE) given by

$$\frac{d\phi}{dt} = \boldsymbol{\nu} \mathbf{T}_0(\phi), \quad (2.3)$$

where $\mathbf{T}_0(\phi)$ is the *flux vector* of all the M reactions and ϕ the concentration vector.

Some of the physical properties of the chemical reaction network can be defined by the following subspaces of $\boldsymbol{\nu}$:

1. **The null space of $\boldsymbol{\nu}$** is defined as the space of all flux vectors \mathbf{f} for which

$$\boldsymbol{\nu} \mathbf{f} = \mathbf{0}. \quad (2.4)$$

This space defines the relationships between the steady-state fluxes of the individual reactions in the network, since Eq. 2.4 implies $\frac{d\phi}{dt} = \mathbf{0}$ (see Eq. 2.3).

2. **The left null space of $\boldsymbol{\nu}$** is defined as the space of all vectors \mathbf{l} for which

$$\boldsymbol{\nu}^T \mathbf{l} = \mathbf{0}. \quad (2.5)$$

To understand the physical meaning of these vectors \mathbf{l} , we multiply Eq. 2.3 on both sides with \mathbf{l}^T :

$$\mathbf{l}^T \frac{d\phi}{dt} = \mathbf{l}^T \boldsymbol{\nu} \mathbf{T}_0(\phi), \quad (2.6)$$

i.e.,

$$\frac{d\phi^T \mathbf{l}}{dt} = \mathbf{T}_0(\phi)^T \boldsymbol{\nu}^T \mathbf{l}. \quad (2.7)$$

Now, Eq. 2.5 implies that

$$\phi^T \mathbf{l} = \text{constant}. \quad (2.8)$$

The left null space thus defines the conservation relations among the species in the reaction network.

2.1.1 Representation of chemical reaction networks

The stoichiometry matrix can also be used to represent the topology of the chemical reaction network by constructing a *dependency graph* (Gibson and Bruck, 2000). The dependency graph of a reaction network is a directed graph with nodes representing reactions and directed edges indicating the couplings between the reactions of the network. A directed edge is drawn from node p to node q if any of the reactants or products of reaction p are involved as a reactant in reaction q (Fig. 2.1). More formally, a directed edge is drawn from node p to

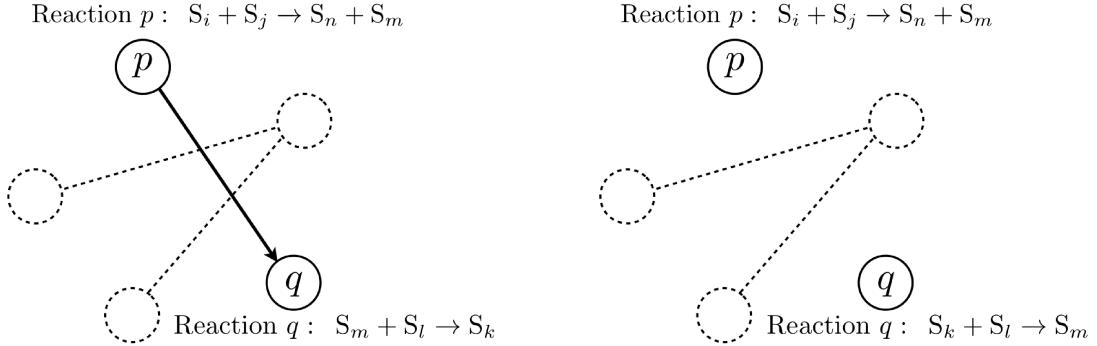


Figure 2.1: Illustration of the dependency graph representation of a chemical reaction network. The nodes in a dependency graph represent reactions. The directed edges (arrows) between the nodes represent the dependencies or the couplings between the reactions in the network. In the left panel, a directed edge is drawn from node p to node q because species S_m is a product of reaction p and a reactant in reaction q . In the right panel, no edge is drawn from node p to node q since none of reactants or products of reaction p are involved as a reactant in reaction q . In both panels, no directed edge is drawn from node q to node p .

node q if the vector obtained by performing an elementwise logical-and operation between the binarized overall stoichiometry vector $\hat{\nu}_p$ and the binarized reactant stoichiometry vector $\hat{\nu}_q^-$ contains non-zero elements. Binarization is defined as:

$$\begin{aligned}\hat{\nu}_{i,p} &= 0 \text{ if } \nu_{i,p} = 0 \\ \hat{\nu}_{i,p} &= 1 \text{ if } \nu_{i,p} \neq 0\end{aligned}\tag{2.9}$$

and

$$\begin{aligned}\hat{\nu}_{i,q}^- &= 0 \text{ if } \nu_{i,q}^- = 0 \\ \hat{\nu}_{i,q}^- &= 1 \text{ if } \nu_{i,q}^- > 0.\end{aligned}\tag{2.10}$$

2.1.2 Degree of coupling of a reaction network

The out-degree d_μ of a node μ in a directed graph is the number of directed edges leaving that node. In a dependency graph of a chemical reaction network, it indicates the number of reaction propensities that change upon firing of the reaction represented by the node. We define

2 Formulations of the stochastic simulation algorithm for chemical reaction networks

the *degree of coupling* d_c of a reaction network as the maximum out-degree of its dependency graph, i.e.,

$$d_c = \max\{d_1, \dots, d_M\}. \quad (2.11)$$

The degree of coupling d_c is equal to the maximum number of propensities that change upon any reaction firing event. The out-degree d_μ of reaction μ in the general chemical reaction system (Eq. 2.1) is

$$d_\mu = \sum_{\mu'=1}^M \frac{\sum_{i=1}^N (\hat{\nu}_{i,\mu} \wedge \hat{\nu}_{i,\mu'}^-)}{\delta\left(\sum_{i=1}^N (\hat{\nu}_{i,\mu} \wedge \hat{\nu}_{i,\mu'}^-)\right) + \sum_{i=1}^N (\hat{\nu}_{i,\mu} \wedge \hat{\nu}_{i,\mu'}^-)} \quad (2.12)$$

where $\delta(\cdot)$ is the unit impulse function.

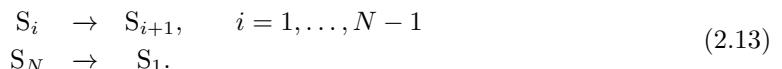
2.1.3 Classification of reaction networks

Based on their *degree of coupling*, we classify reaction networks into *weakly coupled* and *strongly coupled* networks. Weakly coupled networks have a degree of coupling d_c that is bounded by a constant with increasing network size. Formally, in weakly coupled networks d_c is $O(1)$. Strongly coupled networks are those where d_c increases proportionally with network size, i.e., d_c is $O(M)$, where M is the number of reactions in the network.

2.1.4 Examples

2.1.4.1 Cyclic chain model

Consider the following cyclic chain reaction network with N species and $M = N$ reactions:



For $N = 3$ the reaction network is



The stoichiometry matrices for this reaction network are:

$$\boldsymbol{\nu}^- = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.15)$$

$$\boldsymbol{\nu}^+ = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad (2.16)$$

and

$$\boldsymbol{\nu} = \boldsymbol{\nu}^+ - \boldsymbol{\nu}^- = \begin{bmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{bmatrix}. \quad (2.17)$$

The null space \mathbf{f} of $\boldsymbol{\nu}$ is

$$\mathbf{f} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \quad (2.18)$$

Therefore, the steady-state flux vector can be written as

$$\mathbf{T}_0(\phi_{ss}) = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \alpha_1, \quad (2.19)$$

i.e., the steady-state fluxes of all reactions are equal multiples of a positive constant α_1 .

The left null space \mathbf{l} of $\boldsymbol{\nu}$ is

$$\mathbf{l} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad (2.20)$$

which implies:

$$\phi_1 + \phi_2 + \phi_3 = \text{constant}. \quad (2.21)$$

This is the conservation relation enforced by the reactions in Eq. 2.14 at all times (in stationary and non-stationary state). For very large reaction networks, this relation may not be obvious and the left null space useful.

The binarized stoichiometry matrices are

$$\hat{\boldsymbol{\nu}}^- = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.22)$$

$$\hat{\boldsymbol{\nu}}^+ = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.23)$$

and

$$\hat{\boldsymbol{\nu}} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}. \quad (2.24)$$

2 Formulations of the stochastic simulation algorithm for chemical reaction networks

The dependency graph of this reaction network is shown in Fig. 2.2A. A directed edge is drawn from node 1 to node 2 since species S_2 is a product of reaction 1 and a reactant in reaction 2. The drawing of this directed edge can also be decided by performing the elementwise logical-and operation between $\hat{\nu}_1$ and $\hat{\nu}_2^-$:

$$\hat{\nu}_1 \wedge \hat{\nu}_2^- = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \wedge \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \quad (2.25)$$

Since at least one of the elements in the resulting vector is non-zero, we draw a directed edge from node 1 to node 2. In addition, a directed edge is drawn from node 1 to itself, due to the presence of species 1. These two directed edges are the only edges originating from node 1. In a similar manner, directed edges are drawn from nodes 2 and 3 to complete the dependency graph as shown in Fig. 2.2A.

The degree of coupling of this network can be computed from Eqs. 2.11 and 2.12: Visual inspection of the dependency graph in Fig. 2.2A shows that the out-degree $d_\mu = 2$ for every μ . Hence, the degree of the coupling is $d_c = 2$.

This is the degree of coupling for the cyclic chain model with $N = 3$ species. The dependency graph for the same model with $N = 4$ species is shown in Fig. 2.2B, and we observe that also here $d_c = 2$. In general, for the cyclic chain model with N species (Eq. 2.13), $d_c = 2$ independent of the size of the network. Therefore, the cyclic chain model is an example of a *weakly coupled* reaction network.

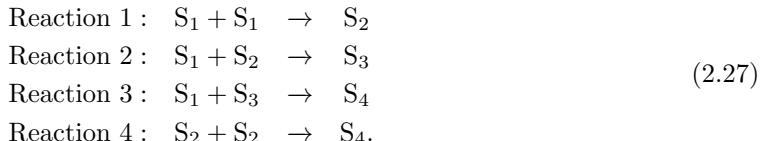
2.1.4.2 Colloidal aggregation model

Consider the following colloidal aggregation reaction network with N species and $M = \left\lfloor \frac{N^2}{4} \right\rfloor$ reactions:



Species S_i can be considered a multimer consisting of i monomers.

For $N = 4$ the reaction network is



The stoichiometry matrices for this reaction network are:

$$\nu^- = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (2.28)$$

$$\boldsymbol{\nu}^+ = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad (2.29)$$

and

$$\boldsymbol{\nu} = \boldsymbol{\nu}^+ - \boldsymbol{\nu}^- = \begin{bmatrix} -2 & -1 & -1 & 0 \\ 1 & -1 & 0 & -2 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \quad (2.30)$$

The null space \mathbf{f} of $\boldsymbol{\nu}$ is

$$\mathbf{f} = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix}. \quad (2.31)$$

Therefore, the steady-state flux vector can be written as

$$\mathbf{T}_0(\phi_{ss}) = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 0 \end{bmatrix} \alpha_1. \quad (2.32)$$

Again, all steady-state fluxes are multiples of a positive constant α_1 . Since α_1 is non-negative, the above relationship indicates that the steady-state fluxes of reactions 2 and 3 in Eq. 2.27 are negative. Since fluxes are always non-negative, this imposes that $\alpha_1 = 0$. Consequently, the above flux relation imposes that all steady-state fluxes in the reaction network of Eq. 2.27 are zero. This can be verified by simulating the RRE with any initial condition and rates. The reasoning can also be as follows: Consider that the initial state of the system is given by 10 molecules of species S_1 and 0 molecules of all other species. Depending on the reaction rates (assuming that they are all non-zero), the steady-state population would be one of two possibilities: 1 molecule of S_2 and 2 molecules of S_4 , or 2 molecules of S_3 and 1 molecule of S_4 . Both of these population vectors would render all fluxes zero as there would not be enough reactant molecules for any reaction to have a non-zero flux (see Eq. 1.41 in Sec. 1.2.2 for the definition of reaction fluxes $\mathbf{T}_0(\phi)$).

The left null space \mathbf{l} of $\boldsymbol{\nu}$ is

$$\mathbf{l} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \quad (2.33)$$

2 Formulations of the stochastic simulation algorithm for chemical reaction networks

implying that

$$\phi_1 + 2\phi_2 + 3\phi_3 + 4\phi_4 = \text{constant}. \quad (2.34)$$

This is the conservation relation enforced by the reactions in Eq. 2.27 at all times (in stationary and non-stationary state). This means that the total number of bound and free monomers in the system is constant.

The binarized stoichiometry matrices are

$$\hat{\nu}^- = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (2.35)$$

$$\hat{\nu}^+ = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad (2.36)$$

and

$$\hat{\nu} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \quad (2.37)$$

The dependency graph of this reaction network is shown in Fig. 2.2C. The degree of coupling can be computed from Eqs. 2.11 and 2.12. Visual inspection of the dependency graph in Fig. 2.2C shows that the out-degrees of the nodes are $d_1 = 4$, $d_2 = 4$, $d_3 = 3$ and $d_4 = 3$. Hence, the degree of coupling is $d_c = 4$ (Eq. 2.12).

Classification: This is the degree of coupling for the colloidal aggregation model with $N = 4$ species. The dependency graph for the same model with $N = 5$ species is shown in Fig. 2.2D, and we observe that then $d_c = 6$. In general, for the aggregation model with N species (Eq. 2.26), $d_c = 2N - 4$. Since d_c increases with network size, $d_c \in O(N) \subseteq O(M)$. Therefore, the colloidal aggregation model is a *strongly coupled* reaction network.

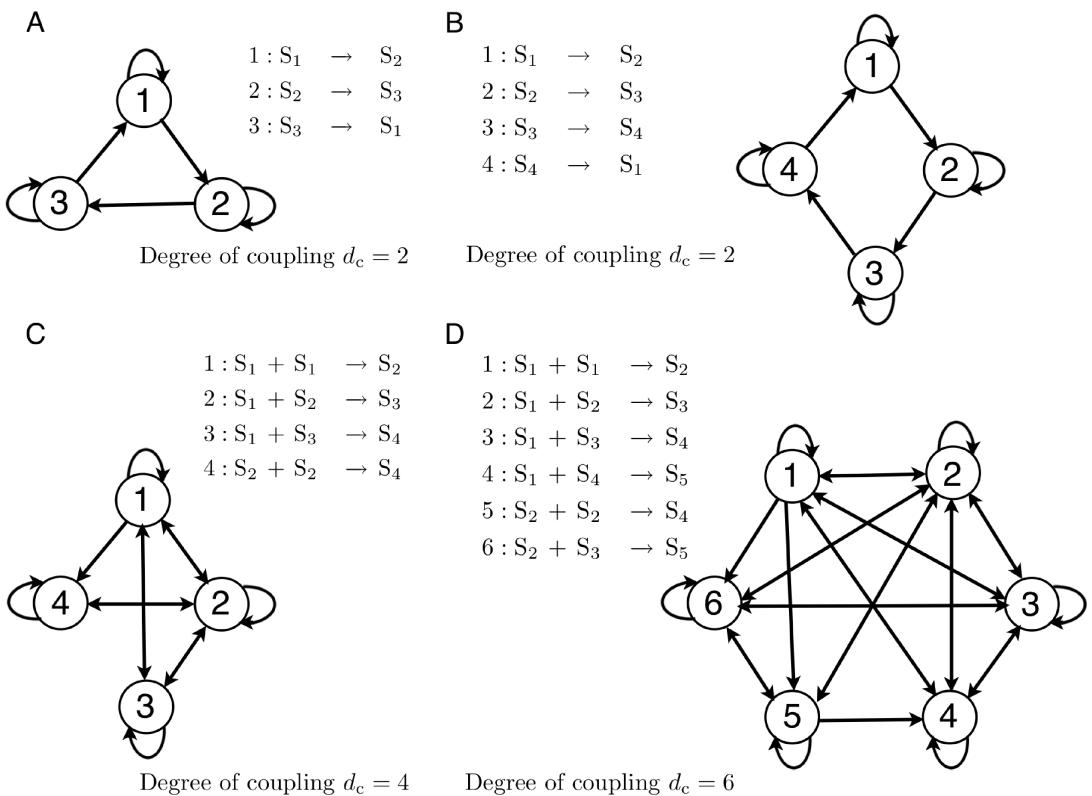


Figure 2.2: Dependency graph representations of four example reaction networks. (A) and (B) show the dependency graph of the cyclic chain model (Eq. 2.13) with $N = 3$ and 4 species, respectively. (C) and (D) show the dependency graph of the colloidal aggregation model (Eq. 2.26) for $N = 4$ and 5, respectively.

2.2 SSA formulations

All formulations of Gillespie's SSA aim to simulate chemical kinetics by sampling the random variables τ (time to the next reaction) and μ (index of the next reaction) according to Eq. 1.49 and propagating the state of the chemical system in time. Since SSA is a Monte Carlo scheme, several independent runs of each simulation need to be performed in order to estimate the state (population) probability function $P(\mathbf{n}, t)$.

Existing SSA formulations can be classified into *exact* and *approximate* methods. Exact methods sample from the joint probability distribution in Eq. 1.49. These formulations include the first reaction method (FRM) (Gillespie, 1976), the direct method (DM) (Gillespie, 1977), Gibson-Bruck's next-reaction method (NRM) (Gibson and Bruck, 2000), a Gibson-Bruck variant of the DM (Gibson and Bruck, 2000), the optimized direct method (ODM) (Cao et al., 2004), the sorting direct method (SDM) (McCollum et al., 2006), the logarithmic direct

method (LDM) (Li and Petzold, 2006) and the composition-rejection formulation (SSA-CR) (Slepoy et al., 2008). Approximate SSA formulations provide better computational efficiency for larger numbers of molecules by sampling from an approximation to the joint probability distribution in Eq. 1.49. These methods include τ -leaping (Gillespie, 2001; Cao et al., 2005a, 2006; Peng et al., 2007), k_α -leaping (Gillespie, 2001), R -leaping (Auger et al., 2006), L -leap (Peng and Wang, 2007), K -leap (Cai and Xu, 2007), the slow-scale method (Cao et al., 2005b), and implicit τ -leaping (Rathinam et al., 2003).

In the following, we focus on *exact* SSA formulations, but briefly also mention the fundamental idea, benefits and limitations of *approximate* SSAs towards the end of this chapter.

2.3 Exact formulations of the SSA and their computational cost

Exact SSA formulations sample μ and τ from Eq. 1.49, and only one reaction μ is executed per time step. The time step size τ is itself a random variable. The population state of the chemical system is subsequently updated. The fundamental steps in every exact SSA formulation are thus:

1. Sample τ and μ from Eq. 1.49.
2. Update population \mathbf{n} .
3. Recompute the reaction propensities a_μ .

Various algorithms have been developed to perform these steps, and they constitute different SSA formulations with different computational costs. We quantify computational cost by the CPU time needed to execute one reaction event. The computational cost of any SSA formulation depends on the coupling class of the simulated reaction network. We provide here a review of various exact SSA formulations and analyze their computational costs as a function of the coupling class of the reaction network.

2.3.1 The first reaction method (FRM)

FRM (Gillespie, 1976) is one of the earliest exact SSA formulations. In this formulation, the time τ_μ when reaction μ fires next is computed according to the probability function

$$p(\tau_\mu \mid \mathbf{n}(t)) = a_\mu e^{-a_\mu \tau_\mu}. \quad (2.38)$$

Subsequently, the next reaction μ is chosen to be the one with the minimum τ_μ , and the time τ to the next reaction is set to the minimum τ_μ . The algorithm is given in Table 2.1.

The computational cost of FRM is $O(M)$ (Gibson and Bruck, 2000; Cao et al., 2004; Ramaswamy et al., 2009; Ramaswamy and Sbalzarini, 2010b) where M is the number of reactions in the network. This is due to steps 2 and 4 (see Table 2.1), both of which have a runtime of $O(M)$: step 2 involves generating M random numbers and step 4 involves recomputing all M reaction propensities.

-
1. Set $t \leftarrow 0$; initialize \mathbf{n} , $a_\mu \forall \mu$, and a
 2. Sample τ_μ according to Eq. 2.38 for each reaction μ : For each reaction generate a uniform random number $r_1 \in [0, 1)$ and compute $\tau_\mu \leftarrow -a_\mu^{-1} \log(r_1)$. $\tau \leftarrow \min\{\tau_1, \dots, \tau_M\}$
 3. $\mu \leftarrow$ the index of $\min\{\tau_1, \dots, \tau_M\}$
 4. Update: $\mathbf{n} \leftarrow \mathbf{n} + \boldsymbol{\nu}_\mu$, where $\boldsymbol{\nu}_\mu$ is the stoichiometry of reaction μ ; recompute all a_μ and a
 5. $t \leftarrow t + \tau$; go to step 2
-

Table 2.1: Algorithm for the first reaction method (FRM)

2.3.2 The direct method (DM)

DM (Gillespie, 1977) samples the next reaction index μ according to Eq. 1.51 using linear search over the reaction propensities. The time τ to the next reaction is sampled according to Eq. 1.50. The algorithm is given in Table 2.2.

-
1. Set $t \leftarrow 0$; initialize \mathbf{n} , $a_\mu \forall \mu$, and a
 2. Sample μ using linear search according to Eq. 1.51: generate a uniform random number $r_1 \in [0, 1)$ and determine μ as the smallest integer satisfying $r_1 < \sum_{\mu'=1}^{\mu} a_{\mu'}/a$
 3. Sample τ according to Eq. 1.50: generate a uniform random number $r_2 \in [0, 1)$ and compute τ as $\tau \leftarrow -a^{-1} \log(r_2)$
 4. Update: $\mathbf{n} \leftarrow \mathbf{n} + \boldsymbol{\nu}_\mu$, where $\boldsymbol{\nu}_\mu$ is the stoichiometry of reaction μ ; recompute all a_μ and a
 5. $t \leftarrow t + \tau$; go to step 2
-

Table 2.2: Algorithm for the direct method (DM)

The computational cost of DM is also $O(M)$ (Gibson and Bruck, 2000; Cao et al., 2004; Ramaswamy et al., 2009; Ramaswamy and Sbalzarini, 2010b). This is due to steps 2 and 4 in the algorithm (see Table 2.2), both of which have a worst-case runtime of $O(M)$. In terms of absolute runtimes, however, DM is more efficient than FRM since it does not involve the expensive step of generating M random numbers for each reaction event.

2.3.3 Next reaction method (NRM)

NRM (Gibson and Bruck, 2000) is an improvement over FRM in which the $M - 1$ unused reaction times are suitably reused, and efficient data structures such as indexed minimum priority queues (Gibson and Bruck, 2000) and dependency graphs are introduced. The indexed priority queue is used to efficiently find the minimum among all τ_μ ; the dependency graph is a data structure that stores for each reaction the indices of the propensities that have to be recomputed upon firing of this reaction (see Sec. 2.1.1). This avoids having to recompute all a_μ after every reaction event. The formulation of NRM is in terms of the absolute time to the next reaction τ^g instead of the relative time to the next reaction τ that is used in DM and FRM. The algorithm of NRM is given in Table 2.3.

1. Set $t \leftarrow 0$; initialize \mathbf{n} , the dependency graph, $a_\mu \forall \mu$, and a
 2. For each reaction μ generate a uniform random number $r_1 \in [0, 1]$ and compute as $\tau_\mu^g \leftarrow -a_\mu^{-1} \log(r_1)$. Store these τ_μ^g 's in a minimum priority queue, indexed by the reaction label
 3. The reaction μ with the minimum τ_μ^g can be found at the top of the priority queue. Set $\tau^g \leftarrow \tau_\mu^g$
 4. Update: $\mathbf{n} \leftarrow \mathbf{n} + \boldsymbol{\nu}_\mu$, where $\boldsymbol{\nu}_\mu$ is the stoichiometry of reaction μ
 5. Set $t \leftarrow \tau^g$
 6. For all reactions i whose propensities change upon firing of reaction μ (i.e., the destination nodes of all directed edges leaving node μ in the dependency graph):
 - 6.1. Recompute the propensity a_i . Store the old propensity in a_i^{old}
 - 6.2. if $i \neq \mu$, set $\tau_i^g \leftarrow t + \frac{a_i^{\text{old}}}{a_i}(\tau_i^g - t)$
 - 6.3. if $i = \mu$, set $\tau_i^g \leftarrow t - a_i^{-1} \log(r_2)$, where r_2 is a uniform random number in $[0, 1]$
 - 6.4. Update the priority queue with the new value of τ_i^g for reaction i
 7. Go to step 3
-

Table 2.3: Algorithm for the next reaction method (NRM)

The data structures in NRM, together with the reuse of reaction times, reduce the computational cost to $O(d_c \log_2 M)$, where d_c is the degree of coupling of the reaction network. This scaling of the computational cost is due to step 6 in Table 2.3. In this step, a maximum of d_c reaction times need to be recomputed. Additionally, every change involves ensuring that the tree structure of minimum priority queue has nodes that always carry smaller reaction times than their children. The computational cost of this update operation is $O(\log_2 M)$ and hence the overall cost of step 6 is $O(d_c \log_2 M)$. For strongly coupled networks, d_c is a function of M and is $O(M)$. The computational cost of NRM is thus $O(M \log_2 M)$ for strongly coupled

2.3 Exact formulations of the SSA and their computational cost

networks. Even for some weakly coupled networks for which $d_c \ll M$ and is $O(1)$, the computational cost of NRM has been empirically shown to be $O(M)$ (Cao et al., 2004). This is due to the additional overhead, memory-access operations, and cache misses introduced by the complex data structures (indexed priority queue, dependency graph) of NRM. The absolute runtime of NRM is, however, still superior to that of FRM and DM.

2.3.4 Optimized direct method (ODM)

ODM (Cao et al., 2004) is an improvement over DM where the reactions are approximately sorted in descending order of firing frequency. Like DM, ODM uses linear search to sample the index of the next reaction and the reordering makes it more probable to find the next reaction close to the beginning of the list, reducing the average search depth. ODM estimates the firing frequencies of all reactions during a short pre-simulation run of about 5–10% of the length of the entire simulation (Cao et al., 2004; McCollum et al., 2006). In order to reduce the cost of updating the propensities after a reaction has fired, ODM also uses a dependency graph. The algorithm of ODM is similar to that of DM (see Table 2.2) except that the propensities of the reactions are stored in the descending order of their estimated firing frequency. Irrespective of the degree of coupling of the network, the computational cost of ODM is $O(M)$, which was also empirically confirmed in benchmarks (Cao et al., 2004). The absolute runtimes of ODM, however, are smaller than those of DM and NRM (Cao et al., 2004), especially for multiscale (stiff) reaction networks whose propensities span several orders of magnitude. In stiff networks, few reactions fire much more frequently than all others, and having the frequent reactions at the top of the list greatly reduces the average search depth in the linear search.

2.3.5 Sorting direct method (SDM)

SDM (McCollum et al., 2006) is a variant of ODM that does not use pre-simulation runs, but dynamically shifts up a reaction in the reaction list whenever it fires (“bubbling up” of the more frequent reactions). Like ODM, SDM is also especially efficient for multiscale reaction networks. Among multiscale reaction networks, the strategy of dynamically sorting the reactions is especially suited to deal with temporal changes in firing frequency that ODM fails to capture. The algorithm of SDM is given in Table 2.4.

The strategy of dynamically sorting the reactions reduces the prefactor of the computational cost of SDM compared to that of ODM, but the scaling remains $O(M)$ (McCollum et al., 2006), irrespective of the degree of coupling of the network (Ramaswamy et al., 2009; Ramaswamy and Sbalzarini, 2010b).

1. Set $t \leftarrow 0$; initialize \mathbf{n} , $a_\mu \forall \mu$, a , the change to the total propensity $\Delta a \leftarrow 0$, the dependency graph and ordering list \mathbf{l} such that $l_j = j$ for $j = 1, \dots, M$.
 2. Sample μ using linear search according to Eq. 1.51: generate a uniform random number $r_1 \in [0, 1)$ and determine j as the smallest integer satisfying $r_1 < \sum_{j'=1}^j a_{l_{j'}}/a$. Set $\mu \leftarrow j$
 3. If $j \neq 1$, swap l_j and l_{j-1}
 4. Sample τ according to Eq. 1.50: generate a uniform random number $r_2 \in [0, 1)$ and compute τ as $\tau \leftarrow -a^{-1} \log(r_2)$
 5. Update: $\mathbf{n} \leftarrow \mathbf{n} + \boldsymbol{\nu}_\mu$, where $\boldsymbol{\nu}_\mu$ is the stoichiometry of reaction μ
 6. For all reactions i whose propensities change upon firing of reaction μ (i.e., the destination nodes of all directed edges leaving node μ in the dependency graph):
 - 6.1. Update $\Delta a \leftarrow \Delta a - a_i$
 - 6.2. Recompute the propensity a_i
 - 6.3. Update $\Delta a \leftarrow \Delta a + a_i$
 7. Set $a \leftarrow a + \Delta a$
 8. $t \leftarrow t + \tau$ and set $\Delta a \leftarrow 0$; go to step 2
-

Table 2.4: Algorithm for the sorting direct method (SDM)

2.3.6 Logarithmic direct method (LDM)

LDM (Li and Petzold, 2006) uses a binary search tree (recursive bisection) on an ordered linear list of cumulative sums of propensities $s_j = \sum_{\mu'=1}^j a_{\mu'}$, $j = 1, \dots, M$ and $s_0 = 0$, to find the next reaction μ . The algorithm for LDM is given in Table 2.5.

The binary search tree to sample the next reaction index reduces the search depth to $O(\log_2 M)$. Irrespective of the degree of coupling, however, the update step (Step 6 in Table 2.5) is $O(M)$ since on average $(M + 1)/2$ and in the worst case M cumulative sums of propensities need to be recomputed, rendering the computational cost of LDM $O(M)$.

-
1. Set $t \leftarrow 0$; initialize \mathbf{n} , $a_\mu \forall \mu$, the dependency graph and the partial sums s_j
 2. Sample μ using recursive bisection search according to Eq. 1.51: generate a uniform random number $r_1 \in [0, 1)$ and perform binary search until $s_{\mu-1} \leq s_M r_1 = ar_1 < s_\mu$ where $a = s_M$ is the total propensity of all reactions
 3. Sample τ according to Eq. 1.50: generate a uniform random number $r_2 \in [0, 1)$ and compute τ as $\tau \leftarrow -a^{-1} \log(r_2)$
 4. Update: $\mathbf{n} \leftarrow \mathbf{n} + \boldsymbol{\nu}_\mu$, where $\boldsymbol{\nu}_\mu$ is the stoichiometry of reaction μ
 5. Using the dependency graph, find the smallest reaction index i whose propensity is affected by μ
 6. for $j = i \dots M$
 - 6.1. Recompute s_j
 7. $t \leftarrow t + \tau$; go to step 2
-

Table 2.5: Algorithm for the logarithmic direct method (LDM)

2.3.7 Composition-rejection method (SSA-CR)

SSA-CR (Slepoy et al., 2008) uses composition-rejection sampling to determine the index of the next reaction. Composition-rejection sampling (Devroye, 1986; Slepoy et al., 2008; Ramaswamy and Sbalzarini, 2010b) is a way of sampling realizations of a random variable according to a given probability function. In SSA, the discrete probability function to sample the next reaction index μ is $p(\mu | \mathbf{n}(t))$ (see Eq. 1.51). The sampling process starts by binning the propensities a_i according to their value, and then proceeds in two steps: The composition step is used to identify the bin by linear search, the rejection step is used to identify the a_μ , and hence the next reaction index μ , inside that bin.

In SSA-CR, the propensities a_i are sorted into $G_a = \log_2(a_{\max}/a_{\min}) + 1$ bins such that bin b contains all a_i where b is determined by the condition: $2^{b-1}a_{\min} \leq a_i < 2^b a_{\min}$. The constants a_{\min} and a_{\max} are the smallest non-zero and largest value that any of the a_i 's can assume during the simulation. The value of a_{\min} is given by the minimum specific probability rate among all reactions in the network. The value of a_{\max} can be estimated by using physical reasoning. In cases where such an estimation is not possible, the number of bins G_a can also be dynamically increased during the simulation.

In SSA-CR, sampling the next reaction index μ proceeds in two steps: (1) composition sampling using linear search over the G_a bins and a uniform random number in $[0, 1)$ to identify the bin b such that

$$b = \min \left[b' : r_1 a < \sum_{i=1}^{b'} \alpha_i \right], \quad (2.39)$$

2 Formulations of the stochastic simulation algorithm for chemical reaction networks

where the total propensity α_b of each bin is computed by summing up the a_i 's in bin b , and subsequently (2) rejection (also known as acceptance-rejection) sampling to identify the reaction index μ in that bin b . This is done by generating a uniformly distributed random number r_2 in $[0, 2^b a_{\min}]$ and a uniformly distributed random integer r_3 between 1 and the number of elements in bin b . If the r_3^{th} element in bin b is greater than or equal to r_2 , the corresponding reaction index is chosen as the index of the next reaction. If the inequality is not satisfied, the rejection step is repeated. This procedure is illustrated in Fig. 2.3. Assume that the composition step has chosen bin 2 as the bin containing the next reaction index. The rejection step then samples uniformly random points inside the rectangle defining the value range of this bin (bold rectangle). A sample is accepted if it falls inside one of the shaded bars representing the propensities. If the first sample (point A in Fig. 2.3 with $r_3 = 3$ and $r_2 > a_2$) is rejected, sampling is repeated until the point falls inside one of the bars (point B in Fig. 2.3 with $r_3 = 2$ and $r_2 < a_4$). By binning the propensities as described above, one ensures that they cover at least 50% of each bin's total area, ensuring that the average number of rejection steps needed is less than or equal to two. Once a reaction is executed, the affected propensities are updated and their bin memberships are reassigned. Reassigning a bin membership can be achieved in $O(1)$ time. The algorithm for SSA-CR is given in Table 2.6.

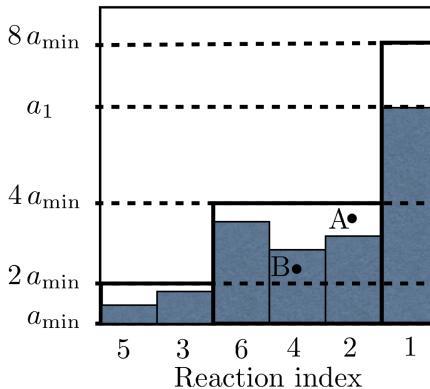


Figure 2.3: Illustration of the binning of the reaction propensities in SSA-CR. In this example the reaction network has six reactions and hence six propensities. The heights of the shaded rectangular bars indicate the magnitudes of the propensities of the reactions with the corresponding index. Points A and B are the examples used in the main text to illustrate rejection sampling.

The computational cost of SSA-CR is $O(G_a)$ for weakly coupled reaction networks. This is due to step 2.1 in Table 2.6 that involves a linear search over the G_a bins to identify the bin containing the index of the next reaction. In cases where the ratio of maximum to minimum non-zero propensity is constant, rendering $G_a O(1)$, the computational cost of SSA-CR reduces to $O(1)$. For strongly coupled reaction networks, the computational cost of SSA-CR is $O(M)$ since the degree of coupling of a strongly coupled network is $O(M)$ and hence the cost of the update step (Step 5 in Table 2.6) because $O(M)$ dominates the overall cost.

1. Set $t \leftarrow 0$; initialize \mathbf{n} , $a_\mu \forall \mu$, a , and the dependency graph. Sort the a_μ 's into G_a bins as described in Sec. 2.3.7. Initialize a_{\min} , a_{\max} and set $\Delta a \leftarrow 0$
2. Sample μ using composition-rejection sampling according to Eq. 1.51:
 - 2.1. Composition step: Generate a uniform random number $r_1 \in [0, 1)$ and perform a linear search over the G_a bins to sample the bin index b containing the next reaction according to Eq. 2.39
 - 2.2. Rejection step: Generate a uniform random number $r_2 \in [0, 2^b a_{\min})$ and a uniform random integer between 1 and the number of elements in bin b . If the r_3^{th} element in bin b is greater than or equal to r_2 , the corresponding reaction index is chosen as the index of the next reaction μ . If the inequality is not satisfied, repeat the rejection step
3. Sample τ according to Eq. 1.50: generate a uniform random number $r_2 \in [0, 1)$ and compute $\tau \leftarrow -a^{-1} \log(r_2)$
4. Update: $\mathbf{n} \leftarrow \mathbf{n} + \boldsymbol{\nu}_\mu$, where $\boldsymbol{\nu}_\mu$ is the stoichiometry of reaction μ
5. For all reactions i whose propensities change upon firing of reaction μ (i.e., the destination nodes of all directed edges leaving node μ in the dependency graph):
 - 5.1. Update $\Delta a \leftarrow \Delta a - a_i$
 - 5.2. Recompute the propensity a_i and reassign the bin membership of a_i if needed
 - 5.3. Update $\Delta a \leftarrow \Delta a + a_i$
6. Set $a \leftarrow a + \Delta a$
7. $t \leftarrow t + \tau$ and set $\Delta a \leftarrow 0$; go to step 2

Table 2.6: Algorithm for the SSA with composition-rejection sampling (SSA-CR)

2.3.8 Summary of the computational costs and memory requirements of these exact SSA formulations

Computational cost: The computational costs of all the aforementioned exact SSA formulations are $O(M)$ for strongly coupled reaction networks (Ramaswamy et al., 2009). For weakly coupled networks, however, some are significantly more efficient and can be $O(\log_2 M)$ or even $O(1)$ (Ramaswamy et al., 2009; Gibson and Bruck, 2000; Slepoy et al., 2008).

Memory requirement: The memory requirements of all aforementioned exact SSA formulations are $O(M)$. For strongly coupled reaction networks, this can be $O(M^2)$. This can be reduced to $O(M)$ by avoiding the use of a dependency graph.

2.4 Approximate SSA formulations

In approximate SSA formulations, the population $\mathbf{n}(t)$ is described by the equation of motion of a jump Markov process given by

$$\mathbf{n}(t + \Delta t) = \mathbf{n}(t) + \Xi(\Delta t; \mathbf{n}(t)). \quad (2.40)$$

The random variable $\Xi(\Delta t; \mathbf{n}(t))$, the Markov propagator, can be written as

$$\Xi(\Delta t; \mathbf{n}(t)) = \boldsymbol{\nu} \begin{bmatrix} \psi_1(\Delta t; \mathbf{n}(t)) \\ \vdots \\ \psi_\mu(\Delta t; \mathbf{n}(t)) \\ \vdots \\ \psi_M(\Delta t; \mathbf{n}(t)) \end{bmatrix}, \quad (2.41)$$

where $\psi_\mu(\Delta t; \mathbf{n}(t))$ is a random variable for the number of times reaction μ fires in the time interval $[t, t + \Delta t]$. Deriving an exact expression for $\psi_\mu(\Delta t; \mathbf{n}(t))$ is equivalent to solving the CME analytically. Hence, we assume that the propensities a_μ do not change in $[t, t + \Delta t]$ leading to an approximate solution of the CME. We start by dividing the time interval $[t, t + \Delta t]$ into k equisized subintervals. The probability $P(\psi_\mu(\Delta t; \mathbf{n}(t)) = \lambda)$ that reaction μ fires in each of $\lambda < k$ subintervals, and does not fire in any of the remaining subintervals, is given by (using Eq. 1.15)

$$P(\psi_\mu(\Delta t; \mathbf{n}(t)) = \lambda) = \lim_{k \rightarrow \infty} \frac{k!}{\lambda!(k - \lambda)!} \left[a_\mu(\mathbf{n}) \frac{\Delta t}{k} + O\left(\frac{\Delta t^2}{k^2}\right) \right]^\lambda \left[1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} + O\left(\frac{\Delta t^2}{k^2}\right) \right]^{(k-\lambda)}. \quad (2.42)$$

Noting that $\lim_{k \rightarrow \infty} O\left(\frac{\Delta t^2}{k^2}\right) = 0$, we get

$$\begin{aligned} P(\psi_\mu(\Delta t; \mathbf{n}(t)) = \lambda) &= \lim_{k \rightarrow \infty} \frac{k!}{\lambda!(k - \lambda)!} \left(a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^\lambda \left(1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^{(k-\lambda)} \\ &= \lim_{k \rightarrow \infty} \frac{k!}{k^\lambda (k - \lambda)!} \frac{(a_\mu(\mathbf{n}) \Delta t)^\lambda}{\lambda!} \left(1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^{(k-\lambda)} \\ &= \lim_{k \rightarrow \infty} \frac{(k - \lambda + 1) \dots k}{k^\lambda} \frac{(a_\mu(\mathbf{n}) \Delta t)^\lambda}{\lambda!} \left(1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^{(k-\lambda)} \\ &= \lim_{k \rightarrow \infty} \left(1 - \frac{\lambda + 1}{k} \right) \dots \left(1 - \frac{1}{k} \right) \frac{(a_\mu(\mathbf{n}) \Delta t)^\lambda}{\lambda!} \left(1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^{(k-\lambda)} \\ &= \lim_{k \rightarrow \infty} \frac{(a_\mu(\mathbf{n}) \Delta t)^\lambda}{\lambda!} \left(1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^k \left(1 - a_\mu(\mathbf{n}) \frac{\Delta t}{k} \right)^{-\lambda} \\ &= \frac{(a_\mu(\mathbf{n}) \Delta t)^\lambda}{\lambda!} e^{-a_\mu(\mathbf{n}) \Delta t}. \end{aligned} \quad (2.43)$$

Therefore, the random variable $\psi_\mu(\Delta t; \mathbf{n}(t))$ is distributed according to the Poisson distribution i.e., $\psi_\mu(\Delta t; \mathbf{n}(t)) \sim \mathcal{P}(a_\mu(\mathbf{n})\Delta t)$. In deriving this we assumed that the propensity $a_\mu(\mathbf{n})$ does not change during the time interval $[t, t + \Delta t]$. This condition can only be satisfied if exactly one reaction fires per time step just as in exact SSA formulation (see Sec. 2.3). Approximate SSA's impose two conditions (Gillespie, 2000, 2001) :

1. Δt must be small enough for the change in the reaction propensities to be small i.e.

$$a_\mu(\mathbf{n}(t + \Delta t)) \approx a_\mu(\mathbf{n}(t)) \quad \forall \mu. \quad (2.44)$$

2. Δt must be large enough for the average number of firings of reaction μ to be much larger than 1. Otherwise the method offers no improvements of the computational performance over exact SSA formulations.

$$\langle \psi_\mu(\Delta t; \mathbf{n}(t)) \rangle = a_\mu(\mathbf{n})\Delta t > 1. \quad (2.45)$$

There are various heuristics for choosing a time step Δt that satisfies the above two conditions (Cao et al., 2006; Gillespie, 2000, 2001). Note that the second condition is not necessary, but it improves the computational performance of approximate SSA's. In general, since many reactions fire within a time step Δt by sampling $\psi_\mu(\Delta t; \mathbf{n}(t))$ from $\mathcal{P}(a_\mu(\mathbf{n})\Delta t)$ for $\mu = 1, \dots, M$, approximate SSA's have a computational cost of $O(M)$ per time step. The computational cost per reaction event is $O(M)$ divided by the number of reactions fired during the time step Δt . This renders the computational cost of approximate SSAs superior to that of exact SSAs. The scaling of the computational cost of approximate SSAs, however, is the same as that of the exact direct method (DM).

Considering that

$$a_\mu(\mathbf{n})\Delta t \gg 1$$

is valid when the population $\mathbf{n}(t)$ is large, the random variables ψ_μ in Eq. 2.41 can be approximates by a Gaussians, i.e.,

$$\psi_\mu(\Delta t; \mathbf{n}(t)) \sim \mathcal{P}(a_\mu(\mathbf{n})\Delta t) \approx \mathcal{N}(a_\mu(\mathbf{n})\Delta t, a_\mu(\mathbf{n})\Delta t) \quad \text{if } a_\mu(\mathbf{n})\Delta t \gg 1. \quad (2.46)$$

Under this approximation the Markov propagator $\Xi(\Delta t; \mathbf{n}(t))$ in Eq. 2.41 can be written as

$$\Xi(\Delta t; \mathbf{n}(t)) \sim \mathcal{N}(\Omega \mathbf{F}(\phi)\Delta t, \Omega \mathbf{D}(\phi)\Delta t), \quad (2.47)$$

where $\mathcal{N}(\mathbf{m}, \Sigma)$ denotes a multivariate Gaussian distribution with mean vector \mathbf{m} and covariance matrix Σ . After changing from population \mathbf{n} to concentration ϕ , Eq. 2.40 hence becomes the *chemical Langevin equation*, which is the equation of motion (see Eq. 1.36) of the *nonlinear chemical Fokker-Planck equation* (see Eq. 1.32) obtained by truncating the *chemical Kramer-Moyal equation* (see Eq. 1.27). This shows that the *nonlinear chemical Fokker-Planck equation* is not just a arbitrary approximation, but one when the population $\mathbf{n}(t)$ is large or, equivalently, when the reactor volume Ω is large (Horsthemke and Brenig, 1977; Gillespie, 2000, 2001).

2 Formulations of the stochastic simulation algorithm for chemical reaction networks

Approximate SSA formulations simulate Eq. 2.40 with different numerical schemes. They are computationally efficient when the population of species is not small (more than a few hundreds). The weak order of convergence of these methods has been shown to be at least $\frac{1}{2}$ (Anderson et al., 2011; Rathinam et al., 2005) for some moments of the state-probability function. There are, however, still some disagreements regarding the order of convergence (Anderson et al., 2011; Rathinam et al., 2005). It is also unclear how quantities that contain information on the path of the stochastic process, such as the time-correlation functions and “higher order quantities”, such as switching frequencies in multistable systems, converge with decreasing time step Δt (Helmuth et al., 2011).

2.5 Summary

We have presented the topological properties and the dependency graph representation of chemical reaction networks. Using the dependency graph representation, we defined the degree of coupling of a reaction network as the maximum out-degree of the graph. This enabled classifying reaction networks into *weakly coupled* and *strongly coupled* ones. Weakly coupled reaction networks are those in which the degree of coupling is bounded by a constant with increasing network size. For strongly coupled networks the degree of coupling increases with the network size.

We further presented various exact formulations of Gillespie’s stochastic simulation algorithm (SSA). We showed that for weakly coupled networks the computational cost of exact SSA has been reduced to $O(\log_2 M)$, or even $O(1)$ under the assumption that the ratio between the maximum and minimum non-zero reaction propensities is bounded by a constant with increasing network size. For strongly coupled networks, however, all presented exact SSA formulations have a computational cost of $O(M)$.

We also presented the concept of approximate SSA formulations that can offer superior computational performance at the expense of accuracy. Approximate SSA formulations can offer significant speed-ups when the population of the chemical species is large. The computational cost of approximate SSAs is $O(M)$, irrespective of the coupling class of the reaction network. The speed-up over exact SSAs is obtained by firing many reactions during a single time step, proportionally reducing the prefactor of the computational cost.

3

Partial-propensity formulations of the stochastic simulation algorithm

“The human brain is incapable of creating anything which is really complex.”
– Andrey Kolmogorov

Stochastic chemical kinetics is described by the chemical master equation (CME). Numerical simulations of the CME can be done using a kinetic Monte Carlo scheme called the stochastic simulation algorithm (SSA) (Gillespie, 1976, 1992) (see Sec. 1.3). The computational cost of exact SSA formulations depends on the property of the chemical reaction network. To predict the computational cost of exact SSA formulations, we classify reaction networks into two classes: *strongly coupled* and *weakly coupled*. Any chemical reaction network with N species and M reactions can be represented by its dependency graph. Each node in this graph represents a chemical reaction and a directed edge is drawn from node p to node q if firing of reaction p affects the copy number of any of the reactants of reaction q . In this representation, we quantify the degree of coupling of the reaction network as the maximum number of edges leaving any node, i.e., the maximum out-degree of the dependency graph. We define weakly coupled networks as those in which the degree of coupling is bounded by a constant with increasing network size. Strongly coupled networks have a degree of coupling that increases unboundedly with network size. The computational cost of exact SSA formulations depends on the coupling class of the reaction network. For weakly coupled reaction networks the computational cost (CPU time) has been reduced to $O(1)$ (Slepoy et al., 2008) under the assumption that the ratio of maximum to minimum propensity is bounded by a constant. For strongly coupled networks, however, the computational cost of exact SSAs remains $O(M)$. For

details see Chapter 2.

We present here a new class of exact SSA formulations using the novel concept of partial propensities. In doing so, we limit ourselves to networks of elementary reactions under the premise that non-elementary reactions can be broken down to elementary ones at the expense of an increase in network size (Gillespie, 1992; Wilhelm, 2000; Schneider and Wilhelm, 2000) (see Appendix A2). In Sec. 3.1 we introduce partial propensities. In Sec. 3.2 we introduce the concept of partial-propensity formulations. Using partial propensities we first describe the partial-propensity direct method (PDM) in Sec. 3.3. The computational cost of PDM is $O(N)$ (i.e., linear in the number of chemical species) irrespective of the degree of coupling of the reaction network. In Sec. 3.4 we present the partial-propensity SSA with composition-rejection sampling (PSSA-CR) that further reduces the computational cost for weakly-coupled networks to $O(1)$. In Sec. 3.5 we extend the partial-propensity formulation to reaction networks with time delays using the delay PDM (dPDM). Finally, we summarize the partial-propensity formulations as a family of SSAs with algorithmic building blocks that naturally constitute the different formulations.

3.1 Partial propensity

We define the partial propensity of a reaction with respect to one of its reactants as the propensity per molecule of this reactant. For example, the partial propensity $\pi_\mu^{(i)}$ of reaction μ with respect to (perhaps the only) reactant S_i is a_μ/n_i , where a_μ is the propensity of reaction μ and n_i is the number of molecules of S_i . The partial propensities of the three elementary reaction types are:

- Bimolecular reactions ($S_i + S_j \rightarrow \text{Products}$): $a_\mu = n_i n_j c_\mu$ and $\pi_\mu^{(i)} = n_j c_\mu$, $\pi_\mu^{(j)} = n_i c_\mu$. If both reactants are of the same species, i.e. $S_i = S_j$, only one partial propensity exists, $\pi_\mu^{(i)} = \frac{1}{2}(n_i - 1)c_\mu$ because the reaction degeneracy is $\frac{1}{2}n_i(n_i - 1)$.
- Unimolecular reactions ($S_i \rightarrow \text{Products}$): $a_\mu = n_i c_\mu$ and $\pi_\mu^{(i)} = c_\mu$.
- Source reactions ($\emptyset \rightarrow \text{Products}$): $a_\mu = c_\mu$ and $\pi_\mu^{(0)} = c_\mu$.

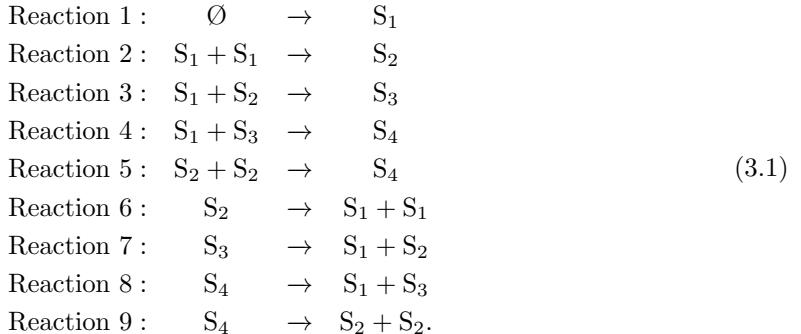
3.2 The concept of partial-propensity SSA formulations

Partial-propensity formulations use partial propensities and group them in order to sample the index of the next reaction and to update the partial propensities after a reaction has fired. For the sampling step, the partial propensities are grouped according to the index of the factored-out reactant, yielding at most $N + 1$ groups of size $O(N)$. Sampling then proceeds in two steps: we first sample the index of the group before sampling the actual partial propensity inside that group. This sampling procedure can also be interpreted as sampling reaction partners. The first step involves sampling a reactant before sampling its reaction partner. This grouping scheme reduces the number of operations needed for sampling the next reaction using a concept that is reminiscent of cell lists (Hockney and Eastwood, 1988).

3.2 The concept of partial-propensity SSA formulations

After the selected reaction has been executed, we use a dependency graph over species, rather than reactions, to find all partial propensities that need to be updated. The dependency graph over species acts as a pointer to the partial propensities that need to be updated upon change in population of a certain species. This is possible because partial propensities depend on the population of at most one species, and it is analogous to a Verlet list (Verlet, 1967). This limits the number of updates to $O(N)$. In addition, partial propensities of unimolecular reactions are constant and never need to be updated. In weakly coupled networks, where the degree of coupling is $O(1)$, the scaling of the computational cost of the update becomes equal to that of methods that use dependency graphs over reactions, such as SSA-CR (Slepoy et al., 2008) (see Sec. 2.3.7), ODM (Cao et al., 2004) (see Sec. 2.3.4), and SDM (McCollum et al., 2006) (see Sec. 2.3.5).

We illustrate the concept of partial propensity methods using a simple protein aggregation example. Consider proteins that aggregate to form at most tetrameric complexes. There are $N = 4$ species in the reaction network: monomers (S_1), dimers (S_2), trimers (S_3), and tetramers (S_4). All species except tetramers can aggregate in all possible combinations to form multimeric complexes (4 bimolecular reactions). In addition, all multimeric complexes can dissociate into any possible combination of two smaller units (4 unimolecular reactions) and monomers are constantly produced (1 source reaction). The reaction network is given by the following 9 reactions



This reaction network is described by $M = 9$ partial propensities $(\pi_1^{(0)}, \pi_2^{(1)}, \pi_3^{(1)}, \pi_4^{(1)}, \pi_5^{(2)}, \pi_6^{(2)}, \pi_7^{(3)}, \pi_8^{(4)}, \pi_9^{(4)})$. Grouping the partial propensities according to the index of the factored-out reactant given in the superscript, we obtain 5 ($= N + 1$) groups as indicated by the parentheses. Along with each group, we store the sum of all partial propensities inside it. We first sample the group that contains the next reaction before finding the corresponding partial propensity inside that group. Assume that in our example reaction 7 is to fire next. Different search algorithms can be used for this task. Using linear search, for instance, the search depth to find the group index is 4 and the search depth to find the partial propensity $(\pi_7^{(3)})$ is 1. Linear search thus requires 5 operations to sample the next reaction in this network of 9 reactions. The average linear search depth for sampling the next reaction in this example is $37/9 \approx 4.1$, if all reaction propensities are equal.

3.3 The partial-propensity direct method (PDM)

In PDM, the index of the next reaction μ is sampled using linear search over groups and subsequently over elements in the group. The sampling procedure is algebraically equivalent to that of Gillespie's direct method (DM). The novelties in PDM are the use of partial propensities and efficient data structures that reduce the number of operations needed to sample μ and to update the partial propensities. The time to the next reaction τ is sampled as in DM.

3.3.1 Detailed description

All partial propensities are stored in the “partial-propensity structure” $\boldsymbol{\Pi} = \{\boldsymbol{\Pi}_i\}_{i=0}^N$ as a one-dimensional array of one-dimensional arrays $\boldsymbol{\Pi}_i$. Each array $\boldsymbol{\Pi}_i$ contains the partial propensities belonging to group i . The partial propensities of source reactions are stored as consecutive entries of the 0^{th} array $\boldsymbol{\Pi}_0$. The partial propensities of all reactions that have species S_1 as one of its reactants are stored as consecutive entries of $\boldsymbol{\Pi}_1$. In general, the i^{th} array $\boldsymbol{\Pi}_i$ contains the partial propensities of all reactions that have S_i as a reactant, provided these reactions have not yet been included in any of the previous $\boldsymbol{\Pi}_{j < i}$. That is, out of the two partial propensities of a reaction μ with S_i and S_j as reactants, $\pi_\mu^{(i)}$ is part of $\boldsymbol{\Pi}_i$ if $i < j$, and $\pi_\mu^{(j)}$ is not stored anywhere. In order for $\boldsymbol{\Pi}$ to be independent of the numbering of the reactants, we first renumber the species such that S_i is the species involved as a reactant in i^{th} -most reactions. This ordering also minimizes the number of required update operations. $\boldsymbol{\Pi}$ needs to be constructed only once, at the beginning of a simulation. The steps to automatically build $\boldsymbol{\Pi}$ from the stoichiometry matrix are outlined in Table 3.1.

Since the different $\boldsymbol{\Pi}_i$'s can be of different length, storing them as an array of arrays is more (memory) efficient than using a matrix (i.e., a two-dimensional array). The reaction indices of the partial propensities in $\boldsymbol{\Pi}$ are stored in a look-up table $\mathbf{L} = \{\mathbf{L}_i\}_{i=0}^N$, which is also an array of arrays. This makes every reaction μ identifiable by a unique pair of indices, a group index I and an element index J , such that the partial propensity of reaction $\mu = \mathbf{L}_{I,J}$ is stored in $\boldsymbol{\Pi}_{I,J}$.

We further define the “group-sum array” $\boldsymbol{\Lambda}$, storing the sums of the partial propensities in each group $\boldsymbol{\Pi}_i$, thus $\Lambda_i = \sum_j \Pi_{i,j}$, $i = 0, \dots, N$. In addition, we also define $\boldsymbol{\Sigma}$, the array of the total propensities of all groups, as $\Sigma_i = n_i \Lambda_i$, $i = 0, \dots, N$, and set the population n_0 of the reservoir in the source reactions to 1. The total propensity of all reactions is then $a = \sum_{i=0}^N \Sigma_i$. The use of $\boldsymbol{\Lambda}$ avoids having to recompute the sum of all partial propensities in $\boldsymbol{\Pi}_i$ after one of them has changed. Rather, the same change is also applied to Λ_i , and computing the new Σ_i only requires a single multiplication by n_i . Using these data structures and a single uniformly distributed random number $r_1 \in [0, 1]$, the next reaction μ can efficiently be sampled in two steps: (1) sampling the group index I such that

$$I = \min \left[I' : r_1 a < \sum_{i=0}^{I'} \Sigma_i \right] \quad (3.2)$$

3.3 The partial-propensity direct method (PDM)

1. Initialize the reactant stoichiometry matrix ν^- , the initial population $\mathbf{n}(0)$, and the specific probability rates \mathbf{c} . Reorder all stoichiometry matrices such that the i^{th} row corresponds to the species involved as a reactant in i^{th} -most reactions.
 2. Using ν^- , build a list of all reactants in each reaction. For reaction μ , the reactants have a non-zero entry in ν_{μ}^- . If all species have a zero reactant stoichiometry then the reactant index is 0 and the reaction is a source reaction.
 3. For each reaction, go through the list of reactants:
 - 3.1 If the number of distinct reactants in a reaction is 2, compute the partial propensity of this reaction by factoring out the population of the species with the *smaller* index i from the full reaction propensity. Append this partial propensity to Π_i .
 - 3.2 If the number of reactants in a reaction is 1, then check
 - 3.2.1 If it is a biomolecular reaction between the same species S_i , store the corresponding partial propensity in Π_i .
 - 3.2.2 If it is a unimolecular reaction with only species S_i as a reactant, store the partial propensity in Π_i .
 - 3.2.3 If it is a source reaction ($i = 0$), store the partial propensity in Π_0 .
 4. Stop.
-

Table 3.1: Algorithm for constructing the partial-propensity structure Π

and (2) sampling the element index J in Π_I such that

$$J = \min \left[J' : r_1 a < \sum_{j=1}^{J'} n_I \Pi_{I,j} + \left(\sum_{i=0}^I \Sigma_i \right) - \Sigma_I \right]. \quad (3.3)$$

(see Appendix A3 for a proof of the equivalence of this sampling scheme to that of DM.) Using the temporary variables

$$\Phi = \sum_{i=0}^I \Sigma_i, \quad \Psi = \frac{r_1 a - \Phi + \Sigma_I}{n_I}, \quad (3.4)$$

Eq. 3.3 can be efficiently implemented as

$$J = \min \left[J' : \Psi < \sum_{j=1}^{J'} \Pi_{I,j} \right]. \quad (3.5)$$

The indices I and J are then translated back to the reaction index μ using the look-up table \mathbf{L} , thus $\mu = L_{I,J}$.

To execute a sampled reaction, \mathbf{n} , Π , Λ , and Σ need to be updated. This is efficiently done using three update structures:

3 Partial-propensity formulations of the stochastic simulation algorithm

$\mathbf{U}^{(1)}$ is an array of M arrays, where the i^{th} array contains the indices of all species involved in the i^{th} reaction.

$\mathbf{U}^{(2)}$ is an array of M arrays containing the corresponding stoichiometry (the change in population of each species upon reaction) of the species stored in $\mathbf{U}^{(1)}$. $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ constitute the sparse representation of the stoichiometry matrix $\boldsymbol{\nu}$.

$\mathbf{U}^{(3)}$ is an array of N arrays, where the i^{th} array contains the indices of all entries in $\mathbf{\Pi}$ that depend on n_i , thus:

$$\mathbf{U}^{(3)} = \begin{cases} \mathbf{U}_1^{(3)} = [(i_1^1, j_1^1) \quad (i_2^1, j_2^1) \quad \dots \quad \dots \quad \dots] \\ \mathbf{U}_2^{(3)} = [(i_1^2, j_1^2) \quad (i_2^2, j_2^2) \quad \dots] \\ \vdots \\ \mathbf{U}_N^{(3)} = [i_1^N, j_1^N \quad i_2^N, j_2^N \quad \dots \quad \dots]. \end{cases} \quad (3.6)$$

When a reaction is executed, the populations of the species involved in this reaction change. Hence, all entries in $\mathbf{\Pi}$ that depend on these populations need to be updated. After each reaction, we use $\mathbf{U}^{(1)}$ to determine the indices of all species involved in this reaction. The stoichiometry is then looked up in $\mathbf{U}^{(2)}$ and the population \mathbf{n} is updated. Subsequently, $\mathbf{U}^{(3)}$ is used to locate the affected entries in $\mathbf{\Pi}$ and recompute them. The two data structures $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are a sparse representation of the stoichiometry matrix, and $\mathbf{U}^{(3)}$ represents the dependency graph over species. Since the partial propensities of unimolecular and source reactions are constant and need never be updated, $\mathbf{U}^{(3)}$ only contains the indices of the partial propensities of bimolecular reactions. The size of $\mathbf{U}^{(3)}$ is a factor of $O(N)$ smaller than that of the corresponding dependency graph over reactions, since partial propensities depend on the population of at most one species. Figure 3.1 summarizes the data structures used in PDM for an example reaction network. The complete algorithm is given in Table 3.2. Overall, PDM's computational cost is $O(N)$ and its memory requirement is $O(M)$, irrespective of the degree of coupling of the simulated network (see Sec. 3.3.3).

1. Initialization: set $t \leftarrow 0$; initialize \mathbf{n} , $\mathbf{\Pi}$, $\mathbf{\Lambda}$, $\mathbf{\Sigma}$; $a \leftarrow \sum_{i=0}^N \Sigma_i$; $\Delta a \leftarrow 0$; generate \mathbf{L} , $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$
2. Sample μ : generate a uniform random number $r_1 \in [0, 1)$ and determine the group index I and the element index J according to Eqs. 3.2, 3.4, and 3.5; $\mu \leftarrow L_{I,J}$
3. Sample τ : generate a uniform random number $r_2 \in [0, 1)$ and compute the time to next reaction τ as $\tau \leftarrow a^{-1} \ln(r_2^{-1})$
4. Update \mathbf{n} : for each index k of $\mathbf{U}_\mu^{(1)}$, $l \leftarrow U_{\mu,k}^{(1)}$ and $n_l \leftarrow n_l + U_{\mu,k}^{(2)}$
5. Update $\mathbf{\Pi}$, $\mathbf{\Lambda}$, $\mathbf{\Sigma}$ and compute Δa , the change in a :

For each index k of $\mathbf{U}_\mu^{(1)}$, do:

 - 5.1. $l \leftarrow U_{\mu,k}^{(1)}$
 - 5.2. For each index m of $\mathbf{U}_l^{(3)}$, do:
 - 5.2.1. $(i_m^l, j_m^l) \leftarrow U_{l,m}^{(3)}$ (Eq. 3.6)
 - 5.2.2. $\Pi_{i_m^l, j_m^l} \leftarrow \Pi_{i_m^l, j_m^l} + c_{\mu'} U_{\mu,k}^{(2)}$, $\mu' = L_{i_m^l, j_m^l}$ if $l \neq i_m^l$
 $\Pi_{i_m^l, j_m^l} \leftarrow \Pi_{i_m^l, j_m^l} + \frac{1}{2} c_{\mu'} U_{\mu,k}^{(2)}$, $\mu' = L_{i_m^l, j_m^l}$ if $l = i_m^l$
 - 5.2.3. $\Lambda_{i_m^l} \leftarrow \Lambda_{i_m^l} + c_{\mu'} U_{\mu,k}^{(2)}$, $\mu' = L_{i_m^l, j_m^l}$ if $l \neq i_m^l$
 $\Lambda_{i_m^l} \leftarrow \Lambda_{i_m^l} + \frac{1}{2} c_{\mu'} U_{\mu,k}^{(2)}$, $\mu' = L_{i_m^l, j_m^l}$ if $l = i_m^l$
 - 5.2.4. $\Sigma_{\text{temp}} \leftarrow \Sigma_{i_m^l}$
 - 5.2.5. $\Sigma_{i_m^l} \leftarrow n_{i_m^l} \Lambda_{i_m^l}$
 - 5.2.6. $\Delta a \leftarrow \Delta a + \Sigma_{i_m^l} - \Sigma_{\text{temp}}$
 - 5.3. $\Delta a \leftarrow \Delta a + n_l \Lambda_l - \Sigma_l$; $\Sigma_l \leftarrow n_l \Lambda_l$
6. Update a and increment time: $a \leftarrow a + \Delta a$; $\Delta a \leftarrow 0$; $t \leftarrow t + \tau$
7. Go to step 2

Table 3.2: Detailed algorithm for the partial-propensity direct method PDM.

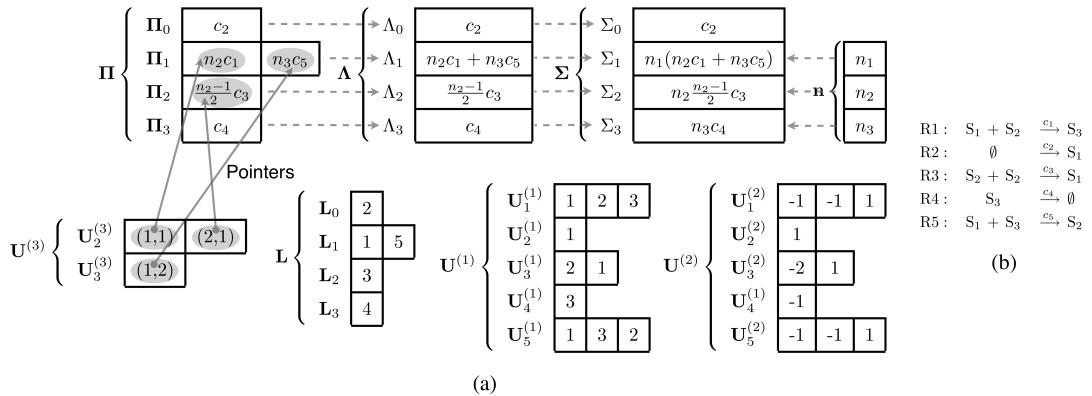


Figure 3.1: (a) Illustration of the data structures in PDM for the example reaction network shown in (b). Note that there may be arrays $\Pi_i, i = 1, \dots, N$, containing at most one negative entry if the corresponding $n_i = 0$. Indeed, in this example, $\Pi_{2,1} < 0$ and $\Lambda_2 < 0$ if $n_2 = 0$. This, however, poses no problem in sampling I and J as all Σ_i for which $n_i = 0$ are zero and hence the corresponding group indices I are never selected.

3.3.2 The sorting partial-propensity direct method (SPDM)

The sorting partial-propensity direct method (SPDM) is the partial-propensity variant of SDM (McCollum et al., 2006). In SPDM, the group and element indices I and J are bubbled up whenever the reaction $\mu = L_{I,J}$ fires. The reordered indices are stored in an array for I , and an array of arrays of the size of Π for the J 's. This requires an additional $N + M$ memory, but further reduces the average search depth to sample the next reaction, especially for multiscale (stiff) networks. The computational cost of SPDM is also $O(N)$, but with a possibly reduced pre-factor (see Sec. 3.3.3).

3.3.3 Computational cost and memory requirements

3.3.3.1 Computational cost

The computational cost of PDM is governed by the following steps: (a) sampling the index of the next reaction and (b) updating the population \mathbf{n} and the partial-propensity structure Π . The scaling of the computational cost of SPDM is the same as that of PDM. In terms of absolute runtimes SPDM, however, is expected to be more efficient than PDM especially for multiscale reaction networks.

Computational cost of sampling the index of the next reaction: For any chemical reaction network with N species, the number of arrays in the partial-propensity structure Π is at most $N + 1$, which is also the maximum length of Σ and Λ . The number of entries in each array Π_i is $O(N)$, since any species can react with at most N species in bimolecular reactions and undergo at most $O(N)$ unimolecular reactions. Sampling the index of the next reaction

3.3 The partial-propensity direct method (PDM)

involves two steps: (a) a linear search for the group index I in Σ and (b) a linear search for the element index J in Π_I . Since Σ is at most of length $N + 1$, the first step is $O(N)$. The second step is also $O(N)$, since the number of elements in Π_i is $O(N)$. The overall computational cost of sampling the next reaction is thus $O(N)$ for networks of any degree of coupling.

Computational cost of the update: Let the maximum number of chemical species involved in any reaction (as reactants or products) be given by the constant s . Let us assume that s is $O(1)$ with increasing system size. This assumption is not restrictive since the number of species involved in a reaction is unlikely to increase with system size. The computational cost of updating \mathbf{n} is thus $s \in O(1)$. In PDM, only the partial propensities of bimolecular reactions need to be updated. The total number of entries in the third update structure $\mathbf{U}^{(3)}$ is, thus, equal to the number of bimolecular reactions. In addition, the total number of entries in Π that depend on any n_i is always less than or equal to N , as any species S_i can only react with itself and the remaining $N - 1$ species in bimolecular reactions. Therefore, the upper bound for the total number of partial propensities in Π to be updated after executing any reaction is $sN \in O(N)$.

In summary, the computational cost of PDM is $O(N)$, irrespective of the degree of coupling of the reaction network (see Sec. 3.3.4 for benchmark results).

3.3.3.2 Memory requirements

The memory requirement of PDM is given by the total size of the data structures \mathbf{n} , Π , \mathbf{L} , Λ , Σ , $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$.

The partial-propensity structure Π and the look-up table \mathbf{L} have the same size. Since every reaction is accounted for exactly once, each structure requires $O(M)$ memory. Λ , \mathbf{n} , and Σ are all at most of length $N + 1$ and thus require $O(N)$ memory. The sizes of $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are $O(M)$, and the size of $\mathbf{U}^{(3)}$ is proportional the number of bimolecular reactions and, hence, $O(M)$ if all reactions are bimolecular.

In summary, the memory requirement of PDM is $O(M)$. SPDM requires an additional $N + M$ memory to store the reordered index lists.

3.3.4 Benchmarks

We benchmark the computational performance of PDM and SPDM using four chemical reaction networks that are prototypical of: (a) strongly coupled reaction networks, (b) strongly coupled reaction networks comprising only bimolecular reactions, (c) weakly coupled reaction networks, and (d) multiscale biological networks. The first two benchmarks consider strongly coupled networks where the degree of coupling d_c scales with system size (see column “Maximum” under “Degree of coupling of nodes (d_μ)” in Table 3.3). The first benchmark consists of a colloidal aggregation model. The second benchmark considers a network of only bimolecular reactions, where none of the partial propensities are constant. In the third benchmark, we compare PDM and SPDM to SDM on the linear chain model, a weakly coupled reaction network with the minimal degree of coupling, for which SDM was reported to be very efficient (Cao et al., 2004; McCollum et al., 2006). The fourth benchmark considers the heat-shock response model, a small multiscale (stiff) biological reaction network of fixed size.

3 Partial-propensity formulations of the stochastic simulation algorithm

Model	Number of species (N)	Number of reactions (M)	Degree of coupling of nodes (d_μ)		
			Minimum ($\min\{d_\mu\}$)	Average $\left(\frac{\sum_{\mu=1}^M d_\mu}{M}\right)$	Maximum ($d_c = \max\{d_\mu\}$)
CA	N	$\left\lfloor \frac{N^2}{2} \right\rfloor$	$\frac{4N-1}{3}$	$2.3N - 4.7$	$3N - 7$
NB	N	$\frac{N(N-1)}{2}$	$4N - 10$	$4N - 10$	$4N - 10$
LC	N	$N - 1$	$1^{(*)}$	$2 - \frac{1}{N-1} \approx 2$	2
HSR	28	61	1	5.9	11

Table 3.3: Properties of the benchmark cases. The number of species, number of reactions, and minimum, average, maximum out-degree (degree of coupling) of the dependency graph are given for the benchmark cases defined in Sections 3.3.4.1, 3.3.4.2, 3.3.4.3 and 3.3.4.4: the colloidal aggregation model (CA), the network of bimolecular reactions (NB), the linear chain model (LC), and the heat-shock response model (HSR). (*) In the linear chain model the degree of coupling is 1 only for the last reaction, since its product is not a reactant anywhere else.

All tested SSA formulations are implemented in C++ using the random-number generator of the GSL library and compiled using the GNU C++ compiler version 4.0.1 with the O3 optimization flag. All timings are determined using a nanosecond-resolution timer (the `mach_absolute_time()` system call) on a MacOS X 10.4.11 workstation with a 3GHz dual-core Intel Xeon processor, 8GB of memory, and a 4MB L2 cache. For each test case we report both the memory requirement and the average CPU time per reaction (i.e., per time step), Θ . Θ is defined as the CPU time (identical to wall-clock time in our case) needed to simulate the system up to final time T , divided by the total number of reactions executed during the simulation, and averaged over independent runs. The time Θ does not include the initialization of the data structures (step 1 in Table 3.2) as this is done only once and is not part of the time loop.

We explain the benchmark results in terms of the computational cost of the individual steps of the algorithms. We distinguish three steps: (a) sampling the index of the next reaction, (b) updating the population, and (c) updating the partial propensities (for PDM and SPDM) or the propensities (for SDM). The computational costs of these steps are quantified separately and the overall timings are then explained as a weighted sum of:

- \mathcal{C}_μ : The number of operations required to sample the index of the next reaction (for PDM, this is step 2 in Table 3.2).
- \mathcal{C}_n : The number of elements of the population n that need to be updated after executing a reaction (for PDM, this is step 4 in Table 3.2).
- \mathcal{C}_P : The number of (partial) propensities that need to be updated after executing a reaction (for PDM, this is step 5.2.2 in Table 3.2).

The expressions for these elementary costs are given in Table 3.4 as determined by independently fitting models for the scaling of the algorithms to the measured operation counts,

3.3 The partial-propensity direct method (PDM)

averaged over 100 independent runs of each test problem. In all cases, the models used for the computational cost explain the data with a correlation coefficient of at least 0.98. The benchmark results are then explained by fitting the weights of the cost superposition $a\mathcal{C}_\mu + b\mathcal{C}_n + c\mathcal{C}_P$ to the measured scaling curves $\Theta(N)$ using the expressions given in Table 3.4. In order to preserve the relative weights of the data points, all fits are done on a linear scale, even though the results are plotted on a logarithmic scale for two of the benchmarks. All these fits also have a correlation coefficient of at least 0.98. Explaining the timing results as a superposition of elementary costs allows determining which part of an algorithm is responsible for a particular speedup or scaling behavior, and what the relative contributions of the three algorithmic steps are to the overall computational cost.

	PDM			SPDM		
	\mathcal{C}_μ	\mathcal{C}_n	\mathcal{C}_P	\mathcal{C}_μ	\mathcal{C}_n	\mathcal{C}_P
CA	$0.49N + 2.0$	3	$5.2N^{0.5} - 8.1$	$0.45N + 0.38$	3	$5.2N^{0.5} - 8.1$
NB	$0.97N - 1.3$	4	$1.6N - 3.2$	$0.94N - 4.7$	4	$1.6N - 3.2$
LC	$0.50N + 1.0$	2	0	$1.0N^{0.5} + 0.79$	2	0
HSR	13	3	2.2	3.7	3	2.2
	SDM					
	\mathcal{C}_μ	\mathcal{C}_n	\mathcal{C}_P			
CA	$0.14N^2 + 1.2N - 9.9$	N	$2.8N - 10$			
NB	$0.33N^2 - 0.044N + 0.51$	N	$4.0N - 10$			
LC	$1.0N^{0.5} - 0.21$	N	2			
HSR	2.9	28	8.2			

Table 3.4: Number of compute operations needed by the different algorithms (PDM, SPDM, SDM) for the different test cases (CA: colloidal aggregation model; NB: network of bimolecular reactions; LC: linear chain model; HSR: heat-shock response model). \mathcal{C}_μ is the average number of operations needed to sample the next reaction μ . \mathcal{C}_n is the average number of entries in the population n that need to be updated after any reaction. \mathcal{C}_P is the average number of partial propensities (or propensities for SDM) that need to be updated after any reaction. The operation counts are averaged over all reactions executed during 100 independent runs of each benchmark over the range of N shown in Fig. 3.2. The average numbers are then fitted with the models given here (with correlation coefficient of at least 0.98 in all cases). See Fig. 3.3 for the distribution of the number of updates.

The memory requirements of the algorithms are reported in Table 3.5 for all benchmark cases. These numbers were derived analytically from the size of the individual data structures.

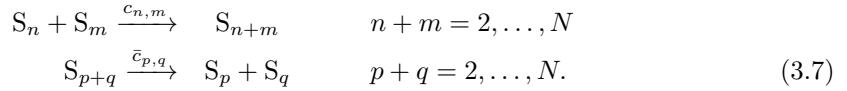
	PDM/SPDM				
	$\mathbf{n}, \boldsymbol{\Lambda}, \boldsymbol{\Sigma}$	$\boldsymbol{\Pi}, \mathbf{L}, \mathbf{c}$	$\mathbf{U}^{(1)}, \mathbf{U}^{(2)}$	$\mathbf{U}^{(3)}$	Total
CA	N	$\left\lfloor \frac{N^2}{2} \right\rfloor$	$3 \left\lfloor \frac{N^2}{2} \right\rfloor$	$2 \left\lfloor \frac{N^2}{4} \right\rfloor$	$O(N^2) = O(M)$
NB	N	$\frac{N(N-1)}{2}$	$4 \frac{N(N-1)}{2}$	$2 \frac{N(N-1)}{2}$	$O(N^2) = O(M)$
LC	N	$N - 1$	$2(N - 1)$	0	$O(N) = O(M)$
HSR	28	61	133	24	557

	SDM				
	\mathbf{n}	\mathbf{c}, \mathbf{a}	dependency graph	$\boldsymbol{\nu}$	Total
CA	N	$\left\lfloor \frac{N^2}{2} \right\rfloor$	$1.2N^3 - 2.5N^2 + 2.3N$	$N \left\lfloor \frac{N^2}{2} \right\rfloor$	$O(N^3) = O(NM)$
NB	N	$\frac{N(N-1)}{2}$	$2N^3 - 7N^2 + 5N$	$\frac{N^2(N-1)}{2}$	$O(N^3) = O(NM)$
LC	N	$N - 1$	$2(N - 1)$	$N(N - 1)$	$O(N^2) = O(NM)$
HSR	28	61	360	1708	2218

Table 3.5: Total amount of computer memory needed by the different algorithms (PDM, SPDM, SDM) for the different test cases (CA: colloidal aggregation model; NB: network of bimolecular reactions; LC: linear chain model; HSR: heat-shock response model). The sizes of all major data structures (\mathbf{c} and \mathbf{a} are the arrays of specific probability rates and reaction propensities, respectively; $\boldsymbol{\nu}$ is the stoichiometry matrix; see Sec. 3.3 for other definitions) as well as the total memory requirements are given as determined analytically for all benchmark simulations. SPDM and SDM need additional memory of size $M + N$ and M , respectively, for the reordered index lists. This, however, does not change the overall scaling of the total memory requirements.

3.3.4.1 Strongly coupled reaction network: colloidal aggregation model

We use the colloidal aggregation model (von Smoluchowski, 1917; van Dongen and Ernst, 1987; van Dongen, 1987; Axford, 1996; Turner et al., 2005) as a first example of a strongly coupled reaction network. The reaction network of the colloidal aggregation model is defined by:



For an even number of species N , the partial-propensity structure for this network is:

$$\Pi = \left\{ \begin{array}{l} \Pi_0 = (\emptyset) \\ \Pi_1 = \left(c_{1,1} \frac{n_1-1}{2} \quad c_{1,2} n_2 \quad c_{1,3} n_3 \quad \dots \quad c_{1,\frac{N}{2}} n_{\frac{N}{2}} \quad \dots \quad c_{1,N-1} n_{N-1} \right) \\ \Pi_2 = \left(\bar{c}_{2,1} \quad c_{2,2} \frac{n_2-1}{2} \quad c_{2,3} n_3 \quad \dots \quad c_{2,\frac{N}{2}} n_{\frac{N}{2}} \quad \dots \quad c_{2,N-2} n_{N-2} \right) \\ \vdots \\ \Pi_{\frac{N}{2}} = \left(\bar{c}_{\frac{N}{2},1} \quad \bar{c}_{\frac{N}{2},2} \quad \dots \quad \bar{c}_{\frac{N}{2},\frac{N}{4}} \quad c_{\frac{N}{2},\frac{N}{2}} n_{\frac{N}{2}} \right) \\ \vdots \\ \Pi_N = \left(\bar{c}_{N,1} \quad \bar{c}_{N,2} \quad \bar{c}_{N,3} \quad \dots \quad \dots \quad \bar{c}_{\frac{N}{2},\frac{N}{2}} \right). \end{array} \right. \quad (3.8)$$

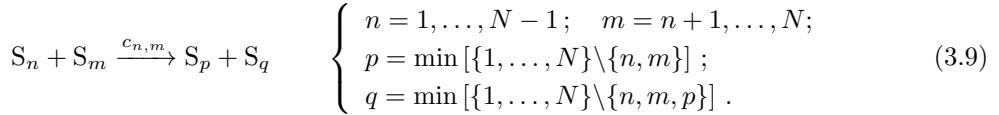
For odd N , the structure looks similar. This reaction network can be used to model, e.g., colloidal aggregation of solvated proteins, nano-beads, or viruses. For N chemical species it consists of $M = \left\lfloor \frac{N^2}{2} \right\rfloor$ reactions and the maximum out-degree of the dependency graph, i.e., the degree of coupling d_c of the network is $3N - 7$ and hence scales with system size (see Table 3.3).

The colloidal aggregation model is simulated up to time $T = 100$ with specific probability rates $c_{n,m} = 1$ and $\bar{c}_{p,q} = 1$. At time $t = 0$, $n_i = N\delta_{1,i}$. The scaling of Θ for PDM, SPDM, and SDM with system size is shown in Fig. 3.2a, averaged over 100 independent runs. Θ^{PDM} and Θ^{SPDM} are $O(N^{0.5})$ for small N (less than about 100) and $O(N)$ for large N . Θ^{SDM} is $O(N^2)$. The pre-factor of Θ^{SPDM} is similar to that of Θ^{PDM} , since in this network \mathcal{C}_μ is not significantly reduced by the dynamic sorting (Table 3.4). The memory requirements of PDM and SPDM are $O(N^2) = O(M)$, that of SDM is $O(N^3) = O(NM)$ (Table 3.5).

In summary, the computational costs of both PDM and SPDM are $O(N)$. This scaling is mediated by all three cost components. The use of partial propensities renders the scaling of the sampling cost \mathcal{C}_μ $O(N)$ (see Table 3.4). The cost \mathcal{C}_P for updating the partial propensities is $O(N^{0.5})$ (Table 3.4), since the use of partial propensities allows formulating a dependency graph over species, rather than reactions, and unimolecular reactions have constant partial propensities. This leads to a smaller number of updates needed as shown in Fig. 3.3a.

3.3.4.2 Strongly coupled network of bimolecular reactions

The following hypothetical network of bimolecular reactions:



consists of $M = \frac{N}{2}(N-1)$ strongly coupled bimolecular reactions, such that none of the partial propensities are constant. The partial-propensity structure for this reaction network is:

$$\Pi = \begin{cases} \Pi_0 = (\emptyset) \\ \Pi_1 = (c_{1,2}n_2 \quad c_{1,3}n_3 \quad c_{1,4}n_4 \quad \dots \quad c_{1,N}n_N) \\ \Pi_2 = (c_{2,3}n_3 \quad c_{2,4}n_4 \quad c_{2,5}n_5 \quad \dots \quad c_{2,N}n_N) \\ \vdots \\ \Pi_{N-1} = (c_{N-1,N}n_N) \\ \Pi_N = (\emptyset). \end{cases} \quad (3.10)$$

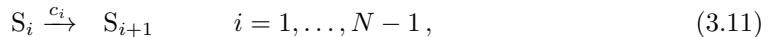
Both the minimum and the maximum out-degrees of the dependency graph in this case are $4N-10$, scaling faster with N than in the colloidal aggregation model (see Table 3.3).

We simulate this network up to time $T = 0.001$ with all specific probability rates $c_i = 1$. At $t = 0$, $n_i = 100(\delta_{N-4,i} + \delta_{N-3,i} + \delta_{N-2,i} + \delta_{N-1,i} + \delta_{N,i})$. The scaling of Θ for PDM, SPDM, and SDM with system size is shown in Fig. 3.2b, averaged over 100 independent runs. Θ^{PDM} and Θ^{SPDM} are $O(N)$, whereas Θ^{SDM} is $O(N^2)$. The pre-factors of PDM and SPDM are comparable. The memory requirements of PDM and SPDM are $O(N^2) = O(M)$, that of SDM is $O(N^3) = O(NM)$ (see Table 3.5).

In summary, the computational costs of PDM and SPDM are $O(N)$ for this strongly coupled, purely bimolecular network. The scaling is again mediated by all three cost components. Grouping the partial propensities renders the sampling cost $\mathcal{C}_\mu O(N)$ (see Table 3.4). Because none of the partial propensities are constant, the update costs \mathcal{C}_P of PDM and SPDM are $O(N)$, as in SDM, albeit with a pre-factor that is ≈ 2.5 times smaller than that in SDM. One reason for this smaller pre-factor is the smaller number of updates needed upon reactions firing, as shown in Fig. 3.3(b). This is due to the fact that partial propensities of bimolecular reactions depend on the population of only one species, which reduces the number of combinations that need to be updated.

3.3.4.3 Weakly coupled reaction network: linear chain model

We benchmark PDM and SPDM on a weakly coupled model in order to assess their limitations in cases where other SSA formulations might be more efficient. We choose the following linear chain model



since it is the most weakly coupled reaction network possible, and it has been used as a model for isolated signal transduction networks (Albert, 2005). For M reactions, it involves

3.3 The partial-propensity direct method (PDM)

the number of species $N = M + 1$, and the maximum out-degree of the dependency graph is constant at the minimum possible value of 2 (see Table 3.3), since every reaction at most influences the population of its only reactant and of the only reactant of the subsequent reaction.

The partial-propensity structure of the linear chain model is given by:

$$\Pi = \begin{cases} \Pi_0 = (\emptyset) \\ \Pi_1 = (c_1) \\ \Pi_2 = (c_2) \\ \vdots \\ \Pi_{N-1} = (c_{N-1}) \\ \Pi_N = (\emptyset). \end{cases} \quad (3.12)$$

We simulate the linear chain model to a final time of $T = 1000$ with all specific probability rates $c_i = 1$. At time $t = 0$, $n_i = 10000\delta_{1,i}$. Figure 3.2c presents the scaling of the CPU time with system size for PDM, SPDM, and SDM, averaged over 100 independent runs. Θ^{PDM} scales linearly with N and Θ^{SPDM} with $N^{0.5}$. Θ^{SDM} is $O(N)$ with a pre-factor that is more than 4 times larger than that of Θ^{PDM} . This difference in pre-factor is mainly caused by PDM having smaller C_n and C_p (Table 3.4). C_μ , however, scales worse for PDM than for SDM due to the dynamic sorting in SDM. This is overcome in SPDM, where C_μ is $O(N^{0.5})$, as in SDM. The memory requirements of SPDM and PDM are $O(N) = O(M)$, that of SDM is $O(N^2) = O(NM)$ (Table 3.5).

In summary, the computational costs of PDM and SPDM on the weakly coupled linear chain model are governed by (a) updating the population \mathbf{n} using a sparse stoichiometry representation and (b) never needing to update the partial propensities of unimolecular reactions. Since the linear chain model contains only unimolecular reactions, none of the partial propensities ever need to be updated, leading to an update cost of $C_p = 0$ (see Table 3.4). While we have implemented SDM according to the original publication (McCollum et al., 2006), we note that if one uses a sparse representation of the stoichiometry matrix also in SDM, point (a) vanishes and $C_n = 2$ also for SDM. A sparse-stoichiometry SDM would thus have the same scaling of the computational cost on the linear chain model as SPDM, outperforming PDM.

3.3.4.4 Multi-scale biological network: heat-shock response in *Escherichia coli*

We assess the performance of PDM and SPDM on a small, fixed-size multiscale reaction network. We choose the heat-shock response model since it has also been used to benchmark previous methods, including ODM (Cao et al., 2004) and SDM (McCollum et al., 2006). The heat-shock response model (Kurata et al., 2001) was obtained from Dr. Hong Li and Prof. Linda Petzold (UCSB) and is publicly available as part of the StochKit package (Li et al., 2008). The model describes one of the mechanisms used by the bacterium *E. coli* to protect itself against a variety of environmental stresses that are potentially harmful to the structural integrity of its proteins. The heat-shock response (HSR) system reacts to this by rapidly synthesizing heat-shock proteins. The heat-shock sigma factor protein σ^{32} activates the HSR by inducing the transcription of heat-shock genes. The heat-shock response model is a small multiscale

3 Partial-propensity formulations of the stochastic simulation algorithm

reaction network (the specific probability rates span 8 orders of magnitude) with $N = 28$ chemical species, $M = 61$ reactions, and a maximum out-degree of the dependency graph of 11 (see Table 3.3). For a detailed description of the model, we refer to Kurata et al. (Kurata et al., 2001)

We simulate the HSR model for $T = 500$ seconds. During this time, approximately 46 million reactions are executed. For a single run, we measure $\Theta^{\text{PDM}} = 0.256 \mu\text{s}$ and $\Theta^{\text{SDM}} = 0.272 \mu\text{s}$. This corresponds to a simulated 3.68 million reactions per second of CPU time for SDM and 3.89 million reactions per second for PDM. Hence, PDM is about 6% faster than SDM. This speed-up is mainly due to a smaller \mathcal{C}_P in PDM (see Fig. 3.3(c) for the distribution of updates over all reactions) since the partial propensities of unimolecular reactions never need to be updated. The speed-up, however, is modest because \mathcal{C}_μ of PDM is ≈ 4.6 times larger than that of SDM (Table 3.4). This is due to the fact that 95% of all reaction firings are caused by a small subset of only 6 reactions. This multiscale network thus strongly benefits from the dynamic sorting used in SDM. This advantage is recovered in SPDM, where \mathcal{C}_μ is comparable to that of SDM, and $\Theta^{\text{SPDM}} = 0.245 \mu\text{s}$ (4.08 million reactions per second). This makes SPDM 11% faster than SDM on this small network.

3.3 The partial-propensity direct method (PDM)

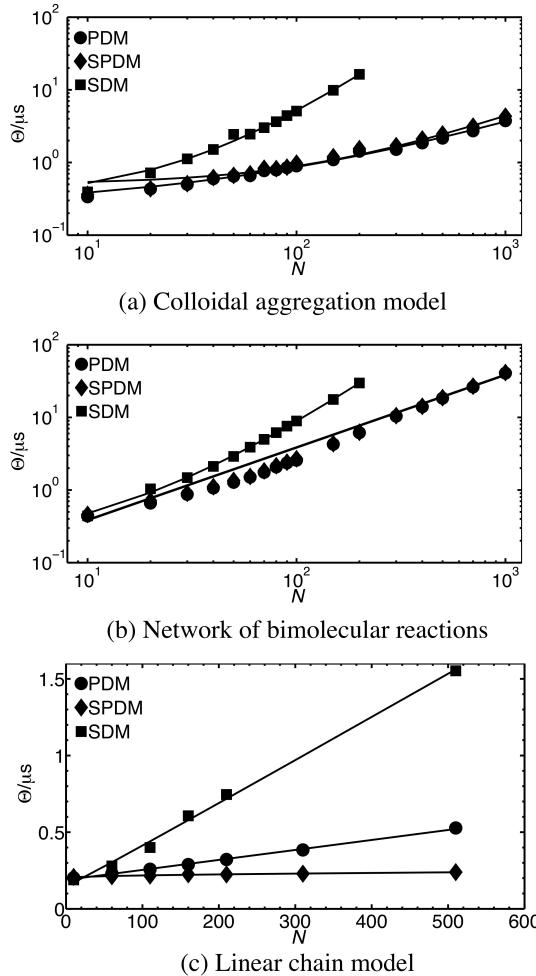


Figure 3.2: Computational costs of PDM (circles), SPDM (diamonds), and SDM (squares). See main text for the simulation parameters and initial conditions used. The average CPU time per reaction (i.e. per time step), Θ , is shown as a function of system size quantified by the number of species N . Θ is defined as the CPU time needed to simulate the system up to final time T , divided by the number of reactions executed during this time, and averaged over 100 independent runs (error bars are smaller than symbol size). The solid lines are the corresponding least-squares fits of the scaling $\Theta(N)$ of PDM, SPDM, and SDM with the model $aC_\mu + bC_n + cC_P$ on a linear scale (see Table 3.4), where a , b , and c are the fitted constants. (a) Logarithmic plot of the results for the colloidal aggregation model. The fits are: $\Theta^{\text{PDM}}/\mu\text{s} = 0.0022N + 0.050N^{0.5} + 0.22$, $\Theta^{\text{SPDM}}/\mu\text{s} = 0.0027N + 0.053N^{0.5} + 0.20$, and $\Theta^{\text{SDM}}/\mu\text{s} = 0.00031N^2 + 0.018N + 0.31$. (b) Logarithmic plot of the results for the network of bimolecular reactions. The fits are: $\Theta^{\text{PDM}}/\mu\text{s} = 0.038N$, $\Theta^{\text{SPDM}}/\mu\text{s} = 0.039N$, and $\Theta^{\text{SDM}}/\mu\text{s} = 0.00061N^2 + 0.027N + 0.15$. (c) Linear plot of the results for the linear chain model. The fits are: $\Theta^{\text{PDM}}/\mu\text{s} = 0.00065N + 0.19$, $\Theta^{\text{SPDM}}/\mu\text{s} = 0.0015N^{0.5} + 0.20$, and $\Theta^{\text{SDM}}/\mu\text{s} = 0.0029N - 0.0025N^{0.5} + 0.15$. In all cases, the computational cost $\Theta(N)$ of PDM and SPDM is $O(N)$.

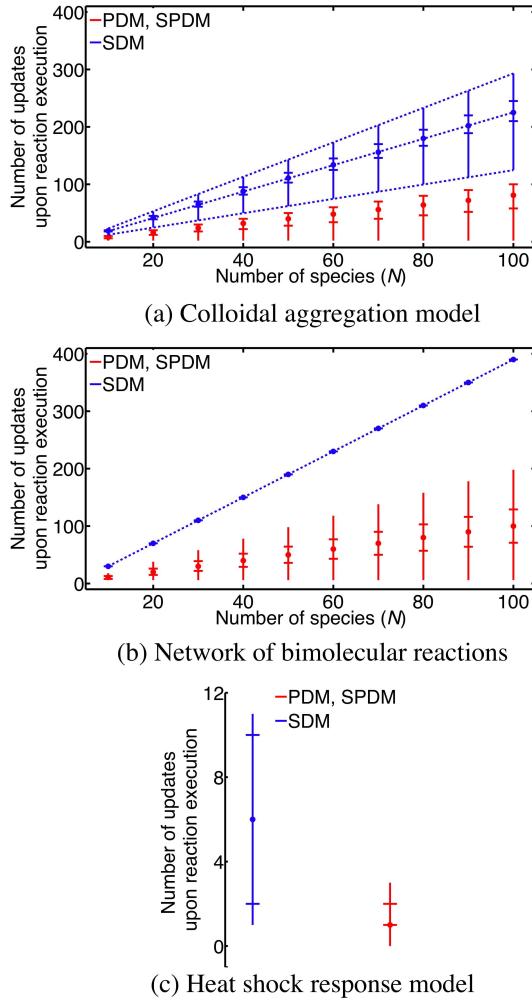


Figure 3.3: Measured distributions of the number of partial propensities (for PDM and SPDM, red line) or propensities (for SDM, blue line) that need to be updated after firing any reaction of: (a) the colloidal aggregation model, (b) the network of bimolecular reactions, and (c) the heat-shock response model. Dots indicate medians, horizontal bars the upper and lower quartiles, and vertical bars the upper and lower extrema (maximum and minimum). The dotted lines denote the minimum, average and maximum degree of coupling d_c of the reaction networks (see Table 3.3). The number of updates in SDM (McCollum et al., 2006) using a dependency graph is governed by the degree of coupling of the network. In PDM and SPDM, less updates need to be performed since partial propensities depend on the population of at most one species and are constant for unimolecular reactions.

3.3.5 Conclusions

When simulating weakly coupled reaction networks, where the maximum number of reactions that are influenced by any reaction is constant with system size, the best computational cost of previous exact SSAs for the sampling step is $O(\log_2 M)$ (Gibson and Bruck, 2000), where M is the total number of reactions, or $O(1)$ under some additional conditions on the propensity distribution (Slepoy et al., 2008). Using dependency graphs, also the update step has been reduced to $O(1)$ for weakly coupled networks (Cao et al., 2004; McCollum et al., 2006; Slepoy et al., 2008). For strongly coupled reaction networks, where the degree of coupling increases with system size and can be as large as the total number of reactions, all previous exact SSA formulations have a computational cost that is $O(M)$.

We have introduced a new quantity called *partial propensity* and have used it to construct two novel formulations of the exact SSA: PDM and its sorting variant SPDM. Both are algebraically equivalent to DM and yield the same population trajectories $\mathbf{n}(t)$ as to those produced by DM. In our formulation of partial propensities, we have limited ourselves to elementary chemical reactions. Since their partial propensities depend on the population of at most one species, both new SSA formulations have a computational cost that scales at most linearly with the number of species rather than the number of reactions, independently of the degree of coupling. This is particularly advantageous in strongly coupled reaction networks, where the number of reactions M grows faster than the number of species N with system size. For networks of fixed size, PDM and SPDM are especially efficient when $M \gg N$. PDM's computational cost is $O(N)$, which is made possible by appropriately grouping the partial propensities in the sampling step and formulating a dependency graph over species rather than reactions in the update step. Moreover, the partial propensities of unimolecular reactions and source reactions are constant and never need to be updated. This further reduces the size of the dependency graph and the computational cost of the update step. To our knowledge, PDM is the first SSA formulation that has a computational cost that is $O(N)$, irrespective of the degree of coupling of the reaction network. In the case of multiscale networks, the absolute computational cost of SPDM is smaller than that of PDM.

We have benchmarked PDM and SPDM on four test cases with various degrees of coupling. The first two benchmarks considered strongly coupled networks, where the degree of coupling scales proportionally to the number of species. The third benchmark considered the most weakly coupled network possible, where several other SSA formulations might be more efficient. Finally, the fourth benchmark considered a small biological multiscale network. These benchmarks allowed estimating the scaling of the computational cost with system size and the cost contributions from reaction sampling, population update, and partial-propensity update. The results showed that (a) the overall computational costs of PDM and SPDM are $O(N)$, even for strongly coupled networks, (b) on very weakly coupled networks, SPDM is competitive compared to SDM, (c) on multiscale networks SPDM outperforms PDM, and (d) the memory requirements of PDM and SPDM are $O(M)$ in all cases, and hence not larger than those of any other exact SSA formulation.

PDM and SPDM, however, have a number of limitations. The most important limitation is that the presented formulation of partial propensities is only applicable to elementary chemical reactions. Higher-order chemical reaction can be broken down into elementary reactions at the expense of increasing system size (see Appendix A2). In applications such as population

3 Partial-propensity formulations of the stochastic simulation algorithm

ecology or social science, the idea of partial propensities can, however, only be used if the (generalized) reactions are at most binary and one species can be factored out, i.e., if the propensity for every reaction between species S_i and S_j can be written as $a_\mu = c_\mu n_i \tilde{h}(n_j)$. Besides this structural limitation, the computational performance of the particular algorithms presented here can be limited in several situations. One of them is the simulation of very small networks, where the overhead of the data structures involved in PDM and SPDM may not be amortized by the gain in efficiency and a simulation using DM may be more efficient. In multiscale networks, where the propensities span several orders of magnitude, PDM is slower than SPDM. In multiscale networks where a small subset ($\ll N$) of all reactions accounts for almost all of the reaction firings, however, the overhead of the data structures involved in SPDM, including their initialization, may not be amortized by the gain in efficiency. Finally, PDM and SPDM were designed to have a computational cost that scales linearly with the number of species rather than the number of reactions. For reaction networks in which the number of reactions grows sub-linearly with the number of species, this becomes a disadvantage. In such cases, SSA formulations that scale with the number of reactions are favorable.

Taken together, our results suggest that PDM and SPDM can potentially offer significant performance improvements especially in strongly coupled networks, including the simulation of colloidal aggregation (von Smoluchowski, 1917; van Dongen and Ernst, 1987; van Dongen, 1987; Axford, 1996; Turner et al., 2005), Becker-Döring-like nucleation-and-growth reactions (Wattis, 2009), and scale-free biochemical reaction networks, where certain hubs are strongly coupled (Jeong et al., 2000; Strogatz, 2001; Albert and Barabási, 2002; Albert, 2005).

3.4 The partial-propensity SSA with composition-rejection sampling (PSSA-CR)

PDM has a computational cost of $O(N)$ irrespective of the degree of coupling of the reaction network. However, in practice, and especially for networks of fixed size, it is often difficult to determine which coupling class a reaction network belongs to. This is because the coupling class is defined as a function of network size. For fixed-size systems, however, only a single point of that function is known, requiring additional knowledge to determine the coupling class. There is thus a need for an exact SSA that combines the favorable scaling of the computational cost of SSA-CR (Sec. 2.3.7) for weakly coupled networks and of PDM for strongly coupled ones. Here, we use the concept of partial propensities (see Sections 3.1 and 3.2) to construct a partial-propensity variant of SSA-CR, called the partial-propensity SSA with composition-rejection sampling (PSSA-CR). We show that PSSA-CR has a computational cost of $O(1)$ for weakly coupled networks and $O(N)$ for strongly coupled networks, thus combining the advantages of PDM and SSA-CR.

The partial-propensity SSA with composition-rejection sampling (PSSA-CR) is based on the idea of factorizing the reaction propensities into partial-propensities, grouping and binning them, and using composition-rejection (CR) sampling (Devroye, 1986; Slepoy et al., 2008) to determine the index of the next reaction (see Sec. 2.3.7 for the principle behind CR sampling). PSSA-CR reduces the computational cost for weakly coupled reaction networks to $O(1)$ under the assumption that the ratio of maximum to minimum non-zero propensity is bounded by a constant. It achieves this superior scaling for weakly coupled networks while maintaining the

3.4 The partial-propensity SSA with composition-rejection sampling (PSSA-CR)

computational cost for strongly coupled reaction networks at $O(N)$.

3.4.1 Detailed description

PSSA-CR uses a composition-rejection sampling strategy over partial propensities in order to sample the index of the next reaction. Since every reaction in a partial-propensity method is identified by its group index and its element index, we apply two composition-rejection steps: one to sample the group index and one to sample the element index. Table 3.6 gives an overview of PSSA-CR. The individual steps are described in detail below.

The principle data structures in PSSA-CR are the same as in PDM. The partial propensities are stored in a partial-propensity structure $\boldsymbol{\Pi} = \{\boldsymbol{\Pi}_i\}_{i=0}^N$ as a one-dimensional array of one-dimensional arrays. The reaction indices μ corresponding to a certain entry in $\boldsymbol{\Pi}$ are stored in a look-up table $\mathbf{L} = \{\mathbf{L}_i\}_{i=0}^N$, making every reaction μ identifiable by its group index I and its element index J as $\mu = \mathbf{L}_{I,J}$. The group-sum array $\boldsymbol{\Lambda}$ stores the sums of the partial propensities in each group $\boldsymbol{\Pi}_i$, i.e. $\Lambda_i = \sum_j \Pi_{i,j}$. We also store the total propensity of each group in an array $\boldsymbol{\Sigma}$, computed as $\Sigma_i = n_i \Lambda_i$, $i = 1, \dots, N$, and $\Sigma_0 = \Lambda_0$. See Sec. 3.3 for more details on the data structures.

In PSSA-CR, the entries of $\boldsymbol{\Sigma}$ are then sorted into $G_\Sigma = \log_2 \frac{\Sigma_{\max}}{\Sigma_{\min}} + 1$ bins such that bin b contains all Σ_i 's with $2^{b-1}\Sigma_{\min} \leq \Sigma_i < 2^b\Sigma_{\min}$. Σ_{\min} and Σ_{\max} are the smallest and largest non-zero values in $\boldsymbol{\Sigma}$ that can possibly occur during a simulation. They are determined as outlined below. The total propensity of each bin b , $\sigma_b^{(\Sigma)}$, is computed by summing up the Σ_i 's in that bin. Similarly, the entries of each $\boldsymbol{\Pi}_i$ are sorted into $G_{\Pi_i} = \log_2 \frac{\Pi_{i,\max}}{\Pi_{i,\min}} + 1$ bins with bin b containing all elements in $\boldsymbol{\Pi}_i$ with $2^{b-1}\Pi_{i,\min} \leq \Pi_{i,j} < 2^b\Pi_{i,\min}$. $\Pi_{i,\min}$ and $\Pi_{i,\max}$ are the smallest and largest non-zero values in $\boldsymbol{\Pi}_i$ that can possibly occur during a simulation. The total partial propensity of bin b is stored in $\sigma_b^{(\Pi_i)}$. The $\Pi_{i,\min}$'s and Σ_{\min} can always be computed *a priori*. $\Pi_{i,\min}$ is the minimum non-zero value in $\boldsymbol{\Pi}_i$ when all partial propensities are calculated with one molecule of each reactant. Σ_{\min} is the minimum among all $n_i \Pi_{i,\min}$'s, where n_i is the population of species S_i used to calculate $\Pi_{i,\min}$. Estimating the $\Pi_{i,\max}$'s and Σ_{\max} *a priori* may be possible using prior knowledge about the chemical reaction network, such as physical constraints. In cases where the $\Pi_{i,\max}$'s and Σ_{\max} cannot be estimated *a priori*, PSSA-CR dynamically updates the $\Pi_{i,\max}$'s and Σ_{\max} over the course of the simulation. If this increases any G_{Π_i} or G_Σ , the corresponding data structures are dynamically enlarged.

We apply the composition-rejection sampling strategy (Devroye, 1986; Slepoy et al., 2008) to obtain the group index I and the element index J of the next reaction μ . The group index I is sampled in two steps: (1) the composition step to find the bin b_I and (2) the rejection step to find Σ_I inside that bin. The composition step is done by linear search, thus:

$$b_I = \min \left[b : r_1 a < \sum_{i=1}^b \sigma_i^{(\Sigma)} \right], \quad (3.13)$$

where a is the total propensity of all reactions in the network and r_1 is a uniform random number in $[0, 1]$. The rejection step samples the group-index I from the elements in bin b_I . For this step, we generate a uniformly distributed random number r_2 in $[0, 2^{b_I}\Sigma_{\min}]$ and a uniformly distributed random integer r_3 between 1 and the number of elements in bin b_I . If

3 Partial-propensity formulations of the stochastic simulation algorithm

the r_3^{th} Σ_i in bin b_I is less than r_2 , the index of that Σ_i is chosen as the group-index I . If this inequality is not satisfied, the rejection step is repeated. This is illustrated in Fig. 3.4 for an example with 6 partial-propensity groups. Assume that, in this example, the composition step has selected bin $b_I = 2$ as the one containing Σ_I . The rejection step then samples uniformly random points inside the rectangle defining this bin's value range (bold rectangle). A sample is accepted if it falls inside one of the bars representing the Σ_i 's. If the first sample (point A in Fig. 3.4 with $r_3 = 2$ and $r_2 > \Sigma_4$) is rejected, sampling is repeated until the point falls inside one of the bars (point B in Fig. 3.4 with $r_3 = 1$ and $r_2 < \Sigma_0$). This determines the group index of the next reaction ($I = 0$ in the example in Fig. 3.4). By binning the Σ_i 's as described, we ensure that the area covered by the Σ_i bars in any bin is at least 50% of the total area of the bin's bounding rectangle. The expected number of iterations of the rejection sampling is hence less than or equal to two.

In order to sample the element index J , the same composition-rejection procedure is also applied within the identified group I . The composition step again involves a linear search for the bin b_J containing the partial propensity of the next reaction, as:

$$b_J = \min \left[b : r_4 \Lambda_I < \sum_{i=1}^b \sigma_i^{(\Pi_I)} \right], \quad (3.14)$$

where r_4 is a uniform random number in $[0,1)$. The rejection step as described above is subsequently used to find the element index J from a uniformly distributed random number r_5 in $[0, 2^{b_J} \Pi_{I,\min})$ and a uniformly distributed random integer r_6 between 1 and the number of elements in bin b_J . In the example in Fig. 3.4, the group index $I = 0$ has been selected. Assume that the composition step for the element index J has selected bin $b_J = 2$ in the group Π_0 . Rejection sampling in this bin is then repeated until a point inside any of the bars representing the partial propensities $\Pi_{0,j}$ is selected (point C in Fig. 3.4 with $r_6 = 2$ and $r_5 < \Pi_{0,1}$). This determines the element index of the next reaction ($J = 1$ in the example in Fig. 3.4). The indices I and J of the next reaction are then translated to the reaction index μ using the look-up table, hence $\mu = L_{I,J}$.

To execute a sampled reaction, \mathbf{n} , $\mathbf{\Pi}$, $\mathbf{\Lambda}$, and $\mathbf{\Sigma}$ are updated using the same update algorithm and data structures as in PDM (see Sec. 3.3):

$\mathbf{U}^{(1)}$ is a array of M arrays, where the i^{th} array contains the indices of all species involved in the i^{th} reaction.

$\mathbf{U}^{(2)}$ is a array of M arrays containing the corresponding stoichiometry (the change in population of each species upon reaction) of the species stored in $\mathbf{U}^{(1)}$.

$\mathbf{U}^{(3)}$ is a array of N arrays, where the i^{th} array contains the indices of all entries in $\mathbf{\Pi}$ that depend on n_i .

After each reaction, we use $\mathbf{U}^{(1)}$ to determine the indices of all species involved in this reaction. The stoichiometry is then looked up in $\mathbf{U}^{(2)}$ and the population \mathbf{n} is updated accordingly. Subsequently, $\mathbf{U}^{(3)}$ is used to locate the affected entries in $\mathbf{\Pi}$ and recompute them. Since the partial propensities of unimolecular and source reactions are constant and need never be updated, $\mathbf{U}^{(3)}$ only contains the indices of the partial propensities of bimolecular reactions.

3.4 The partial-propensity SSA with composition-rejection sampling (PSSA-CR)

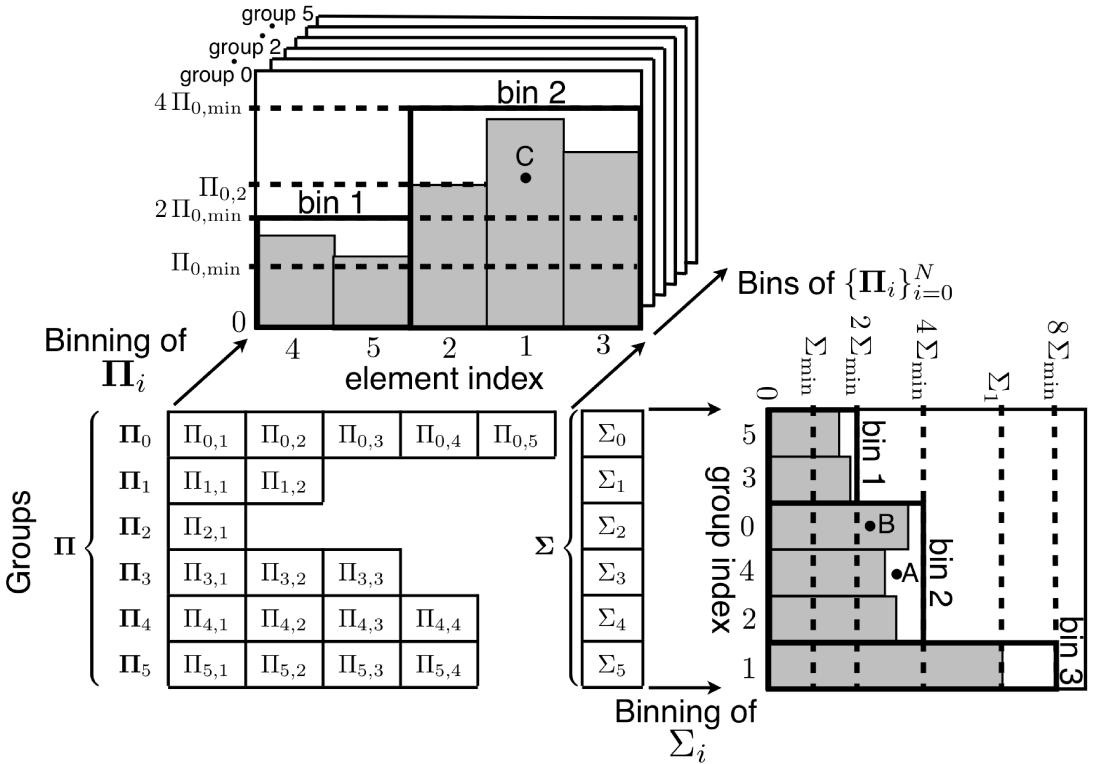


Figure 3.4: Illustration of the two composition-rejection sampling steps used in PSSA-CR. The example shown is for a network with $N = 5$ species and $M = 19$ reactions. The partial propensities are grouped into 6 ($=N+1$) groups in the partial-propensity structure $\{\Pi\}_{i=0}^N$. The sum of propensities in group Π_i is stored in Σ_i . The elements of Σ and of each Π_i are sorted into dyadic bins. The shaded bars represent the values of the corresponding entries. The extent of each bin is shown by a bold rectangle. Due to the dyadic binning, the bars always cover at least 50% of the area of any bin's rectangle. In order to sample the index of the next reaction, two composition-rejection sampling steps are used: one for the group index I and another one for the element in index J in Π . Points A, B, and C refer to the example given in the main text.

3 Partial-propensity formulations of the stochastic simulation algorithm

After updating the partial propensities, the bin memberships of all modified $\Pi_{i,j}$'s and Σ_i 's need to be updated. This requires locating the bin assignment of any $\Pi_{i,j}$ and Σ_i in a one-step operation. We implement this by having every $\Pi_{i,j}$ and Σ_i store two additional integers: the bin membership and the location inside that bin. Depending on their new value, the changed $\Pi_{i,j}$'s and Σ_i 's are kept inside the same bin or moved to a different bin. Then, the corresponding bin sums are updated by adding the total change. This can be done in $O(1)$ operations since the ordering of elements in a bin does not matter. Elements that are removed from a bin are simply replaced by the last element in that bin, which is then removed.

The computational cost of PSSA-CR is $O(G_\Sigma + \max\{G_{\Pi_0}, \dots, G_{\Pi_N}\} + N)$ for strongly coupled reaction networks and $O(G_\Sigma + \max\{G_{\Pi_0}, \dots, G_{\Pi_N}\})$ for weakly coupled ones (see Sec. 3.4.2). If the dynamic range of propensities is bounded over the time of a simulation, the computational cost on weakly coupled networks reduces to $O(1)$ (see Sec 3.4.2).

The memory requirement of PSSA-CR is larger than that of PDM. In addition to the data structures required in PDM, PSSA-CR needs an additional $O(N + M)$ memory for the binning of the Σ_i 's and $\Pi_{i,j}$'s. This renders the memory requirement of PSSA-CR $O(N + M)$, which is equivalent to $O(M)$ since M usually scales faster than N for large reaction networks.

1. Initialize the data structures. Set time $t \leftarrow 0$.
2. While $t < t_f$, where t_f is the final simulation time, repeat:
 - 2.1. Sample the group index I using composition-rejection sampling.
 - 2.2. Sample the element index J using composition-rejection sampling.
 - 2.3. Read the index of the reaction identified by the group index I and the element index J from the look-up table \mathbf{L} .
 - 2.4. Compute the time to the next reaction $\tau \leftarrow a^{-1} \ln(r^{-1})$, where a is the total propensity of all reactions and r a uniformly distributed random number in $[0, 1)$.
 - 2.5. Update the population of species and the partial propensity structure using the dependency graph over species. Update the bin assignments of changed partial propensities.
 - 2.6. Increment time: $t \leftarrow t + \tau$.
3. Stop.

Table 3.6: Overview of PSSA-CR.

3.4.2 Computational cost

The computational cost of PSSA-CR is determined by the sampling and update steps of the algorithm. Composition-rejection sampling of the group-index I has a cost that is $O(G_\Sigma)$. This is because (a) the composition step involves a linear search over at most G_Σ elements and (b) the computational cost of the rejection step is $O(1)$ with increasing network size. The

3.4 The partial-propensity SSA with composition-rejection sampling (PSSA-CR)

reasoning is as follows: The present binning strategy ensures that at least $1/2$ of the area of each bin is covered by the Σ_i 's in that bin (see Fig. 3.4). Therefore, the probability of acceptance after i iterations of rejection sampling is $1 - 2^{-i}$. This is independent of network size and hence the rejection step is $O(1)$. The probability of acceptance is at least 0.9999 after 13 iterations of the rejection sampling step*. Likewise, the computational cost of the composition-rejection sampling of the element index J is $O(\max\{G_{\Pi_0}, \dots, G_{\Pi_N}\})$.

The computational cost of the update step is $O(N)$ like in PDM, albeit with a larger prefactor due to additional overhead associated with reassigning bin memberships. In summary, the total computational cost of PSSA-CR is $O(G_\Sigma + \max\{G_{\Pi_0}, \dots, G_{\Pi_N}\} + N)$.

For weakly coupled reaction networks, the update step becomes $O(1)$, since the number of entries in Π that need to be updated is independent of system size. This reduces the computational cost of PSSA-CR for weakly coupled networks to $O(G_\Sigma + \max\{G_{\Pi_0}, \dots, G_{\Pi_N}\})$. In addition, if Σ_{\max} and $\Pi_{i,\max}$ are bounded for all i , the number of bins $G_\Sigma = \log_2 \frac{\Sigma_{\max}}{\Sigma_{\min}} + 1$ and $G_{\Pi_i} = \log_2 \frac{\Pi_{i,\max}}{\Pi_{i,\min}} + 1$ are also bounded. This renders the computation cost of PSSA-CR $O(1)$ for weakly coupled networks that have a bounded dynamic range of propensities. Even if G_Σ and G_{Π_i} are not bounded by a constant it is unlikely that they equal to N irrespective of system size. This would require requires that the ratios $\frac{\Sigma_{\max}}{\Sigma_{\min}}$ and $\frac{\Pi_{i,\max}}{\Pi_{i,\min}}$ scale proportionally to 2^N . In practice we observe that G_Σ and G_{Π_i} scale only weakly with increasing system size N . We present this empirical evidence for the weakly coupled cyclic chain model in Sec. 3.4.3.1.

3.4.3 Benchmarks

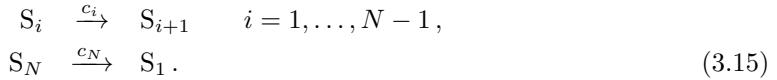
We benchmark the computational performance of PSSA-CR on both a weakly coupled and a strongly coupled reaction network. We choose the cyclic chain model (Cao et al., 2004; Heuett and Qian, 2006) and the colloidal aggregation model (von Smoluchowski, 1917; van Dongen and Ernst, 1987; van Dongen, 1987; Axford, 1996; Turner et al., 2005) as representative networks, respectively. We compare the performance of PSSA-CR with that of SDM, the sorting direct method (McCollum et al., 2006), and SPDM, the analogous sorting variant of PDM (see Sections 3.3 and 3.3.2).

All tested SSA formulations are implemented in C++ using the random number generator of the GSL library and compiled using the Intel C++ compiler version 11.1 with the O3 optimization flag. All timings are measured on a Linux 2.6 workstation with a 2.8 GHz quad-core Intel Xeon E5462 processor, 8 GB of memory and 4 MB L2 cache. For all test cases, we simulate the reaction network until 10^7 reactions have been executed and report the average CPU time Θ per reaction. All simulations are run without any *a priori* estimate of the $\Pi_{i,\max}$'s and Σ_{\max} . Instead, the $\Pi_{i,\max}$'s and Σ_{\max} are constantly updated over the course of a simulation and the number of bins is dynamically increased when necessary.

*The number $1 - 2^{-i}$ evaluates to 1 in the standard double-precision representation for $i = 51$.

3.4.3.1 A weakly coupled reaction network: Cyclic chain model

The cyclic chain model is given by the reaction network



For N chemical species, this network has $M = N$ reactions. The degree of coupling (maximum out-degree of the dependency graph) of this reaction network is 2, independently of system size.

At time $t = 0$, we set all $n_i = 1$ and all specific probability rates $c_i = 1$. Fig. 3.5A shows $\Theta(N)$ for PSSA-CR, SPDM, and SDM. As expected from the theoretical cost analysis, Θ is $O(1)$ for PSSA-CR and $O(N)$ for SPDM and SDM. PSSA-CR outperforms SPDM for N above a certain break-even point ($N > 700$ here; Fig. 3.5A) and is faster than SDM for all N tested. Below the break-even point, the overhead of the additional data structures and the binning involved in PSSA-CR is not amortized by the better scaling of the computational cost. The $O(1)$ scaling for PSSA-CR in this case is realized because the reaction network is weakly coupled (degree of coupling is independent of N) and all G_{Π_i} 's and G_Σ are constant with system size.

In order to test the efficiency of PSSA-CR for a weakly coupled reaction network with *increasing* number of bins, we simulate this test case with specific probability rates c_i randomly chosen between 1 and 10^6 from an exponential distribution. All other simulation parameters are unchanged. Fig. 3.5B shows the scaling of Θ for PSSA-CR, SPDM, and SDM. In this multi-scale case, G_Σ increases slowly with system size (by 2% over a 16-fold increase in N), leading to a very slow increase in Θ (proportional to $N^{0.028}$ in this case) of PSSA-CR, as predicted by the theoretical cost analysis. Nevertheless, PSSA-CR is more efficient than SPDM for N above a certain break-even point ($N > 500$ here; Fig. 3.5B) and more efficient than SDM for all N tested.

In summary, the measured computational cost of PSSA-CR is $O(1)$ for the cyclic chain model if the number of bins is bounded. If G_Σ or G_{Π_i} increase with system size, the computational cost is $O(G_\Sigma + \max\{G_{\Pi_0}, \dots, G_{\Pi_N}\})$, as shown in Sec. 3.4.2.

3.4.3.2 A strongly coupled reaction network: Colloidal aggregation model

The colloidal aggregation model is given in Eq. 3.7. For N chemical species, the number reactions is $M = \left\lfloor \frac{N^2}{2} \right\rfloor$. The degree of coupling of this reaction network is $3N - 7$ and hence scales with system size (Table 3.3).

At time $t = 0$, we set all $n_i = 1$ and all specific probability rates to 1. Fig. 3.5C shows $\Theta(N)$ for PSSA-CR, SPDM, and SDM. Θ is $O(N)$ for PSSA-CR and SPDM, and it is $O(N^2)$ for SDM. The Θ of PSSA-CR is always larger than that for SPDM. This constant offset is caused by the additional overhead of binning and bin reassessments in PSSA-CR, which is not necessary in SPDM. The break-even point of PSSA-CR with SDM is around $N > 160$. For systems larger than this, the extra overhead in PSSA-CR is amortized.

3.4 The partial-propensity SSA with composition-rejection sampling (PSSA-CR)

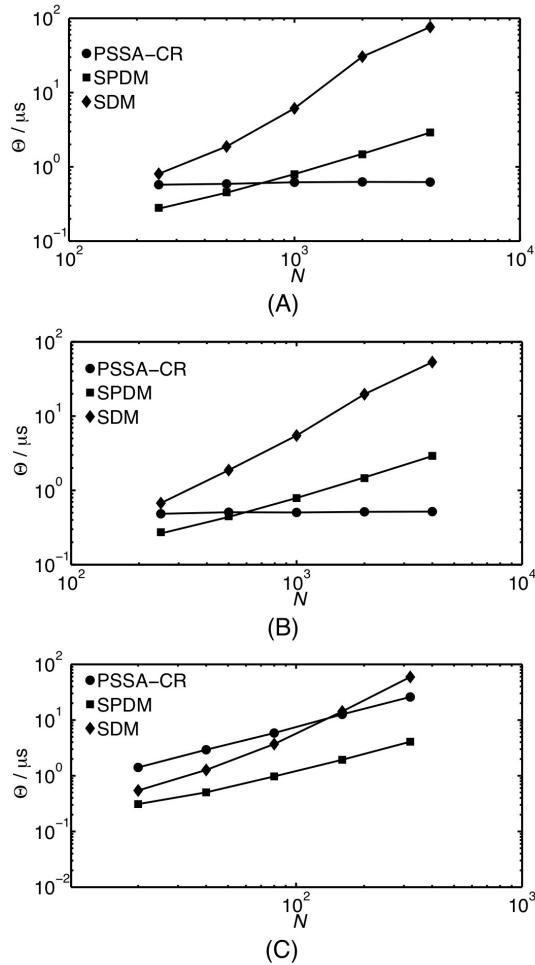


Figure 3.5: Computational cost of PSSA-CR (circles), SPDM (squares), and SDM (diamonds). The average CPU time Θ per reaction, averaged over 100 independent runs, is shown as a function of the number of species N . (A) $\Theta(N)$ for the weakly coupled cyclic chain model with bounded dynamic range of non-zero reaction propensities. Θ is $O(1)$ for PSSA-CR and $O(N)$ for SPDM and SDM. (B) $\Theta(N)$ for the weakly coupled cyclic chain model with increasing dynamic range of non-zero reaction propensities. $\Theta \propto N^{0.028}$ for PSSA-CR and $\Theta \propto N^1$ for SPDM and SDM. (C) $\Theta(N)$ for the strongly coupled colloidal aggregation model. Θ is $O(N)$ for both PSSA-CR and SPDM, whereas it is $O(N^2)$ for SDM.

3.4.4 Conclusions

We have introduced PSSA-CR, a partial propensity variant of the stochastic simulation algorithm with composition-rejection sampling (SSA-CR) (Slepoy et al., 2008). PSSA-CR uses two composition-rejection sampling steps over partial propensities in order to determine the index of the next reaction. Computational efficiency is achieved by grouping the partial propensities and using dyadic binning in the sampling.

PSSA-CR is an exact SSA formulation whose computational cost is $O(N)$ for strongly coupled reaction networks and $O(1)$ for weakly coupled networks with a bounded range of propensities. We have presented a theoretical cost analysis of PSSA-CR and benchmarked it on three prototypical test cases: (1) a non-stiff weakly coupled reaction network, (2) a multi-scale (stiff) weakly coupled reaction network, and (3) a strongly coupled reaction network. All benchmarks confirmed the theoretically predicted scaling of the computational cost.

PSSA-CR, however, inherits the limitations of PDM (Sec. 3.3.5) and of SSA-CR (Slepoy et al., 2008). For small networks, PSSA-CR is outperformed by other methods due to the additional overhead involved in the composition-rejection sampling. SSA formulations such as SDM (McCollum et al., 2006), NRM (Gibson and Bruck, 2000), SSA-CR (Slepoy et al., 2008), PDM, or SPDM (see Sections 3.3 and 3.3.2) might be more efficient here. In addition, PSSA-CR only achieves the $O(1)$ scaling for weakly coupled networks for which the ratio of maximum to minimum non-zero reaction propensity is bounded by a constant throughout a simulation.

To our knowledge, PSSA-CR has the best scaling of the computational cost on any class of reaction networks. This, however, does not imply that the actual computational cost of PSSA-CR is lowest in all cases, since the pre-factor depends on the data structures involved. If the coupling class of a particular network is not known in practice, however, PSSA-CR seems a reasonable choice for exact stochastic simulations of large reaction networks. Compared to other partial propensity methods, such as SPDM, the better computational scaling of PSSA-CR for weakly coupled networks is paid for by a larger pre-factor in the computational cost for strongly coupled networks.

3.5 The delay partial-propensity direct method (dPDM)

All the exact SSA formulations presented so far assume instantaneous execution of reactions according to the description by the CME. The population of species is instantaneously updated at the time of reaction firing. In many systems, such as gene expression networks in biological cells, the initiation of the reaction and the formation of the products, however, is not instantaneous. In gene expression, the initiation of a reaction corresponds to the binding of the transcription factor to the gene. The product, the ribonucleic acid (RNA), is only formed once the RNA-polymerase finished scanning the entire gene. There is hence a time delay between the initiation of the reaction and formation of the products. The average transcription and translation speeds in eukaryotic cells, for example, are 20 nucleotides per second and 2 codons per second, respectively (Alberts et al., 1997; Bratsun et al., 2005; Cai, 2007). This amounts to a coarse-grained modeling of the fundamental processes that mediate the delay, lumping them into a single delay time. Nevertheless, such approaches are valuable and have been used, for instance, to implicate delay along with intrinsic noise in chemical reactions to tune or

3.5 The delay partial-propensity direct method (dPDM)

induce circadian-rhythmic oscillations in drosophila (Li and Lang, 2008; Xu and Cai, 2009), oscillations in other biomolecular clocks such as in the dynamics of messenger-RNA for Notch signaling molecules (Barrio et al., 2006) and oscillations in gene regulatory networks (Bratsun et al., 2005).

Delays in chemical reactions render the kinetics non-Markovian. This is because the next population state of the system depends on the current population as well as the population of the system at previous times when still unfinished reactions have been initiated. This non-Markovian process can be accounted for by the delay CME (Bratsun et al., 2005; Barrio et al., 2006; Tian et al., 2007). The delay CME (dCME) is analogous to the classical CME and in the limit of infinite reactor volume Ω (or infinite population) it tends to the corresponding delay RRE (dRRE), which is a delay differential equation (Bratsun et al., 2005; Barrio et al., 2006; Tian et al., 2007). In order to simulate stochastic chemical kinetics of reaction networks with delays, the delay stochastic simulation algorithm (dSSA) is available (Bratsun et al., 2005; Barrio et al., 2006; Cai, 2007; Tian et al., 2007), extending SSAs to properly account for the effects of non-zero reaction durations. The dSSA samples exact trajectories from the dCME (Barrio et al., 2006; Cai, 2007; Tian et al., 2007). Accelerated approximate methods to sample from the dCME are also available (Zhou et al., 2008; Bayati et al., 2009). The main difference between the dSSA and the conventional SSA is that the reaction propensities (i.e., the probability rate of a reaction firing) may change in the time *between* two reaction initiations (firings), as a result of pending reactions finishing meanwhile. This renders the random variable for the time to the next reaction τ and the index of the next reaction μ mutually dependent. In contrast to conventional SSAs, τ and μ are independent random variables (see Eq. 1.52 in Sec. 1.3).

3.5.1 The delay stochastic simulation algorithm (dSSA)

Consider a network of M chemical reactions among N species. Assume that a subset of these M reactions incur a delay. If a reaction involves no delay (hereafter denoted as R_{D0}), it completes instantaneously and the populations of reactants and products are immediately updated. If a reaction incurs a delay, its products are formed only after a delay d_μ from reaction initiation. We classify delay reactions depending on when the reactants are consumed into *non-consuming* (denoted R_{D1}) and *consuming* (denoted R_{D2}) ones (Cai, 2007). In non-consuming delay reactions, the population of reactants is only updated once the products have formed, thus after the delay d_μ . In consuming delay reactions, the population of the reactants is updated immediately upon reaction initiation, but the products only form after the delay d_μ . In the following, we measure time globally, i.e. relative to time $t = 0$. This is in contrast to the local (relative to the current time t) times used in other SSA formulations. We denote the global time of firing (initiation) of the next reaction as $\tau^g = t + \tau$ and the global time at which the products of a delay reaction μ are formed as $d_\mu^g = t + \tau + d_\mu$.

Assume that at some time t there are Δ pending (ongoing) delay reactions that will finish at later global times $T_1^g, T_2^g, T_3^g, \dots, T_\Delta^g$. We assume that the list of pending reactions is ordered according to ascending global completion times, thus $T_i^g \leq T_{i+1}^g$ $i = 1, \dots, \Delta - 1$. Furthermore, we define $T_0^g = t$ and $T_{\Delta+1}^g = \infty$. As in classical SSA, the time to the next reaction τ (or the global time of firing of the next reaction, τ^g) and the index of the next reaction μ are sampled in order to propagate the system from reaction event to reaction event. In classical

3 Partial-propensity formulations of the stochastic simulation algorithm

SSA, all reactions complete instantaneously, i.e., reaction initiation and completion happen at the same time. Therefore, the reaction propensities remain unchanged during the time interval $[t, t + \tau]$. This, however, is not the case in delay SSAs, where the reaction propensities change whenever a pending reaction completes. Accounting for these inter-firing changes of the propensities, the probability distribution functions for the global time of firing (initiation) of the next reaction $f_\tau(\tau^g)$ and of the index of the reaction $f_\mu(\mu)$ are given by: (Cai, 2007)

$$\begin{aligned} f_\tau(\tau^g) &= a(T_i^g) \exp \left(- \sum_{j=0}^{i-1} a(T_j^g)(T_{j+1}^g - T_j^g) - a(T_i^g)(\tau^g - T_i^g) \right), \\ \tau^g &\in [T_i^g, T_{i+1}^g], \quad i = 0, \dots, \Delta, \end{aligned} \quad (3.16)$$

and

$$f_\mu(\mu) = \frac{a_\mu(T_i^g)}{a(T_i^g)}, \quad \mu = 1, \dots, M, \quad \tau^g \in [T_i^g, T_{i+1}^g]. \quad (3.17)$$

Here, $a_\mu(t)$ is the reaction propensity of reaction μ at global time t and $a(t)$ is the total propensity of all reactions at global time t .

3.5.1.1 The delay direct method (dDM)

In dDM, as presented by Cai et al. (Cai, 2007) and summarized in Table 3.7, the global time of firing of the next reaction τ^g is obtained from Eq. 3.20 using linear search in order to sample the interval p such that

$$p = \max [i : r_1 \geq F(T_i^g)] \quad (3.18)$$

with $\tau^g \in [T_p^g, T_{p+1}^g]$ and r_1 a uniform random number in $[0,1]$. Here, $F(\cdot)$ is the cumulative distribution function of the probability density function $f_\tau(\tau^g)$ (Eq. 3.16). It is given by:

$$\begin{aligned} F(\tau^g) &= 1 - \exp \left(- \sum_{j=0}^{i-1} a(T_j^g)(T_{j+1}^g - T_j^g) - a(T_i^g)(\tau^g - T_i^g) \right), \\ \tau^g &\in [T_i^g, T_{i+1}^g], \quad i = 0, \dots, \Delta. \end{aligned} \quad (3.19)$$

Note that in order to find p , we have to keep track of the change in a whenever a pending reaction finishes. This is done by successively updating the propensities a_μ and the total propensity a every time a pending reaction completes. Therefore, p is the search depth needed to sample τ^g .

Once the interval p is determined, τ^g is calculated as

$$\tau^g = T_p^g + \frac{-\log(1 - r_1) - \sum_{j=0}^{p-1} a(T_j^g)(T_{j+1}^g - T_j^g)}{a(T_p^g)}, \quad (3.20)$$

such that

$$\tau^g \in [T_p^g, T_{p+1}^g].$$

3.5 The delay partial-propensity direct method (dPDM)

The index μ of the next reaction is also obtained by linear search. Unlike in Gillespie's original direct method (DM), however, the probability distribution function of μ depends on the interval p , where p is an integer such that $0 \leq p \leq \Delta$ (In Gillespie's non-delay DM, p is fixed to 0 since there are no pending reactions). The next reaction is hence always sampled after p has been found. Using a uniform random number $r_2 \in [0, 1]$, μ is found such that

$$\mu = \min \left[\mu' : r_2 a(T_p^g) < \sum_{i=1}^{\mu'} a_i(T_p^g) \right]. \quad (3.21)$$

The algorithm of the delay direct method (dDM) (Cai, 2007) is summarized in Table 3.7. It is built around a list of global completion times of the pending delay reactions, maintained in ascending order. The computational cost of this algorithm is determined by the following steps:

Update step: In a strongly coupled reaction network, firing of one reaction can potentially affect all propensities. Hence, the computational cost of updating the reaction propensities is $O(M)$, where M is the number of reactions in the network. For a weakly coupled reaction network the update step is $O(1)$ since the number of propensities affected by any reaction is (by definition of a weakly coupled network) bounded by a constant.

Sampling the global time of the next reaction: The computational cost of sampling the global time of firing (initiation) of the next reaction, τ^g , is $O(pM)$ for a strongly coupled reaction network. Here, p is the search depth to locate τ^g according to Eq. 3.18. This is because the number of times the propensities need to be updated due to pending reactions finishing is p when $\tau^g \in [T_p^g, T_{p+1}^g]$. In each of these p updates, $O(M)$ propensities need to be updated. Deleting the p pending reactions that finish is $O(p)$. Similarly, for a weakly coupled reaction network, the computational cost of sampling τ^g is $O(p)$. In C++, we store the list of the global finishing times of pending reactions in the `multiset` standard template library (STL) container and use the provided methods to add and remove pending reactions.

Sampling the index of the next reaction: The index of the next reaction is found by linear search across the M propensities. The computational cost of this operation is $O(M)$. If the sampled reaction is a delay reaction, it is added to the list of pending reactions, along with its global completion time. Inserting a new reaction such a way that the global times of completion of pending reactions remain sorted in ascending order is $O(\log_2 \Delta)$, where Δ is the number of pending reactions currently in the list.

In summary, the computational cost of dDM is $O(pM + M + \log_2 \Delta)$ for strongly coupled reaction networks. This is equivalent to $O(pM + \log_2 \Delta)$ for $p > 0$. For weakly coupled reaction networks, the computational cost is $O(p + M + \log_2 \Delta)$. Note that when there are no delay reactions, and hence no global pending times need to be inserted in the list, the computational cost of dDM is $O(M)$, as for Gillespie's DM.

We present a partial-propensity formulation of the exact dSSA for chemical reaction networks with delays: the delay partial-propensity direct method (dPDM). The computational cost of dPDM is $O(pN + \log_2 \Delta)$ for strongly coupled networks and $O(p + N + \log_2 \Delta)$ for

1. Initialization: set $t \leftarrow 0$, $\delta a \leftarrow 0$, and the number of pending reactions $\Delta \leftarrow 0$; initialize the population vector \mathbf{n} , the propensities a_μ , and the total propensity $a = \sum_\mu a_\mu$.
 2. Sample the global time of firing of the next reaction, τ^g : First, perform linear search to find the search depth p such that $p \in [T_p^g, T_{p+1}^g)$ according to Eq. 3.18. Then compute τ^g according to Eq. 3.20. Update $\Delta \leftarrow \Delta - p$ and set $t \leftarrow \tau^g$.
 3. Sample the index of the next reaction μ according to Eq. 3.21 using linear search.
 4. If μ is a delay reaction, insert $t + d_\mu$ into the list that stores the global finishing times of the pending reactions. Use bisection search to insert at the proper position such that the list is maintained in ascending order; increment $\Delta \leftarrow \Delta + 1$.
 5. Update the affected a_μ 's using a dependency graph and calculate the change in total propensity δa .
 6. Update $a \leftarrow a + \delta a$.
 7. Go to step 2.
-

Table 3.7: Outline of the algorithm for the delay direct method (dDM) with global times. In C++, the list of global finishing times of pending reactions can conveniently be stored in a `multiset` standard template library container.

3.5 The delay partial-propensity direct method (dPDM)

weakly coupled ones. As a result of using partial propensities, the number of reactions M in the computational cost of dDM is replaced by the usually smaller number of species N . The dPDM formulation is thus especially efficient when p and Δ , which are network-specific parameters that are independent of the simulation method, do not scale faster than $O(N)$. In addition, the linear dependence of the computational cost on N makes dPDM especially efficient for strongly coupled reaction networks, where M grows much faster than N with network size.

3.5.2 Detailed description

Like in PDM, the partial propensities in dPDM are stored in a partial-propensity structure $\boldsymbol{\Pi} = \{\boldsymbol{\Pi}_i\}_{i=0}^N$, the reaction indices in look-up table $\mathbf{L} = \{\mathbf{L}_i\}_{i=0}^N$, the sum of partial propensity in a group in the array $\boldsymbol{\Lambda}$ and sum of propensities in a group in the array $\boldsymbol{\Sigma}$.

After each reaction event (reaction initiation or completion) the population \mathbf{n} , the partial propensities $\Pi_{i,j}$, the Λ_i 's, and the Σ_i 's need to be updated. Which values need to be updated depends on the type of event that happened (firing of a non-delay reaction, initiation of a non-consuming delay reaction, initiation of a consuming delay reaction, or completion of a delay reaction). We efficiently implement the updates using the following data structures:

- $\mathbf{U}^{(1)}$: an array of M arrays, where the i^{th} array contains the indices of all species involved in the i^{th} reaction.
- $\mathbf{U}^{(2)}$: an array of M arrays containing the corresponding stoichiometry (the change in population of each species upon reaction) of the species stored in $\mathbf{U}^{(1)}$.
- $\mathbf{U}_{(-)}^{(1)}$: an array of M arrays, where the i^{th} array contains the indices of all species that are reactants in the i^{th} reaction.
- $\mathbf{U}_{(-)}^{(2)}$: an array of M arrays containing the corresponding stoichiometry of the reactant species stored in $\mathbf{U}_{(-)}^{(1)}$. $\mathbf{U}_{(-)}^{(1)}$ and $\mathbf{U}_{(-)}^{(2)}$ constitute the sparse-representation of the reactant stoichiometry matrix $\boldsymbol{\nu}^-$.
- $\mathbf{U}_{(+)}^{(1)}$: an array of M arrays, where the i^{th} array contains the indices of all species that are products in the i^{th} reaction.
- $\mathbf{U}_{(+)}^{(2)}$: an array of M arrays containing the corresponding stoichiometry of the product species stored in $\mathbf{U}_{(+)}^{(1)}$. $\mathbf{U}_{(+)}^{(1)}$ and $\mathbf{U}_{(+)}^{(2)}$ constitute the sparse-representation of the product stoichiometry matrix $\boldsymbol{\nu}^+$.
- $\mathbf{U}^{(3)}$: an array of N arrays, where the i^{th} array contains the indices of all entries in $\boldsymbol{\Pi}$ that depend on n_i .

We also maintain a list \mathbf{T} that stores the global times (T_i^g , $i = 0, \dots, \Delta$) of all Δ pending reactions in ascending order. The corresponding indices and delay types (R_{D0} , R_{D1} , or R_{D2}) of the reactions are stored in the lists $\boldsymbol{\mu}^{(D)}$ and \mathbf{D} , respectively.

3 Partial-propensity formulations of the stochastic simulation algorithm

In dPDM, the global time of firing (initiation) of the next reaction, τ^g , and the index of the next reaction are mutually dependent. First, the interval p is found according to Eq. 3.18 using linear search such that the global time of firing of the next reaction $\tau^g \in [T_p^g, T_{p+1}^g]$. This tells us between which two reaction completion events the next firing or initiation event happens (see Fig. 3.6A). The difference between dPDM and dDM in sampling p is the mechanism of updating the total propensity $a(T_i^g)$ each time a pending reaction completes and is removed from the queue of pending reactions. In dPDM, we make use of the partial propensities Π and the associated data structures to update a . For instance, assume that $\tau^g \in [T_1^g, T_2^g]$ and the reaction type associated with the global completion time T_1^g is R_{D2} (consuming delay reaction). In this case, we update \mathbf{n} using $\mathbf{U}_{(+)}^{(1)}$ and $\mathbf{U}_{(+)}^{(2)}$. If the finishing reaction is of type R_{D1} (non-consuming delay reaction), \mathbf{n} is updated using $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$. Subsequently, Π and the associated data structures are updated using $\mathbf{U}^{(3)}$, thereby obtaining δa (the change in a) and hence the new a . All these updates are done at the completion times of each pending reaction until the interval containing the global time of firing (initiation) of the next reaction is reached and all p pending reactions that have completed are removed from the list \mathbf{T} . Then, the global time of firing (initiation) of the next reaction, τ^g , within that interval is calculated according to Eq. 3.20.

For sampling the index μ of the next reaction, we use a single uniformly distributed random number $r_2 \in [0, 1)$ to (a) sample the group index I using linear search such that

$$I = \min \left[I' : r_1 a(T_p^g) < \sum_{i=0}^{I'} \Sigma_i(T_p^g) \right] \quad (3.22)$$

and (b) sample the element index J in Π_I using linear search such that

$$J = \min \left[J' : r_1 a(T_p^g) < \sum_{j=1}^{J'} n_I \Pi_{I,j}(T_p^g) + \left(\sum_{i=0}^I \Sigma_i(T_p^g) \right) - \Sigma_I(T_p^g) \right] \quad (3.23)$$

if $\tau^g \in [T_p^g, T_{p+1}^g]$ (see Fig. 3.6B). The sampling of J can be performed efficiently as described in Eq. 3.5. The indices I and J are then translated back to the reaction index μ using the look-up table \mathbf{L} , thus $\mu = L_{I,J}$.

Once the index of the next reaction is sampled, we ascertain the type of the reaction and initiate it. If μ is a non-delay (type R_{D0}) reaction, the population \mathbf{n} is immediately updated using $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$. Subsequently, Π is updated using $\mathbf{U}^{(3)}$. If μ is a non-consuming delay reaction (type R_{D1}), \mathbf{n} and Π are not updated at the time of reaction initiation. Instead, the attributes of this delay reaction (its global time of completion, index, and type) are inserted into \mathbf{T} , $\boldsymbol{\mu}^{(D)}$, and \mathbf{D} , respectively. We ensure that the global completion times in \mathbf{T} are maintained in ascending order by inserting at the appropriate location, which is found using bisection search. If μ is a consuming delay reaction (type R_{D2}), \mathbf{n} is immediately updated using $\mathbf{U}_{(-)}^{(1)}$ and $\mathbf{U}_{(-)}^{(2)}$. Subsequently, Π is updated using $\mathbf{U}^{(3)}$. In addition, the attributes of this reaction are inserted into \mathbf{T} , $\boldsymbol{\mu}^{(D)}$, and \mathbf{D} at the appropriate location, again found by bisection search.

In summary, dPDM is an exact formulation of dSSA, generalizing PDM to handle reactions with delays according to the probability distribution functions of dSSA (Eqs. 3.16 and 3.17).

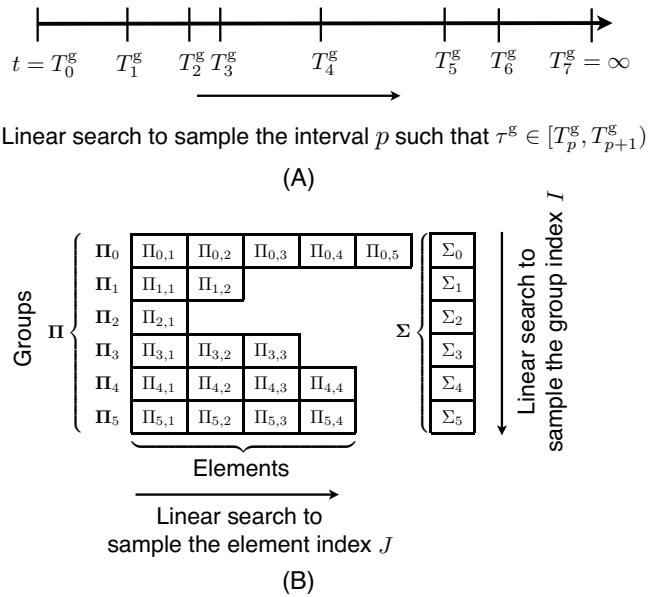


Figure 3.6: Illustration of the main steps in dPDM. (A) Illustration of the linear search to find the interval p such that the global time of firing (initiation) of the next reaction $\tau^g \in [T_p^g, T_{p+1}^g]$. In this figure, the number of pending reactions $\Delta = 6$. (B) Illustration of the partial-propensity structure Π and the grouping based on the index of the common factored-out reactant. The group index I of the next reaction is sampled using linear search over the total propensities of the groups, Σ_i . The element index J within the selected group is found using linear search over the partial propensities stored in group I .

3 Partial-propensity formulations of the stochastic simulation algorithm

The detailed algorithm of dPDM is given in Table 3.8. The computational cost of dPDM is $O(pN + \log_2 \Delta)$ for strongly coupled reaction networks and $O(p + N + \log_2 \Delta)$ for weakly coupled ones, as shown in Sec. 3.5.3.

-
1. Initialization: set $t \leftarrow 0$, $\delta a \leftarrow 0$, and the number of pending reactions $\Delta \leftarrow 0$; initialize the population vector \mathbf{n} , the partial propensities $\boldsymbol{\Pi}$, the group sum array $\mathbf{\Lambda}$, $\mathbf{\Sigma}$, and the total propensity $a \leftarrow \sum_{i=0}^N \Sigma_i$; initialize \mathbf{T} , \mathbf{D} , and $\boldsymbol{\mu}^{(D)}$ (these are empty at this stage); initialize the update structures $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}_{(-)}^{(1)}$, $\mathbf{U}_{(-)}^{(2)}$, $\mathbf{U}_{(+)}^{(1)}$, and $\mathbf{U}_{(+)}^{(2)}$.
 2. Sample the global time of firing of the next reaction, τ^g :
 - 2.1. Generate a uniform random number r_1 in $[0,1)$.
 - 2.2. If $\Delta == 0$ (i.e., \mathbf{T} is empty) then $t \leftarrow t - \log r_1/a$
 - 2.3. else
 - 2.3.1. $\lambda_1 \leftarrow t$; $\lambda_2 \leftarrow T_1$; $a_t \leftarrow a(\lambda_2 - \lambda_1)$; $F \leftarrow 1 - \exp(-a_t)$
 - 2.3.2. While $F < r_1$
 - 2.3.2.1. Get current delay reaction and its type from $\mu_1^{(D)}$ and D_1 , respectively. Update \mathbf{n} , $\boldsymbol{\Pi}$, $\mathbf{\Lambda}$, and $\mathbf{\Sigma}$ accordingly using the proper subset of update structures $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}_{(-)}^{(1)}$, $\mathbf{U}_{(-)}^{(2)}$, $\mathbf{U}_{(+)}^{(1)}$, and $\mathbf{U}_{(+)}^{(2)}$, $\mathbf{U}^{(3)}$ (see Section 3.5.2). Calculate δa and set $a \leftarrow a + \delta a$.
 - 2.3.2.2. $\lambda_1 \leftarrow T_1$. Remove T_1 , $\mu_1^{(D)}$, and D_1 from the corresponding lists and decrement $\Delta \leftarrow \Delta - 1$.
 - 2.3.2.3. If $\Delta == 0$ then exit from the while loop 2.3.2.
 - 2.3.2.4. else $\lambda_2 \leftarrow T_1$
 - 2.3.2.5. $a_t \leftarrow a_t + a(\lambda_2 - \lambda_1)$; $F \leftarrow 1 - \exp(-a_t)$
 - 2.3.3. if $\Delta == 0$ then $\tau^g \leftarrow \lambda_1 + \frac{-\log(1-r_1)-a_t-a(\lambda_2-\lambda_1)}{a}$; set $t \leftarrow \tau^g$
 - 2.3.4. else $\tau^g \leftarrow \lambda_1 + \frac{-\log(1-r_1)-a_t}{a}$; set $t \leftarrow \tau^g$.
 3. Sample the index of the next reaction, μ : Using linear search, sample the group index I and element index J of the next reaction according to Eqs. 3.22 and 3.23, respectively. Look up the index of the next reaction as $\mu = L_{I,J}$.
 4. If μ is a delay reaction, increment $\Delta \leftarrow \Delta + 1$. Insert $t + d_\mu$ into \mathbf{T} , μ into $\boldsymbol{\mu}^{(D)}$, and the type of the delay reaction into \mathbf{D} . Use bisection search to ensure that the entries in \mathbf{T} are in ascending order and maintain the correspondence between \mathbf{T} , $\boldsymbol{\mu}^{(D)}$, and \mathbf{D} .
 5. Update \mathbf{n} depending on reaction μ 's type:
 - 4.1. If μ is R_{D0} , then update \mathbf{n} using $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$
 - 4.2. else if μ is R_{D1} , then do not update \mathbf{n}
 - 4.2. else if μ is R_{D2} , then update \mathbf{n} using $\mathbf{U}_{(-)}^{(1)}$ and $\mathbf{U}_{(-)}^{(2)}$
 6. Update $a \leftarrow a + \delta a$.
 7. Go to step 2.

Table 3.8: Detailed algorithm for the delay partial-propensity direct method (dPDM), explicitly describing all sub-steps. Using the `multiset` container of the C++ STL, the list of pending reactions is conveniently maintained.

3.5.3 Computational cost

The computational cost of dPDM as detailed in Table 3.8 is determined by the following steps:

Update step: The computational cost of the update step is $O(N)$ and $O(1)$ for strongly and weakly coupled reaction networks, respectively, where N is the number of species in the network.

Sampling the global time of the next reaction: The computational cost of sampling the global time of firing (initiation) of the next reaction, τ^g , is $O(pN)$ and $O(p)$ for strongly and weakly coupled reaction networks, respectively. This is because the number of times the partial propensities need to be updated due to a finishing pending reaction is p , where p is search depth to locate τ^g . During each of these p updates, the number of partial propensities that need to be recomputed is $O(N)$ and $O(1)$ for strongly and weakly coupled reaction networks, respectively. Removing the p completed reactions from the list of pending reactions is $O(p)$.

Sampling the index of the next reaction: Sampling the group index is performed using linear search across at most $N + 1$ groups. Subsequently, the element index is sampled using linear search across the $O(N)$ partial propensities within the selected group. The computational cost of sampling the index of the next reaction hence is $O(N)$. If the sampled reaction is a delay reaction, it is added to the list of pending reactions, along with its attributes. Inserting a new reaction such a way that the global completion times of pending reactions are maintained in ascending order is $O(\log_2 \Delta)$, where Δ is the number of pending reactions currently in the list.

In summary, the computational cost of dPDM is $O(pN + N + \log_2 \Delta)$ for strongly coupled reaction networks. This is equivalent to $O(pN + \log_2 \Delta)$ for $p > 0$. For weakly coupled reaction networks, the computational cost is $O(p + N + \log_2 \Delta)$. In general, the search depth p is $O(\Delta)$. The worst case is realized when the time to the next reaction is past the last pending reaction. In this case, $p = \Delta$. The computational cost for subsequently sampling the next reaction is then $O(N)$, without the $O(\log_2 \Delta)$ term. This is because the queue of pending reactions is empty and cost of inserting the new pending reaction is $O(1)$. The overall cost of dPDM then is $O(\Delta N + N)$ and $O(\Delta + N)$ for strongly coupled and weakly coupled reaction networks, respectively.

3.5.4 Benchmarks

We benchmark the computational performance of dPDM on both a weakly coupled and a strongly coupled prototypical reaction network. We again choose the cyclic chain model (Cao et al., 2004; Heuett and Qian, 2006) and the colloidal aggregation model (von Smoluchowski, 1917; van Dongen and Ernst, 1987; van Dongen, 1987; Axford, 1996; Turner et al., 2005) as representative networks for which we compare the performance of dPDM with that of dDM (Cai, 2007). In the benchmarks, we only consider consuming delay reactions since they require updates at both the time of reaction initiation as well as completion.

All tested SSA formulations are implemented in C++ using the random number generator of the GSL library and compiled using the Intel C++ compiler version 11.1 with the O3

3.5 The delay partial-propensity direct method (dPDM)

optimization flag. All timings are measured on a Linux 2.6 workstation with a 2.8 GHz quad-core Intel Xeon E5462 processor, 8 GB of memory and 4 MB L2 cache. For all test cases, we simulate the reaction network until 10^7 reactions have been initiated, and we report the average CPU time Θ per reaction initiation (i.e., the average time to execute steps 2 through 7 in Table 3.8 for dPDM and Table 3.7 for dDM).

3.5.4.1 A strongly coupled reaction network: Colloidal aggregation model

The colloidal aggregation model is given in Eq. 3.7. For N chemical species, the number of reactions is $M = \left\lfloor \frac{N^2}{2} \right\rfloor$. The degree of coupling of this reaction network is $3N - 7$ and hence scales with system size.

At time $t = 0$, we set all $n_i = 1$ and all specific probability rates $c_\mu = 1$. We set all reactions with an even index to be consuming delay reactions (R_{D2}), each with a delay of $d_\mu = 0.1$. The rest of the reactions are non-delay reactions (R_{D0}). The benchmarks confirm that the search depth p to sample the global time of firing (initiation) of the next reaction is $O(1)$, and that the logarithm of the number of pending delay reactions, $\log_2 \Delta$, is $O(\log_2 N)$. Hence, the computational cost of this simulation is $O(N)$ for dPDM and $O(N^2)$ for dDM. This is shown in Fig. 3.7A, where $\Theta(N)$ for dPDM and dDM are compared.

Figure 3.7B shows the results for larger networks on a linear scale. Here, we consider networks of up to $N = 2000$ species and $M = 2$ million reactions in order to reveal memory contention effects. Around $N = 1000$ species, the slope of the cost curve increases, while remaining $O(N)$. This is probably due to the partial-propensity structure not fitting into cache any more. The machine used for the benchmark has a 4 MB L2 cache. At $N = 1000$ the partial-propensity structure for this network contains 500 000 double-precision floating-point numbers of 8 bytes each, amounting to exactly 4 MB.

In summary, for a strongly coupled reaction network, the computational cost of dPDM is $O(pN + \log_2 \Delta)$ as predicted by the theoretical analysis.

3.5.4.2 A weakly coupled reaction network: Cyclic chain model

The cyclic chain model is given by Eq. 3.15. For N chemical species, this network has $M = N$ reactions. The degree of coupling of this reaction network is 2, independent of system size.

At time $t = 0$, we set all $n_i = 1$ and all specific probability rates $c_\mu = 1$. We set all reactions with an even index to be consuming delay reactions (R_{D2}), each with a delay $d_\mu = 0.1$. The rest of the reactions are non-delay reactions (R_{D0}). The benchmarks confirm that the search depth p to sample τ^g is $O(1)$ and that $\log_2 \Delta$ is $O(\log_2 N)$. Hence, the computational cost of this simulation is $O(N)$ for dPDM as well as for dDM. The corresponding $\Theta(N)$ for dPDM and dDM are shown in Fig. 3.7C.

In summary, for a weakly coupled reaction network, the computational cost of dPDM is $O(p + N + \log \Delta)$ as predicted by the theoretical analysis.

3 Partial-propensity formulations of the stochastic simulation algorithm

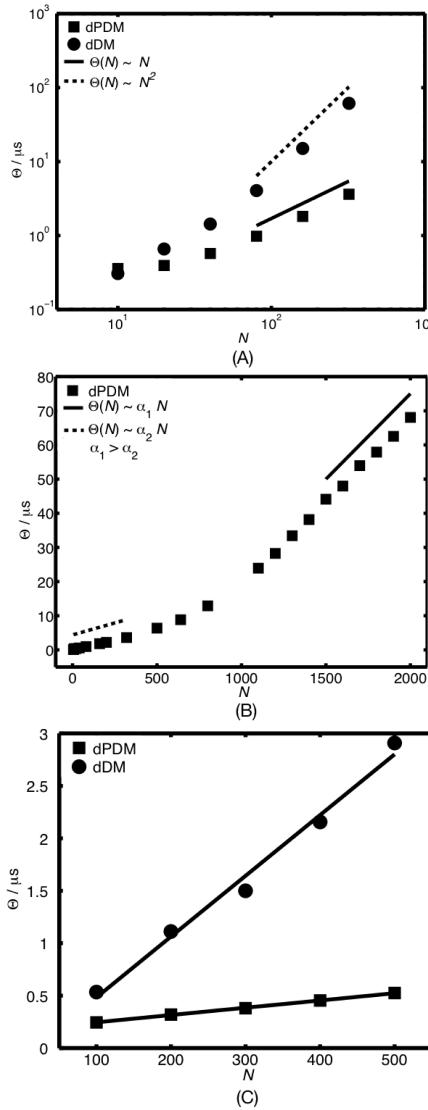


Figure 3.7: Computational cost of dPDM (squares) and dDM (circles). The average (over 100 independent runs) CPU time Θ per reaction initiation (i.e., the average time to execute steps 2 through 7 in Table 3.8 for dPDM, and Table 3.7 for dDM) is shown as a function of the number of species N in the reaction network. (A) Logarithmic plot of $\Theta(N)$ for the strongly coupled colloidal aggregation model, considering systems of size up to $N = 320$. Θ is $O(N)$ for dPDM and $O(M) = O(N^2)$ for dDM. (B) Linear plot of $\Theta(N)$ for the strongly coupled colloidal aggregation model, considering systems of size up to $N = 2000$ (2 million reactions). While the scaling of the computational cost remains linear for all system sizes tested, the slope increases around $N = 1000$. This is the system size beyond which the partial-propensity structure does not fit into the computer's cache memory any more. (C) Linear plot of $\Theta(N)$ for the weakly coupled cyclic chain model. The solid lines are linear least square fits. Θ is $O(N)$ for both dPDM and dDM, but with a smaller slope for dPDM.

3.5.5 Conclusions

We have presented the delay partial-propensity direct method (dPDM), a partial-propensity formulation of the delay stochastic simulation algorithm (dSSA) (Cai, 2007) to simulate chemical reaction networks with delays. dPDM uses partial propensities and reaction groups in order to improve computational efficiency. For reaction networks with no delays, dPDM becomes identical to the partial-propensity direct method (PDM) (see Sec. 3.3).

The presented dPDM is an exact dSSA formulation with a computational cost of $O(pN + N + \log_2 \Delta)$ for strongly coupled reaction networks and $O(p + N + \log_2 \Delta)$ for weakly coupled networks. Here, N is the number of chemical species, p is the search depth to sample the time to the next reaction, and Δ is the number of pending delay reactions at a given time. We have presented a theoretical cost analysis of dPDM and confirmed its results in two benchmark cases prototypical of strongly and weakly coupled reaction networks. Since p and Δ are properties of the chemical reaction network alone, and the only other variable that the computational cost depends on is linear in N , dPDM is especially efficient for strongly coupled reaction networks with delays. This is because in these networks the number of chemical species N grows much slower with network size than the number of chemical reactions M .

However, dPDM inherits the limitations of PDM (see Sec. 3.3.5). Like PDM and PSSA-CR, it is limited to chemical reaction networks composed of elementary reactions involving at most two reactants. For small networks, dPDM is outperformed by other methods due to the overhead of the additional data structures. Other dSSA formulations, such as the delay direct method (dDM) (Cai, 2007), might be more efficient there.

The computational cost of dPDM can be further reduced to $O(p + \log_2 \Delta)$ for weakly coupled reaction networks by using composition-rejection sampling (Devroye, 1986; Slepoy et al., 2008) (see Sections 2.3.7 and 3.4) instead of linear search (Gillespie, 1976) to sample the index of the next reaction. This is analogous to PSSA-CR. For multi-scale (stiff) reaction networks, prototypical of biochemical networks where the propensities span several orders of magnitude, dynamic sorting (McCollum et al., 2006) (see Sec. 3.3.2) can further reduce the computational cost, even though its scaling with N remains the same. These two improved formulations of dPDM have been implemented in the partial-propensity SSA software package as dPSSA-CR and dSPDM respectively (see Appendix A4).

3.6 The family of partial-propensity methods

We present the different partial propensity methods like PDM, SPDM, PSSA-CR, dPDM, dSPDM and dPSSA-CR as realizations of a fixed set of modules. We show that by modifying these modules one can flexibly obtain different partial-propensity formulations, each of which being particularly efficient on a certain class of reaction networks. For example, on weakly coupled reaction networks, the partial propensity SSA with composition-rejection sampling (PSSA-CR) has a computational cost of $O(1)$ under the assumption that the ratio of maximum to minimum non-zero propensity is bounded by a constant. On strongly coupled reaction networks, the partial propensity direct method (PDM) is particularly efficient with a computational cost of $O(N)$. On multi-scale strongly coupled networks, the sorting variant of PDM (SPDM) is recommended. For networks with delays, dSPDM is efficient for strongly

3 Partial-propensity formulations of the stochastic simulation algorithm

coupled networks and for weakly coupled networks dPSSA-CR (the delay variant of PSSA-CR) is efficient.

3.6.1 Modules of partial-propensity algorithms

The use of partial propensities can be interpreted as follows: Let \mathbf{X} be the diagonal matrix of the population vector \mathbf{n} , such that $\mathbf{X} = \text{diag}(\mathbf{n})$. Further, let \mathbf{B} be the symmetric, positive definite $N \times N$ matrix of specific probability rates of all bimolecular reactions. Element $B_{i,j} = B_{j,i} > 0$ is the specific probability rate c of the reaction of species i with species j . Similarly, the specific probability rates of all unimolecular reactions are collected in the $N \times N$ diagonal matrix \mathbf{U} . The propensities of bimolecular reactions are then given by the product $\mathbf{A}_B = \mathbf{X}\mathbf{B}\mathbf{X}$, those of unimolecular reaction by $\mathbf{A}_U = \mathbf{X}\mathbf{U}$. Traditional SSA formulations amount to first explicitly computing all propensities and then sampling over all the elements in $\mathbf{A}_{B,U} = [\mathbf{A}_B, \mathbf{A}_U]$. Partial-propensity methods first sample over the vector $\mathbf{A}_{B,U} \mathbf{1}$ to obtain the group index I , where $\mathbf{1}$ is a vector of 1's. Subsequently sampling the element index J is performed over the elements of the I^{th} row of the matrix $\mathbf{X}^{-1}\mathbf{A}_{B,U} = [\mathbf{B}\mathbf{X}, \mathbf{U}]$. This is implemented using three algorithmic modules: grouping the partial propensities, sampling the next reaction, and updating the values. These modules of partial propensity SSAs are summarized in Fig. 3.8 together with their respective computational costs. Different partial-propensity methods with different computational costs can be constructed by using different algorithms in the sampling module.

1. **Grouping module:** Partial-propensity methods group the partial propensities of all reactions according to the index of the factored-out reactant, i.e., the common reaction partner. Each group thus contains the partial propensities of all reactions having this species as a reactant. The different partial propensities within a group correspond to the various possible reaction partners of the common, factored-out reactant. For any reaction network, there are at most $N+1$ groups (including group 0 for source reactions) and the number of partial propensities in each group is at most $O(N)$. For higher-order reactions (trimolecular and more), multi-dimensional grouping can be used with one dimension per reactant. Again, the total number of groups in each dimension is $O(N)$ and the sampling module can be independently applied in each dimension in order to sample the reaction partners.
2. **Sampling module:** The key building block of partial propensity methods is the algorithm used to sample the time to the next reaction as in DM and the index of the next reaction. Given the grouping of partial propensities, sampling the index of the next reaction involves sampling the index of the group and then the index of the element within that group. Sampling the index of the group amounts to sampling the first reactant of the next reaction. In order to find out which partner this reactant is going to react with, the partial propensity within the group is sampled. For unimolecular and source reactions, the partial propensities are constants and the second step is obsolete.

All sampling algorithms used in standard SSAs can also be used in partial-propensity methods. Instead of applying them over reactions, however, they are first applied over partial-propensity groups and then over the elements within the selected group. For

3.6 The family of partial-propensity methods

example, using linear search (as in Gillespie’s direct method (Gillespie, 1976)) leads to a sampling step that is $O(N)$ on all classes of networks. Replacing linear search by composition-rejection sampling (Devroye, 1986) reduces the computational cost of the sampling step to $O(1)$. Other sampling strategies, such as search trees or a first-reaction-method-like sampling over the reaction times can also be used straightforwardly. Depending on the sampling strategy and the associated algorithmic overhead, certain partial-propensity formulations are particularly well suited for certain classes of reaction networks. Also for networks with time delays, different algorithms can be combined for sampling the time to the next reaction and the index of the next reaction. The key difference, however, is that the time to the next reaction and the index of the next reaction are not independent random variables, and hence the time to the next reaction needs to be computed first. See Fig. 3.8 for a summary of different algorithms that can be used. Note that the partial-propensity formulations for networks with delays seamlessly reduce to partial-propensity formulations for network without delays when the delay for each reaction is set to 0.

3. **Update module:** After the selected reaction has fired and the populations of the involved species have been updated, the affected partial propensities are recomputed using a dependency graph over species. Since any partial propensity is a function of the population of at most one species, the number of partial propensities to be updated is at most $O(N)$. In weakly coupled reaction networks, the number of partial propensities to be updated is $O(1)$, since the degree of coupling is bounded by a constant. However, depending on the data structures that are used in the sampling module, the computational cost of the update module varies. Figure 3.8 shows the computational cost of the update step depending on the sampling method used.

Using this modular approach, different algorithms can be combined to construct different partial-propensity formulations (see Fig. 3.8). Certain formulations may be well-suited for reaction networks with certain properties. The classification of reaction networks according to their “difficulty”, however, is still largely an open question. Besides system size, degree of coupling, and multiscaling (spectrum of time scales), there might also be other network properties that influence the computational cost of the various SSA formulations. Automatized selection of the most efficient SSA formulation for a given network would require both a systematic classification of networks that goes beyond merely classifying networks as being weakly or strongly coupled. In addition, a prediction of the computational cost of SSA formulations based on network properties would be required. This might involve a more detailed cost analysis of the algorithms and a set of standard benchmark problems that are designed to cover a wide range of performance-relevant parameters.

Implementing the generic modules in C++, we have developed the partial-propensity SSA (pSSA) software package for simulating stochastic chemical kinetics of reaction networks with or without delays. pSSA reads the reaction network in the SBML (Systems Biology Markup Language) (Hucka et al., 2003) input format and is equipped with a user-friendly MATLAB interface. Exact stochastic simulation algorithms supported by the pSSA include DM, PDM, SPDM, PSSA-CR, dDM, dPDM, dSPDM and dPSSA-CR. More details on the pSSA software package can be found in Appendix A4.

3 Partial-propensity formulations of the stochastic simulation algorithm

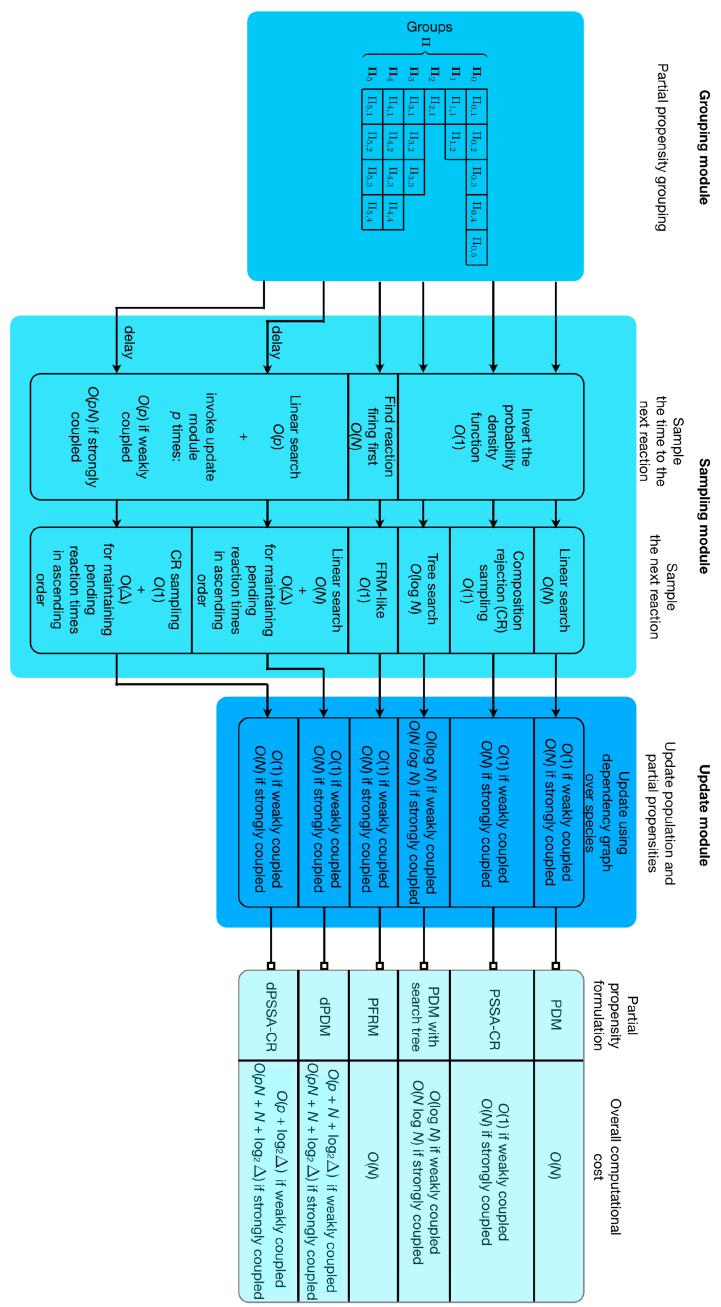


Figure 3.8: Illustration of the three modules of partial propensity SSAs: the grouping module, the sampling module, and the update module. The grouping module groups reactions according to their partial propensities. The sampling module samples the time to the next reaction and the index of the next reaction. The update module updates the population and the affected partial propensities after a reaction has fired. Using different algorithms in the sampling module we obtain different partial propensity formulations, such as PDM (the partial-propensity direct method), PSSA-CR (the partial-propensity SSA with composition-rejection sampling), PDM with search trees, PFPM (the partial-propensity first reaction method), the delay PDM and the delay PSSA-CR. These formulations have varying overall computational costs, depending on the coupling class of the reaction network. N is the number of chemical species in a reaction network, p is the search depth to find the interval for the time to the next reaction is delay SSAs and Δ is the number of pending reactions in delay SSAs. If the delays of all reactions in a chemical reaction network are zero the computational cost of any delay partial-propensity methods becomes equal to that of the corresponding non-delay formulation. Sorting variants of PDM (SPDM) and of dPDM (dSPDM) is obtained by dynamically bubbling up reactions in the next reaction sampling module.

3.7 Summary

We have introduced partial-propensity formulations of Gillespie's exact stochastic simulation algorithm (SSA). All presented partial-propensity formulations sample trajectories from the exact solution of the chemical master equation (CME). In addition, we also presented a partial-propensity formulation of the delay SSA (dSSA) for chemical reaction networks with delays. We showed that all partial-propensity formulations can be composed from three modules: the grouping module, the sampling module and the update module. Different algorithms and data structures can be used in these modules to obtain partial-propensity formulations. These formulations have varying computational cost depending on the algorithms used and on the coupling class of the simulated reaction network.

Limiting ourselves to networks with elementary reactions, all partial-propensity formulations have a computational cost that scales at most linear with the number of chemical species in the reaction network. Partial-propensity formulations are therefore efficient for reaction networks where the number of chemical species is much smaller than the number of chemical reactions. Due to the overhead of the additional data structures, partial-propensity formulations may not be efficient for small reaction networks, where the additional cost from creating and operating on these data structures may not be amortized.

For strongly and weakly coupled reaction networks without delays, the partial-propensity direct method (PDM) has a computational cost of $O(N)$ where N is the number of chemical species. Due to the dynamic sorting strategy in the sampling module, the sorting variant of PDM (SPDM) is especially efficient for multiscale (stiff) reaction networks, without any significant trade-off in the computational cost for non-multiscale reaction networks. For weakly coupled reaction networks, the computational cost of partial-propensity formulations has been reduced to $O(1)$ using composition-rejection sampling (PSSA-CR). For reaction networks with delays, the delay variants of the these partial-propensity formulations have the same scaling of computational cost with increasing network size.

The favorable scaling of the computational cost of partial-propensity formulations, however, does not necessarily make them the most efficient in terms of absolute runtimes. Even though we have demonstrated that partial-propensity methods can offer significant speed-ups for relatively large reaction networks, this does not imply superior performance on any given, particular network. Based on empirical evidence and on theoretical analysis of the computational costs, however, we recommend SPDM for strongly coupled networks and PSSA-CR for weakly coupled networks. In special cases, SPDM can be worse than PSSA-CR even on strongly coupled reaction networks. If the coupling-class of the reaction network is unknown, we recommend PSSA-CR. For reaction networks with time delay, the corresponding delay variant is recommended.

4

Discreteness-induced concentration inversion in mesoscopic chemical reaction systems

“NATURE! . . . She performs a play; we know not whether she sees it herself, and yet she acts for us, the lookers-on.”

– *Nature: Aphorisms by Goethe, Thomas H. Huxley (1869)*

Mesoscopic chemical reaction systems are typically realized in compartments or reactors with length scales ranging from a few tens of nanometers to millimeters. The same concentration realized in compartments of decreasing size implies a decreasing total number of molecules. This leads to an apparent increase in molecular discreteness and increases concentration fluctuations (van Kampen, 1963; Kurtz, 1972; Gardiner et al., 1976; Berg et al., 2000; Pedraza and Paulsson, 2008; van Kampen, 2001; Shibata, 2004; Miller et al., 2005; Raj et al., 2006; Beard and Qian, 2007; Perc et al., 2007; Grima and Schnell, 2008; Mouri et al., 2009; Nishikawa and Shibata, 2010; Grima, 2010a). Thus, chemical reaction dynamics in small-scale compartments must be considerably influenced by intrinsic (or internal) noise (Qian et al., 2002; Perc et al., 2008; Shimoni et al., 2009; Grima, 2009b, 2010a). The effect of intrinsic noise on simple chemical reaction systems has been introduced in Chapter 1, Sec. 1.4.

Biological cells are natural and ubiquitous examples of highly confined chemically reactive systems. Biochemical reactions proceed in sub-micron compartments containing one to several thousand molecules per compartment (Ghaemmaghami et al., 2003; Ishihama et al., 2008) and hence intrinsic noise is expected to play an important functional role in biochemical cir-

cuits (Berg et al., 2000; Springer and Paulsson, 2006; Eldar and Elowitz, 2010). Such systems are also subjected to extrinsic noise originating outside the immediate system of interest. For example, the rate of chemical reactions inside a compartment may depend on the abundance of an upstream species that is translocated from another compartment (Bar-Even et al., 2006; Pedraza and Paulsson, 2008; Cai et al., 2008; Eldar and Elowitz, 2010). The relevance of intrinsic noise is, however, not limited to biological systems. Molecular capsules (Vriezema et al., 2005), carbon nanotubes (Ugarte et al., 1996), and crystalline zeolites (Smit and Maesen, 2008) are other examples of nanospaces confining chemical reactions. Recent studies have shown that consideration of intrinsic noise leads to a considerable modification of the temperature dependence of the equilibrium constants of reactions in such artificially confined spaces (Polak and Rubinovich, 2008).

There is an extensive modeling literature investigating the effects of intrinsic noise on chemical kinetics. The effect of intrinsic noise is taken into account by the CME, which is an exact mesoscopic description for any well-stirred and thermally equilibrated gas-phase chemical system (Gillespie, 1992), and for chemical reactions in well-stirred dilute solutions (Gillespie, 2009). Unfortunately, the CME is generally analytically intractable. To overcome this limitation, a large proportion of studies (van Dongen, 1987; Elf and Ehrenberg, 2003; Paulsson, 2004; Bruggeman et al., 2009; Grönlund et al., 2010; Elf et al., 2003; Hayot and Jayaprakash, 2004; McKane et al., 2007) make use of the linear-noise approximation (LNA)—a linear approximation of the nonlinear Fokker-Planck equation—of the CME by using Langevin or linear Fokker-Planck equations to analytically estimate the magnitude of intrinsic noise and to investigate various noise-induced phenomena. One should, however, bear in mind that the LNA is valid in the limit of large volumes (van Kampen, 2001, 1961, 1976). This limit in van Kampen’s system-size expansion is taken at fixed macroscopic concentrations and thus the LNA is valid in the asymptotic limit of very large molecule numbers. In other words, use of the LNA presumes that effects due to molecular discreteness do not dominate the phenomena under study (See Appendix A5 for more details on the van Kampen expansion and the LNA). This is, for example, valid for pathways involving at most first-order reactions, for which it is known that including the effects of molecular discreteness does not influence the predictions of the mean concentrations (McQuarrie, 1967; Heuett and Qian, 2006). However, this is not the case if some reactions are bimolecular; these effects originate from the nonlinearity inherent in the law of mass action describing such reactions (van Kampen, 1963, 2001; Grima, 2010a). It hence stands to reason that the predicted functionality and operation of a chemical circuit involving small copy numbers of some species may in some cases be considerably different from standard predictions based on the LNA. It is furthermore of interest to understand how extrinsic noise influences a circuit’s low-copy-number properties, since experimental studies suggest that extrinsic noise is frequently comparable to or larger than intrinsic noise (Bar-Even et al., 2006; Newman et al., 2006).

Here, we investigate the dependence of the nonequilibrium steady-state properties of a monostable chemical circuit on the apparent discreteness of the interacting chemical species. We consider independent realizations of the same chemical reaction system in compartments of different volumes. Given some fixed rate constants, reaction rate equations (RRE) predict the same steady-state concentrations for all realizations. However, the predicted average molecule numbers (concentration multiplied by the volume) decrease with compartment volume. It is thus clear that one can study the relationship between molecular discreteness and the steady

state of a chemical system by studying the relationship of the latter with compartment volume. RRE are based on an implicit assumption of large numbers of interacting molecules and hence one expects their breakdown for systems realized in small volumes. The linear Fokker-Planck equation associated with the LNA leads to the same prediction of volume-independent mean concentrations (van Kampen, 2001; Grima, 2010a). In contrast, we show that accounting for molecular discreteness leads to volume-dependent mean concentrations, and that the RRE and LNA predictions are qualitatively correct only for systems above a critical volume. This describes a new phenomenon: *discreteness-induced concentration inversion*. For systems in sub-critical volumes, the RRE and LNA predict that the concentration of a species S_i is larger than that of another species S_j , whereas accounting for discreteness leads to the opposite prediction. We present a theory to explain this novel discreteness-induced inversion effect and to predict the values of the critical volumes. We illustrate and verify the theory by comparing its predictions to exact stochastic simulations of the CME for the model system of trimerization (see Fig. 4.1 for a cartoon illustration). Furthermore, we show that the inversion effect occurs already at larger numbers of molecules in the presence of extrinsic noise. In general, extrinsic noise in biological systems models stochastic influences from the environment. Here, we specifically model extrinsic noise as a bursty influx of a reactant into the reactor.

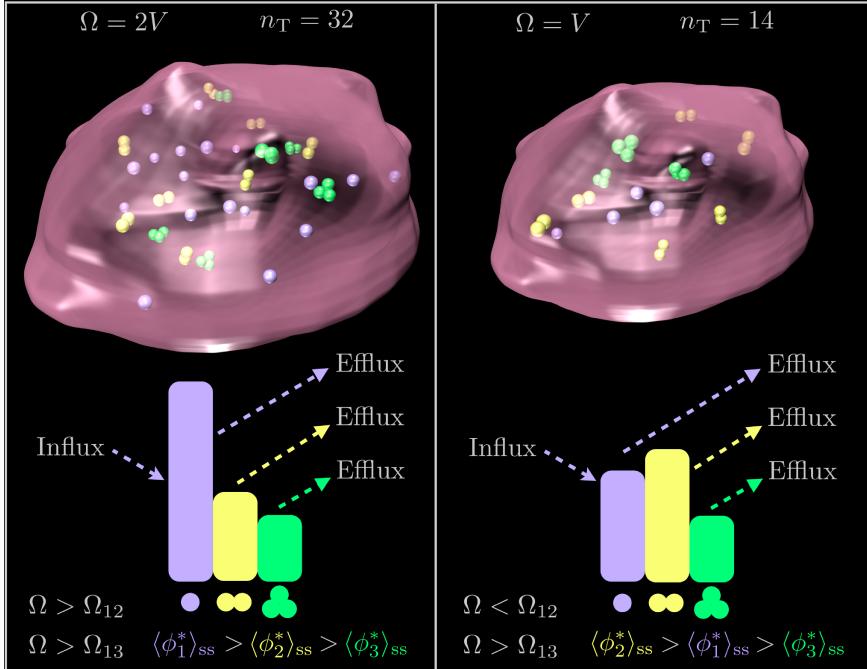
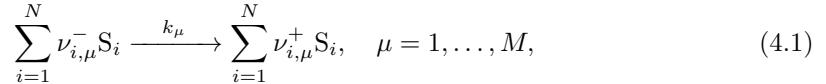


Figure 4.1: Illustration of the change in the nonequilibrium steady-state concentrations of monomers (species 1, purple), dimers (species 2, yellow), and trimers (species 3, green) with compartment volume Ω to which an open trimerization reaction system is confined. Monomers are pumped into the compartment. They then bind to other monomers to form dimers and to dimers to form the final product, trimers. All molecules leave the compartment at a constant rate. The spheres represent the molecules, while the semi-transparent pink surface represents the arbitrarily shaped compartment to which the molecules are confined. The bar graphs reflect the mean steady-state concentration values (denoted by $\langle \phi_i^* \rangle_{ss}$ for species i). When the volume decreases below a certain critical value (Ω_{12}), the ratio of monomer and dimer concentrations changes from greater than one to less than one. This “inversion effect” is induced by an increase in the apparent discreteness of the system as the volume is decreased from $2V$ to V . The apparent discreteness is quantified by the total number of molecules in the compartment, n_T . The rate constants and the average occupied volume fraction of the system are the same at both volumes. In contrast, RRE and the LNA predict no change in the steady state of the system as the volume is decreased.

4.1 General theory

We present a theory that predicts the discreteness-induced concentration inversion in mesoscopic chemical systems. We start from a general molecular reaction network proceeding in a well-stirred volume Ω . The network can be written as a set of N distinct chemical species interacting via M reactions of the type described in Eq. 1.1:



where the macroscopic reaction rate of reaction μ is k_μ . The constraints $\sum_{i=1}^N \nu_{i,\mu}^- \leq 2 \forall \mu$ ensure that each reaction is at most bimolecular and hence elementary. The RRE for this system is

$$\frac{\partial \phi}{\partial t} = \boldsymbol{\nu} \mathbf{T}_0(\phi), \quad (4.2)$$

where $\phi(t) = [\phi_1(t), \dots, \phi_N(t)]^T$ is the vector of macroscopic concentrations at time t , and $T_{\mu,0}(\phi(t))$ is the macroscopic rate function of reaction μ given by $T_{\mu,0}(\phi(t)) = k_\mu \prod_{i=1}^N \phi_i^{\nu_{i,\mu}^-}(t)$ (see Chapter 1, Eq. 1.41). The details of the RRE can be encapsulated in the $N \times M$ stoichiometric matrix $\boldsymbol{\nu}$ with entries $\nu_{i,\mu} = \nu_{i,\mu}^+ - \nu_{i,\mu}^-$ and the $M \times 1$ rate function (also called reaction flux) vector $\mathbf{T}_0(\phi)$.

The CME is the corresponding mesoscopic description of the molecular network. It is a differential-difference equation (McQuarrie, 1967) whose solution gives the probability that the system is in a certain state at time t . The state of the system is given by the vector of the absolute numbers of molecules of each species (the molecular position and velocity variables do not enter into the state description because of the assumption of well-mixing). Thus, the CME provides us with a description of chemical processes factoring in the discrete nature of molecules in well-mixed compartments of mesoscopic to macroscopic sizes. Unfortunately, the stochastic description of a general reaction network using a CME is not easily amenable to calculation. The problem, however, simplifies if we are only interested in the mean concentrations as predicted by the CME. Approximate time-evolution equations (Grima, 2010a) have recently been derived for these mean concentrations using the volume expansion of the CME (van Kampen, 2001), including terms of the order $\Omega^{-1/2}$. Terms of order Ω^0 correspond to the LNA, while terms of order $\Omega^{-1/2}$ and beyond capture the effects of molecular discreteness. These equations are called the *effective mesoscopic reaction rate equations* (EMRE). In Appendix A5 we provide a detailed description of the van Kampen system-size expansion that leads to the LNA as a first approximation (Appendix A5.1) and subsequently to the EMRE when terms of the order $\Omega^{-1/2}$ are included (Appendix A5.2).

The EMRE has the following form for a general reaction network:

$$\frac{\partial \langle \phi^* \rangle}{\partial t} = \frac{\partial \phi}{\partial t} + \mathbf{J}^{(0)} (\langle \phi^* \rangle - \phi) + \Omega^{-1} \Delta(\mathbf{C}) + O(\Omega^{-3/2}) \quad (4.3)$$

$$\frac{\partial \mathbf{C}(t)}{\partial t} = \mathbf{J}^{(0)} \mathbf{C} + \mathbf{C} \mathbf{J}^{(0)T} + \mathbf{D}^{(0)} + O(\Omega^{-1/2}), \quad (4.4)$$

where we have suppressed the time dependence of variables for clarity. The vector of mean concentrations is $\langle \phi^* \rangle = [\langle n_1 \rangle / \Omega, \dots, \langle n_N \rangle / \Omega]^T$, where n_i is the number of molecules of species i in volume Ω . The matrix $\mathbf{J}^{(0)}$ is the Jacobian of the RRE (Eq. 4.2) and \mathbf{C} is the covariance matrix, with entries in the i^{th} row and j^{th} column equal to $\langle \epsilon_i \epsilon_j \rangle$, where $\Omega^{-1/2} \epsilon_i$ is the noise about the macroscopic concentration of species i (van Kampen, 2001). The matrix $\mathbf{D}^{(0)}$ is given by $\mathbf{D}^{(0)} = \boldsymbol{\nu} \text{diag}(\mathbf{T}_0(\phi)) \boldsymbol{\nu}^T$ (Elf and Ehrenberg, 2003) and provides a measure of the strength of noise. The vector Δ is the mean–covariance coupling vector. Its l^{th} components is defined as (see also Eq. A5.37):

$$\Delta_l = \frac{1}{2} \left(\sum_{w,z=1}^N \frac{\partial J_{lw}}{\partial \phi_z} \langle \epsilon_w \epsilon_z \rangle - \sum_{w=1}^N \phi_w \frac{\partial J_{lw}}{\partial \phi_w} \right), \quad (4.5)$$

where $\langle \epsilon_w \epsilon_z \rangle$ is a short-hand notation for $\langle \epsilon_w \epsilon_z \rangle$. In the limit of macroscopically large volumes, the Ω^{-1} term in Eq. 4.3 tends to zero, implying that in this limit the EMRE reduce to the RRE. For finite volumes, the solution of the EMRE is in general different from that of the RRE. The macroscopic RRE estimate for the copy number of molecules of a species equals the macroscopic concentration of that species multiplied by the compartment or reactor volume; hence it is clear that the Ω^{-1} term in the EMRE formulation arises due to the discreteness of the system. Furthermore, inspection of the vector Δ shows that the corrections to the RRE exist only if some Jacobian elements in Eq. 4.5 are functions of the macroscopic concentrations, i.e., if the chemical network has at least one bimolecular reaction (See Appendix A5.2.1 and A5.2.2). Although the EMRE are derived from the CME, they can be constructed from sole knowledge of the RRE. We emphasize that the correct way of interpreting the EMRE (Eqs. 4.3–4.4) is that given some set of reactions occurring in a volume Ω , ϕ is the concentration prediction if we ignore molecular discreteness, and $\langle \phi^* \rangle$ is the approximate mean-concentration prediction if we take discreteness into account. This interpretation stems from the fact that terms of order Ω^0 in the system-size expansion, which lead to the LNA, do not explicitly depend on the volume, whereas the terms of order $\Omega^{-1/2}$, from which the EMRE are obtained, do depend on the volume (see Eq. A5.17 in Appendix A5).

We obtain steady-state values by setting the time derivatives in Eqs. 4.3 and 4.4 to zero, leading to

$$\langle \phi^* \rangle_{\text{ss}} - \phi_{\text{ss}} = -\Omega^{-1} \boldsymbol{\delta}, \quad (4.6)$$

where $\boldsymbol{\delta} = \mathbf{J}_{\text{ss}}^{(0)-1} \Delta (\mathbf{C}_{\text{ss}})$ and the subscript ss denotes evaluation at steady-state. None of these matrices, nor the steady-state solution ϕ_{ss} of the RRE, explicitly depend on Ω . Consider the case where the macroscopic steady state of the system is such that $\langle \phi_1^* \rangle = \phi_1 > \langle \phi_2^* \rangle = \phi_2$. It follows from Eq. 4.6 that if $\delta_1 > \delta_2$ then there exists a critical compartment volume $\Omega_{12} = (\delta_2 - \delta_1) / (\phi_2 - \phi_1)$ at which the concentrations of the two species become equal, $\langle \phi_1^* \rangle = \langle \phi_2^* \rangle$. For volumes smaller than critical, the state of the system is described by $\langle \phi_1^* \rangle < \langle \phi_2^* \rangle$. In other words, although accounting for molecular discreteness always leads to some correction to the RRE solutions, it is only below a certain critical copy number of molecules that these corrections lead to qualitative changes in the mean steady-state concentrations of a chemical network. We refer to this novel transition as a *discreteness-induced inversion effect* due to the “inversion” in the ratio of the concentrations of two species as the volume crosses the critical threshold. This inversion effect is not predicted by the LNA because the effect originates from

terms of higher order than Ω^0 . For a system of N distinct chemical species, there are at most $\frac{1}{2}N(N-1)$ different critical volumes, one for each unique pair of species. The general condition for the existence of a critical volume Ω_{ij} for species i and j is: $\text{sign}(\delta_i - \delta_j) = \text{sign}(\phi_i - \phi_j)$. RRE and the LNA are qualitatively valid for volumes larger than the maximum of all critical volumes, whereas discreteness-induced effects dominate the steady-state behavior for subcritical volumes.

A general three-step recipe for predicting the critical volumes of a chemical network is as follows:

1. An expression for the vector $\boldsymbol{\delta} = \mathbf{J}_{ss}^{(0)^{-1}} \Delta(\mathbf{C}_{ss})$ is derived using the Jacobian of the RRE rate function $\mathbf{T}_0(\boldsymbol{\phi})$ and the mean-covariance coupling vector as given by Eq. 4.5.
2. An explicit expression for the steady-state covariance matrix, \mathbf{C}_{ss} , in terms of the steady-state concentrations is obtained by solving Eq. 4.4 with the time derivative set to zero. This also requires that one derives an expression for the matrix $\mathbf{D}_{ss}^{(0)}$ using $\mathbf{D}^{(0)} = \boldsymbol{\nu} \text{diag}(\mathbf{T}_0(\boldsymbol{\phi})) \boldsymbol{\nu}^T$.
3. By substituting the resulting expression for \mathbf{C}_{ss} into the one previously obtained for $\boldsymbol{\delta}$, and inserting the latter in

$$\Omega_{ij} = (\delta_i - \delta_j)/(\phi_i - \phi_j), \quad i \neq j, \quad i, j = 1, \dots, N, \quad (4.7)$$

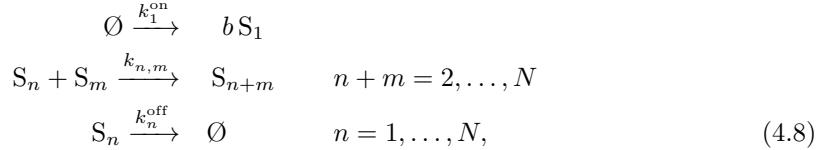
one obtains the final expressions for the critical volumes of the system.

4.2 Model

As a representative model of nonlinear reaction networks out of equilibrium we consider driven colloidal aggregation, for three reasons: First, it is a complete model since this reaction network comprises all three types of elementary reactions: bimolecular, source (input), and unimolecular (Gillespie, 1992), rendering the results obtained here valid also for other reaction networks. Second, it is a well-characterized model as it has been studied for decades, notably from the 1917 works of Smoluchowski on coagulation and fragmentation (von Smoluchowski, 1917). Third, it is a relevant model for many real-world phenomena of practical importance, e.g., in biological cells (receptor oligomerization, protein and prion-peptide aggregation, cytoskeletal actin & tubulin polymerization), in nanotechnology (nano-particle clustering, colloidal crystallization), in food engineering and the oil industry (emulsion stabilization, emulsification in porous media), and in metallurgy (dealloying).

We use the CME to describe the reaction kinetics, neglecting molecular aspects underlying nucleation and growth. Our system is spatially homogeneous (well stirred) as we disregard structural, spatial, or solvent effects. We also factor out the role of (i) densification upon decrease in system volume, as the total volume fraction is kept constant, and (ii) conformational kinetics, as we do not consider intra-molecular degrees of freedom. In addition, we study our system at a steady state that may be arbitrarily far away from thermodynamic equilibrium.

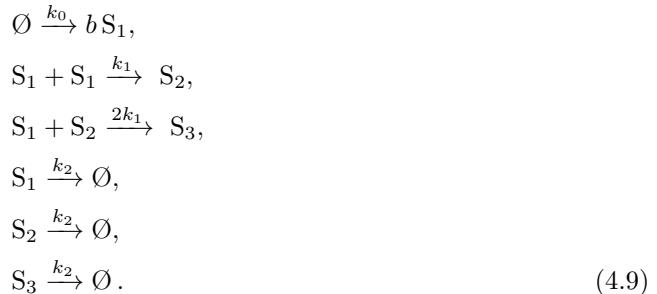
Denoting aggregates or multimers containing n monomers as species S_n , the aggregation reaction network is:



where the k 's are macroscopically measurable reaction rates (Gillespie, 1976). This system describes the aggregation of monomers S_1 into multimers S_n of maximum size N . Monomers are input into the finite reactor volume Ω in bursts of arbitrary size b . They then form dimers, which can further aggregate with other monomers or multimers to form larger aggregates.

We now demonstrate and verify our theoretical predictions by considering the molecular assembly of trimers from monomers (i.e., $N = 3$ in Eq. 4.8) in a small confined space. This is an example of particular relevance in various biological contexts, such as the trimerization of receptor proteins in the plasma membrane (Klemm et al., 1998; Choudhary and Mann, 2010) that tunes the activity and sensitivity of many signal-transduction pathways, and the trimerization of heat-shock factors, which is responsible for regulating a multitude of cellular processes ranging from the expression of stress genes to cell development and lifespan control (Akerfelt et al., 2010). Further, experiments indicate that approximately 35% of the proteins in *E. coli* are homodimers or homotrimers (Goodsell and Olson, 2000). Finally, trimerization (i.e., $N = 3$ in Eq. 4.8) is the simplest colloidal aggregation model with all types elementary chemical reactions: source reactions (input), unimolecular reactions, and the two types of bimolecular reactions: homodimerisation and heterodimerisation.

The specific reaction scheme considered here is:



The above choice of rate constants significantly reduces the dimensionality of the parameter space and simplifies the algebra. Monomers (S_1) are input into the compartment in bursts of size b . The monomers form dimers (S_2) that can in turn react with more monomers to form trimers (S_3). All three species leave the system at some constant rate. The CME and the RRE associated with the reactions in Eq. 4.9 are given in Appendix A8. We also show in Appendix A8 that this trimerization system is monostable.

The burst-input mechanism is a model for molecules entering the compartment by active means of transport, rather than diffusion (Dobrzański and Bruggeman, 2009; Grima, 2009a). This is a common mechanism in biological cells (Alberts et al., 1994; Cai et al., 2008). In

particular, this mechanism occurs in the dynamics of membrane-protein domains (rafts) in contact with a metabolic network (Turner et al., 2005; Zhao and Keen, 2008). For example, in the case of receptor protein trimerization in the plasma membrane, the burst input could stem from the vesicular transport of receptor monomers from exocytic compartments. Experimental evidence suggests that the burst size b can be as large as 1000 (Cai et al., 2008). Our system is influenced by intrinsic noise stemming from the inherent molecular discreteness, and also by extrinsic noise due to the burst-input process (Pedraza and Paulsson, 2008; Eldar and Elowitz, 2010). The parameter b controls the magnitude of extrinsic noise, since a larger b naturally implies larger fluctuations in the monomer concentration upon input into the compartment. Generally, in a reaction system with burst input $\emptyset \xrightarrow{k} b A$ into a reactor of finite volume Ω (k is the macroscopic reaction rate), the variance at a nonequilibrium steady-state is $O(b/\Omega)$ (see Eq. A7.10 in Appendix A7). We emphasize that these input fluctuations are not induced by a chemical reaction process inside the compartment; rather, they are caused by the input process, which is external to the compartment and hence a source of extrinsic noise to the reactions in the compartment. For the moment we consider b to be a time-independent constant; later we also study the case where b is itself a stochastic quantity, as it typically is in bursty phenomena associated with protein production (Cai et al., 2006).

We start by calculating the critical volume sizes for this system using the above theoretical framework and then validate these predictions using PDM simulations. Using the Jacobian of the deterministic RRE for this system together with Eq. 4.5, we find:

$$\delta = \frac{k_1}{\alpha} \begin{pmatrix} 2[\langle \epsilon_{11} \rangle_{ss} - \phi_{1,ss}] (3k_1\phi_{1,ss} + k_2) + 2k_2\langle \epsilon_{12} \rangle_{ss} \\ -[\langle \epsilon_{11} \rangle_{ss} - \phi_{1,ss}] (6k_1\phi_{2,ss} + k_2) + 2(k_2 + 6k_1\phi_{1,ss})\langle \epsilon_{12} \rangle_{ss} \\ 2k_1[\langle \epsilon_{11} \rangle_{ss} - \phi_{1,ss}] (2\phi_{2,ss} - \phi_{1,ss}) - 2(4k_1\phi_{1,ss} + k_2)\langle \epsilon_{12} \rangle_{ss} \end{pmatrix}, \quad (4.10)$$

where $\alpha = 12k_1^2\phi_{1,ss}^2 + 2k_1k_2(3\phi_{1,ss} + \phi_{2,ss}) + k_2^2$. The steady-state values of the relevant correlators are obtained by solving the Lyapunov Eq. 4.4 with the time derivative set to zero, leading to:

$$\begin{aligned} \langle \epsilon_{11} \rangle_{ss} = & \beta^{-1} [(8k_1^2\phi_{1,ss}^2 + (5\phi_{1,ss} + \phi_{2,ss})k_2k_1 + k_2^2)D_{1,1,ss}^{(0)} \\ & - 2k_1\phi_{1,ss}(2k_1\phi_{1,ss} + k_2)D_{1,2,ss}^{(0)} + 2k_1^2\phi_{1,ss}^2D_{2,2,ss}^{(0)}], \end{aligned} \quad (4.11)$$

$$\begin{aligned} \langle \epsilon_{12} \rangle_{ss} = & \beta^{-1} [((2\phi_{1,ss}^2 - 2\phi_{1,ss}\phi_{2,ss})k_1^2 + (\phi_{1,ss} - \phi_{2,ss})k_2k_1)D_{1,1,ss}^{(0)} \\ & + (4\phi_{1,ss}(\phi_{2,ss} + 2\phi_{1,ss})k_1^2 + 2(3\phi_{1,ss} + \phi_{2,ss})k_2k_1 + k_2^2) \\ & \times D_{1,2,ss}^{(0)} - k_1\phi_{1,ss}(2(2\phi_{1,ss} + \phi_{2,ss})k_1 + k_2)D_{2,2,ss}^{(0)}], \end{aligned} \quad (4.12)$$

$$\begin{aligned} \langle \epsilon_{22} \rangle_{ss} = & \beta^{-1} [2(\phi_{1,ss}^2 - 2\phi_{1,ss}\phi_{2,ss} + \phi_{2,ss}^2)k_1^2D_{1,1,ss}^{(0)} + ((-\phi_{1,ss}\phi_{2,ss} \\ & - \phi_{2,ss}^2 + 2\phi_{1,ss}^2)4k_1^2 + (\phi_{1,ss} - \phi_{2,ss})2k_2k_1)D_{1,2,ss}^{(0)} \\ & + ((\phi_{2,ss}^2 + 7\phi_{1,ss}^2 + 4\phi_{1,ss}\phi_{2,ss})2k_1^2 + (3\phi_{2,ss} + 7\phi_{1,ss}) \\ & \times k_2k_1 + k_2^2)D_{2,2,ss}^{(0)}], \end{aligned} \quad (4.13)$$

where

$$\beta = 2(12k_1^2\phi_{1,ss}^2 + 6k_1\phi_{1,ss}k_2 + 2k_1\phi_{2,ss}k_2 + k_2^2)(k_1\phi_{2,ss} + 3k_1\phi_{1,ss} + k_2). \quad (4.14)$$

The entries of the symmetric, positive semi-definite matrix $\mathbf{D}^{(0)}$ can be computed from the stoichiometry matrix $\boldsymbol{\nu}$ and the rate-function vector $\mathbf{T}_0(\boldsymbol{\phi})$:

$$D_{1,1,ss}^{(0)} = b^2k_0 + \phi_{1,ss}(4k_1\phi_{1,ss} + 2k_1\phi_{2,ss} + k_2), \quad (4.15)$$

$$D_{1,2,ss}^{(0)} = 2k_1\phi_{1,ss}(\phi_{2,ss} - \phi_{1,ss}), \quad (4.16)$$

$$D_{2,2,ss}^{(0)} = k_1\phi_{1,ss}^2 + 2k_1\phi_{1,ss}\phi_{2,ss} + k_2\phi_{2,ss}. \quad (4.17)$$

The vector $\boldsymbol{\delta}$ in Eq. 4.10 can then be expressed in terms of the macroscopic rate constants and the steady-state solution of the RRE. The three critical volumes of the system are obtained by substituting the components of $\boldsymbol{\delta}$ in Eq. 4.7.

4.3 Simulation details

The theoretical prediction of the critical volume (see Eq. 4.7) at which the trimerization system undergoes inversion is tested using PDM (see Sec. 3.3). The volume dependence enters the stochastic simulation through the propensities of the input and bimolecular reactions, which scale as Ω and $1/\Omega$, respectively, in accordance with the CME (Gillespie, 1976, 1992; van Kampen, 2001) (see Eq. A8.3). The molecular concentrations ϕ_i^* are computed from the simulations by ensemble-averaging the absolute number of molecules of species i over 20,000 independent realizations, and dividing by Ω .

Steady-state conditions require that the influx of monomers equals their efflux, i.e., $bk_0 = k_2(\phi_1^* + 2\phi_2^* + 3\phi_3^*)$. The efflux is equal to the sum of three expressions: the first is the efflux of monomers, the second is the efflux of monomers which are part of a dimer, and the third is the efflux of monomers which are part of a trimer. This condition is also valid for the deterministic RRE, i.e., $bk_0 = k_2(\phi_1 + 2\phi_2 + 3\phi_3)$. Denoting the volume of a single monomer by v , the average occupied volume fraction (at steady state) is

$$\Phi = vbk_0/k_2. \quad (4.18)$$

Here, we choose to fix $\Phi = 0.1$ and $v = 0.01$ (hence $bk_0/k_2 = 10$), such that the comparison between the solutions of the RRE and the EMRE at different volumes is performed at constant volume fraction. Imposing the mass-balance condition has some important consequences:

1. It helps us isolate the role of Ω from that of densification by keeping Φ constant as we vary Ω across systems of fixed b , v , k_1 and k_2 .
2. It helps isolate the effect of burst size b from that of influx $bk_0\Omega$ by keeping bk_0 constant as we vary b and Ω across systems of fixed Φ , v , k_1 and k_2 .
3. The RRE does not discriminate between (i) a burst input $\emptyset \xrightarrow{k} bA$ at rate k and a non-burst input $\emptyset \xrightarrow{bk} A$ at rate bk , (ii) different volumes Ω (see Appendix A7). This renders

the kinetics predicted by the RRE independent of the position in (Ω, b) parameter space for fixed k_1 and k_2 . Hence, the deviation of the stochastic kinetics arises solely due to the noise sources b and Ω^{-1} .

Given the steady-state condition and the definition of the average occupied volume fraction, it can be deduced that the volume of the compartment and the total number of monomers (free and bound in dimers and trimers, i.e., $N_m = \langle n_1^* \rangle + 2\langle n_2^* \rangle + 3\langle n_3^* \rangle$) are related as: $\Omega = (v/\Phi)N_m$. Since we have fixed the volume fraction, the total number of monomers N_m decreases as the volume decreases. The quantity N_m is an upper bound for the actual number of discrete molecules in the system. Hence, instead of using N_m as a measure of system discreteness, we use the total number of molecules in the compartment as predicted by the RRE, $n_T = \Omega(\phi_1 + \phi_2 + \phi_3)$. A smaller n_T indicates a higher level of discreteness, and vice versa.

4.4 Results

Using PDM simulations we first verify the existence of the predicted discreteness-induced concentration inversion for a particular set of parameters. This example also helps clarify the nature of the effect, since our presentation so far has been purely abstract. We then present the results of a large-scale scan of parameter space using both simulations and the theoretical expressions previously derived, with the aim of validating the theory's ability of delineating the regions of parameter space where RRE predictions qualitatively fail.

Figure 4.2 shows the time evolution of the concentrations from both ensemble-averaged PDM simulations (solid lines) and the RRE (dashed lines) for three different compartment volumes $\Omega = 1000, 12, 3$ with identical rate constants $k_0 = 1/3$, $k_1 = 0.5$, and $k_2 = 1.0$ and a fixed burst size $b = 30$. The discreteness of the system as quantified by n_T becomes more apparent as the volume decreases. The concentrations from the RRE and the large-volume, large-copy-number ($\Omega = 1000$, $n_T = 5195$) stochastic simulations are in good agreement (Fig. 4.2a). The two critical volumes of this system as predicted by our theory (i.e., Eq. 4.7 together with Eqs. 4.10–4.13) are $\Omega_{12} = 2.79$ and $\Omega_{13} = 11.83$. Indeed, at a compartment volume of $\Omega = 12$ (Fig. 4.2b), the stochastic simulations show that the mean steady-state concentrations of species 1 and 3 have become equal. For smaller volumes, the mean steady-state concentration of species 3 is larger than that of species 1 (Fig. 4.2c). The simulations thus verify the existence of the postulated discreteness-induced inversion effect. It is interesting that the inversion occurs at a copy number of $n_T = 62$, which is relatively large considering that stochastic effects are usually deemed significant only in conditions characterized by a mere few molecules. The inversion effect between species 1 and 2, which is theoretically predicted to occur at a smaller volume than that between species 1 and 3, is not found in our simulations (Fig. 4.2c). We will come back to this point later.

In order to further test the theory, we consider the (Ω, b) parameter space, where Ω and b take values between 1 and 1000 in increments of 10. These two variables are convenient because Ω provides a measure of the intrinsic noise, while b measures extrinsic noise. The constants k_1 and k_2 are fixed to the same values as in the previous example. The parameter k_0 is determined by the condition $bk_0/k_2 = 10$, which guarantees that the comparison between the RRE and EMRE is performed at constant volume fraction at all points in parameter space

(see Sec. 4.3). Since $k_0 b$ is a constant, the deterministic RRE prediction is independent of (Ω, b) . In contrast, the magnitudes of intrinsic and extrinsic noise vary from one (Ω, b) to another, leading us to expect inversions in some regions of parameter space, but not in others.

We perform simulations for 100×100 pairs of (Ω, b) values. The solid black and gray regions in Fig. 4.3a show where the simulations gave zero and one inversions, respectively. The orange dashed line shows the theoretical prediction of the critical volume Ω_{13} as a function of b , calculated using Eq. 4.7 together with Eqs. 4.10–4.13. The line accurately demarcates the solid black and grey regions, thus verifying the theory’s ability to predict where the deterministic RRE model qualitatively fails. This is also significant from a computational point of view since the theoretical calculation can be performed much faster than scanning the parameter space (in Fig. 4.3) using an SSA simulation. The red dashed line shows the theoretically predicted critical volume Ω_{12} as a function of b , which should demarcate the regions of one and two inversions. However, the simulations show that there is no second inversion in the parameter space (a similar observation, but for specific parameters, was already pointed out in the previous example of Fig. 4.2). This disagreement with theory is not unexpected since EMRE break down for sufficiently small volumes (Grima, 2010a). The largest theoretical critical volume is hence likely to be the only reliable prediction, but also the most important one. This is because the line in parameter space associated with the maximum of all critical volumes of a system demarcates the two most important regions: the region where the steady-state predictions of the RRE and the LNA are qualitatively valid, and the region where these predictions are incorrect for at least one pair of species.

The fact that Ω_{13} is a linear function of b can be deduced from theory as follows: Our scan of parameter space is at constant $k_0 b$, which implies that the macroscopic concentrations are constant and that $D_{1,1}^{(0)} = b^2 k_0 + \phi_1(4k_1\phi_1 + 2k_1\phi_2 + k_2)$ is a linear function of b (the other elements of $\mathbf{D}^{(0)}$ are independent of b ; see Section 4.1). Thus, it follows from Eq. 4.7 and Eqs. 4.10–4.13 that all critical volumes of the system are linear functions of b . This can be shown to be generally true for any chemical reaction network with burst input at steady-state. The increase in critical volume with b shows that the addition of extrinsic noise increases the threshold copy number of molecules below which stochastic effects become determinant to the steady-state behavior of the system.

We also numerically compute maps of the type shown in Fig. 4.3a for different values of the decay constant k_2 . For each value we compute the fraction $f_{i,j}$ of the (Ω, b) -space ($1 \leq (\Omega, b) \leq 1000$) where there is at least one inversion in the concentrations of species i and j (see Fig. 4.3b). The agreement between EMRE (solid lines) and numerical predictions (data points) is very good. Consider first the variation of $f_{1,3}$ (yellow) with k_2 : it is zero for $k_2 \lesssim 0.8$, has a sharp peak up to one at $k_2 \simeq 0.8$, and then decreases smoothly back to zero with further increasing k_2 . This implies that the effect due to molecular discreteness can be felt across all of the considered parameter space at that particular value of k_2 . This can be explained as follows: At $k_2 = 0.77$, the macroscopic concentrations of species 1 and 3 are precisely equal. Discreteness invariably induces a correction to the RRE concentrations. Even if this correction is very small, for example at very large volumes, it is sufficient to break the strict equality between the macroscopic concentrations of species 1 and 3, and hence to induce an inversion across all of parameter space. Finite-volume corrections to the macroscopic concentrations are always non-zero, but they can either amplify or diminish the existing difference between the

4.4 Results

two macroscopic concentrations. Of these two cases, the former cannot lead to inversion and occurs for $k_2 < 0.77$, while the latter naturally leads to inversion and occurs for $k_2 > 0.77$. This also explains the discontinuous rise in $f_{1,3}$ at $k_2 = 0.77$ and the smooth decay beyond this point.

The peak in $f_{2,3}$ (red) at $k_2 = 4.45$ similarly coincides with the value of k_2 at which the RRE predict equal macroscopic steady-state concentrations for species 2 and 3. In contrast, we find no peak in $f_{1,2}$ (blue), which monotonically increases as k_2 approaches zero. For the chosen parameter values, the macroscopic steady-state concentration of species 2 monotonically approaches (from below) that of species 1 as k_2 approaches zero. The size of the correction needed to cause inversion hence becomes smaller as k_2 approaches zero. This also corresponds to a larger critical volume, which explains the monotonic increase of $f_{1,2}$ in the same limit. The general significance of the existence of peaks in $f_{i,j}$ at the point where $\phi_i = \phi_j$ is that the region of parameter space where discreteness-induced effects are most conspicuous can be deduced directly from the deterministic RRE.

So far we have assumed that the burst-size parameter b is a sure variable. In our model, b could for example represent the number of monomers carried by a vesicle in active intracellular transport. Clearly, different vesicles may carry different numbers of monomers, rendering b itself a stochastic variable. In the absence of experimental information about the probability distribution of b , we use data acquired on bursty phenomena in protein synthesis. Experiments have shown that proteins are produced in bursts, where the number of molecules per burst is distributed according to an exponential distribution (Cai et al., 2006). We therefore also study the situation where the burst size b is sampled, independently for each input event, from a geometric probability distribution, $P(b) = p(1-p)^{b-1}$, where $p = \langle b \rangle^{-1}$, and $\langle b \rangle$ is the average burst size. This distribution is the discrete analog of the exponential distribution. Simulations of the CME and of the RRE are rerun with stochastic b ; the results are shown in Fig. 4.4. Comparing Fig. 4.3a and Fig. 4.4 shows that the fraction of parameter space characterized by an inversion when b is a stochastic variable is significantly larger than when b is a sure variable. In the present example, this fraction is almost 1.8 times larger when burst is stochastic. In other words, increasing the magnitude of extrinsic noise in the system enlarges the region of parameter space where the RRE are qualitatively incorrect.

The new line demarcating the region of no inversion from that characterized by a single concentration inversion can be approximately predicted by a simple, albeit heuristic modification of the EMRE, which we now discuss. The EMRE predictions are based on the solution of the RRE. The RRE in the case where b is stochastic are also stochastic themselves, and hence their average concentrations must be obtained by ensemble-averaging over a large number of independent realizations. The concentrations ϕ_1 and ϕ_2 to be used in the EMRE (Eqs. 4.7, 4.10–4.13) are set to equal the averaged concentrations obtained from solving the stochastic RRE. Furthermore, the EMRE explicitly depend on b via $D_{1,1}$. We hence replace b^2 in $D_{1,1}$ by its average computed over the geometric distribution, i.e., $\langle b^2 \rangle = \langle b \rangle(2\langle b \rangle - 1)$. The two critical volume predictions of the so-modified EMRE are shown as dashed lines in Fig. 4.4. The predicted line separating the regions of zero and one inversion (orange dashed line) is in rough agreement with the border between the black and gray regions obtained from simulations. The decreased accuracy of the theoretical predictions compared to the case of constant b clearly stems from the fact that the present modifications to the EMRE to include stochastic b are heuristic and not derived from first principles as for the case of constant b .

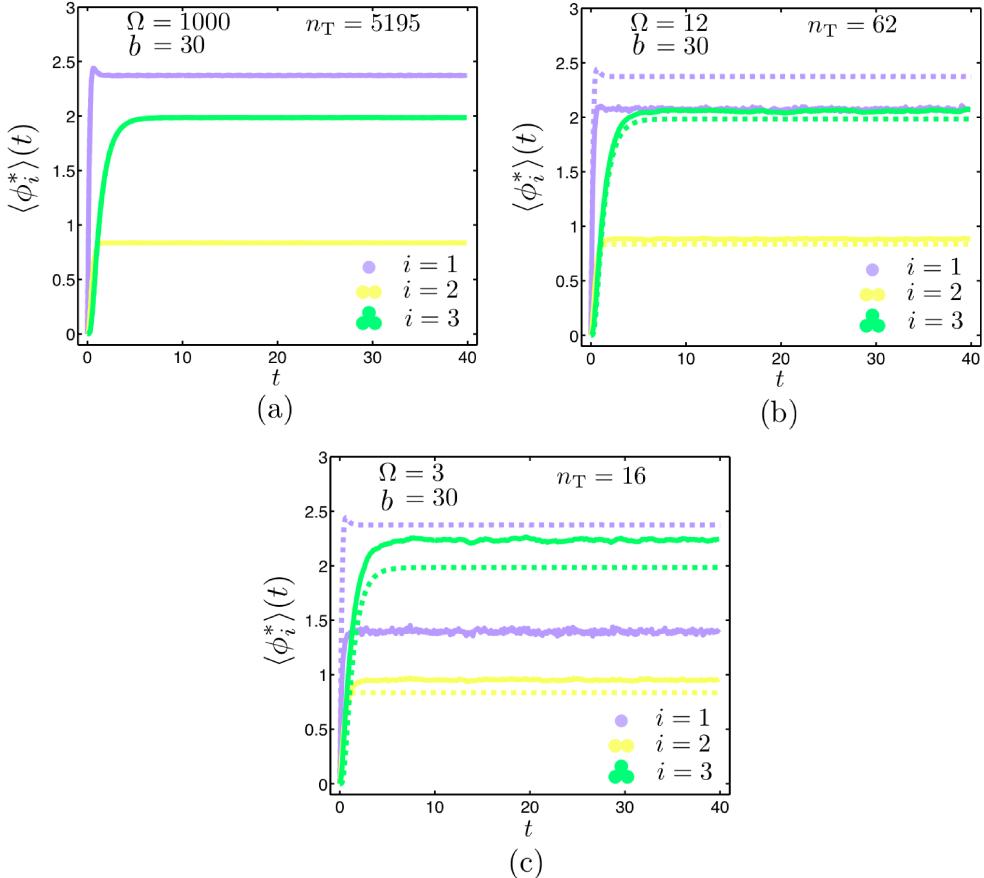


Figure 4.2: Ensemble-averaged concentrations (from exact stochastic simulations) versus time for the trimerization reactions in different compartment volumes: (a) $\Omega = 1000$, (b) $\Omega = 12$, (c) $\Omega = 3$. The rate constants are fixed to $k_0 = 1/3$, $k_1 = 0.5$, and $k_2 = 1$ and the input burst-size to $b = 30$. The dashed lines show the RRE predictions for the same parameter values. The simulations confirm the theoretical prediction of a discreteness-induced inversion below a critical volume, $\Omega_{13} = 11.83$, for species 1 and 3. The total concentration of monomers in free and bound states is constant at all volumes, $\phi_1^* + 2\phi_2^* + 3\phi_3^* = bk_0/k_2 = 10$ (as is also the occupied volume fraction), a condition enforced by steady-state conditions.

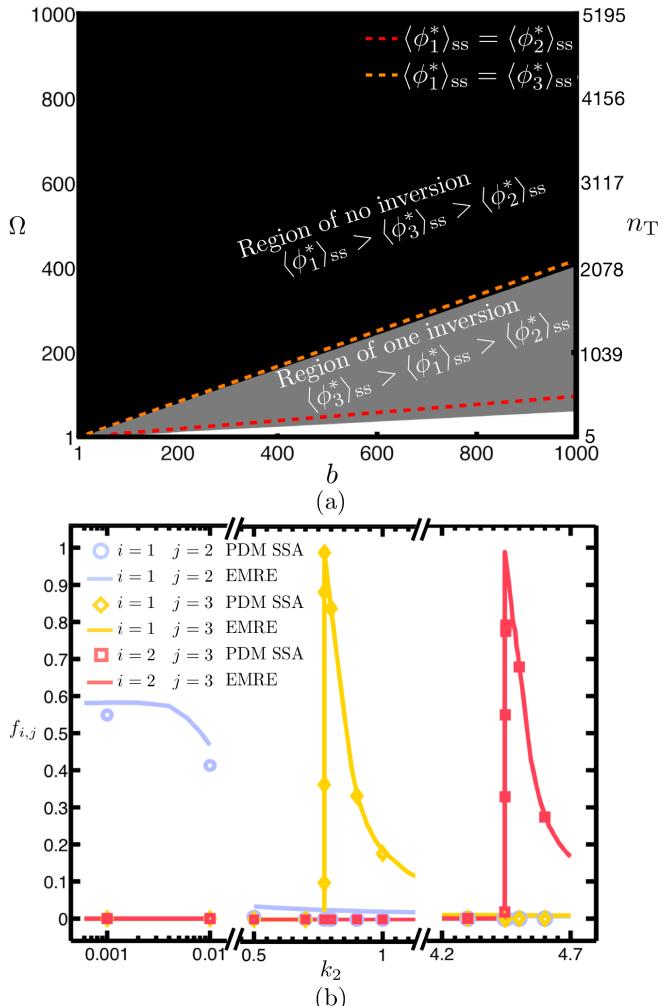


Figure 4.3: Comparison of theory and simulations for the regions of parameter space where the deterministic RRE model fails, i.e., regions with at least one inversion. (a) We consider the (Ω, b) parameter space. The dashed lines are the theoretical predictions, while the solid background color shows the simulation results. The demarcation of parameter space by the dashed lines and by the solid colors is almost coincident, highlighting the agreement between theory and simulation. In the white region, the EMRE breaks down since it predicts negative mean steady-state concentrations. This breakdown is induced by very large fluctuations due to a combination of very low copy numbers and large-burst input. The red line is the equation of the curve for $\Omega_{12}(b) = 0.09658b - 0.1107$ obtained from EMRE. The orange line is the equation of the curve for $\Omega_{13}(b) = 0.4189b - 0.7395$ also obtained from EMRE. The simulations do not show the existence of Ω_{12} , whereas the least-square fit for $\Omega_{13}(b)$ from the SSA simulations is given by $\Omega_{13}(b) = 0.4035b + 4.418$. (b) Plot of the fraction $f_{i,j}$ of parameter space with at least one inversion between species i and j versus the decay rate constant k_2 . The peaks identify the conditions for which there is maximum difference between the predictions of deterministic and stochastic models (see text for discussion). In both figures, the rate constants are $k_0 = 10k_2/b$ and $k_1 = 0.5$ with $k_2 = 1$ in (a) and varying between 10^{-3} and 4.7 in (b).

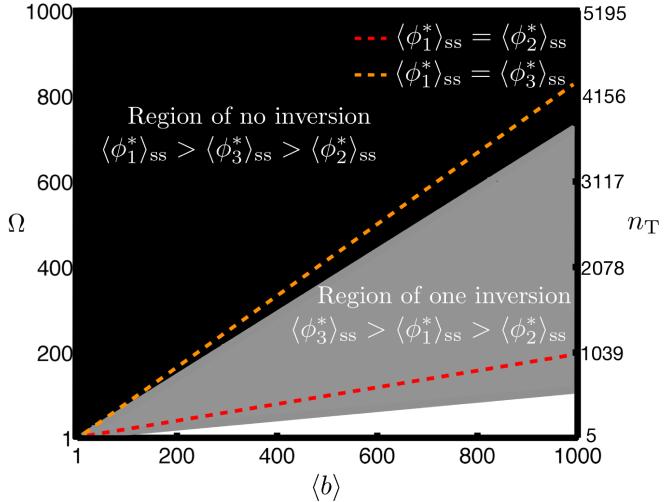


Figure 4.4: Comparison of theory and simulations for the regions of parameter space where the deterministic RRE model fails when the burst-size parameter b is a stochastic variable. The probability distribution of the latter is a geometric distribution with mean $\langle b \rangle$. All parameter values are the same as for Fig. 4.3a. The dashed lines are the theoretical predictions, while the solid background color shows the simulation results. Comparing Fig. 4.3a with the present shows that the increase in extrinsic noise caused by making b stochastic leads to a larger proportion of parameter space where the RRE are qualitatively incorrect (i.e., larger regions of inversion). The theoretical predictions here are obtained using a heuristic modification of the EMRE to approximately account for the randomness in burst size. The red line is the equation of the curve for $\Omega_{12}(b) = 0.1954\langle b \rangle - 0.305$ obtained from the modified EMRE. The orange line is the equation of the curve for $\Omega_{13}(b) = 0.8594\langle b \rangle - 1.756$ also obtained from the modified EMRE. The simulations do not show the existence of Ω_{12} and the least-square fit for $\Omega_{13}(b)$ from the SSA simulations is given by $\Omega_{13}(b) = 0.7283\langle b \rangle + 3.691$.

4.5 Conclusions and Summary

We have elucidated the effect of discreteness on the mean concentrations in a confined monostable chemical reaction network at steady state. Our analysis considered a setup in which the same chemical system is realized in a number of different volumes and the mean steady-state concentrations are studied as a function of the volume. Molecule numbers decrease with volume, and the apparent discreteness is proportional to the inverse volume. Since both RRE and the LNA are only valid in the limit of large-enough numbers of molecules, we surmised that their predictions break down in small volumes containing small numbers of molecules. Our analysis based on EMRE, a recently developed framework providing corrections to RRE due to system discreteness or, equivalently, due to a finite compartment volume, confirmed our initial hypothesis. The steady-state mean concentrations of a nonlinear chemical system was found to depend on the volume of the compartment in which it is realized. For volumes above a critical value, the mean steady-state concentrations as predicted by our theory qualitatively agree with those predicted by RRE, i.e., the ranking of chemical species according to their mean concentrations is the same for both predictions. For sub-critical systems this is not the case, i.e., the ranking is changed for some pair of species. This discreteness-induced concentration inversion was confirmed by exact stochastic simulations. Furthermore, we have found that the critical compartment volume at which inversion occurs increases with the magnitude of extrinsic noise due to a bursty input process. Our analysis also clarifies that the system size, Ω in van Kampen's system-size expansion, is related to the average total number of molecules in the system, but not in a linear manner. Rather, Ω is directly proportional to a quantity that is an upper bound of the average total number of molecules. While the concentration-inversion effect described here is just one of a wealth of possible discreteness-induced effects, the theoretical framework we have developed provides a relatively simple means of investigating such phenomena and is therefore expected to provide a useful tool for probing the physics of mesoscopic chemical systems.

Our analysis is based on the EMRE and on the CME, both of which impose certain constraints on the conditions that can be reasonably captured within our analytical and simulation approach. The EMRE is based on the volume expansion of the CME, which is only valid for monostable chemical reaction networks (van Kampen, 2001). This is a limitation of the volume expansion, but not of the CME itself. This also implies that the qualitative change of steady state at the critical compartment volume is not due to noise-induced transitions between two or more fixed points of the RRE (see Appendix A8). The inversion effect purely stems from a noise-induced change in the shape of the unimodal probability distribution of copy numbers: deviations from Gaussian increase with system discreteness, leading to corresponding changes in the moments of the distribution. Although a few recent papers (Scott et al., 2007; Samoilov and Arkin, 2006; Grima, 2009b, 2010a, 2009a; Thomas et al., 2010) have studied the renormalization of the probability distribution or of the steady-state concentrations with volume, to our knowledge, this is the first time that a discreteness-induced inversion effect has been found and studied in detail. In other words, the inversion effect can be explained as follows: According to the CME, the mean rate of a bimolecular reaction between species i and j with concentrations $n_i\Omega^{-1}$ and $n_j\Omega^{-1}$, respectively (n_i is number of molecules of species i), is proportional to the mean of the product of the concentrations, $\Omega^{-2}\langle n_i n_j \rangle$. Equivalently, it is proportional to the sum of the covariance of concentration fluctuations, $\Omega^{-2}(\langle n_i n_j \rangle - \langle n_i \rangle \langle n_j \rangle)$, and of the

product of the mean concentrations, $\Omega^{-2}\langle n_i \rangle \langle n_j \rangle$. In the limit of large molecule numbers, the covariance becomes very small and the mean rate is simply proportional to the product of the mean concentrations, $\Omega^{-2}\langle n_i \rangle \langle n_j \rangle$; this is the RRE. As molecule numbers decrease, the covariance grows and the mean rates of bimolecular reactions deviate from those predicted by the RRE, leading to different concentrations. Rates and product concentrations increase if the covariance is positive and decrease if it is negative. Hence, it is possible that if the RRE predict the steady-state concentration of a species to be larger than that of another species, the reverse may be true for small molecule numbers. This is the presented discreteness-induced inversion effect.

Two other implicit assumptions of our approach are that the system is dilute and well stirred, both necessary conditions for the validity of the CME (Gillespie, 2007). While in biochemical systems the latter condition is not always true over the length scale of an entire cell, it may hold in smaller compartments, particularly those smaller than the Kuramoto length (Kuramoto and Yamada, 1976; van Kampen, 2001; Grima and Schnell, 2008). Thus, the assumption of well-mixedness is not problematic for the scenarios studied here. The implicit assumption of diluteness is more difficult to uphold, since the intracellular environment is crowded with various macromolecules that occupy 5% to 40% of the cell's volume (Schnell and Turner, 2004) without participating in a given chemical reaction pathway. Hence, our CME-based approach is only valid for describing reactions in lightly crowded compartments; an extension of the study to intermediately crowded conditions may be possible through the use of CMEs with renormalized propensity functions (Grima, 2010b).

One may ask whether the concentration inversion described here is of importance in any real-world systems. As we show in Appendix A6, discreteness-induced inversion effects also exist in the protein concentration output of a genetic network with negative feedback. This motif is ubiquitous in biology, appearing in such diverse contexts as metabolism (Selkov, 1968), signaling (Goldbeter, 1975), somitogenesis (Lewis, 2003), and circadian clocks (Tyson et al., 1999). In biological systems the gene network considered feeds into more complicated metabolic or signal-transduction networks and it is plausible that the discreteness-induced concentration inversions at the level of the gene network are propagated into these downstream networks.

Practical relevance of the inversion effect requires that the critical volumes be in a physically or physiologically meaningful range. Substituting typical parameter values in Eqs. (4.7)–(4.13): $k_0 \in [10^2, 10^4] \text{ Ms}^{-1}$, $k_1 \in [5 \times 10^4, 5 \times 10^5] \text{ M}^{-1}\text{s}^{-1}$, $k_2 \in [10^5, 10^6] \text{ s}^{-1}$ (Fersht, 1998), and $m = 1000$ (Cai et al., 2008) results in critical volumes in the range 28 nm^3 to $6 \times 10^7 \text{ nm}^3$ (corresponding to spheres of diameters 4 to 500 nm). This suggests that RE predictions may become qualitatively incorrect when modeling chemical reactions inside, e.g., lipid rafts (10 to 200 nm (Pike, 2009)), endosomes and endocytic vesicles (20 to several hundred nanometers (Helmut et al., 2009)), sub-organellar structures in the endoplasmic reticulum (few hundred nanometers (Sbalzarini et al., 2005; Luedke et al., 2005)) and mitochondria (few to hundred nanometers (Tam et al., 2010)), and inside “effective” cytoplasmic compartments (35 to 50 nm (Provance et al., 1993)) created by molecular sieving effects. Similar predictions are expected for reactions occurring in artificial nanoreactors, such as nanofibers, and various biomimetic reactors that typically have diameters of less than a few hundred nanometers (Anzenbacher and Palacios, 2009; Karlsson et al., 2004).

5

Effect of noise on the relaxation kinetics of mesoscopic chemical systems

“NATURE! . . . She is vanity of vanities; but not to us, to whom she has made herself of the greatest importance. She allows every child to play tricks with her; every fool to have judgement over her; thousands to walk stupidly over her and see nothing; and takes her pleasure and finds her account in them all.”

– *Nature: Aphorisms by Goethe, Thomas H. Huxley (1869)*

We investigate the characteristics of fluctuation at a nonequilibrium steady-state of mesoscopic chemical reaction systems. We assume that intrinsic noise is realized by decreasing reactor volume, and extrinsic noise is realized by burst input as described in Chapter 4.

The effect of noise manifests itself differently in different types of chemical reaction networks: In linear reaction networks, RRE predictions of the mean concentrations from the CME are correct, regardless of the reactor volume (van Kampen, 2001; Gillespie, 1991). In nonlinear reaction networks, however, noise induces quantitative differences from the concentrations predicted by RRE (Gillespie, 1991; van Kampen, 2001). This quantitative difference can become large enough to render RRE even qualitatively invalid in certain regimes (Chapter 4). In multi-stable systems, intrinsic noise can lead to switching behavior between the multiple fixed points of the system (Gillespie, 1991; van Kampen, 2001), as shown in Sec. 1.4.4. This phenomenon has, e.g., been used to explain spontaneous switching behavior in biochemical systems (Carrier and Keasling, 1999; Tian and Burrage, 2006; Samoilov et al., 2005) and the switching of genetic programs in response to environmental changes (Kashiwagi et al., 2006). More remarkably, intrinsic noise can induce oscillatory behavior at steady state, even when

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

the corresponding RRE is away from Hopf bifurcation and hence do not exhibit oscillatory behavior (Qian et al., 2002; Baxendale and Greenwood, 2011). This has, e.g., been used to explain circadian rhythms in biological organisms (Barkai and Leibler, 2000; Li and Lang, 2008; Ko et al., 2010).

In this chapter, we investigate the relaxation kinetics of the fluctuations at a nonequilibrium steady-state of both a non-oscillating and an oscillating nonlinear mesoscopic reaction network.

In Sec. 5.1 we study the characteristics of the time-autocorrelation of the steady-state concentration fluctuations in the trimerization model system considered in Chapter 4. The RRE of the trimerization system has a single fixed point. In addition, the mesoscopic system parameterized as in Chapter 4 does not exhibit oscillatory behavior. This can be seen by the relaxation of a small perturbation around the fixed point of the RRE. The eigenvalues corresponding to this relaxation have negative real parts and, importantly, the absolute values of the real parts of the eigenvalues are much larger than the absolute values of their imaginary parts (see Appendix A8). Under this condition, oscillatory behavior is not expected even in the presence of noise (Baxendale and Greenwood, 2011). Moreover, the RRE of the trimerization system are always asymptotically stable in the Lyapunov sense for all rate constants and hence does not exhibit any bifurcations (see Appendix A8).

In Sec. 5.2 we study the frequency spectrum of the steady-state concentration fluctuations of an oscillatory mesoscopic system. We use the Brusselator (Nicolis and Prigogine, 1977; Prigogine, 1980) as a model system. The Brusselator is a chemical reaction system that exhibits Hopf bifurcation. Like the trimerization system, it has a single fixed point. The characteristic of the fixed point is determined by the eigenvalues corresponding to the relaxation kinetics of a small perturbation around the fixed point. These eigenvalues are functions of the rate constants. When these eigenvalues have negative real parts, the Brusselator is monostable. For some values of the rate constants the real parts of the eigenvalues vanish, the Brusselator hence undergoes a Hopf bifurcation and the RRE exhibits limit cycle oscillations when the real parts of the eigenvalues are positive (see Appendix A9). Interestingly, it has been observed that even when the real parts of the eigenvalues are negative, and hence away from Hopf bifurcation, the mesoscopic Brusselator can exhibit oscillatory behavior. These oscillations are especially prominent when the absolute value of the real part of an eigenvalue is comparable to the absolute value of its imaginary part (Qian et al., 2002; Baxendale and Greenwood, 2011). We study the frequency spectrum of the fluctuations around a nonequilibrium steady-state of the mesoscopic Brusselator under this condition. We will restrict ourselves to the effect of intrinsic noise as realized by low copy numbers upon decrease in reactor volume Ω .

5.1 Steady-state relaxation kinetics of mesoscopic non-oscillatory chemical systems

We study the signatures of the two noise sources that cooperate to produce the discreteness-induced concentration inversion observed in Chapter 4: (i) low copy number as created by finite volume Ω and (ii) extrinsic noise due to the burst input b . We quantify their influence on the relaxation kinetics of nonlinear reaction networks. We use the time-autocorrelation function (ACF) of concentration fluctuations around a nonequilibrium steady-state via as

5.1 Steady-state relaxation kinetics of mesoscopic non-oscillatory chemical systems

integral (here we call it lifetime) and derivatives*. For this we use (i) the LNA of the CME via the van Kampen expansion in the system volume (van Kampen, 2001; van Dongen, 1987) and (ii) the full CME via the partial-propensity direct method (PDM) (see Sec. 3.3).

We show that the lifetimes of chemical species are modulated by burst input b and volume Ω (or confinement Ω^{-1}). We quantify lifetime by the integral of the ACF of the concentration fluctuations. The integral or the lifetime of the ACF is the fraction of the power of the fluctuations at zero frequency. Experimentally, in mesoscopic biochemical systems, the ACF can be measured using fluorescence-lifetime imaging microscopy (FLIM) or fluorescence-correlation spectroscopy (FCS) (Lakowicz, 2006).

Analysis of FLIM and FCS spectra, however, is based on deterministic RRE, which is adequate for linear reaction networks (Elson, 1974; Magde, 1974; Qian and Elson, 2004). For nonlinear reaction networks in small volumes and with bursty input the RRE-based analysis might be inadequate. We show that confinement increases the lifetimes of all reactants in a nonlinear reaction. Burst either increases or decreases the lifetimes. Furthermore, we show that the derivatives of the ACF of the concentration fluctuations are affected in opposite ways by burst b and confinement Ω^{-1} , thus discriminating between the two noise source. This directly links the present results to experimental applications in two ways: (i) Knowing the lifetime modulation introduced by confinement and burst may allow estimating reaction rates in experimental systems, since the lifetime is a function of reaction fluxes and covariances, which are in turn functions of burst and confinement. (ii) Derivatives of the ACF can be used to discriminate between the confinement- and burst-induced effects.

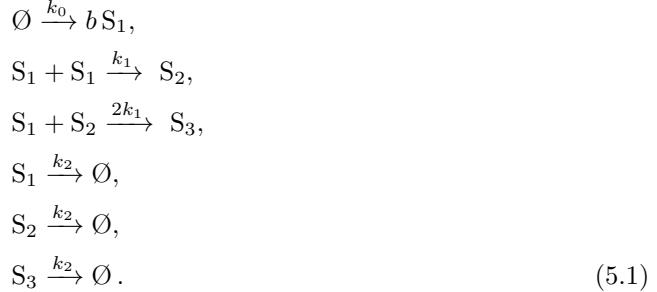
We believe that our findings are useful to: (i) Use FLIM or FCS to measure burst size b and volume Ω when the reaction rates are known. (ii) Correct for the effects of burst input and confinement when experimentally measuring reaction rates. (iii) Understand the mechanisms that deviate stochastic kinetics from its deterministic counterpart and choose the right level of description when modeling nonlinear reaction networks. (iv) Account for the influences of confinement and burst when formulating coarse-grained governing equations of nonlinear reaction models.

In Sec. 5.1.1 we introduce the model. In Sec. 5.1.2 we study the ACF from the LNA, which shows modulation by the burst b alone. In Sec. 5.1.3, using the PDM SSA we numerically generate population trajectories from the full CME as system volume Ω is shrunk and burst b is increased. The ACF of these trajectories has that of the LNA as a baseline. Sec. 5.1.4 provides an analysis of the results and concludes.

*Note that ACF is normalized by the variance such that the total power of the fluctuations is one.

5.1.1 Model

We consider the same reaction network as in Chapter 4, namely the trimerization network described in Eq. 4.9:



We again impose the mass-balance condition as prescribed by Eq. 4.18: $\Phi = v b k_0 / k_2$, where v is the volume of a single monomer and Φ is the average occupied volume fraction at steady state. We solve the CME (Eq. A8.3) approximately using (i) the LNA (Appendix. A5.1), and (ii) numerically generating exact trajectories of the CME using PDM. We then compute the ACF of the concentration fluctuations of species S_n at steady state as

$$R_n(t) = \langle \widetilde{\phi}_n^*(0) \widetilde{\phi}_n^*(t) \rangle_{ss} / \sigma_{\phi_n^*}^2. \tag{5.2}$$

Here, 0 is a time origin at steady-state, i.e. after the initial relaxation period $-\infty < t < 0$, where $-\infty$ represents an arbitrary origin in the past. The average $\langle \cdot \rangle_{ss}$ is taken at steady state at time origins and independent stochastic trajectories, $\widetilde{\phi}_n^* = \phi_n^* - \langle \phi_n^* \rangle_{ss}$ is the fluctuation, and $\sigma_{\phi_n^*}^2 = \langle \phi_n^*(0) \widetilde{\phi}_n^*(0) \rangle_{ss}$ is the steady-state variance.

We compute the lifetime of an aggregate of size n as

$$\tau_n = \int_0^\infty R_n(t) dt \approx \int_0^{t_n^\times} R_n(t) dt, \tag{5.3}$$

where t_n^\times is the first zero crossing of $R_n(t)$. This is a measure of the fraction of the power of concentration fluctuations of the aggregate S_n at zero frequency, in accordance with the Wiener-Khintchine theorem (Gillespie, 1991). Since the fixed point of the trimerization system is a stable spiral, the ACF becomes negative due to oscillatory relaxation, hence the approximation in Eq. 5.3 may not be valid. The frequency of these oscillatory relaxation, however, is small enough compared to the rate of decay of the ACF to justify the approximation in Eq. 5.3 (see Appendix A8).

We also compute the decay-rate function of the ACF as

$$\chi_n(t) = -\frac{d}{dt} R_n(t) \tag{5.4}$$

and the initial curvature of the ACF

$$Z_n = \frac{d^2}{dt^2} R_n(0). \tag{5.5}$$

These quantities serve as (curve) characteristics to study the effects of b and Ω on the kinetics. In addition, they provide a connection with experiments since they can directly be calculated from standard FCS or FLIM read-outs.

In our model, we set $k_2 = 1$, $v = 0.01$, $k_1 = 0.5$, and $\Phi = 0.1$ and compute k_0 using Eq. 4.18. We also limit ourselves to (b, Ω) -regimes where population fluctuations are not larger than their mean. We estimate the bounds of this regime as follows: The mean number of particles at steady state is $\Phi\Omega/v = 10\Omega$. From Eq. A7.10 we see that the standard deviation at steady state without any aggregation, i.e., for a system containing only monomers, is proportional to $(b\phi\Omega/(2v))^{1/2}$ (see Appendix A7). We impose the mean as an upper bound for twice the standard deviation. This imposes a b -dependent lower bound on the system volume: $\phi\Omega/v > 2b$. Interestingly, we observe that below this lower bound for the volume Ω , the EMRE fail as indicated by the white region in Fig. 4.3a.

5.1.2 Low confinement: the linear-noise approximation (LNA)

We approximate the CME associated with our model by the LNA (van Kampen, 2001). The LNA of the master equation is valid at low confinement, i.e., for large enough reactor volumes. We do this in order to (i) obtain baseline kinetics relative to which to interpret the full-CME kinetics provided in the next section (see Sec. 5.1.3), (ii) obtain analytical functions for the ACF, and (iii) reach the large-volume, low-confinement limit where the modulation of the ACF by Ω vanishes, thus isolating the dependence on b .

The details of the LNA are presented in Appendix A5.1. In short, the LNA consists of retaining leading-order terms in a Taylor expansion of the concentration probability function $P(\phi^*, t)$ in the small parameter $\Omega^{-1/2}$. The latter enters after assuming that the fluctuations around the concentration of aggregate S_n scales with reactor volume Ω as $\Omega^{-1/2}\epsilon_n$, where ϵ_n is a random variable evolved by a master equation (van Kampen, 2001; van Dongen and Ernst, 1987; van Dongen, 1987; Grima, 2009b).

In the LNA, (i) the noise ϵ_n is Gaussian, (ii) the mean concentration $\langle\phi_n^*\rangle$ obeys the deterministic RRE, and (iii) the ACF as defined in Eq. 5.2 does not depend on Ω (van Kampen, 2001). Despite this, the LNA remains useful as the covariances do depend on the burst b , as we show in this section.

Considering that in the LNA the covariances $\langle\widetilde{\phi}_n^*\widetilde{\phi}_m^*\rangle$ coincide with the second moments $\langle\phi_n^*\phi_m^*\rangle$, because the mean noise is zero, we solve the time evolution of the first and second moments (see Eqs. A5.27 and A5.31 in Appendix A5.1) at steady state to obtain the ACF at steady state,

$$\begin{aligned} R_1(t) &= a_{1,1}e^{-\gamma_1 t} + a_{1,2}e^{-\gamma_2 t} + a_{1,3}e^{-\gamma_3 t} \\ R_2(t) &= a_{2,1}e^{-\gamma_1 t} + a_{2,2}e^{-\gamma_2 t} + a_{2,3}e^{-\gamma_3 t} \\ R_3(t) &= a_{3,1}e^{-\gamma_1 t} + a_{3,2}e^{-\gamma_2 t} + a_{3,3}e^{-\gamma_3 t}. \end{aligned} \quad (5.6)$$

The coefficients $a_{i,j}$, $i, j = 1, \dots, 3$, are ratios of two functions that are linear in the covariances.

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

The rates γ_n , $n = 1, \dots, 3$, are (Appendix A8)

$$\begin{aligned}\gamma_1 &= k_2 + 3k_1\phi_{1,ss} + k_1\phi_{2,ss} - k_1\sqrt{-3\phi_{1,ss}^2 + 6\phi_{1,ss}\phi_{2,ss} + \phi_{2,ss}^2} \\ \gamma_2 &= k_2 + 3k_1\phi_{1,ss} + k_1\phi_{2,ss} + k_1\sqrt{-3\phi_{1,ss}^2 + 6\phi_{1,ss}\phi_{2,ss} + \phi_{2,ss}^2} \\ \gamma_3 &= k_2,\end{aligned}\tag{5.7}$$

where $\phi_{n,ss}$ is the steady-state RRE concentration of species S_n . Note that γ_1 and γ_2 may have imaginary parts that give the ACF an oscillatory contribution, introducing anticorrelation at late times. These imaginary parts, however, do not point at limit cycle oscillations. They merely show the nature of the fixed point as a stable spiral. This makes the ACF have properties of a weakly underdamped behavior (i.e., almost overdamped) in this case, since $\text{Re } \gamma_n \gtrsim 5 \text{Im } \gamma_n$ (see Appendix A8). By integrating Eq. 5.6 over $[0, \infty)$ we get the lifetimes,

$$\begin{aligned}\tau_1 &= a_{1,1}\gamma_1^{-1} + a_{1,2}\gamma_2^{-1} + a_{1,3}\gamma_3^{-1} \\ \tau_2 &= a_{2,1}\gamma_1^{-1} + a_{2,2}\gamma_2^{-1} + a_{2,3}\gamma_3^{-1} \\ \tau_3 &= a_{3,1}\gamma_1^{-1} + a_{3,2}\gamma_2^{-1} + a_{3,3}\gamma_3^{-1},\end{aligned}\tag{5.8}$$

where the integrals of Eq. 5.6 from their first zero-crossings up to infinity are negligibly small. The corresponding integrals over $[t_n^\times, \infty)$ for the SSA-computed ACFs remain small.

The pre-factors $a_{i,j}$, $i, j = 1, \dots, 3$, are ratios of two functions linear in the burst b because each covariance is linear in b . This is seen by solving Eq. A5.31 at steady state with the mass-balance condition. As a consequence, $a_{i,j}$, $i, j = 1, \dots, 3$, become b -independent at large enough b , and so do the lifetimes. Figure 5.1a shows how the lifetimes depend on burst. As burst increases from the no-burst case $b = 1$, monomer lifetimes decrease and multimer lifetimes increase. As seen from Eq. 5.8, the lifetimes become b -independent at large enough b , Fig. 5.1b. This thus defines a high- b region above $b \approx 300$. It can also be seen from the general form of Eq. 5.8 for N species that, for a nonlinear reaction network at a nonequilibrium steady-state, τ_n will either increase or decrease with b .

Figure 5.2 shows the decay-rate functions $\chi_n(t)$ for several burst values. For monomers, $\chi_1(t)$ remains monotonic as burst increases, with its maximum at $t = 0$. For dimers, $\chi_2(t)$ becomes non-monotonic above a threshold burst $b \approx 10$, while for trimers the threshold sets in before, at $b \approx 6$. In other words, the decay-rate function of the non-aggregating multimers (trimers) is more sensitive to burst than that of the aggregating multimers (dimers). Note that the maximum that develops shifts from being at $t = 0$ towards later times as burst increases the time $t^{\chi_n, \max}$ at which $\chi_n(t)$ reaches its maximum. We interpret $t^{\chi_n, \max}$ as the time of fastest decay, since the (absolute value of the) ACF slope is maximum at this time.

In this section we have calculated the ACF from the linear-noise approximation of the CME, from which we obtained the lifetimes analytically. We observed that the ACF is a superposition of exponentials with pre-factors modulated by the bursty driving, thereby establishing the baseline of the burst-induced modulation of the kinetics.

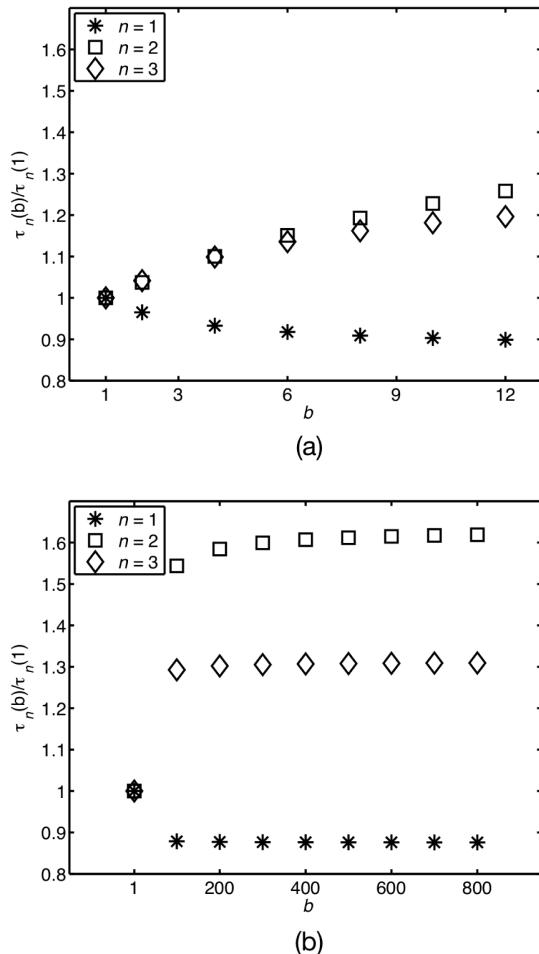


Figure 5.1: Lifetimes from the linear-noise Fokker-Planck approximation at low confinement. Lifetime as a function of burst for (a) small and (b) large bursts, normalized to the no-burst, unit-stoichiometry case $b = 1$ for monomers $n = 1$, dimers $n = 2$ and trimers $n = 3$. The region above ca. $b = 300$ defines the high- b region, where lifetimes become insensitive to b . Note that the lifetime of monomers decreases, whereas that of the dimers and trimers increases.

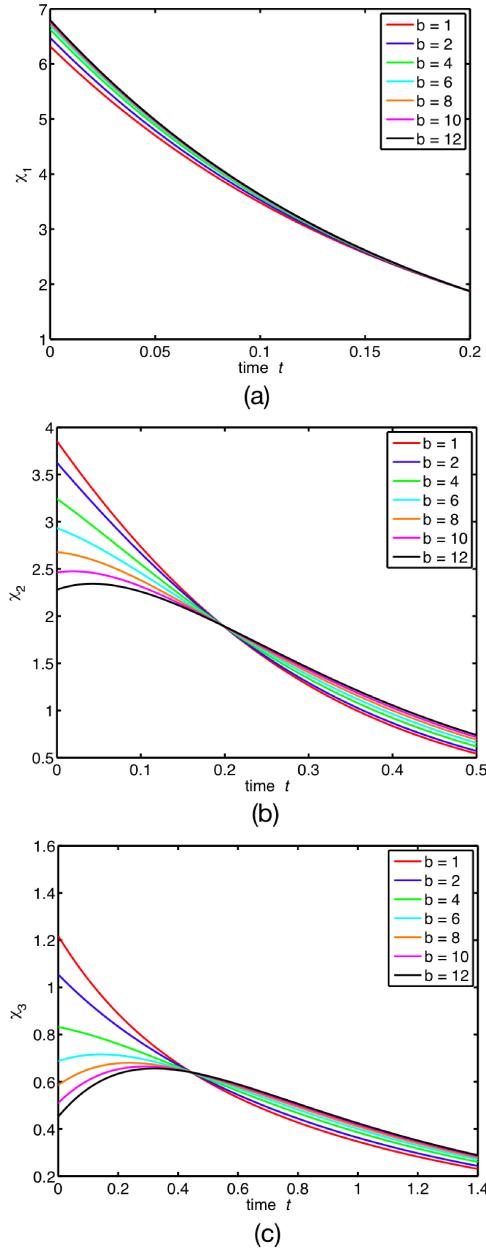


Figure 5.2: Decay-rate functions from the linear-noise Fokker-Planck approximation at low confinement for several burst values b . (a) Monomers $n = 1$; (b) Dimers $n = 2$; (c) Trimers $n = 3$. For dimers and trimers there is a threshold burst above which $\chi_n(t)$ becomes non-monotonic in t . Furthermore, the maximum shifts towards later times with increasing burst b .

5.1.3 Beyond the LNA: the full CME

We showed in the previous section how the ACF depends on burst in the low-confinement limit. In this section we show how higher confinement further modulates this ACF. We compute the stochastic trajectories of the populations n_i as given by the CME to show that shrinking the volume at high-enough confinement further modulates lifetimes and the time of fastest decay. In addition, we introduce the ACF's initial curvature as a further characteristic.

To generate stochastic trajectories from the CME we use the PDM SSA (Sec. 3.3). For each parameter set we generate an ensemble of 20 000 independent trajectories at steady state. Each trajectory is roughly $20(k_2)^{-1}$ long and we record 4 000 equi-spaced data points with a time resolution of $0.005(k_2)^{-1}$. The initial condition for each trajectory is $n_i(-\infty) = 0$, where $-\infty$ represents an arbitrary origin in the past and $-\infty < t < 0$ is a period of relaxation to steady state.

5.1.3.1 Lifetime

Figure 5.3 shows the lifetimes $\tau_n(\Omega)$ as a function of volume Ω for both no burst ($b = 1$) and a burst value in the high-burst regime observed in the LNA limit ($b = 500$). We see that shrinking Ω increases τ_1 and τ_2 , but not τ_3 , and that this effect is more appreciable at larger Ω as the burst b increases.

Figure 5.4 shows maps of lifetime versus volume for a burst range. The trimers' map shows that volume does not affect lifetime, as also seen in Fig. 5.3. Figure 5.4 shows that for monomers and dimers, increasing burst b extends the Ω -interval over which the lifetimes vary with Ω . This can also be seen in Fig. 5.3. In other words, burst seems to act as an amplifier for confinement-induced lifetime modulation. This is analogous to the discreteness-induced inversion occurring at larger volumes with increasing burst size (see Fig. 4.3a).

The monomer lifetime τ_1 deserves special attention because it is the only lifetime that is non-monotonic in the burst b , see Fig. 5.4a. For any Ω fixed in the interval $100 < \Omega \leq 1000$, τ_1 decreases with b and then increases back for b beyond some threshold b^{τ_1} . The threshold b^{τ_1} , in turn, decreases with confinement Ω^{-1} . The non-monotonicity of $\tau_1(b)$ is a high-confinement effect because it does not occur in the LNA, see Fig. 5.1. The existence of the threshold b^{τ_1} , nonetheless, is not surprising because for monomers, confinement and burst cause opposing modulations: confinement increases lifetimes whereas, as seen from the LNA, burst decreases them. Since burst amplifies the confinement-induced modulation of the lifetimes, it acts as a $-/+$ switch for it. Another interesting observation is the close proximity of the line $\Omega(b^{\tau_1})$ to the critical volume $\Omega_{13}(b)$ that demarcates the region of inversion from the region of no inversion in Fig. 4.3a. At this point the relationship between the proximity of these lines remains unexplored and unexplained.

We can also view the problem from the perspective of how confinement affects burst-induced lifetime modulation: varying b while we fix Ω below the LNA limit, see Fig. 5.4. In other words, by looking into a hypothetical volume-dependent, high-confinement version of Eq. 5.8. Note also that the lifetimes $\tau_2(b)$ and $\tau_3(b)$ are the only lifetimes increasing with burst b in the LNA limit. Recall that further confinement Ω^{-1} allows the decreasing function $\tau_1(b)$ to acquire a slope of the same sign of that of $\tau_2(b)$ and $\tau_3(b)$ for large enough burst b . This suggests that confinement is an amplifier of burst-induced lifetime modulation. This amplification, in turn, must result from $O(\Omega^{-\alpha})$ terms entering $a_{i,j}$, $i, j = 1, \dots, 3$, and/or $O(\Omega^\alpha)$ terms

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

entering γ_j , $j = 1, \dots, 3$, in Eq. 5.8 for some $\alpha > 0$. This is in agreement with van Kampen's observation that nonlinearity in chemical reaction networks gives rise to additional Debye terms in the fluctuation spectrum (van Kampen, 2001), making them more relevant at higher confinement.

In summary, we have shown that confinement increases the lifetimes of all species that are reactants in a bimolecular reaction. Confinement-induced modulation acts on top of the burst-induced modulation seen in the LNA limit. It provides an effective modulation that may lead to non-monotonic behavior.

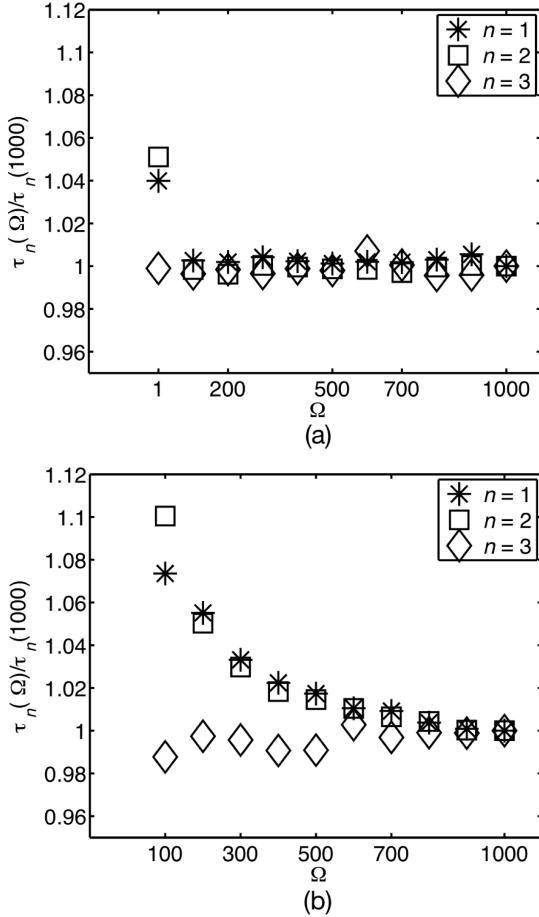


Figure 5.3: Lifetime from the full-CME trajectories. We show the lifetimes as a function of system volume Ω for constant burst b , normalized to its corresponding $\Omega = 1000$ system. (a) No burst, $b = 1$; (b) Higher burst, $b = 500$ for monomers $n = 1$, dimers $n = 2$ and trimers $n = 3$. Note that the system becomes insensitive to Ω at large-enough Ω , as the linear-noise approximation predicts (see Sec. 5.1.2). As volume decreases, the system departs from linear-noise behavior. Note that trimers are insensitive to volumes in the present example as they are not reactants in any nonlinear reaction.

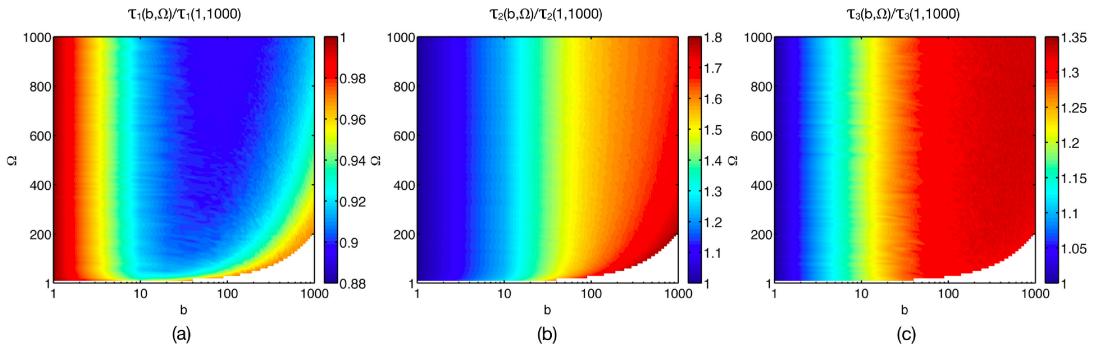


Figure 5.4: Lifetimes from the full-CME trajectories. We show the lifetimes normalized to their value at $(b, \Omega) = (1, 1000)$. (a) Monomers $n = 1$; (b) dimers $n = 2$; (c) trimers $n = 3$. The void region for small Ω and large b corresponds to population fluctuations becoming larger than the mean. The plot show an interpolation of data sampled at intervals $(\Delta b, \Delta \Omega) = (10, 10)$.

5.1.3.2 Derivatives of the time-autocorrelation function

Figure 5.5 shows representative samples of how the decay-rate function $\chi_n(t)$ responds to volume shrinking at burst $b = 500$. This burst value corresponds to a large burst regime for the multimers ($n = 2, 3$) at low confinement, see Fig. 5.2. Our aim here is to study how confinement alters this low-confinement behavior. We look for qualitative features that correlate with changes in volume Ω and burst b . These features may possibly be used to develop quantitative methods to characterize local volume and burst from FCS-sampled ACFs.

From Fig. 5.5 we can see that for monomers, $\chi_1(t)$ is monotonic. For multimers ($n = 2, 3$), $\chi_n(t)$ is non-monotonic, making the time $t^{\chi_n,\max}$ at which $\chi_n(t)$ is maximum greater than zero i.e., $t^{\chi_1,\max} = 0$, $t^{\chi_2,\max} > 0$ and $t^{\chi_3,\max} > 0$. This change in monotonicity of $\chi_2(t)$ and $\chi_3(t)$ is a purely burst-induced modulation and exists already in the LNA limit (see Sec. 5.1.2). Note that confinement reduces $t^{\chi_n,\max}$, as opposed to burst, which increases it, see Fig. 5.2.

Up to now we have studied two-dimensional datasets $\{(t, \chi_n)\}$. To facilitate feature detection in an FCS experiment, it would be desirable to reduce the dimensionality to one. To this end we study the ACF's initial curvature Z_n . Since $Z_n = -\frac{d}{dt}\chi_n(0)$, from Fig. 5.5 we see that Z_n is monotonic for all species as the volume shrinks.

Figure 5.6 shows the ACF's initial curvature Z_n for burst and volume ranges. For monomers, confinement *increases* Z_1 , more noticeably at larger burst. Moreover, $Z_1 > 0$, reflecting the monotonicity of $\chi_1(t)$. For multimers ($n = 2, 3$), on the contrary, confinement *reduces* the ACF's initial curvature from a positive to a negative value as we go from the small- b -large- Ω region to the large- b -small- Ω region. This reflects the non-monotonicity of $\chi_n(t)$ for $n > 1$, beyond a burst threshold. In other words, the change of monotonicity is a purely burst-induced modulation, also at high confinement. There is no qualitative difference between aggregating ($n = 2$) and non-aggregating ($n = 3$) multimers.

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

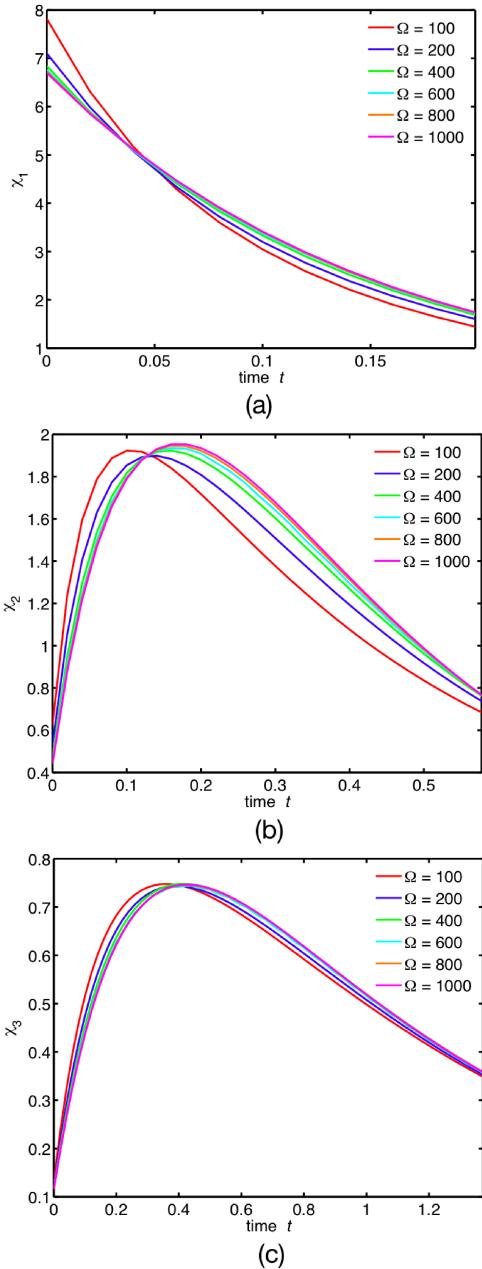


Figure 5.5: Decay-rate functions from full-CME trajectories. We show the decay-rate functions $\chi_n(t)$ for (a) monomers $n = 1$, (b) dimers $n = 2$, and (c) trimers $n = 3$ as volume shrinks at $b = 500$. The time $t_2^{\chi, \text{max}}$ is defined as the position of the maximum. Shrinking volume alone reduces $t_2^{\chi, \text{max}}$, as opposed to increasing b , see Fig. 5.1. A similar trend is also shown by the trimers.

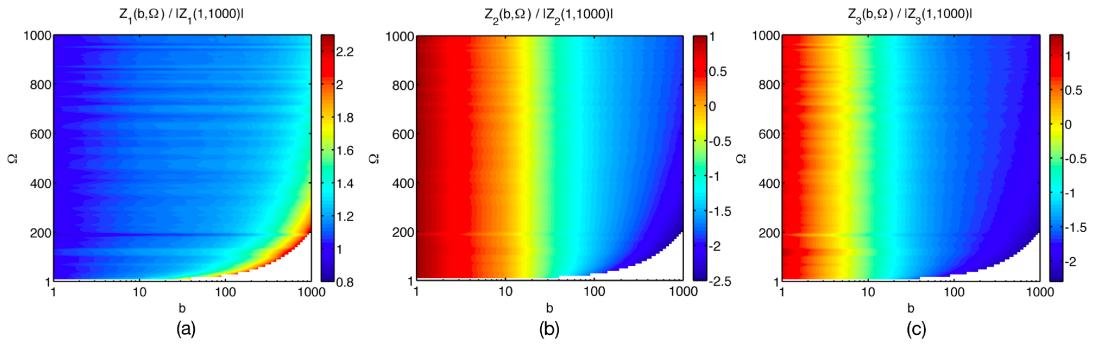


Figure 5.6: ACF initial curvature from full-CME trajectories. We show the ACF's initial curvature, $Z_n \equiv \frac{d^2}{dt^2} C_{nn}(0) = -\frac{d}{dt} \chi_n(0)$, normalized by its absolute value at $(b, \Omega) = (1, 1000)$; (a) for monomers $n = 1$, (b) dimers $n = 2$, and (c) trimers $n = 3$. This quantity serves as a lower-dimensional read-out of the decay-rate function $\chi_n(t)$. The void region for small Ω and large b corresponds to population fluctuations becoming larger than the mean. The plots show an interpolation of data sampled at intervals $(\Delta b, \Delta \Omega) = (10, 10)$.

5.1.4 Conclusions

In Table 5.1 we summarize the behavior of the most relevant characteristics we studied, which can be obtained *a posteriori* from standard FCS or FLIM read-outs. This table may serve as a reference for contrasting burst-induced and confinement-induced modulations and may be useful for later studies of the mechanisms causing them. An immediate use may be to help discern whether the noise source is burst-induced or confinement-induced.

Characteristic	LNA $(\Delta_b, \Delta_{\Omega^{-1}})$	CME $(\Delta_b, \Delta_{\Omega^{-1}})$
τ_1	$(-, 0)$	$(\curvearrowleft, +) \spadesuit$
τ_2	$(+, 0)$	$(+, +)$
τ_3	$(+, 0)$	$(+, 0) \star$
Z_1	$(+, 0)$	$(+, +)$
Z_2	$(-, 0) \blacktriangle$	$(-, +) \blacktriangle \spadesuit$
Z_3	$(-, 0) \blacktriangle$	$(-, +) \blacktriangle \spadesuit$
$t^{x_1, \max}$	$(0, 0)$	$(0, 0)$
$t^{x_2, \max}$	$(+, 0)$	$(+, -)$
$t^{x_3, \max}$	$(+, 0)$	$(+, -)$

Characteristics upon increasing burst b and confinement Ω^{-1} , encoded as pairs $(\Delta_b, \Delta_{\Omega^{-1}})$, where $\Delta_b, \Delta_{\Omega^{-1}} \in \{+, -, 0, \curvearrowleft\}$ is the modulation of the relevant characteristic as b or Ω^{-1} increases, respectively, while keeping the other constant. Here τ_n is the lifetime, Z_n is the initial curvature of the ACF and $t^{x_n, \max}$ is the time at which the decay rate of the ACF is maximum for monomers $n = 1$, dimers $n = 2$ and trimers $n = 3$ (see Sec. 5.1.1). The modulation states are positive (+), negative (-), negligible or zero (0), and decreasing-then-increasing (\curvearrowleft). \spadesuit : \curvearrowleft because there exists a competition of burst-induced versus confinement-induced modulation. \star : $\Delta_{\Omega^{-1}} \tau_n \equiv 0$ for species reacting only unimolecularly. \blacktriangle : $\Delta_b Z_n$ decreases from positive to negative, reflecting the role of burst in changing $\chi_n(t)$ monotonicity. \spadesuit : $\Delta_{\Omega^{-1}} Z_n$ does not change sign, hence Ω^{-1} does not change $\chi_n(t)$ monotonicity.

Table 5.1: ACF characteristics upon increasing burst b and confinement Ω^{-1} .

The presence of anti-correlations implies that care must be taken when relating lifetimes as we define them to the correlation time τ_c in the expression for the ACF $R(t) \propto e^{-t/\tau_c} \cos \omega_0 t$.

5.1 Steady-state relaxation kinetics of mesoscopic non-oscillatory chemical systems

The quantity we refer to as lifetime τ is the integral of the ACF. In other words, the lifetime is the power spectral density of the fluctuations at zero frequency. Only in the absence of oscillations $\omega_0 = 0$ can we say that the lifetime τ is the same as the correlation time τ_c . In our case, τ_c^{-1} is at least five times larger than ω_0 in the LNA, and hence the oscillatory relaxation is minimal. Truncating the ACF at its first zero crossing is therefore not such a restrictive assumption. In cases where τ_c^{-1} is comparable to or less than ω_0 , the power spectral density of the fluctuations, which is the Fourier transform of the ACF, would be more appropriate. In Sec. 5.2 we thus investigate the power spectral density of a mesoscopic reaction system where τ_c^{-1} is less than ω_0 . We summarize the relationship between the lifetime, correlation time and the oscillatory frequency ω_0 in the summary section of this chapter.

Including scission of aggregates or multimers as a backward reactions in Eq. 4.8 would not modify the qualitative behavior of the results. This is because scission is a unimolecular reaction, whose reaction degeneracy, and hence its propensity, is linear in the population, while the degeneracy for aggregation is nonlinear (Gillespie, 1976, 1992; van Kampen, 2001). Consequently, scission would modify the populations at the same rate for all reactants S_{n+m} and would not introduce any additional nonlinearities. Note that scission is not negligible for aggregates of low-enough interfacial tension, whose equilibrium in the absence of driving is not totally displaced to the product side of the aggregation reactions.

In summary, we have characterized fundamental properties of the relaxation kinetics of a nonlinear and non-oscillating stochastic reaction network around a nonequilibrium steady-state. We have chosen as a model a confined, open colloidal aggregation system in a finite volume Ω . The system is driven by a monomer influx in bursts of b monomers and a non-burst multimer outflux. Specifically, we studied the trimer aggregation network as the simplest aggregation network comprising all types of elementary reactions. This makes our observations on the relaxation kinetics applicable also to larger aggregation networks and to other nonlinear reaction networks around a nonequilibrium steady-state. We studied the role of (i) low copy number created by confinement Ω^{-1} at constant volume fraction, and (ii) burst influx b . Both of these are noise sources that increase concentration fluctuations.

We accounted for these stochastic effects using (i) a linear-noise, Fokker-Planck approximation, valid in the low-confinement limit, and (ii) exact trajectories of the CME from a stochastic simulation algorithm, modeling high confinement. We used the time autocorrelation function (ACF) of species concentrations to study the relaxation kinetics towards the nonequilibrium steady-state.

We have proposed the following curve characteristics to study the response of the ACF of a species n to confinement (inverse volume) and burst: (i) the lifetime $\tau_n = \int_0^\infty C_n(t)dt$, (ii) the decay-rate function $\chi_n(t) = -\frac{d}{dt}C_n(t)$, and (iii) the ACF's initial curvature $Z_n = \frac{d^2}{dt^2}C_n(0)$.

We observed that increasing burst b monotonically increases or decreases the lifetimes (or the fraction of the power of the fluctuations at zero frequency) of all species. On the other hand, confinement Ω^{-1} increases the lifetimes of those species undergoing bimolecular reactions (monomers and dimers), but does not modulate those undergoing only unimolecular reactions (trimers). This can lead to a competition between confinement-induced and burst-induced modulations. From these observations we hypothesize that the ACF is modulated through terms of the form $b^\alpha\Omega^{-\beta}$ for some $\alpha \geq 0$, $\beta \geq 0$, in agreement with van Kampen's observation of nonlinearity giving rise to additional Debye terms in the fluctuation spectrum (van Kampen,

2001) rendering the additional terms systematically more relevant at higher confinement.

Burst alone is responsible for making $\chi_n(t)$ non-monotonic for some species. The peak in the non-monotonic $\chi_n(t)$, reflected by Z_n , is shifted in opposite directions by burst b and confinement Ω^{-1} . This observation might be useful to differentiate between extrinsic noise due to burst b and intrinsic noise due to confinement Ω^{-1} . Recently, Hilfinger and Paulsson have proposed using read-outs of fluctuations from two independent and identical reaction systems embedded in the same environment to differentiate between extrinsic noise due to dynamic environments and intrinsic noise due to low population (Hilfinger and Paulsson, 2011). They have shown that for nonlinear reaction networks the effect of intrinsic and extrinsic are coupled. We observe this coupling in our analysis. Additional relationships between the results presented by Hilfinger and Paulsson, and ours will be investigated in the future.

We believe that our results are useful to measure volume and burst in systems with known reaction rates, or, alternatively, correct for the effects of volume and burst when experimentally measuring reaction rates using fluorescence-lifetime imaging microscopy (FLIM) or fluorescence-correlation spectroscopy (FCS). Furthermore, our results might help understand the mechanisms that deviate the stochastic kinetics of nonlinear reaction networks at high confinement and burst from their deterministic counterpart.

5.2 Steady-state relaxation kinetics of mesoscopic oscillatory chemical reaction systems

We investigate the effect of volume Ω or confinement Ω^{-1} on the frequency spectrum of the fluctuations in a mesoscopic chemical reaction system at a nonequilibrium steady state. For this purpose we choose a reaction network that exhibits oscillatory behavior in the presence of intrinsic noise. Oscillatory chemical reaction networks are appealing systems to study as they can exhibit a wide range of complex behaviors, such as bifurcations, limit cycles, and chaos in different parts of their phase spaces. Consequently, they have been shown to be involved in a number of fundamental phenomena, including pattern formation (Kuramoto and Yamada, 1976), turbulence (Mertens et al., 1994; Kuramoto, 2003), chemical waves (Kuramoto, 2003), and vortex dynamics (Wu et al., 1991). Chemical oscillators also play important roles in biological systems, ranging from circadian clocks (Barkai and Leibler, 2000; Li and Lang, 2008; Ko et al., 2010; O'Neill et al., 2011) to rhythmic gene expression and metabolism (Schibler and Naef, 2005), glycolytic oscillators (Hess, 1979; Chandra et al., 2011), embryonic segmentation clocks (Pourquié, 2003), and cell-division control in both space and time (Hu and Lutkenhaus, 1999; Novak and Tyson, 1993; Tyson et al., 1996).

Here, we use an exact SSA and a mesoscopic oscillatory model system away from Hopf bifurcation to characterize the effect of intrinsic noise on the frequency spectrum of the steady-state concentration fluctuations. We use Gillespie's exact SSA (Gillespie, 1977) to sample trajectories governed by the corresponding CME. The impact of noise is quantified by changes in the power spectral density (PSD) of the concentration fluctuations at a nonequilibrium steady state. Larger intrinsic noise is realized by decreasing the reactor volume Ω at constant macroscopic concentration, hence decreasing the total number of reactive molecules in the system. We observe that the PSD has a Lorentzian-like form, confirming an earlier study (Xiao et al., 2007). We, further, show that the frequency at which the PSD is maximum depends on

the reactor volume and hence the noise magnitude. We observe that the peak shifts toward lower frequencies with increasing intrinsic noise. This shift is also accompanied by an increase in the bandwidth of the fluctuation spectrum.

Interactions between several oscillatory systems may lead to non-trivial effects (Cohen and Neu, 1979). We hence also study cascades of downstream-coupled mesoscopic chemical systems and compare the results to a single-stage system and to the large- Ω regime (Shibata, 2004; van Kampen, 2001). We show that due to intrinsic noise, each cascade stage further amplifies the shift in the peak of the PSD towards lower frequencies. For a fixed volume, the bandwidth decreases along the cascade, rendering the peak sharper and more relevant with every additional stage. In biology, downstream-coupled cascades of reaction networks are found, e.g., as cascades of autocatalytic reactions or in signaling cascades. This includes the up to six downstream-coupled autocatalytic stages of MAPK signaling cascades (Seger and Krebs, 1995; Angeli et al., 2004), which can exhibit oscillatory behavior under global feedback (Kholodenko, 2000). Another example of an oscillatory autocatalytic biochemical system is found in cell cycle control (Novak and Tyson, 1993; Tyson et al., 1996).

We believe that our findings shed light on the effect of intrinsic noise on mesoscopic oscillatory chemical reaction systems. We show that intrinsic noise can not only induce oscillatory behavior in systems away from Hopf bifurcation (Qian et al., 2002; Baxendale and Greenwood, 2011), but that it also alters the frequency spectrum of the oscillations. The key novelty in our findings is the shift of the peak of the Lorentzian-like PSD with increasing intrinsic noise. Our findings can be used to understand the fundamental effects of intrinsic noise in (cascades of) mesoscopic chemical reaction networks. Furthermore, it has been shown that steady-state fluctuation spectra can aid parameter identification of stochastic chemical reaction networks (Munsky et al., 2009). Under this premise, our results can be used to more reliably identify the parameters of mesoscopic chemical reaction networks by using the corrected fluctuation-relaxation kinetics as an additional fingerprint of the effects of intrinsic noise, where the time series of steady-state fluctuations can, e.g., be obtained using fluorescence correlation spectroscopy (FCS) (Lakowicz, 2006; Qian and Elson, 2004; Rigler and Elson, 2001). See Chapter 7.

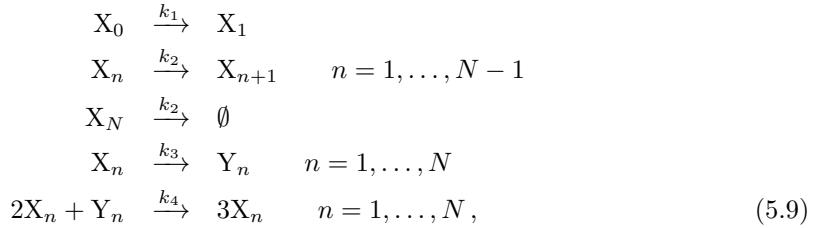
In Sec. 5.2.1 we present the model system. We present the results in Sec. 5.2.2 and conclude in Sec. 5.2.3.

5.2.1 Model

We consider a chain of downstream-coupled Brusselators (Shibata, 2004) in a reactor of volume Ω (see Fig. 5.7) as a model system. The Brusselator is a model system for autocatalytic reactions. Examples of autocatalytic reactions include the Belousov-Zhabotinsky reaction, MAPK signaling cascades (Seger and Krebs, 1995; Angeli et al., 2004), and activation of the M-phase promoting factor in cell-cycle control (Novak and Tyson, 1993; Tyson et al., 1996).

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

The reaction network of our model system is:



where n denotes the stage of the cascade and $N \geq 1$ the total number of stages. The k 's are the macroscopic reaction rates. Each stage $n \geq 1$ involves reactions between two species, X_n and Y_n . The first stage of the cascade is driven by a buffer (species X_0) whose concentration is fixed at all times. Every subsequent stage of the cascade is driven by species X_{n-1} of the previous stage. Therefore, the first stage $n = 1$ is independent of the subsequent downstream stages ($n > 1$). In general, any stage $n = q$ is independent of all subsequent downstream stages ($n > q$).

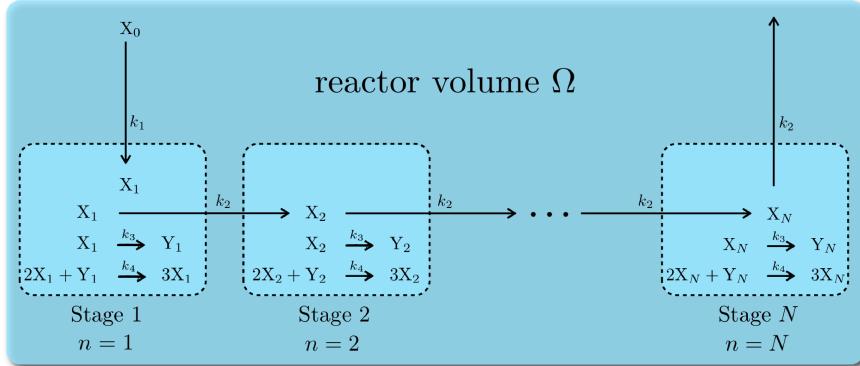


Figure 5.7: Illustration of the model system of a series of N downstream-coupled Brusselators. The cascade of Brusselator reactions progresses in a mesoscopic reactor volume Ω .

The CME corresponding to the reaction system in Eq. 5.9 is given by

$$\begin{aligned}
 \frac{\partial P(X_1, Y_1)}{\partial t} &= x_0 \Omega k_1 (E_{X_1}^{-1} - 1) P(X_1, Y_1) + k_2 (E_{X_1}^1 - 1) X_1 P(X_1, Y_1) \\
 &+ k_3 (E_{X_1}^1 E_{Y_1}^{-1} - 1) X_1 P(X_1, Y_1) \\
 &+ \frac{k_4}{\Omega^2} (E_{X_1}^{-1} E_{Y_1}^1 - 1) X_1 (X_1 - 1) Y_1 P(X_1, Y_1) \\
 \frac{\partial P(X_n, Y_n)}{\partial t} &= k_2 (E_{X_n}^1 - 1) X_n P(X_n, Y_n) + k_3 (E_{X_n}^1 E_{Y_n}^{-1} - 1) X_n P(X_n, Y_n) \\
 &+ \frac{k_4}{\Omega^2} (E_{X_n}^{-1} E_{Y_n}^1 - 1) X_n (X_n - 1) Y_n P(X_n, Y_n), \quad n = 2, \dots, N,
 \end{aligned} \tag{5.10}$$

5.2 Steady-state relaxation kinetics of mesoscopic oscillatory chemical reaction systems

where X_n and Y_n are the populations (copy numbers) of species X_n and Y_n , respectively, x_0 is the concentration of the buffer X_0 , $P(X_n, Y_n)$ is the probability of having X_n molecules of species X_n and Y_n molecules of species Y_n , and E_X is a step operator defined as $E_X^m f(X, Y) = f(X + m, Y)$ for any function $f(\cdot)$. The corresponding RRE describing the deterministic time evolution of the concentrations $x_n = X_n/\Omega$ and $y_n = Y_n/\Omega$ is given in Eq. A10.1 (see Appendix A10).

For simplicity, we set $k_2 = k_4 = 1$. We also enforce the concentration x_0 of the buffer X_0 to be 1 at all times. Under these conditions, the fixed point of the deterministic RRE (Eq. A10.1) is given by $x_n(t \rightarrow \infty) = k_1$ and $y_n(t \rightarrow \infty) = k_3/k_1$ for all $n \geq 1$. This fixed point is exponentially stable if $k_3 < k_1^2 + 1$, undergoes a Hopf bifurcation at $k_3 = k_1^2 + 1$, and becomes unstable for $k_3 > k_1^2 + 1$ (see Appendix A9). In the latter case, $x_n(t)$ and $y_n(t)$ exhibit oscillations for each $n = 1, \dots, N$. In the deterministic RRE description, the condition for a limit cycle has to be strictly fulfilled in order to observe oscillatory behavior, whereas in the stochastic description oscillations may be observed even when the limit-cycle condition for the RRE is not fulfilled (Qian et al., 2002; Baxendale and Greenwood, 2011).

We numerically sample trajectories from the CME (Eq. 5.10) using the direct method (Gillespie, 1976), an exact formulation of Gillespie's stochastic simulation algorithm (Gillespie, 1977), for different reactor volumes Ω . The reactor volume determines the magnitude of intrinsic noise, as smaller Ω decrease the total number of reactive molecules in the system and hence increase the intrinsic noise magnitude. We set $N = 20$, thus simulating a cascade of 20 Brusselator stages, which includes the single-stage case when looking at the concentrations in stage 1, since they are independent of all subsequent stages. This value for N is chosen arbitrarily and the results would not change if a different N were chosen. The concentrations of all species are 0 at time $t = 0$. We also set $k_1 = 1$ and $k_3 = 1$. For these parameters, the fixed point of the system is stable, and the deterministic system does hence not exhibit limit-cycle oscillations, reaching the fixed point $(x_n, y_n) = (1, 1)$ for all $n = 1, \dots, N$ (see Appendix A9). The mesoscopic system, however, shows oscillations due to intrinsic noise. This can be seen in Fig. 5.8, where the deterministic RRE trajectories from the first and last cascade stages are shown in panels (b) and (d), respectively, and a single trajectory sampled from the CME in panels (a) and (c). Starting from the above initial condition, the stochastic trajectory shows oscillations, whereas the deterministic trajectory reaches the stable fixed point.

We study the normalized steady-state PSD $S_n(\omega)$ of the concentration fluctuations of species X_n (species X of the n -th stage) for $n = 1, \dots, N$ as a function of the angular frequency ω for different reactor volumes Ω . The normalized steady-state PSD $S_n(\omega)$ is defined as the Fourier transform of the ACF $R_n(t)$:

$$S_n(\omega) = \mathcal{F}(R_n(t)), \quad (5.11)$$

where the ACF is

$$R_n(t) = \langle \tilde{x}_n(0)\tilde{x}_n(t) \rangle_{\text{ss}} / \sigma_n^2. \quad (5.12)$$

The subscript ss denotes quantities computed at steady state, $\tilde{x}_n = x_n - \langle x_n \rangle_{\text{ss}}$, and the variance at steady state $\sigma_n^2 = \langle \tilde{x}_n(0)\tilde{x}_n(0) \rangle_{\text{ss}}$. The normalization of $R_n(t)$ with σ_n^2 factors out the total energy of the fluctuations, so that $\int_0^\infty S_n(\omega)d\omega = 1$. We hence call $S_n(\omega)$ the *normalized steady-state PSD*. It quantifies the fraction of energy of the fluctuations at

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

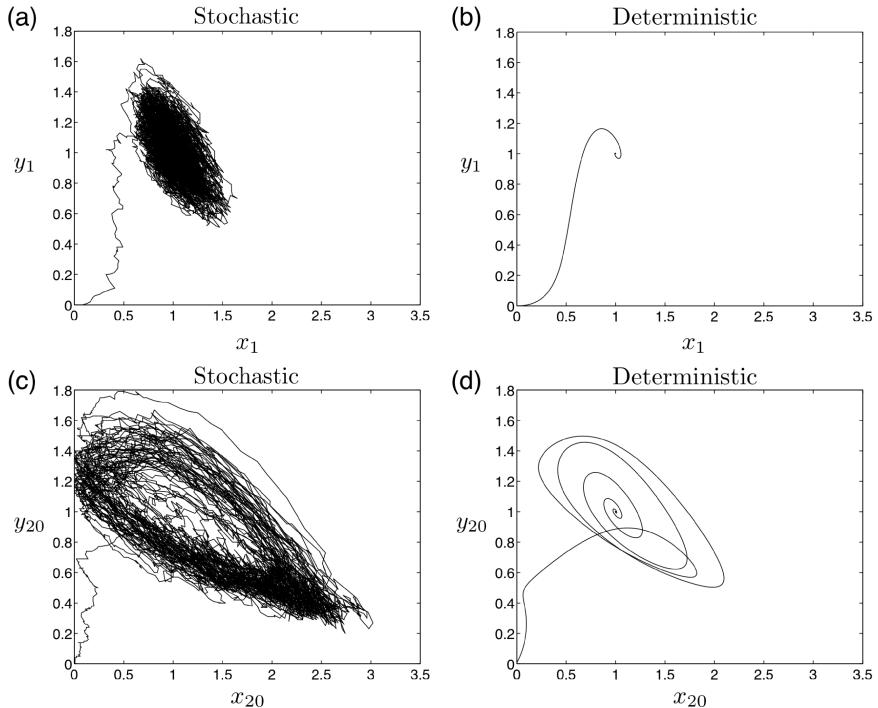


Figure 5.8: Plot of a single trajectory in phase space (x_n, y_n) for the first and last stages, $n = 1$ and $n = 20$, of the model system (Eq. 5.9) with $N = 20$ and reactor volume $\Omega = 100$. For this system, the RRE predict an exponentially stable fixed point at $(x_n, y_n) = (1, 1)$. (a) A single stochastic trajectory sampled from the CME for stage $n = 1$; (b) the corresponding deterministic RRE prediction for the same parameters. (c, d) Stochastic and deterministic trajectories, respectively, for the last stage of the cascade.

a specific frequency, namely, $S_n(\omega)d\omega$ gives the fraction of the energy of the fluctuations contained between ω and $\omega + d\omega$. We compute $S_n(\omega)$ by recording a single long trajectory of $x_n(t)$ at steady state. We sample $8\,404\,992 (2^{23} + 2^{14})$ data points of $x_n(t)$ starting from $t = 2000$ with a time resolution of $\delta t = 0.1$. We then compute the ACF (Eq. 5.12) from a minimum lag of $t = 0$ up to a maximum lag of $t = 2^{14}\delta t$. $S_n(\omega)$ is obtained by fast Fourier transform (Eq. 5.11).

We quantify the effect of intrinsic noise by the PSD's peak frequency and bandwidth. The peak frequency ω_n^m is defined as the angular frequency at which $S_n(\omega)$ is maximum, hence

$$\omega_n^m = \arg \max_{\omega} (S_n(\omega)). \quad (5.13)$$

Since $S_n(\omega)$ is generated by a stochastic process and hence is noisy, we smooth $S_n(\omega)$ before computing ω_n^m . Smoothing is done using a moving-average filter with a window diameter of 10 data points, corresponding to a frequency-space resolution of $\delta\omega = 2 \cdot 10^{-2}$ (data points in frequency space are uniformly spaced with a distance of $2 \cdot 10^{-3}$).

The bandwidth ω_n^b of the steady-state PSD is defined as the difference between the two frequencies (ω_1, ω_2) where the steady-state PSD drops to half of its maximum value, i.e., $\omega_n^b = \omega_2 - \omega_1$ so that $S_n(\omega_1) = S_n(\omega_2) = \frac{1}{2}S_n(\omega_n^m)$ with $\omega_2 > \omega_1$. Also ω_n^b is computed on the smoothed PSD.

In the large-volume regime, the quantities $S_n(\omega)$ and ω_n^m can be calculated analytically (Shibata, 2004; van Kampen, 2001) (see Appendix A10). We use the results from the large-volume regime as a baseline to understand the effects of intrinsic noise in our model system.

5.2.2 Results

We present the normalized steady-state PSD $S_n(\omega)$, the peak frequency ω_n^m , and the bandwidth ω_n^b for different reactor volumes Ω of our model system. These quantities are numerically computed from exact SSA trajectories (Gillespie, 1976, 1992) as described above. In the large-volume limit, the quantities are analytically computed as described in Appendix A10.

First, we assess $S_n(\omega)$ as a function of reactor volume Ω for the stage $n = 1$ alone and then extend our results to cascades of coupled Brusselators. Figure 5.9a shows $S_1(\omega)$ for three different reactor volumes $\Omega = 1, 5, 50$. It can be seen that the steady-state PSD is indeed a function of Ω and that it has a Lorentzian-like form with a peak frequency and an associated bandwidth. This form of the steady-state PSD is expected, since the time-autocorrelation function is $R_n(t) \propto e^{-\alpha_n t} \cos(\beta_n t)$ (Xiao et al., 2007). As the volume Ω is reduced, the peak ω_1^m shifts to lower frequencies. For $\Omega = 50$ the peak frequency is $\omega_1^m = 0.86$. This reduces to $\omega_1^m = 0.73$ and $\omega_1^m = 0.45$ for $\Omega = 5$ and $\Omega = 1$, respectively.

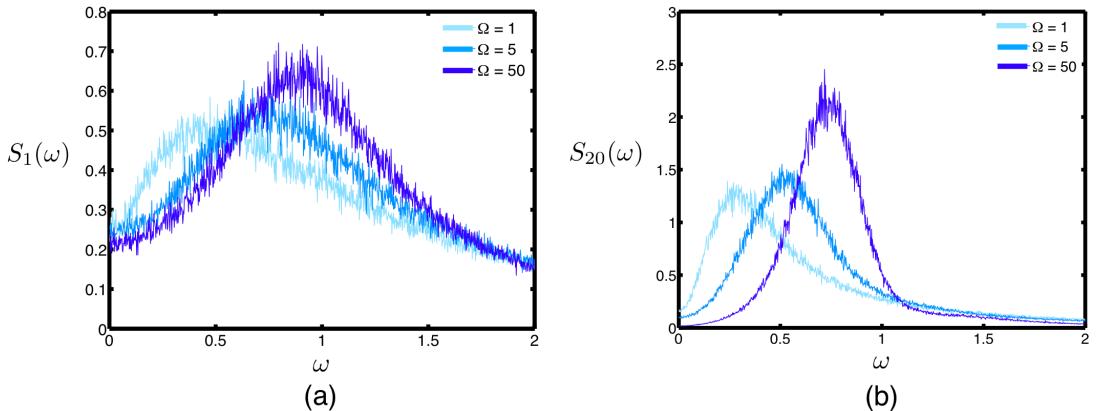


Figure 5.9: (a) Normalized steady-state power spectral density $S_1(\omega)$ of the fluctuations of species X_1 (see Eq. 5.11) for a single Brusselator in different reactor volumes $\Omega = 1, 5, 50$. (b) Normalized steady-state power spectral density $S_{20}(\omega)$ of the fluctuations of species X_{20} at the last stage of a cascade of 20 downstream-coupled Brusselators in different reactor volumes $\Omega = 1, 5, 50$.

The bandwidth for $\Omega = 50$ is $\omega_1^b = 1.19$. For $\Omega = 5$, it increases to $\omega_1^b = 1.29$, and for $\Omega = 1$ further to $\omega_1^b = 1.46$. The bandwidth of the steady-state PSD hence increases with

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

decreasing reactor volume Ω . This is in agreement with the observation that even though noise induce oscillations in mesoscopic chemical systems, it also results in a loss of synchrony of the oscillations (Barkai and Leibler, 2000).

We now consider how these results change along a cascade of downstream-coupled Brusselators. Figure 5.9b shows $S_{20}(\omega)$ at stage $n = 20$ for the same three reactor volumes $\Omega = 1, 5, 50$. Similar to what is observed in the first stage, the peak ω_n^m also shifts toward lower frequencies as Ω decreases. For $\Omega = 50$ the peak frequency is $\omega_{20}^m = 0.72$, which reduces to $\omega_{20}^m = 0.51$ and $\omega_{20}^m = 0.28$ for $\Omega = 5$ and $\Omega = 1$, respectively. For a given Ω , the peak frequency is successively reduced by each cascade, hence $\omega_{n+1}^m < \omega_n^m$.

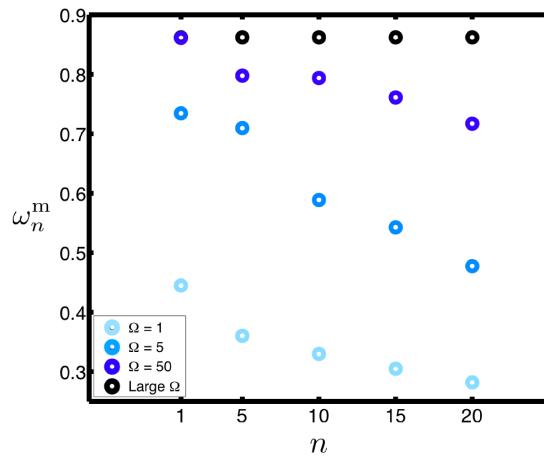


Figure 5.10: Peak frequency ω_n^m of the normalized steady-state power spectral density $S_n(\omega)$ as a function of cascade stage n in different reactor volumes Ω . The results for “large Ω ” are obtained from the analytical expressions presented in Appendix A10.

The bandwidth ω_n^b at stage $n = 20$ also increases with decreasing volume, just as it did for the first stage (see Fig. 5.9b). For $\Omega = 50, 5$, and 1 , we find $\omega_{20}^b = 0.37$, $\omega_{20}^b = 0.48$, and $\omega_{20}^b = 0.50$, respectively. Comparing ω_n^b and ω_{20}^b , we observe that for fixed reactor volume Ω the bandwidth *decreases* with every stage along the cascade.

These results are summarized in Table 5.2 and shown also for intermediate cascade stages in Fig. 5.10. We observe that for a given cascade stage n the peak frequency decreases with decreasing reactor volume and that for a given reactor volume the peak frequency also decreases with every stage along the cascade. In the large-volume limit, the peak frequency is independent of the reactor volume and also of the cascade stage (see Eq. A10.9). This indicates that the effects observed for smaller volumes are indeed caused by intrinsic noise in the system.

It should also be noted that $S_n(\omega = 0)$ increases with decreasing reactor volume Ω (see Fig 5.9). This corroborates the observation made in Sec. 5.1 that the lifetime increases with decreasing Ω . This is because the lifetime as defined in Eq. 5.3 is the equivalent to the PSD at $\omega = 0$ according to the definition of the PSD given in Eq. 5.11.

Ω	Stage 1 ($n = 1$)		Stage 20 ($n = 20$)	
	ω_1^m	ω_1^b	ω_{20}^m	ω_{20}^b
Large (linear noise)	0.86	1.18	0.86	0.19
50	0.86	1.19	0.72	0.37
5	0.73	1.29	0.51	0.48
1	0.45	1.46	0.28	0.50

Table 5.2: The effect of intrinsic noise, realized by decreasing the reactor volume Ω , on the peak frequency ω_n^m and the bandwidth ω_n^b of the normalized steady-state PSD $S_n(\omega)$ for stages $n = 1$ and $n = 20$ of a cascade of downstream-coupled Brusselators (see Eq. 5.9).

5.2.3 Conclusions

We have studied the effect of intrinsic noise due to molecular discreteness in a mesoscopic oscillatory chemical reaction network. We considered the model system of a Brusselator away from Hopf bifurcation, where the deterministic RRE do not exhibit oscillatory behavior. Oscillations can nevertheless be induced by intrinsic noise, the magnitude of which was tuned by changing the reactor volume. We also studied propagation of noise-induced effects along a downstream-coupled cascade of Brusselators. Noise-induced effects were quantified using the frequency spectrum of concentration fluctuations, given by the normalized power spectral density (PSD) at a nonequilibrium steady state. Specifically, we used the peak frequency and the bandwidth of the PSD as fingerprints of noise-induced effects. We used an exact stochastic simulation algorithm (Gillespie, 1976, 1992) to study the kinetics of the system as governed by the corresponding CME.

The results have shown that the frequency at which the steady-state PSD is maximum decreases with decreasing reactor volume. This effect is further amplified with every stage of a cascade of coupled Brusselators, leading to a further decrease along the cascade. This is in contrast to the large-volume limit, which does not predict any effect of intrinsic noise on the *normalized* steady-state PSD. We also showed that for a given cascade stage the bandwidth of concentration fluctuations decreases with increasing reactor volume, which is in agreement with the loss of synchrony of oscillations with increasing noise in mesoscopic chemical reaction systems (Barkai and Leibler, 2000; Gaspard, 2002; Xiao et al., 2007). For a fixed volume, the bandwidth also decreases along the cascade, rendering the peak sharper and more pronounced with every stage. One may hence speculate whether intrinsic noise plays a role in “tuning” the output frequency of biochemical oscillators. This tuned output frequency could in turn drive further downstream reaction networks, qualitatively changing their behavior. It is for example known that the behavior of glycolytic oscillator models can change from periodic to quasi-periodic to chaotic upon small changes in the driving input frequency (see p. 33, Fig. 2.24, in (Holden, 1986), or (Tomita and Daido, 1980; Chandra et al., 2011)).

We believe that our study sheds light on the role of intrinsic noise in chemical oscillators. Even though we studied a simple model system, the results show how intrinsic noise qualitatively influences the frequency spectrum of the oscillations. We have shown that mesoscopic chemical reaction networks, and cascades thereof, exhibit different output spectra depending

5 Effect of noise on the relaxation kinetics of mesoscopic chemical systems

on the magnitude of intrinsic noise. We expect this effect to be present also in mesoscopic oscillatory reaction systems where the RRE show oscillatory behavior. This is because there is no qualitative difference in the concentration trajectories of mesoscopic systems just before and after Hopf bifurcation (Qian et al., 2002).

Our findings are relevant for identifying the reactor volume of (cascades of) mesoscopic reaction networks when the concentration fluctuations are measured experimentally, e.g. using fluorescence correction spectroscopy. In addition, we believe that the effects reported here can be used as fingerprints to more reliably identify parameters of stochastic chemical reaction networks in systems biology models (see Chapter 7). Finally, the presented results here might contribute towards developing a general understanding of how noise influences the kinetics of different chemical systems, when deterministic RRE predictions are valid, and what deviations are to be expected otherwise.

5.3 Summary

We studied the effect of noise on the relaxation kinetics of open mesoscopic chemical systems. Specifically, we investigated the effect of two noise sources: (1) the effect of intrinsic noise due to decreasing copy numbers realized by decreasing reactor volume Ω and (2) the effect of extrinsic noise modeled as a bursty driving. We studied the effects of both noise sources on a mesoscopic non-oscillatory reaction system and the effect of intrinsic noise on a mesoscopic oscillatory system.

We observed that for both mesoscopic systems, the fraction of the power of the fluctuations at zero frequency increase with decreasing reactor volume Ω . Increasing burst can either increase or decrease this fraction. These observations help characterize the change in the nonequilibrium steady-state fluctuations as a function of noise for any given mesoscopic chemical system. In general, the time autocorrelation function (ACF) $R(t)$ of the fluctuations at steady state, of any mesoscopic reaction network is of the form $e^{-t/\tau_c} \cos(\omega_m t)$ (van Kampen, 2001; Xiao et al., 2007). This general model relates the relaxation time of the exponential envelope of $R(t)$, or the autocorrelation time τ_c , of the fluctuations and the oscillatory behavior quantified by the angular frequency ω_m . The Fourier transform of normalized $R(t)$ is the normalized power spectral density $S(\omega)$ of the fluctuations. $S(\omega)$ is Lorentzian-like (Xiao et al., 2007) and it can hence be approximated by the Cauchy-Lorentz distribution

$$S(\omega) = \frac{(\tau_c)^{-1}}{\pi [(\tau_c)^{-2} + (\omega - \omega_m)^2]} \quad (5.14)$$

with the peak frequency ω_m and bandwidth (or width at half maximum) $2\tau_c^{-1}$. In our study, we observed that the peak frequency ω_m decreases and the bandwidth $2\tau_c^{-1}$ increases with decreasing Ω , i.e.,

$$\omega_m = f(\Omega) \omega_{m,\infty} \quad (5.15)$$

and

$$\tau_c = g(\Omega) \tau_{c,\infty}, \quad (5.16)$$

5.3 Summary

where $f(\Omega)$ and $g(\Omega)$ are functions that monotonically increase with Ω and tend to 1 for asymptotically large Ω . The peak frequency and bandwidth of the fluctuation spectrum for asymptotically large Ω is $\omega_{m,\infty}$ and $2\tau_{c,\infty}^{-1}$ respectively. The normalized power spectral density (PSD) $S(\omega)$ can thus be written as

$$S(\omega) = \frac{g(\Omega) \tau_{c,\infty}}{\pi \left\{ 1 + [\omega - f(\Omega) \omega_{m,\infty}]^2 [g(\Omega) \tau_{c,\infty}]^2 \right\}}. \quad (5.17)$$

The fraction of power at zero frequency

$$S(\omega = 0) = \frac{g(\Omega) \tau_{c,\infty}}{\pi \left\{ 1 + [f(\Omega) \omega_{m,\infty}]^2 [g(\Omega) \tau_{c,\infty}]^2 \right\}}, \quad (5.18)$$

and hence the lifetime of fluctuations increases with decreasing Ω . This illustrates that for systems where the RRE show an oscillatory relaxation to their only fixed point, there is no qualitative difference between the nonequilibrium steady-state fluctuation spectra of oscillatory and non-oscillatory mesoscopic chemical reaction systems. The differences are merely quantitative. For systems where the RRE exhibit a non-oscillatory relaxation to steady state, i.e. $\omega_{m,\infty} = 0$, the $S(\omega = 0)$ is expected to decrease with decreasing Ω .

Our observations shed light on the effects of intrinsic noise on the fluctuations at steady state and may be used to understand the effects of noise on stochastic chemical kinetics of nonlinear reaction networks. Furthermore, our observations may be useful when using ACFs measured in fluorescence correlation spectroscopy experiments to estimate reaction rates of mesoscopic chemical systems.

6

Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

“There are the rushing waves...
mountains of molecules,
each stupidly minding its own business...
trillions apart
...yet forming white surf in unison.”
– Richard P. Feynman

Chemically reactive systems exhibiting spatial heterogeneity are often modeled using reaction-diffusion equations (Fisher, 1937; Kolmogorov et al., 1937; Pearson, 1993; Chaplain et al., 2001; Elf and Ehrenberg, 2004; Fange and Elf, 2006; Altschuler et al., 2008; Takahashi et al., 2010; Bergdorf et al., 2010; Holloway et al., 2011). Reaction-diffusion models explicitly capture spatial variations of the concentration fields, accounting for diffusive transport of reactants and products to and from reaction sites. Spatial heterogeneity is sustained when diffusion of chemicals is slower than reactions between them. In the limit of large numbers of molecules, reaction-diffusion processes can be modeled continuously as systems of coupled partial differential equations (frequently called *reaction-diffusion equations* or *Fisher-KPP equations* (Fisher, 1937; Kolmogorov et al., 1937)) governing the spatiotemporal evolution of the smooth concentration fields of all chemical species. Continuum reaction-diffusion models can exhibit nontrivial spatiotemporal dynamics, such as traveling concentration fronts

6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

(Benguria and Depassier, 1996) and inhomogeneous stationary concentration distributions (“Turing patterns”) (Turing, 1952; Gierer and Meinhardt, 1972; Pearson, 1993; Koch and Meinhardt, 1994). These phenomena have been successful in explaining a number of experimental observations, including localization of cell division sites in *E. coli* (Meinhardt and Piet, 2001) and “black eyes” patterns in the chlorite-iodide-malonic acid reaction (Zhou et al., 2002; Yang et al., 2002; Gunaratne et al., 1994). For low molecular copy numbers, however, continuum models fail to provide an accurate description of the spatiotemporal dynamics of reaction-diffusion systems. In particular, intrinsic noise from the apparent molecular discreteness, leading to stochasticity of chemical reactions, alters front propagation dynamics (Panja, 2004) and Turing patterns (Elf and Ehrenberg, 2004; Fange and Elf, 2006) in a nontrivial way. This is because fluctuations in the molecule populations may no longer be negligible, and correlated fluctuations may lead to deviations from deterministic behavior (Gardiner et al., 1976; Elf and Ehrenberg, 2004; Fange and Elf, 2006; Takahashi et al., 2010; Springer and Paulsson, 2006). These effects can be accounted for by stochastic reaction diffusion (SRD) simulations.

There are mainly two types of SRD simulations: on-lattice (or compartment-based) simulations and off-lattice (or particle-based) simulations. On-lattice simulations include the Next Subvolume Method (NSM) (Elf and Ehrenberg, 2004), whereas Greens-Function Reaction Dynamics (GFRD) (van Zon and ten Wolde, 2005) and Brownian Dynamics (BD) (Andrews and Bray, 2004) are examples of off-lattice schemes. On-lattice SRD simulations (Elf and Ehrenberg, 2004; Hattne et al., 2005; Rossinelli et al., 2008; Iyengar et al., 2010; Ferm et al., 2010; Koh and Blackwell, 2011; Jeschke et al., 2011) are based on dividing (discretizing) the computational domain into subvolumes, in each of which the chemical reaction system is assumed to be well mixed (spatially homogeneous). It is further postulated that only molecules within the same subvolume can react with each other, effectively treating molecules of the same chemical in different subvolumes as different species. Diffusion is modeled as unimolecular “diffusion reactions” representing jumps of molecules between neighboring subvolumes. The on-lattice approach hence describes the reaction-diffusion system as a large chemical reaction network with the number of species proportional to the product of the actual number of chemical species and the number of subvolumes used to discretize space. The kinetics of this enlarged reaction network can be mathematically described by the on-lattice reaction-diffusion master equation (RDME), analogous to the chemical master equation (CME) (Gillespie, 1992). Off-lattice SRD simulations (Andrews and Bray, 2004; van Zon and ten Wolde, 2005; Morelli and ten Wolde, 2008; Hellander and Lotstedt, 2011) are based on computational particles mimicking the Brownian motion of molecules, whereby the molecules involved in a bimolecular reaction react with a certain probability when the distance between them is smaller than a pre-defined reaction radius.

Here, we focus on-lattice SRD simulations in order to avoid computationally expensive collision detection and time-step adaptation mechanisms (Fange et al., 2010). Since on-lattice SRD is described by a system of chemical reactions modeled by the RDME, it can be exactly simulated using Gillespie’s stochastic simulation algorithm (SSA) (Gillespie, 1976, 1992). SSA samples trajectories from the exact solution of the master equation by sampling the index of the next reaction, the time to the next reaction, and updating the reaction probability rates (called “propensities”). Different SSA formulations are available that use different sampling and update algorithms, including the direct method (DM) (Gillespie, 1976, 1977), the first

reaction method (FRM) (Gillespie, 1976, 1977), the next reaction method (NRM) (Gibson and Bruck, 2000), the optimized direct method (ODM) (Cao et al., 2004), the sorting direct method (SDM) (McCollum et al., 2006), the SSA with composition-rejection sampling (SSA-CR) (Slepoy et al., 2008) (see Sec. 2.3), and partial-propensity methods (see Chapter 3) such as the partial-propensity direct method (PDM) (Sec. 3.3), the sorting PDM (SPDM) (Sec. 3.3.2), and the partial-propensity SSA with composition-rejection sampling (PSSA-CR) (Sec. 3.4). Directly using any of these SSA formulations for the RDME, without adapting it to the specifics of on-lattice SRD simulations, would be correct, but computationally and/or memory inefficient since the number of species and the number of reactions increase linearly with the number of subvolumes. A more efficient way of performing on-lattice SRD simulation is to first sample the subvolume in which the next reaction will happen and then sample the index of the reaction within that subvolume. This is, for example, done in NSM (Elf and Ehrenberg, 2004) as implemented in the MesoRD software package (Hattne et al., 2005). NSM uses NRM for sampling the subvolume and DM for sampling the reaction within that subvolume (Elf and Ehrenberg, 2004; Hattne et al., 2005). For a chemical reaction network with N species and M reactions, the computational cost (here formalized using the Bachmann-Landau “big-O” notation) of NSM to perform an on-lattice SRD simulation in a three-dimensional (3D) computational domain divided into N_v subvolumes is $O(\log_2 N_v + Mf_r + 6N(1 - f_r))$ (Elf and Ehrenberg, 2004; Hattne et al., 2005), where f_r is the fraction of firings accounted for by “real” reactions and $(1 - f_r)$ the fraction of firings of “diffusion reactions”. This is composed of the $O(\log_2 N_v)$ cost for maintaining the subvolume priority queue and the $O(Mf_r + 6N(1 - f_r))$ cost for sampling the next reaction. $M + 6N$ is the number of reactions in each subvolume, composed of the M “real” chemical reactions and the $6N$ “diffusion reactions” to the 6 face-connected neighboring subvolumes in a uniform Cartesian 3D mesh (in 2D this would be $4N$).

If the molecular population increases, the time step of exact SSAs decreases, increasing the runtime of the simulations. This can be alleviated by approximate SSAs that use a fixed time step to sample trajectories from an approximate solution of the master equation. In this spirit, on-lattice SRD simulations involving larger population sizes can be accelerated using approximate and hybrid SSAs (Rossinelli et al., 2008; Iyengar et al., 2010; Ferm et al., 2010; Koh and Blackwell, 2011; Jeschke et al., 2011). Here, we focus on exact on-lattice SRD formulations since they are parameter free and do not require prescription of a time step size or a target error level.

We propose a novel exact on-lattice SRD simulation method, called PSRD, using partial propensities. Partial-propensity methods are exact SSAs with a computational cost that scales at most linearly with the number of species in the reaction network (see Chapter 3). For weakly coupled reaction networks, where the number of reactions influenced by any other reaction is asymptotically independent of system size, the computational cost of partial-propensity methods is bounded by a constant (see Chapter 3). This is particularly advantageous for on-lattice SRD simulations, where the number of neighboring subvolumes influenced by any subvolume is constant (6 in 3D, 4 in 2D), independent of the total number of subvolumes used to discretize space. PSRD hence uses composition-rejection sampling to find the next subvolume with an $O(G_a)$ cost, and then uses SPDM inside that subvolume to sample the next reaction with a cost of $O(N)$. G_a is the logarithm of the ratio of the maximum to the minimum non-zero subvolume propensities, which is at most $O(\log_2 N_v)$. PSRD thus has an overall computational cost of $O(G_a + N)$, which is asymptotically bounded from above by

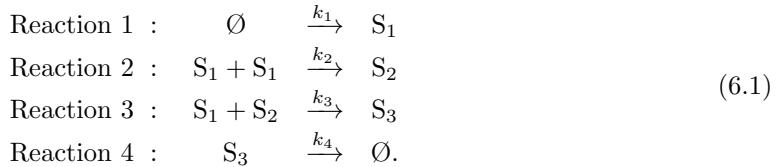
$O(\log_2 N_v + N)$ and independent of the ratio between “real” and “diffusion” reactions. This is achieved by restricting the method to elementary chemical reactions, under the premise that any non-elementary reactions can be broken down into elementary reactions (Wilhelm, 2000; Schneider and Wilhelm, 2000; Gillespie, 1992) at the expense of an increase in network size (see Appendix A2). We demonstrate the scaling of the computational cost of PSRD on two types of reaction networks: one in which the number of reactions M increases super-linearly with the number of species N (a strongly coupled network), and a second in which M is proportional to N (a weakly coupled network). Finally, we demonstrate the application of PSRD to pattern-forming stochastic Gray-Scott systems (Gray and Scott, 1983, 1984, 1985; Pearson, 1993; Lee et al., 1993), highlighting the effect of intrinsic noise on the resulting Turing patterns.

6.1 On-lattice stochastic reaction-diffusion

We recall the concept of on-lattice SRD simulations using an example reaction network. In the benchmarks presented below we assume that the boundary of the computational domain is reflective (no-flux boundary condition), except for the showcases in Sec. 6.2.5, where we use periodic boundary conditions. Other boundary conditions can be treated as described by Erban and Chapman (2007) (Erban and Chapman, 2007) (see Sec. 2 in their article). The scaling of the computational cost of on-lattice SRD simulations, however, is independent of the type of the boundary condition.

6.1.1 General concept

Consider the example of the following trimerization reaction in a 3D cuboidal reactor of dimension $L_x \times L_y \times L_z$ and volume $\Omega = L_x L_y L_z$:



The k 's are the macroscopic reaction-rate constants. This reaction network has $N = 3$ species and $M = 4$ reactions. We choose this reaction network as an example since it contains all types of elementary reactions: reaction 1 is a source reaction, reaction 2 a bimolecular reaction between the same species (homo-bimolecular reaction), reaction 3 a bimolecular reaction between two different species (hetero-bimolecular reaction), and reaction 4 is a unimolecular reaction. Any non-elementary reaction involving $r > 2$ reactants can be broken down to a set of $2r - 3$ elementary reactions by introducing additional $r - 2$ auxiliary species (see Appendix A2). The reaction-propensity a_μ of reaction μ is defined as the probability rate of firing of that reaction. Each a_μ is computed as the product of the reaction degeneracy and the specific probability rate c_μ of that reaction. The reaction degeneracy is the number of distinct combinations (collision pairs) of reactant molecules that can be formed, and the specific probability rate is the probability rate of the reaction when only one molecule of each reactant

6.1 On-lattice stochastic reaction-diffusion

is present (see Chapter 1 for a more elaborate presentation of these quantities). According to these definitions, the reaction propensities for the reaction network in Eq. 6.1 are:

$$a_\mu = \begin{cases} c_\mu, & c_\mu = k_1\Omega, \quad \text{if } \mu = 1 \\ \frac{1}{2}n_1(n_1 - 1)c_\mu, & c_\mu = 2k_2\Omega^{-1}, \quad \text{if } \mu = 2 \\ n_1n_2c_\mu, & c_\mu = k_3\Omega^{-1}, \quad \text{if } \mu = 3 \\ n_3c_\mu, & c_\mu = k_4, \quad \text{if } \mu = 4, \end{cases} \quad (6.2)$$

where n_i denotes the population of species S_i , i.e., the number of molecules of S_i present in the system.

If the characteristic time of diffusion of the species is comparable to or larger than the characteristic time of reaction, the system will exhibit spatial inhomogeneities and diffusion of the species in the reaction network needs to be explicitly accounted for. In on-lattice SRD methods, this is done by dividing the computational domain into subvolumes within which the system is assumed to be well mixed. The chemical species in each subvolume can (i) react with each other in bimolecular reactions, (ii) undergo unimolecular reactions, or (iii) be produced through source reactions. In both cases, the products are formed in the same subvolume and species from different subvolumes can not react with each other. Diffusion of molecules is modeled as a jump process from a subvolume to any of the face-connected neighboring subvolumes.

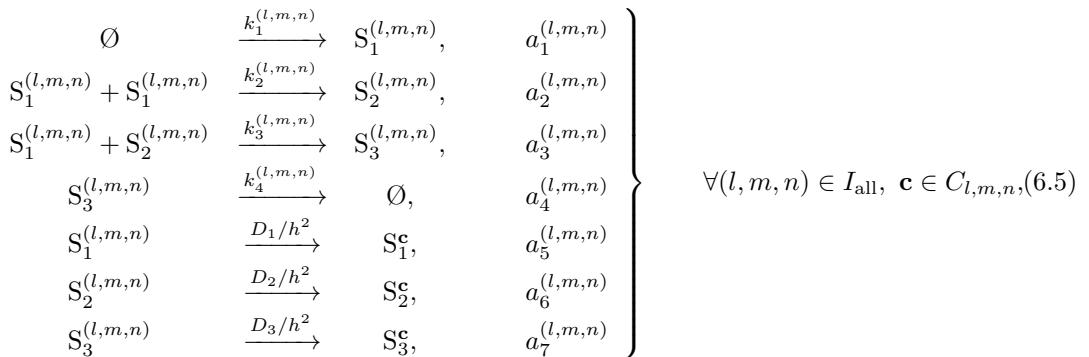
Assume that we divide the 3D computational domain into $N_v = K_x K_y K_z$ equi-sized cubic subvolumes of edge length $h = L_x/K_x = L_y/K_y = L_z/K_z$ and volume $\Omega_c = h^3$ (see Fig. 6.1C; the one- and two-dimensional cases are illustrated in Figs. 6.1A and 6.1B, respectively). The subvolumes are indexed by their Cartesian mesh coordinates over the set (Erban and Chapman, 2009)

$$I_{\text{all}} = \{(l, m, n) \mid l, m, n \text{ are integers such that } 1 \leq l \leq K_x; 1 \leq m \leq K_y; 1 \leq n \leq K_z\} \quad (6.3)$$

and the set of face-connected neighbors of a subvolume with index (l, m, n) is

$$C_{l,m,n} = \{(l, m, n) + \boldsymbol{\delta} \mid (l, m, n) + \boldsymbol{\delta} \in I_{\text{all}}\} \quad (6.4)$$

such that $\boldsymbol{\delta} \in E = \{(1, 0, 0), (-1, 0, 0), (0, 1, 0), (0, -1, 0), (0, 0, 1), (0, 0, -1)\}$. Hence, the on-lattice reaction-diffusion system of the reaction network in Eq. 6.1 can be written as:



6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

where $S_i^{(l,m,n)}$ denotes species S_i in subvolume (l, m, n) , the k 's the macroscopic reaction rates, and the a 's the corresponding propensities. In general, the k 's can be different in different subvolumes, which is explicitly shown in Eq. 6.5 by indexing them with the subvolume index. Diffusion of species S_i with diffusion constant D_i is modeled as jumps to face-connected neighboring subvolumes as illustrated in Fig. 6.1D. Equation 6.5 models the on-lattice reaction-diffusion system as a system of chemical reactions composed of $3K_x K_y K_z = 3N_v$ species and $22K_x K_y K_z - 6(K_x K_y + K_y K_z + K_x K_z) = 22N_v - 6N_v(\frac{1}{K_x} + \frac{1}{K_y} + \frac{1}{K_z})$ reactions composed of $4K_x K_y K_z$ “real” reactions and $3(6K_x K_y K_z - 6(K_x K_y + K_y K_z + K_x K_z))$ “diffusion reactions”, accounting for the missing neighboring subvolumes at the domain boundary. In general, the 3D SRD dynamics of N species and M reactions in a computational domain with reflective boundaries and $K_x \times K_y \times K_z$ subvolumes can be modeled by a chemical reaction network consisting of $NK_x K_y K_z = NN_v$ species and $(M + 6N)K_x K_y K_z - 2(K_x K_y + K_y K_z + K_x K_z)N = (M + 6N)N_v - 2\left(\frac{1}{K_x} + \frac{1}{K_y} + \frac{1}{K_z}\right)N_v N$ reactions. For other boundary conditions, the number of reactions is $(M + 6N)N_v$, accounting for the diffusive fluxes across the boundary.

For inhomogeneous diffusion, D_i additionally depends on the subvolume index (l, m, n) . For anisotropic diffusion, D_i depends on the direction of the jump reaction. These extensions are straightforward to include in any on-lattice SRD framework.

The propensities of the reactions in Eq. 6.5 are:

$$a_\mu^{(l,m,n)} = \begin{cases} c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = k_1^{(l,m,n)}\Omega_c, \quad \text{if } \mu = 1 \\ \frac{1}{2}n_1^{(l,m,n)}(n_1^{(l,m,n)} - 1)c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = 2k_2^{(l,m,n)}\Omega_c^{-1}, \quad \text{if } \mu = 2 \\ n_1^{(l,m,n)}n_2^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)}k_3^{(l,m,n)}\Omega_c^{-1}, \quad \text{if } \mu = 3 \\ n_3^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = k_4^{(l,m,n)}, \quad \text{if } \mu = 4 \\ n_1^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = D_1 h^{-2}, \quad \text{if } \mu = 5 \\ n_2^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = D_2 h^{-2}, \quad \text{if } \mu = 6 \\ n_3^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = D_3 h^{-2}, \quad \text{if } \mu = 7, \end{cases} \quad (6.6)$$

where $n_i^{(l,m,n)}$ is the population of species $S_i^{(l,m,n)}$ (i.e., species S_i in subvolume (l, m, n)) and $c_\mu^{(l,m,n)}$ is the specific probability rate of reaction μ in subvolume (l, m, n) . These formulations for the propensities directly follow from the same argument as the propensities in Eq. 6.2 for the reaction system given in Eq. 6.1. The rates of the “diffusion reactions” always scale as h^{-2} , irrespective of the dimension of the subvolumes.

6.1.2 Discretization-corrected propensities

The propensity formulations in Eq. 6.6 may lead to artifacts in the kinetics introduced by the spatial discretization (Erban and Chapman, 2009; Fange et al., 2010). This is due to the subdivision of the reaction space into several small subvolumes. This subdivision is fundamentally different from the one used in spatial discretization of continuum models (e.g., finite difference or finite volume methods). While in discretizing continuum models more resolution (smaller subvolumes) is always better, this is not necessarily the case in SRD simulations. This is because in SRD simulations, the subvolumes introduce spurious physical boundaries; molecules in one subvolume cannot react with molecules in a neighboring subvolume, even if for

6.1 On-lattice stochastic reaction-diffusion

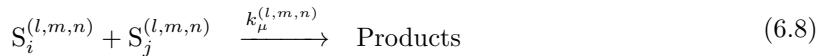
molecules close to a subvolume boundary the closest collision partner could be in a neighboring subvolume. The subvolumes thus define closed, well-mixed reaction spaces of volume $\Omega_c \ll \Omega$. In order for the reaction system to be well mixed within each subvolume, the subvolume edge lengths have to be much smaller than the Kuramoto length (Kuramoto and Yamada, 1976), hence

$$h \ll h_{\max} = \sqrt{2dDt_r}, \quad (6.7)$$

where d is the dimension of the subvolume. The characteristic time t_r of the fastest reactions in the system can be estimated from the time autocorrelation function of species populations simulated using exact SSA. While t_r depends on the reactor volume Ω , an estimate for it can be obtained from the linear-noise approximation of the CME (see Chapter 5). At length scales above h_{\max} the subvolumes are no longer spatially homogeneous (well mixed).

In addition to this upper bound on h , there may also be a lower bound. It is, for example, known that chemical kinetics in small volumes is quantitatively and qualitatively altered (see Chapters 4 and 5). It has further been shown that the RDME gives different results depending on the level of spatial discretization of the computational domain (Erban and Chapman, 2009). If the discretization becomes too fine, the RDME even yields unphysical results (Erban and Chapman, 2009; Fange et al., 2010). These artifacts are introduced by the artificial subdivision of space and lead to propensities in the RDME becoming inconsistent with Smoluchowski's microscopic reaction-diffusion framework (von Smoluchowski, 1917; Fange et al., 2010). The propensities in on-lattice SRD simulations hence need to be corrected for the spatial discretization.

At present, there are two strategies toward deriving discretization-corrected propensities. The first is based on the premise that for a well-stirred system of reactions the kinetics of the reaction-diffusion process should not depend on the resolution of the spatial discretization (Erban and Chapman, 2009). In this strategy, only the propensities of bimolecular reactions need to be corrected. Assuming a 3D cubic computational domain of size $L \times L \times L$ that is divided into K^3 subvolumes of edge length $h = L/K$, Erban and Chapman (2009) (Erban and Chapman, 2009) have derived discretization-corrected propensities for bimolecular reactions. For a hetero-bimolecular reaction μ occurring in subvolume (l, m, n)



the discretization-corrected propensity is given by:

$$a_\mu^{(l,m,n)} = n_i^{(l,m,n)} n_j^{(l,m,n)} c_\mu^{(l,m,n)}, \quad c_\mu^{(l,m,n)} = \frac{(D_i + D_j) k_\mu^{(l,m,n)}}{(D_i + D_j) h^3 - \beta k_\mu^{(l,m,n)} h^2}. \quad (6.9)$$

For a homo-bimolecular reaction μ occurring in subvolume (l, m, n)



the discretization-corrected propensity is given by:

$$a_\mu^{(l,m,n)} = \frac{n_\mu^{(l,m,n)} (n_\mu^{(l,m,n)} - 1)}{2} c_\mu^{(l,m,n)}, \quad c_\mu^{(l,m,n)} = \frac{D_i k_\mu^{(l,m,n)}}{D_i h^3 - \beta k_\mu^{(l,m,n)} h^2}. \quad (6.11)$$

6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

In Eqs. 6.9 and 6.11, the factor β is given by:

$$\beta = \frac{1}{2K^3} \sum_{l,m,n=0; (l,m,n) \neq (0,0,0)}^{K-1} \frac{1}{3 - \cos(l\pi/K) - \cos(m\pi/K) - \cos(n\pi/K)}. \quad (6.12)$$

This discretization-correction framework imposes a lower bound on the admissible subvolume size, given by the constraint that the corrected reaction propensities have to be non-negative (Erban and Chapman, 2009). The RDME with corrected propensities is only physically valid for

$$h > h_{\min} = \max_{\mu} \left[\frac{\beta k_{\mu}^{(l,m,n)}}{D_i + D_j} (1 - \delta_{ij}) + \frac{\beta k_{\mu}^{(l,m,n)}}{D_i} \delta_{ij} \right], \quad (6.13)$$

where δ_{ij} is the Kronecker delta. The maximum is taken over all bimolecular reactions μ where D_i and D_j are the diffusion coefficients of the two respective reactants.

The second strategy derives discretization-corrected propensities such that the RDME becomes consistent with Smoluchowski's microscopic reaction-diffusion framework (von Smoluchowski, 1917; Fange et al., 2010). In this strategy, the discretization-corrected propensities depend on the population of reactant molecules in the neighboring subvolumes, rendering the correction non-local and reaction-dependent. This approach is valid also for non-cubic computational domains and non-3D simulations, and it does not impose any lower bound on the subvolume size h . It has been shown to provide a seamless transition between Smoluchowski's microscopic framework and that of on-lattice SRD as based on the RDME (von Smoluchowski, 1917; Fange et al., 2010).

In summary, the propensities of all reactions in a system obtained by on-lattice discretization of a reaction-diffusion process need to be corrected. Erban and Chapman (2009) (Erban and Chapman, 2009) have derived the discretization correction only for cubic computational domains where the number of subvolumes in each spatial dimension is the same. We thus use the above discretization-corrected propensities only in these cases. Extending the present on-lattice SRD method to the framework proposed by Fange et al. (2010) (Fange et al., 2010) should be possible. The scaling of the computational cost of on-lattice SRD methods, however, is independent of the formulation used for the propensities.

We note that similar corrections are also necessary in off-lattice SRD simulations, where the artificial spatial discretization is, e.g., introduced by the reaction radius (Erban and Chapman, 2009).

6.1.3 The Next Subvolume Method (NSM) for on-lattice stochastic reaction-diffusion simulations

NSM simulates the on-lattice SRD system by sampling from the conditional joint probability distribution function (PDF) for the time τ to the next reaction, the index μ of the next reaction, and the subvolume (l, m, n) containing the next reaction, given the current population $\mathbf{n}(t)$ at time t . This joint PDF results from the on-lattice RDME and is given by:

$$p(\tau, \mu, l, m, n | \mathbf{n}(t)) = p(\tau) p(l, m, n) p(\mu | l, m, n). \quad (6.14)$$

6.1 On-lattice stochastic reaction-diffusion

Here, $p(\tau)$ is the continuous PDF for the time to the next reaction, τ , given by:

$$p(\tau) = ae^{a\tau}, \quad (6.15)$$

where a is the total propensity of the system. The discrete PDF $p(l, m, n)$ for the subvolume (l, m, n) of the next reaction is given by:

$$p(l, m, n) = \frac{a^{(l, m, n)}}{a}, \quad (6.16)$$

where $a^{(l, m, n)}$ is the propensity of subvolume (l, m, n) . The discrete PDF $p(\mu | l, m, n)$ for the next reaction μ within subvolume (l, m, n) is given by

$$p(\mu | l, m, n) = \frac{a_{\mu}^{(l, m, n)}}{a^{(l, m, n)}}, \quad (6.17)$$

$a_{\mu}^{(l, m, n)}$ the propensity of reaction μ in subvolume (l, m, n) . Formally, $\mathbf{n}(t) = [n_1^{(1,1,1)}, \dots, n_N^{(1,1,1)}, \dots, n_1^{(K_x, K_y, K_z)}, \dots, n_N^{(K_x, K_y, K_z)}](t)$, where $n_i^{(l, m, n)}(t)$ is the population of species S_i in subvolume (l, m, n) at time t , $a^{(l, m, n)} = \sum_{\mu} a_{\mu}^{(l, m, n)}$ the total propensity of all reactions in subvolume (l, m, n) , and $a = \sum_{l=1}^{K_x} \sum_{m=1}^{K_y} \sum_{n=1}^{K_z} a^{(l, m, n)}$ the total propensity of all reactions across all subvolumes.

NSM (Elf and Ehrenberg, 2004) is a popular and efficient algorithm for sampling trajectories of $\mathbf{n}(t)$ from the above PDF, which is the exact solution of the RDME. In NSM, the subvolume (l, m, n) in which the next reaction will occur is sampled first according to Eq. 6.16 and subsequently one of the reactions μ in that subvolume is sampled according to Eq. 6.17. The latter is done by first deciding whether the next reaction is a “real” or a “diffusion” reaction and then using linear search only over the corresponding reaction group (Hattne et al., 2005). The algorithm used in NSM to sample the next subvolume is inspired by the indexed priority queues used in the next reaction method (Gibson and Bruck, 2000). Sampling a reaction within a subvolume is done using linear search as in Gillespie’s original direct method (Gillespie, 1976, 1977). The time to the next reaction is calculated from Eq. 6.15. After the chosen reaction fired, the population and the propensities of some of the reactions need to be updated. In NSM, the population is updated using a sparse representation of the stoichiometry matrix, and the propensities are updated using a dependency graph (Gibson and Bruck, 2000).

The computational cost of NSM is: (i) $O(1)$ for sampling the subvolume; (ii) $O(Mf_r + 6N(1 - f_r))$ for sampling the next reaction within that subvolume, where M is the number of “real” reactions, $6N$ the number of “diffusion reactions” ($4N$ in 2D), and f_r the fraction of reaction firings accounted for by “real” reactions; (iii) $O(1)$ for updating the population; (iv) at most $O(M)$ for updating the propensities within a subvolume; (v) $O(\log_2 N_v)$ for updating the subvolume priority queue, where N_v is the number of subvolumes. The overall computational cost of NSM thus is $O(\log_2 N_v + Mf_r + 6N(1 - f_r))$. The fraction f_r of “real” reaction firings decreases with increasing N_v . For small N_v , almost all reactions are “real” and the computational cost of NSM is $O(\log_2 N_v + M)$. In particular, for $N_v = 1$ the fraction $f_r = 1$ and the computational cost of NSM is $O(M)$, as for Gillespie’s DM (Gillespie, 1976, 1977). For large N_v , the computational cost of NSM is $O(\log_2 N_v + 6N)$ since $f_r \ll 1$ and almost all reaction events pertain to “diffusion reactions”.

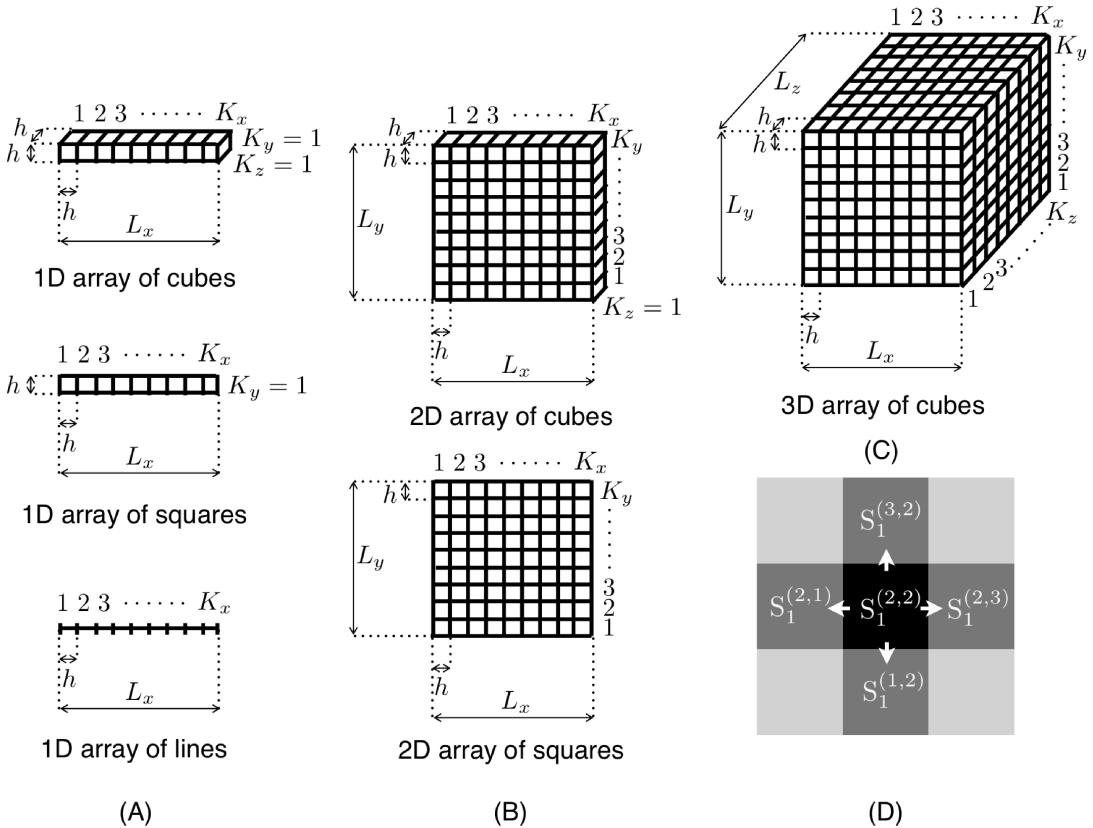


Figure 6.1: Division of the computation domain into subvolumes. (A–C) Different possibilities of subdividing a box-shaped computational domain in one, two, and three dimensions, respectively. L_x , L_y , and L_z are the lengths of the computational domain in each direction. K_x , K_y , and K_z are the numbers of subvolumes of edge length h after subdivision in each direction. (D) Diffusion is modeled as jump “reactions” to face-connected subvolumes. The same chemical in different subvolumes is treated as a different species in stochastic on-lattice reaction-diffusion simulations. Unimolecular “diffusion reactions” convert species as shown.

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

Combining ideas from NSM and partial-propensity SSAs (Chapter 3), we introduce a novel on-lattice SRD simulation method, the partial-propensity stochastic reaction-diffusion method (PSRD). PSRD is based on the idea of binning the subvolumes and determining the next subvolume using composition-rejection sampling (Devroye, 1986; Slepoy et al., 2008) (see Sections 2.3.7 and 3.4). Then, we use the concept of partial propensities (see Sec. 3.2) to sample the index of the next reaction within the selected subvolume.

6.2.1 General concept of PSRD

We summarize the general concepts of binned composition-rejection sampling and partial propensities in the context of SRD simulations.

6.2.1.1 Composition-rejection sampling to select the subvolume

Composition-rejection sampling (Devroye, 1986; Fox, 1990; Rajasekaran and Ross, 1993; Hagerup et al., 1993) is an efficient algorithm to sample realizations of a random variable according to a given discrete probability distribution. In on-lattice SRD simulations, the discrete PDF for the subvolume index (l, m, n) is $p(l, m, n)$ (see Eq. 6.16). The sampling process starts by binning the $a^{(l,m,n)}$ according to their values and then proceeds in two steps: The composition step is used to identify the bin by linear search, and the rejection step is used to identify the $a^{(l,m,n)}$, and hence the index of the subvolume (l, m, n) , inside that bin.

6.2.1.2 Partial propensities to sample the next reaction within a subvolume

Partial propensities: The partial propensity of a reaction is defined as the propensity per molecule of one of its reactants (see Sec. 3.1). For example, the partial propensity $\pi_\mu^{(l,m,n);(i)}$ of reaction μ within a subvolume (l, m, n) with respect to (perhaps the only) reactant $S_i^{(l,m,n)}$ is $a_\mu^{(l,m,n)} / n_i^{(l,m,n)}$, where $a_\mu^{(l,m,n)}$ is the propensity of reaction μ in subvolume (l, m, n) and $n_i^{(l,m,n)}$ the population of $S_i^{(l,m,n)}$ (i.e., the number of molecules of species S_i in subvolume (l, m, n)). The partial propensities of the three elementary reaction types within each subvolume (l, m, n) are:

- Bimolecular reactions $S_i^{(l,m,n)} + S_j^{(l,m,n)} \xrightarrow{c_\mu^{(l,m,n)}} \text{Products}$: $a_\mu^{(l,m,n)} = n_i^{(l,m,n)} n_j^{(l,m,n)} c_\mu^{(l,m,n)}$ and $\pi_\mu^{(l,m,n);(i)} = n_j^{(l,m,n)} c_\mu^{(l,m,n)}$, $\pi_\mu^{(l,m,n);(j)} = n_i^{(l,m,n)} c_\mu^{(l,m,n)}$. If both reactants are of the same species, i.e. $S_i^{(l,m,n)} = S_j^{(l,m,n)}$, only one partial propensity exists, $\pi_\mu^{(l,m,n);(i)} = \frac{1}{2}(n_i^{(l,m,n)} - 1)c_\mu^{(l,m,n)}$, because the reaction degeneracy is $\frac{1}{2}n_i^{(l,m,n)}(n_i^{(l,m,n)} - 1)$.
- Unimolecular reactions $S_i^{(l,m,n)} \xrightarrow{c_\mu^{(l,m,n)}} \text{Products}$: $a_\mu^{(l,m,n)} = n_i^{(l,m,n)} c_\mu^{(l,m,n)}$ and $\pi_\mu^{(l,m,n);(i)} = c_\mu^{(l,m,n)}$. The “diffusion reactions” representing the jumps from a subvolume to one of its neighbors fall into this category.

- Source reactions $\emptyset \xrightarrow{c_\mu^{(l,m,n)}} \text{Products: } a_\mu^{(l,m,n)} = c_\mu^{(l,m,n)}$ and $\pi_\mu^{(l,m,n);(0)} = c_\mu^{(l,m,n)}$.

We use the specific probability rates given in Eq. 6.6. In cases where the 3D computational domain is cubic with equal numbers of subvolumes in each dimension, we use the discretization-corrected specific probability rates as given by Erban and Chapman (2009) (Erban and Chapman, 2009) for the bimolecular reactions. The computational cost and the formalism of PSRD, however, are independent of the formulation used for the specific probability rates.

Sampling using partial propensities: Within the selected subvolume we use partial propensity methods to sample the next reaction according to Eq. 6.17. We group the partial propensities of all reactions within each subvolume according to the index of the factored-out reactant. This results in at most $N + 1$ groups of size $O(N)$. Every reaction in a subvolume, and its corresponding partial propensity, are then identifiable by two indices: a *group index* and an *element index*. The group index identifies the partial-propensity group to which a reaction belongs and the element index identifies the position of the reaction inside that group. Determining the index of the next reaction is thus done by first sampling its group index and then the element index.

After the selected reaction has fired and the populations of the involved species have been updated according to the reaction stoichiometry, the affected partial propensities are updated using a dependency graph over species (see Eq. 3.6). This dependency graph points to all partial propensities within the subvolume that need to be updated due to the change in population. If the executed reaction was a “diffusion reaction” modeling the jump of a molecules from a subvolume to one of its neighbours, we additionally update the population of that species in the corresponding neighboring subvolume and update the affected partial-propensities in the neighboring subvolume using the respective dependency over species. Since any partial propensity is a function of the population of at most one species, the number of updates is at most $O(N)$. For more details, see Chapter 3.

6.2.2 Detailed description of the PSRD algorithm

We provide a detailed description of the algorithms and data structures used in PSRD. The workflow of the algorithm is summarized in Table 6.1.

6.2.2.1 Data structures

The population of species in each subvolume (l, m, n) is stored in an array $\mathbf{n}^{(l,m,n)}$. The partial propensities of the reactions within each subvolume (l, m, n) are stored in “partial-propensity structures” $\mathbf{\Pi}^{(l,m,n)} = \left\{ \mathbf{\Pi}_i^{(l,m,n)} \right\}_{i=0}^N$ as one-dimensional arrays of one-dimensional arrays $\mathbf{\Pi}_i^{(l,m,n)}$. Each array $\mathbf{\Pi}_i^{(l,m,n)}$ contains the partial propensities belonging to group i in subvolume (l, m, n) . The partial propensities of source reactions are stored as consecutive entries of the 0th array $\mathbf{\Pi}_0^{(l,m,n)}$. The partial propensities of all reactions in subvolume (l, m, n) that have species $S_1^{(l,m,n)}$ as the factored-out reactant are stored as consecutive entries of $\mathbf{\Pi}_1^{(l,m,n)}$. In general, the i th array $\mathbf{\Pi}_i^{(l,m,n)}$ contains the partial propensities of all reactions

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

in subvolume (l, m, n) that have $S_i^{(l,m,n)}$ as the common factored-out reactant, provided these reactions have not yet been included in any of the previous $\Pi_{j < i}^{(l,m,n)}$. That is, out of the two partial propensities of a bimolecular reaction with $S_i^{(l,m,n)}$ and $S_j^{(l,m,n)}$ as its reactants and $i < j$, $\pi_\mu^{(l,m,n);(i)}$ is part of $\Pi_i^{(l,m,n)}$, and $\pi_\mu^{(l,m,n);(j)}$ is not stored anywhere. In order to save memory, we lump the “diffusion reactions” of each species within a subvolume into one reaction with no products. The specific probability rate of the lumped reaction is the sum of the specific probability rates of all “diffusion reactions” in that subvolume. Therefore, instead of storing 6 partial propensities in 3D (4 in 2D), we only store 1 partial propensity for the “diffusion reactions” of each species. This reduces the total number of reactions per subvolume from $M + 6N$ in 3D ($M + 4N$ in 2D) to $M + N$. For convenience, we define all reactions $\mu \leq M$ as “real” reactions and the reaction with index $\mu = M + i$ as the lumped “diffusion reaction” of species $S_i^{(l,m,n)}$. See Sec. 6.2.2.2 for how the direction of a “diffusion jump” is resolved when a lumped “diffusion reaction” has been selected.

The reaction indices of the partial propensities in $\Pi^{(l,m,n)}$ are stored in a look-up table $\mathbf{L} = \{\mathbf{L}_i\}_{i=0}^N$, which is also an array of arrays. For subvolumes containing the same reaction network, we store the look-up table only once. In case the reaction network is the same in all subvolumes, only a single, global look-up table is needed. Subvolumes that host different reaction networks have different look-up tables. The look-up table renders every reaction within each subvolume identifiable by a unique pair of indices, a group index I and an element index J , such that the partial propensity of reaction $\mu = L_{I,J}$ is stored in $\Pi_{I,J}^{(l,m,n)}$ for subvolume (l, m, n) .

The “group-sum array” $\Lambda^{(l,m,n)}$ stores the sums of the partial propensities in each group $\Pi_i^{(l,m,n)}$, i.e. $\Lambda_i^{(l,m,n)} = \sum_j \Pi_{i,j}^{(l,m,n)}$. We also store the total propensity of each group in an array Σ , computed as $\Sigma_i^{(l,m,n)} = n_i^{(l,m,n)} \Lambda_i^{(l,m,n)}$, $i = 1, \dots, N$, and $\Sigma_0^{(l,m,n)} = \Lambda_0^{(l,m,n)}$. The total propensity of the reactions in subvolume (l, m, n) is then $a^{(l,m,n)} = \sum_{i=0}^N \Sigma_i^{(l,m,n)}$. The total propensity of all reactions across all subvolumes is stored in $a = \sum_{l=1}^{K_x} \sum_{m=1}^{K_y} \sum_{n=1}^{K_z} a^{(l,m,n)}$ and is used to calculate the time to the next reaction according to Eq. 6.15 as $\tau = -a^{-1} \log_2 r_0$, where r_0 is a uniformly distributed random number in $[0, 1)$.

6.2.2.2 Algorithms

In PSRD, like in NSM, the subvolume containing the next reaction is sampled first. To this end, the total propensities $a^{(l,m,n)}$ of all subvolumes are sorted into $G_a = \log_2(a_{\max}/a_{\min}) + 1$ bins, such that bin b contains all $a^{(l,m,n)}$ in the interval $2^{b-1}a_{\min} \leq a^{(l,m,n)} < 2^b a_{\min}$. The bounds a_{\min} and a_{\max} are the smallest non-zero and the largest value that any of the $a^{(l,m,n)}$ can assume during the simulation. They are determined as follows: The lower bound a_{\min} is the minimum propensity of any reaction in any subvolume when the number of molecules of all reactants is one (minimum non-zero population). For elementary reactions, this is the smallest specific probability rate across all subvolumes. The largest possible value of $a^{(l,m,n)}$ may be ascertained using physical reasoning or prior knowledge about the reaction-diffusion system. In cases where this cannot be evaluated *a priori*, PSRD initially sets a_{\max} to the maximum $a^{(l,m,n)}$. If during the course of the simulation the maximum $a^{(l,m,n)}$ increases, PSRD updates a_{\max} and G_a , and the corresponding data structures are dynamically enlarged.

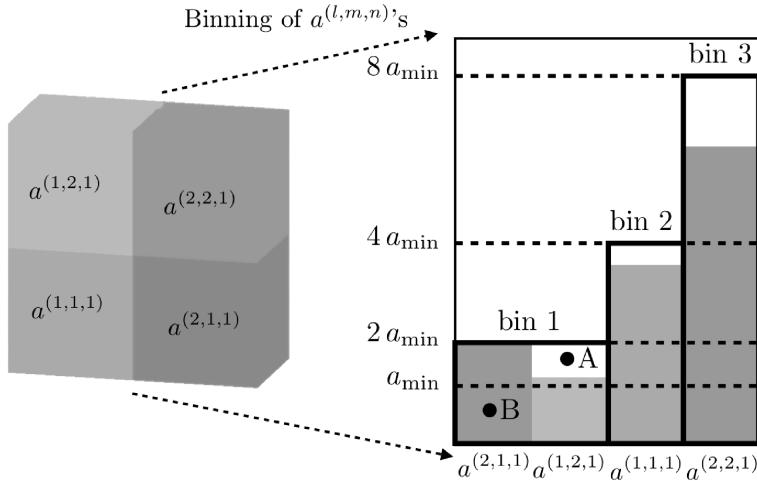


Figure 6.2: Illustration of the binning of the total propensities of the subvolumes used for composition-rejection sampling of the next subvolume. The illustration shows a computational domain divided into 4 subvolumes. Points A and B refer to the example in main text used to explain rejection sampling.

PSRD uses composition-rejection sampling to determine the subvolume of the next reaction in two steps: (i) composition step to find the bin b and (ii) rejection step to find $a^{(l,m,n)}$ inside that bin. The composition step uses linear search to determine

$$b = \min \left[b' : r_1 a < \sum_{i=1}^{b'} \alpha_i \right], \quad (6.18)$$

where r_1 is a uniform random number in $[0, 1]$ and α_i is the total propensity in bin i computed by summing up the $a^{(l,m,n)}$ in that bin. The rejection step samples the subvolume (l, m, n) among the entries in the selected bin b . For this, we first generate a uniformly distributed random number r_2 in $[0, 2^b a_{\min}]$ and a uniformly distributed random integer r_3 between 1 and the number of entries in bin b . If the r_3 -th element in bin b is greater than or equal to r_2 , the corresponding subvolume is selected. If the inequality is not satisfied, the rejection step is repeated. This procedure is illustrated in Fig. 6.2 for a computational domain divided into 4 subvolumes with indices $(1,1,1)$, $(2,1,1)$, $(1,2,1)$ and $(2,2,1)$. Assume that the composition step has chosen bin 1 as the bin containing the next subvolume. The rejection step then samples uniformly random points inside the rectangle defining the range of this bin (bold rectangle). A sample is accepted if it falls inside one of the shaded bars representing the $a^{(l,m,n)}$'s. If the first sample (point A in Fig. 6.2 with $r_3 = 2$ and $r_2 > a^{(1,2,1)}$) is rejected, sampling is repeated until the point falls inside one of the shaded bars (point B in Fig. 6.2 with $r_3 = 1$ and $r_2 < a^{(2,1,1)}$). By binning the $a^{(l,m,n)}$ as described above, it is guaranteed that the area covered by the $a^{(l,m,n)}$ bars in each bin is at least 50% of the bin's total area. This ensures that the expected number of rejection steps required is ≤ 2 . The probability of

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

needing more than k rejection steps is $\leq 2^{-k}$ and hence exponentially small.

Once the subvolume (l, m, n) containing the next reaction has been chosen, PSRD samples the index of the next reaction μ within that subvolume in two steps: (i) perform linear search for the group index I such that

$$I = \min \left[I' : r_4 a^{(l,m,n)} < \sum_{i=0}^{I'} \Sigma_i^{(l,m,n)} \right] \quad (6.19)$$

and (ii) perform linear search for the element index J inside group $\Pi_I^{(l,m,n)}$ such that

$$J = \min \left[J' : r_4 a^{(l,m,n)} < \sum_{j=1}^{J'} n_I^{(l,m,n)} \Pi_{I,j}^{(l,m,n)} + \left(\sum_{i=0}^I \Sigma_i^{(l,m,n)} \right) - \Sigma_I^{(l,m,n)} \right], \quad (6.20)$$

where r_4 is a uniform random number in $[0,1]$ (see Eqs. 3.4 and 3.5 for the procedure to sample J efficiently). The indices I and J are then translated to the reaction index μ in subvolume (l, m, n) using the look-up table \mathbf{L} , thus $\mu = L_{I,J}$. In order to reduce the average search depth, the group and element indices are dynamically rearranged such that frequent reactions accumulate at the beginning of the list, i.e., have low index values. This is done by dynamically bubbling up a reaction whenever it fires by performing a single iteration of a bubble-sort algorithm. The permutation lists for the reordered indices in each subvolume are stored in an array for the I 's, and one-dimensional array of one-dimensional arrays of the size of $\Pi^{(l,m,n)}$ for the J 's. PSRD thus uses the sorting partial-propensity direct method (SPDM) to sample the next reaction within a subvolume (see Sec. 3.3.2). This renders the sampling procedure more efficient (in the sense that it reduces the prefactor in the scaling of the computational cost) when the reaction network in a subvolume is multi-scale (stiff), without compromising on the efficiency in non-stiff cases. In SRD simulations the reaction networks inside the subvolumes tend to be stiff since the specific probability rates of bimolecular reactions scale as h^{-3} (in 3D subvolumes) whereas those of source reactions scale as h^3 (see Eq. 6.6). Using SPDM instead of PDM may hence lead to significant computational savings.

Once a reaction has been executed, $\mathbf{n}^{(l,m,n)}$, $\Pi^{(l,m,n)}$, $\Lambda^{(l,m,n)}$, and $\Sigma^{(l,m,n)}$ need to be updated. This is efficiently done using three update structures. If the reaction network is the same in each subvolume, the same update structures can be used for all subvolumes and they do not have to be stored separately for different subvolumes. Subvolumes containing different reaction networks have different update structures.

- $\mathbf{U}^{(1)}$ is an array of M arrays, where the i^{th} array contains the indices of all species involved in the i^{th} “real” reaction. The index of the species involved in the i^{th} lumped “diffusion reaction” does not need to be stored as it is simply i itself.
- $\mathbf{U}^{(2)}$ is a array of M arrays containing the corresponding stoichiometries (the change in population of each species upon reaction) of the species stored in $\mathbf{U}^{(1)}$. The stoichiometries of the “diffusion reactions” are not stored since they are all -1 .
- $\mathbf{U}^{(3)}$ is a array of N arrays, where the i^{th} array contains the indices of all entries in the $\Pi^{(l,m,n)}$'s that depend on $n_i^{(l,m,n)}$.

6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

When a reaction is executed in subvolume (l, m, n) , the populations of the species involved in this reaction change. Hence, all entries in $\Pi^{(l,m,n)}$ that depend on these populations need to be updated. After each reaction, we use $\mathbf{U}^{(1)}$ to determine the indices of all species involved in this reaction. The stoichiometry is then looked up in $\mathbf{U}^{(2)}$ and the population $\mathbf{n}^{(l,m,n)}$ is updated. Subsequently, $\mathbf{U}^{(3)}$ is used to locate the affected entries in $\Pi^{(l,m,n)}$ and recompute them. The two data structures $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ hence amount to a sparse representation of the stoichiometry matrix; $\mathbf{U}^{(3)}$ represents the dependency graph over species. Since the partial propensities of unimolecular and source reactions are constant and never need to be updated, $\mathbf{U}^{(3)}$ only contains the indices of the partial propensities of bimolecular reactions. Along with updating the partial propensities in subvolume (l, m, n) , the change in the total propensity of that subvolume is also calculated and incrementally applied to $a^{(l,m,n)}$. This may require the bin membership of $a^{(l,m,n)}$ to be updated, for which the current bin assignment of $a^{(l,m,n)}$ must be known. We implement this by storing two additional integers for every $a^{(l,m,n)}$: one for the bin membership and the other for the location inside that bin. Depending on its new value, $a^{(l,m,n)}$ may remain in the same bin or move to a different one. Removal of an element from a bin is done by replacing it with the last element in that bin and reducing the bin size by one. Addition of an element into a bin is done by appending it at end of the bin. The computational cost of both of these operations is $O(1)$ (Slepoy et al., 2008) (see Sec. 3.4).

If the index of the next reaction is greater than M , then the sampled reaction is a lumped “diffusion reaction” and additional steps need to be taken to resolve the direction of the jump as follows: First, the index of the species undergoing diffusion is computed as $i = \mu - M$. Second, a uniform random number between 0 and the lumped specific probability rate of the lumped “diffusion reaction” is generated. Third, linear search over the specific probability rates of individual directional diffusion events is used to determine the target subvolume of the jump. The jump is executed by increasing the population of species S_i in the target subvolume by 1 (the reduction in the source subvolume has already been done above) and updating the entries in the partial propensity structure of the target subvolume as given by the indices in $\mathbf{U}_i^{(3)}$. Finally, the total propensity of the target subvolume and its bin membership are updated.

Figure 6.3 summarizes the data structures used in PSRD for the example reaction network given in Eq. 6.1. The complete algorithm is given in Table 6.1. The computational cost of PSRD to sample the subvolume is $O(1)$ if the ratio of maximum to minimum non-zero total propensity in each subvolume is independent of the number of subvolumes and of the size of the reaction network. In cases where this ratio is not bounded by a constant, the computational cost to sample the subvolume is $O(G_a)$, where the total number of bins G_a depends on the logarithmic span of the subvolume propensities as $G_a = \log_2(a_{\max}/a_{\min}) + 1$. The computational cost of sampling the index of the next reaction within a subvolume is $O(N)$. The overall computational cost of PSRD hence is $O(G_a + N)$, which is at most $O(\log_2 N_v + N)$. The memory requirement of PSRD is $O((M + N)N_v)$. For more details on the computational cost and the memory requirement, see Sec. 6.2.3.

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

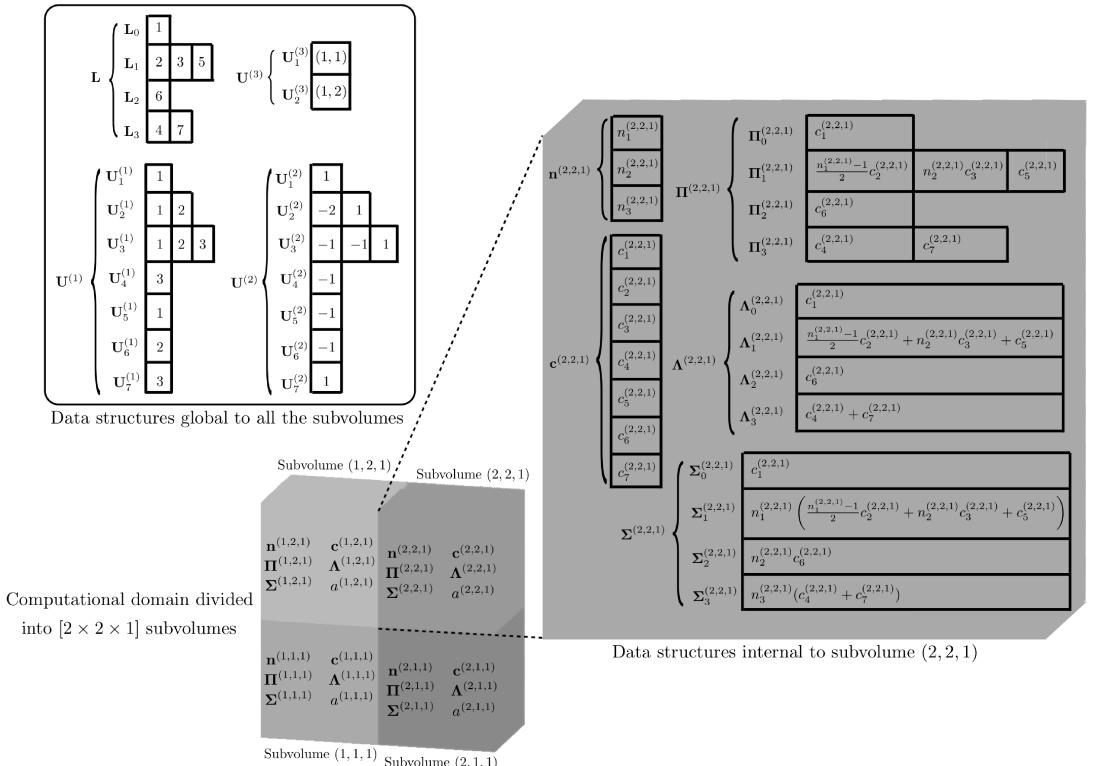


Figure 6.3: The data structures in PSRD. The contents of the data structures shown corresponds to the example reaction network in Eq. 6.1 with 3 species and 4 reactions. We assume that the computational domain is divided into 4 subvolumes. In the illustration, $c_5^{(2,2,1)}$, $c_6^{(2,2,1)}$, and $c_7^{(2,2,1)}$ are the lumped specific probability rates of the “diffusion reactions” of species 1, 2, and 3 respectively. See main text for further details.

0. Assume that the reaction network has N species and M reactions. Divide the computational box into $N_v = K_x K_y K_z$ cubic subvolumes of edge length h . Formulate the reaction network modeling the reaction-diffusion system by including the jump reactions. The resulting reaction network for the reaction-diffusion system has at most $M + 6N$ reactions and N species in each subvolume. Lump the “diffusion reactions” of each species in each subvolume into one reaction with no products, such that number of reactions in each subvolume is $M + N$. Make sure that the reaction index of the lumped “diffusion reactions” is $> M$ and that reaction $\mu = M + i$ is the lumped “diffusion reaction” of species i .
1. Set time $t \leftarrow 0$. Initialize the data structures in each subvolume (l, m, n) : the partial-propensity structure $\Pi^{(l,m,n)}$, the group-sum array $\Lambda^{(l,m,n)}$, $\Sigma^{(l,m,n)}$, the population $\mathbf{n}^{(l,m,n)}$, the specific probability rates $\mathbf{c}^{(l,m,n)}$, and the total propensity in the subvolume $a^{(l,m,n)}$. Also initialize the data structures global to all subvolumes: the look-up table \mathbf{L} , the sparse representation of the stoichiometry matrix $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$, the dependency graph over species $\mathbf{U}^{(3)}$, and the total propensity of all subvolumes a . Bin the $a^{(l,m,n)}$ into G_a bins as described in Sec. 6.2.2.2.
2. While $t < t_f$, repeat:
 - 2.1. Compute the time to the next reaction $\tau \leftarrow a^{-1} \ln(r^{-1})$, where a is the total propensity of all reactions and r a uniformly distributed random number in $[0, 1)$.
 - 2.2. Use composition-rejection sampling to determine the subvolume (l, m, n) containing the next reaction. Use linear search (Eq. 6.18) in the composition step to locate the bin containing $a^{(l,m,n)}$ and use the rejection step to locate $a^{(l,m,n)}$ inside that bin.
 - 2.3. Sample the next reaction μ in subvolume (l, m, n) by sampling its group and element indices. Sample the group index I using linear search over $\Sigma^{(l,m,n)}$ (Eq. 6.19). Subsequently, sample the element index J using linear search over $\Pi_I^{(l,m,n)}$ (Eq. 6.20). The reaction index μ is then identified from the look-up table as $\mu = \mathbf{L}_{I,J}$.
 - 2.4. Update the internal data structures in subvolume (l, m, n) and the total propensity a using $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$.
 - 2.5. Increase the number of bins G_a if necessary and update the bin membership of $a^{(l,m,n)}$
 - 2.6. If $\mu > M$ (i.e., the sample reaction is a lumped “diffusion reaction”), compute the index of the diffusing species as $s = \mu - M$. Resolve the diffusion event to identify the neighboring target subvolume (l', m', n') to which species s of subvolume (l, m, n) is diffusing. Update the population of species s in the target subvolume. Subsequently, update the other internal data structures of the target subvolume and the total propensity a using $\mathbf{U}_s^{(3)}$. Increase the number of bins G_a if necessary and update the bin membership of $a^{(l',m',n')}$.
 - 2.6. Advance time: $t \leftarrow t + \tau$.
3. Stop.

6.2.3 Computational cost

The steps that define the scaling of the computational cost of PSRD are the sampling of the subvolume containing the next reaction, the sampling of the next reaction within that subvolume, and the update of the data structures after firing the sampled reaction.

The computational cost of the composition-rejection sampling of the next subvolume is $O(G_a)$. This is because (i) the composition step is a linear search over G_a bins, and (ii) the rejection step is $O(1)$ since the average number of iterations for this step is bounded by a constant thanks to the dyadic binning (Slepoy et al., 2008) (see Sec. 3.4).

The computational cost for sampling the next reaction within the selected subvolume (l, m, n) is $O(N)$. This step involves sampling the group index I and the element index J of the next reaction in the partial-propensity structure. Sampling the group index involves a linear search over the at most $N + 1$ elements of $\Sigma^{(l,m,n)}$ and hence has a computational cost of $O(N)$. Sampling the element index involves a linear search over the $O(N)$ elements of $\Pi_I^{(l,m,n)}$ and hence has a computational cost of $O(N)$ as well.

The computational cost for updating the data structures within a subvolume is at most $O(N)$. Assuming that the number of distinct species involved in any one chemical reaction is $O(1)$ (i.e., does not increase beyond a constant bound as the number of species in the network increases), the cost of updating the population of species is $O(1)$. Under the same assumption, the number of entries in $\Pi^{(l,m,n)}$ that need to be updated after any reaction has fired is at most $O(N)$.

By the same argument, the cost of updating the partial-propensity structure of any neighboring subvolume upon firing of a “diffusion reaction” is at most $O(N)$.

Overall, the computational cost of PSRD thus is $O(G_a + N)$, irrespective of the fraction f_r of “real” reaction firings.

The asymptotically (for large N_v) worst case for PSRD is when half of the subvolumes contain bimolecular reactions and the other half source reactions. In 3D subvolumes, the propensity of the bimolecular reactions is proportional to h^{-3} whereas that of the source reactions is proportional to h^3 , where h is the edge length of the subvolumes. As N_v increases, the logarithmic span of the propensities in the system hence increases. This leads to an increase in the number of bins G_a that is proportional to $\log_2 h^{-6} = 2 \log_2 N_v - 2 \log_2 \Omega$, where Ω is the (constant) volume of the reactor. Therefore, $G_a \in O(\log_2 N_v)$, rendering the computational cost of PSRD $O(\log_2 N_v + N)$ in the worst case, independent of f_r . This worst-case scaling of PSRD’s computational cost can be reduced to $O(\log_2 \log_2 N_v + N)$ by using a tree search (Gibson and Bruck, 2000) to sample b in Eq. 6.18.

The data structures of PSRD require $O(M + N)$ memory per subvolume. Therefore, the total memory requirement of PSRD is $O((M + N)N_v)$.

6.2.4 Benchmarks

We analyze the computational cost of PSRD as quantified by the average simulation (CPU) time Θ taken per reaction event of the chemical reaction model of a reaction-diffusion system. We compare it to the time expected from the theoretical cost analysis (see Sec. 6.2.3) for two different types of reaction networks: (i) an aggregation model where the number of reactions increases super-linearly with the number of species and (ii) a linear chain model where the

number of reactions is almost the same as the number of species. We simulate the corresponding reaction-diffusion processes in a three-dimensional cubic computational domain (reactor) of dimensions $L \times L \times L$ and volume $\Omega = L^3$ from a initial time $t = 0$ until a final time $t = t_f$. For simulating the reaction-diffusion process we divide the computational domain into $N_v = K^3$ equi-sized cubic subvolumes of edge length $h = L/K$, such that K is the number of subvolumes along each spatial dimension.

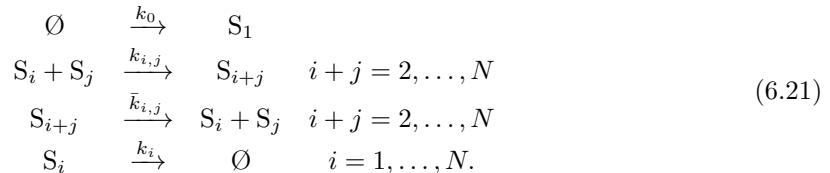
For each of these networks we report Θ as a function of the number of subvolumes N_v for a fixed size of the reaction network and as a function of the reaction network size for a fixed number of subvolumes. We use the number of species N in the reaction network to quantify the size of the network. All timings are compared to those obtained on the same systems and the same computer using NSM.

Both PSRD and NSM were implemented in C++ using the random number generator of the GSL library and compiled using the Intel C++ compiler version 12.0.2 with the O3 optimization flag. NSM is implemented according to the details provided on the MesoRD webpage (Algorithm 7 in (Hattne, 2006)). All timings were measured on a Linux 2.6 workstation with a 2.8 GHz quad-core Intel Xeon E5462 processor, 8 GB of memory and 4 MB L2 cache. For all test cases, we simulate until a final time t_f and report the average CPU time Θ per reaction event. The time Θ does not include the initialization of the data structures as this is done only once. We explain the measurements by least-squares fits of $\Theta(N_v, N)$ with the corresponding theoretical cost models. For PSRD and NSM, we hence fit Θ with $\alpha_1 \log_2 N_v + \alpha_2 N$ and $\alpha_1 \log_2 N_v + \alpha_2 f_r M + \alpha_3 (1 - f_r) 6N$, respectively. Before fitting, we estimate the functional dependence of f_r on N_v or N by performing simulations. Subsequently, we fit Θ to determine the coefficients α_i .

All simulations are run without any *a priori* estimate of the maximum total propensity a_{\max} across all subvolumes. Instead, a_{\max} is constantly updated during a simulation and the number of bins G_a is dynamically increased when required (see Sec. 6.2.2.2).

6.2.4.1 Colloidal aggregation model

We consider the nonequilibrium colloidal aggregation model as a prototype of a strongly coupled reaction network in which the number of reactions increases super-linearly with the number of species:

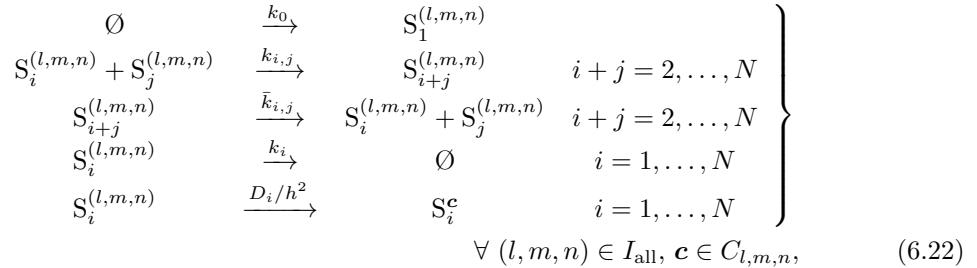


The k 's are the macroscopic reaction rates. This system of reactions models the influx of monomers (S_1) into a reactor where multimers (S_i) fuse with each other to form larger multimers. Multimers in the reactor also break to form smaller units in all possible combinations, and all of the multimers can leave the reactor. Such a system of reactions models driven colloidal aggregation and is relevant for a variety of phenomena of practical importance, e.g., nano-particle clustering and colloidal crystallization (nanotechnology), emulsifi-

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

cation and emulsion stabilization in porous media (oil industry), and oligomerization of proteins (biochemistry). For N chemical species, the aggregation reaction network consists of $M = \lfloor N^2/2 \rfloor + N + 1$ reactions.

We divide the cubic computational domain (reactor) into $N_v = K^3$ subvolumes, such that the on-lattice reaction-diffusion process with reflective boundary conditions is described by the following set of “reactions”:



where D_i is the diffusion constant of species S_i and h the edge length of the cubic subvolumes. The propensities of these reactions are computed as described in Sec. 6.1. For all bimolecular reactions, we use discretization-corrected propensities (see Eqs. 6.9 and 6.11). The above network consists of $NK^3 = NN_v$ species and $MK^3 + N(6K^3 - 6K^2) = (\lfloor N^2/2 \rfloor + 1)N_v + N(7N_v - 6N_v^{2/3})$ reactions.

For the present benchmarks, we set the macroscopic reaction rates $k_{i,i} = 0.5$, all other rates and all diffusion constants to 1, and the reactor volume to $\Omega = 10$. At time $t = 0$, the populations of all species in all subvolumes, i.e. all $n_i^{(l,m,n)}$, are set to 0. From this initial condition we simulate the reaction-diffusion system until $t_f = 100$.

Figure 6.4A shows the computational cost Θ as a function of the number of subvolumes N_v using PSRD and NSM for two fixed-size aggregation networks with $N = 10$ and $N = 100$, respectively. The corresponding numbers of reactions M are 61 and 5101, respectively. In both cases we estimate f_r and use it for fitting Θ . We observe that f_r decreases as $N_v^{-0.34}$ with increasing N_v . For PSRD, $\Theta(N_v, N = 10) \approx 0.02861 \log N_v$ at large N_v . This scaling of Θ is caused by the dynamic increase in the number of bins G_a . For NSM, $\Theta(N_v, N = 10) \approx 0.1095 \log N_v$ at large N_v . For the larger network with $N = 100$, we find for PSRD $\Theta(N_v, N = 100) \approx 0.04401 \log N_v$ at large N_v . For NSM, $\Theta(N_v, N = 100) \approx 0.288 \log N_v$ at large N_v . For smaller $1 \leq N_v < 512$, Θ of NSM decreases with increasing N_v . This is mediated by the decrease in f_r . At $N_v = 1$, $f_r = 1$ and the cost is dominated by that of sampling the next “real” reaction. As N_v increases, f_r decreases. This decrease in f_r progressively reduces the cost of sampling a reaction in a subvolume from being linear in M to linear in $6N$. At large-enough N_v , the cost of sampling a reaction in a subvolume is dominated by the cost of sampling “diffusion reactions”. For a fixed network size, the increase in $\Theta(N_v, N = 100)$ at large N_v is $O(\log_2 N_v)$. This scaling is asymptotically (for large N_v) the same as that of NSM, but with a smaller prefactor.

Figure 6.4B shows the computational cost Θ as a function of the size N of the aggregation reaction network using PSRD and NSM with $N_v = 512$ and $N_v = 1000$ subvolumes. We

6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

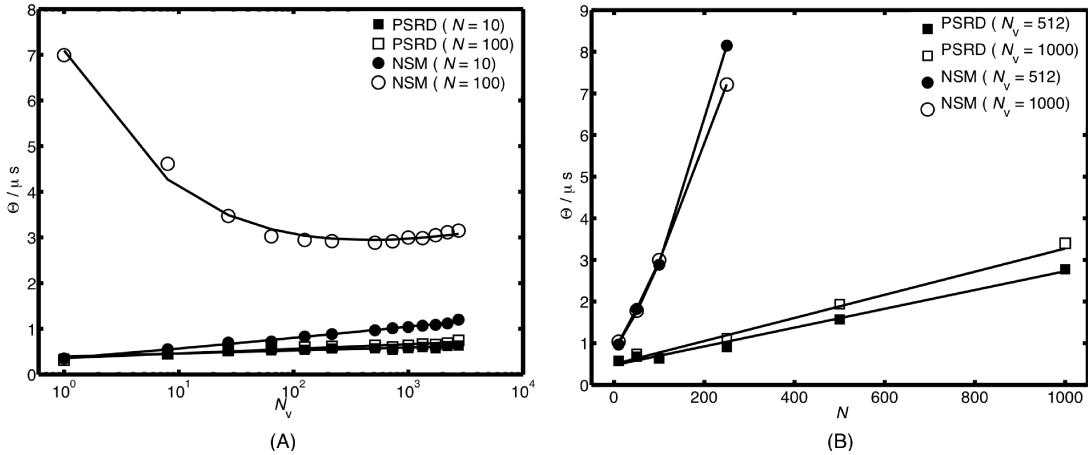
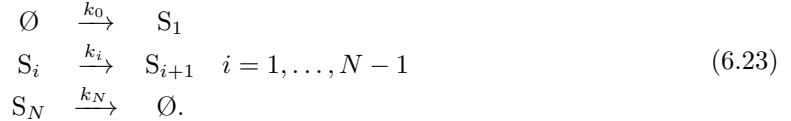


Figure 6.4: Computational cost of PSRD and NSM for the aggregation model (Eq. 6.21). (A) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of subvolumes N_v with the size of the reaction network fixed to $N = 10$ (filled symbols) and $N = 100$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N = 10$, $\Theta^{\text{PSRD}} \approx 0.02861 \log N_v + 0.03925N$, $\Theta^{\text{NSM}} \approx 0.1095 \log N_v + 0.00581 f_r M + 0.00481(1 - f_r)6N$; for $N = 100$, $\Theta^{\text{PSRD}} \approx 0.04401 \log N_v + 0.003579N$, $\Theta^{\text{NSM}} \approx 0.288 \log N_v + 0.001375 f_r M + 0.001418(1 - f_r)6N$. We estimate $f_r \approx 1.096 N_v^{-0.3353} - 0.08263$ for $N = 10$ and $f_r = 1.097 N_v^{-0.3372} - 0.0825$ for $N = 100$. (B) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of species N in the reaction network with the number of subvolumes fixed to $N_v = 512$ (filled symbols) and $N_v = 1000$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N_v = 512$, $\Theta^{\text{PSRD}} \approx 0.07559 \log N_v + 0.002258N$, $\Theta^{\text{NSM}} \approx 0.1356 \log N_v + 0.002784 f_r (\lfloor N^2/2 \rfloor + N + 1) + 0.002633(1 - f_r)6N$; for $N_v = 1000$, $\Theta^{\text{PSRD}} \approx 0.07205 \log N_v + 0.002777N$, $\Theta^{\text{NSM}} \approx 0.1198 \log N_v + 0.002762 f_r (\lfloor N^2/2 \rfloor + N + 1) + 0.003163(1 - f_r)6N$. The fraction $f_r = 0.04$ for $N_v = 512$ and $f_r = 0.02$ for $N_v = 1000$.

observe that for both N_v the ratio f_r does not depend on the size N of network. For $N_v = 512$, $f_r = 0.04$, decreasing to $f_r = 0.02$ for $N_v = 1000$. For PSRD, $\Theta(N_v = 512, N) \approx 0.002258 N$, confirming the linear dependence on N predicted by the theoretical cost analysis. For NSM, $\Theta(N_v = 1000, N) \approx 0.00011M + 0.0152N$. For the larger number subvolumes, $N_v = 1000$, $\Theta(N_v = 1000, N) \approx 0.002777N$ for PSRD. For NSM, $\Theta(N_v = 1000, N) \approx 0.000055M + 0.0186N$. In summary, the scaling of the computational cost of PSRD with respect to the size N of the reaction network is $O(N)$.

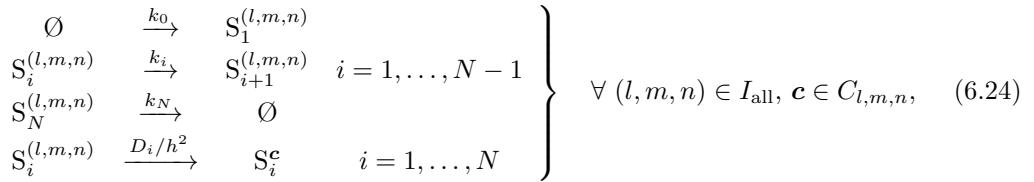
6.2.4.2 Linear chain model

As a prototypical reaction network in which the number of reactions is almost the same as the number of species, we consider the nonequilibrium linear chain model:



Again, the k 's are the macroscopic reaction rates. This linear chain of reactions can, e.g., be used to model signal transduction pathways in biological cells (Albert and Barabási, 2002; Albert, 2005). For N species, this network contains $M = N + 1$ reactions.

Again dividing the cubic computational domain into $N_v = K^3$ subvolumes, the resulting reaction-diffusion system with reflective boundary conditions is given by:



where D_i is the diffusion constant of species S_i and h is the edge length of the cubic subvolumes. The propensities of these reactions are computed as described in Sec. 6.1. There are no bimolecular reactions in this network, and we do not use discretization-corrected propensities (Erban and Chapman, 2009). This system of reactions modeling the reaction-diffusion process contains $NK^3 = NN_v$ species and $MK^3 + N(6K^3 - 6K^2) = (7N + 1)N_v - 6NN_v^{2/3}$ reactions.

For the benchmarks we set all macroscopic reactions rates and all diffusion constants D_i to 1, and the volume of the reactor to $\Omega = 100$. At time $t = 0$, the populations of all species in all subvolumes are 0, and the simulation is run until $t_f = 100$.

Figure 6.5A shows the computational cost Θ as a function of the number of subvolumes N_v using PSRD and NSM for two fixed-size linear chain networks with $N = 10$ and $N = 100$. The corresponding numbers of reactions M are 11 and 101, respectively. In both cases we estimate f_r and use it for fitting Θ . We observe that f_r decreases as $N_v^{-0.22}$ with increasing N_v . For PSRD, $\Theta(N_v, N = 10) \approx 0.03312 \log N_v$. This scaling of Θ is caused by the increase in the number of bins G_a . For NSM, $\Theta(N_v, N = 10) \approx 0.08256 \log N_v$. For the larger network with $N = 100$, the computational cost of PSRD is $\Theta(N_v, N = 100) \approx 0.04842 \log N_v$ for $N_v \lesssim 512$ and $\Theta(N_v, N = 100) \approx 0.2923 \log N_v$ for $N_v \gtrsim 512$. For NSM, $\Theta(N_v, N = 100) \approx 0.1428 \log N_v$ for $N_v \lesssim 512$ and $\Theta(N_v, N = 100) \approx 0.5929 \log N_v$ for $N_v \gtrsim 512$. The abrupt increase in the prefactor of the scaling around $N_v \approx 512$ is likely caused by cache-memory effects. In summary, the scaling of the computational cost of PSRD with respect to the number of subvolumes N_v is $O(\log_2 N_v)$. Again, this is the same scaling as that of NSM, but with a smaller prefactor.

Figure 6.5B shows the computational cost Θ as a function of the size N of the linear chain network using PSRD and NSM with $N_v = 512$ and $N_v = 1728$ subvolumes. We observe that

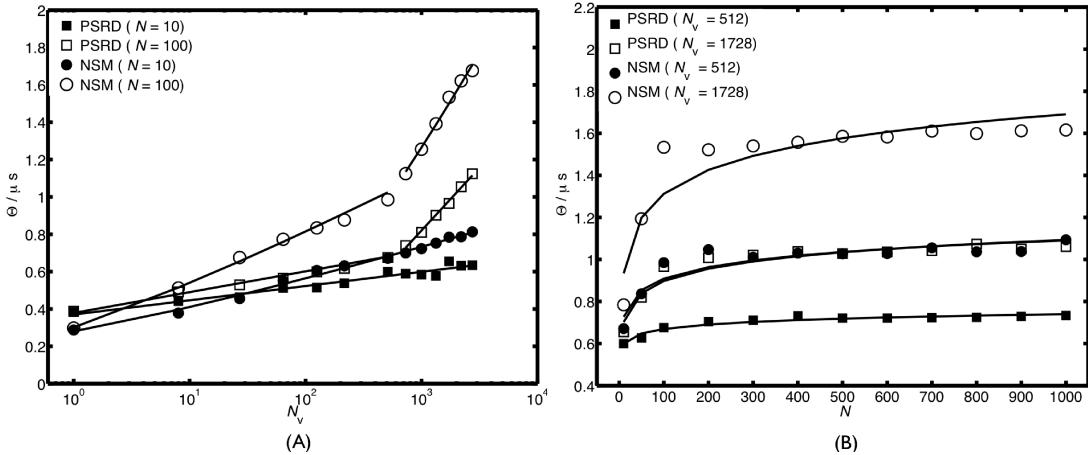


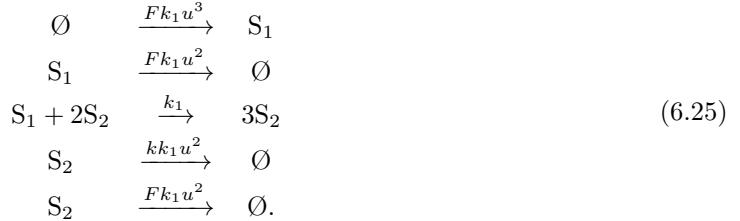
Figure 6.5: Computational cost of PSRD and NSM for the linear chain model (Eq. 6.23). (A) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of subvolumes N_v with the size of the reaction network fixed to $N = 10$ (filled symbols) and $N = 100$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N = 10$, $\Theta^{\text{PSRD}} \approx 0.03312 \log N_v + 0.03703N$, $\Theta^{\text{NSM}} \approx 0.08256 \log N_v + 0.02504f_r M + 0.002615(1 - f_r)6N$; for $N = 100$, $\Theta^{\text{PSRD}} \approx 0.04842 \log N_v + 0.003786N$, $\Theta^{\text{NSM}} \approx 0.1428 \log N_v + 0.002934f_r M + 0.0001978(1 - f_r)6N$ for $N_v \lesssim 512$ and $\Theta^{\text{PSRD}} \approx 0.2923 \log N_v - 0.01199N$, $\Theta^{\text{NSM}} \approx 0.5929 \log N_v + 0.0000008f_r M - 0.004924(1 - f_r)6N$ for $N_v \gtrsim 512$. We estimate $f_r \approx 1.281N_v^{-0.2121} - 0.2505$ for $N = 10$ and $f_r \approx 1.241N_v^{-0.2291} - 0.2138$ for $N = 100$. (B) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of species N in the reaction network with the number of subvolumes fixed to $N_v = 512$ (filled symbols) and $N_v = 1728$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N_v = 512$, $\Theta^{\text{PSRD}} \approx 0.03051 \log N + 0.5291$, $\Theta^{\text{NSM}} \approx 0.07885 \log N + 0.5458$; for $N_v = 1728$, $\Theta^{\text{PSRD}} \approx 0.08479 \log N + 0.5073$, $\Theta^{\text{NSM}} \approx 0.1642 \log N + 0.5561$. The fraction $f_r = 0.06$ for $N_v = 512$ and $f_r = 0.03$ for $N_v = 1728$.

for both N_v the ratio f_r is independent of the size N of the network. For $N_v = 512$, $f_r = 0.06$, decreasing to $f_r = 0.03$ for $N_v = 1728$. We observe that the scaling of Θ is slower than predicted by the theoretical cost analysis. This is not a violation of the theory since the theoretical analysis only provides an upper bound for the scaling. The slower scaling in the present case is specific to the particular reaction network. We obtain reasonable fits with a function linear in $\log N$. The asymptotic plateau is due to “diffusion reactions” of species S_1 accounting for the majority of all reaction firings. Since this reaction is on top of the list (species index 1), it is found in $O(1)$ time. For PSRD, $\Theta(N_v = 512, N) \approx 0.03051 \log N$. For NSM, $\Theta(N_v = 512, N) \approx 0.07885 \log N$. For the larger number subvolumes $N_v = 1728$, Θ of PSRD is $\Theta(N_v = 1728, N) \approx 0.08479 \log N$. For NSM, $\Theta(N_v = 1728, N) \approx 0.1642 \log N$. In summary, the scaling of the computational cost of PSRD with respect to the size N of the reaction network is $O(N)$, since $\log N \in O(N)$.

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

6.2.5 Two- and three-dimensional SRD simulations using PSRD

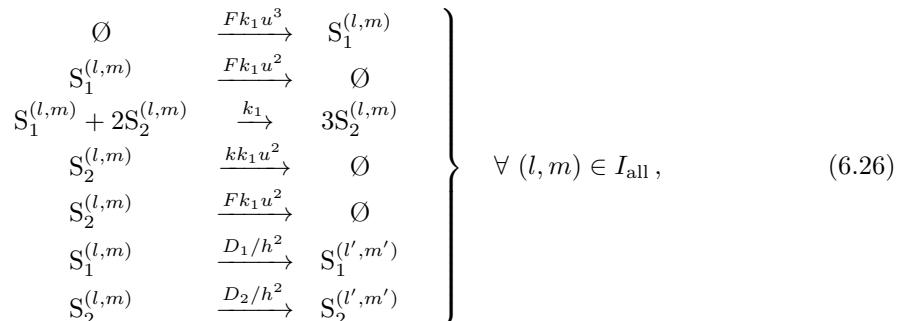
As an example application we use PSRD for two- and three-dimensional SRD simulations of the Gray-Scott reaction system (Gray and Scott, 1983, 1984, 1985; Pearson, 1993; Lee et al., 1993), given by:



This system is widely used to study the formation of Turing patterns (Turing, 1952) in reaction-diffusion systems (Pearson, 1993). The trivial steady state of the system is $n_1 = u\Omega$ and $n_2 = 0$, where Ω is the volume of the reactor. For a fixed reactor volume, a larger u hence corresponds to a larger number of molecules in the reactor, reducing the effect of noise. In the limit of very large u , the kinetics of the stochastic system tends to that of the deterministic one.

The third reaction in the system is not elementary since it involves three reactant molecules. We therefore extend PSRD to also handle trimolecular reactions by using a three-dimensional partial-propensity structure and factoring out two other reactants. We choose this strategy over expanding the network into elementary reactions in order to render the parameters k and F comparable to the deterministic limit case. We do not use discretization-corrected propensities since no theoretical framework is available for trimolecular reactions (Fange et al., 2010).

For the simulations we fix the dimensionless constants such that $F = 0.04$ and $k = 0.06$, and we choose the macroscopic rate $k_1 = 1$. In 2D we simulate the reaction-diffusion system in a computational domain of area $\Omega = 0.64^2$, divided into $K^2 = 64^2$ subvolumes (or subareas) of edge length $h = 0.01$. At the boundary of the computational domain, periodic boundary conditions are used for the jump reactions. The resulting reaction-diffusion system in 2D thus is:



where I_{all} is the set of all possible subvolume indices in 2D and (l', m') are the neighboring subvolumes of (l, m) taking into account the periodic boundary conditions, hence $l' \in \{[(l -$

6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

$1) \pm 1 + 2K) \bmod K] + 1\}$ and $m' \in \{[(m - 1) \pm 1 + 2K) \bmod K] + 1\}$. At $t = 0$, the initial population is:

$$\begin{aligned} n_1^{(l,m)} &= \begin{cases} \frac{uh^2}{2} + \lfloor 0.04(r - 0.5)uh^2 + 0.5 \rfloor, & \text{for } 24 \leq l, m \leq 40 \\ uh^2, & \text{otherwise.} \end{cases} \\ n_2^{(l,m)} &= \begin{cases} \frac{uh^2}{4} + \lfloor 0.02(r - 0.5)uh^2 + 0.5 \rfloor, & \text{for } 24 \leq l, m \leq 40 \\ 0, & \text{otherwise,} \end{cases} \end{aligned} \quad (6.27)$$

where r is a uniform random number in $[0, 1)$ that acts as an initial perturbation. We use the diffusion constants $D_1 = 2 \cdot 10^{-5}$ and $D_2 = D_1/2$.

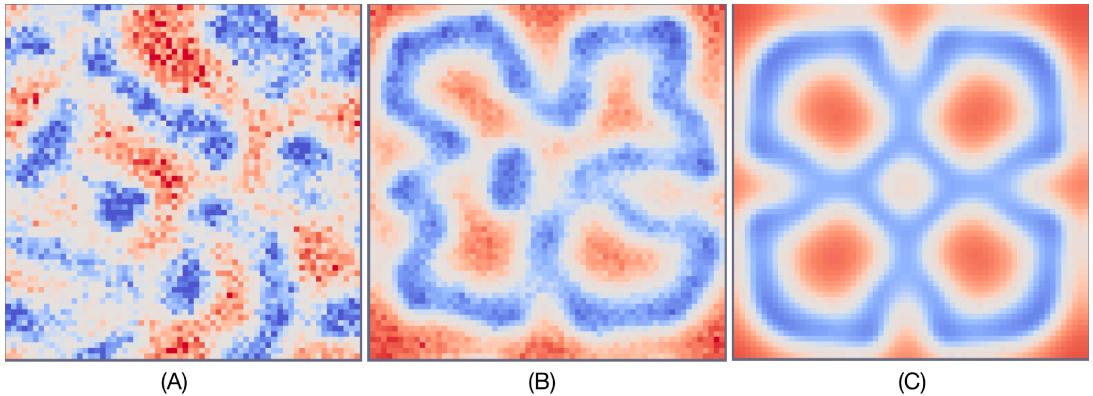


Figure 6.6: Normalized spatial concentration distribution of species S_1 in the two-dimensional Gray-Scott reaction-diffusion system (Eq. 6.26) for $F = 0.04$, $k = 0.06$, $k_1 = 1$, and $D_1 = 2D_2 = 2 \cdot 10^{-5}$ in a square computational domain of area 0.64^2 , divided into $N_v = 64^2$ subvolumes (or subareas) of edge length $h = 0.01$. The concentration in each subvolume is shown as a color ranging from blue (concentration zero) to red (concentration one). (A)+(B) Concentration distributions, normalized by u , obtained using PSRD for $u = 10^6$ and $u = 10^7$, respectively. (C) Normalized concentration distribution obtained from a deterministic simulation using the same parameters, simulated using second-order finite differences. All snapshots are taken at time $t = 2000/(k_1 u^2)$.

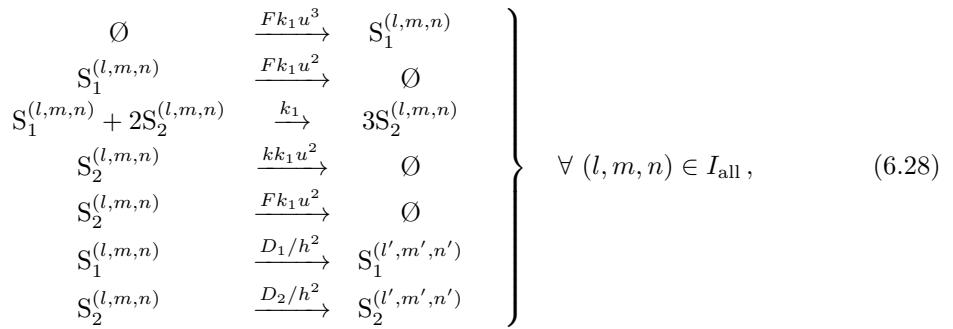
Figure 6.6 shows the 2D simulation results from PSRD and from a deterministic simulation. It shows the spatial concentration distribution of species S_1 , normalized with u , at time $t_f = 2000/(k_1 u^2)$. Figures 6.6A and 6.6B show the normalized concentration distributions for $u = 10^6$ and 10^7 , respectively, as obtained using PSRD. The maximum number of molecules of S_1 in any subvolume is on the order of $h^2 u = 0.01 u$. For $u = 10^6$, approximately $0.3 \cdot 10^9$ reaction events are simulated until t_f with $f_r \approx 0.12$ and a total runtime (CPU time) of 157 s for PSRD and 200 s for NSM. For $u = 10^7$, the number of reaction events happening during the simulation increases to $\approx 3 \cdot 10^9$ with $f_r \approx 0.14$ and a total runtime (CPU time) of 1854 s for PSRD and 2290 s for NSM.

Increasing u increases the total number of molecules in the reactor and hence decreases the noise in the system. The normalized concentration distribution obtained from a deterministic

6.2 The partial-propensity stochastic reaction-diffusion method (PSRD)

simulation is independent of u and is shown in Fig. 6.6C. The deterministic simulation is done using the same numerical scheme as Pearson (Pearson, 1993) in order to render the results comparable. This is a second-order finite-difference discretization of the Laplacian for the diffusion part and a first-order explicit Euler scheme with time-step size $\Delta t = 1.125$ for time stepping. The results show that as the number of molecules in the reactor increases with increasing u , the stochastic spatial pattern tends toward the deterministic one. The intrinsic noise in the stochastic system, however, breaks the symmetry of the pattern.

We also simulate the reaction-diffusion system in Eq. 6.26 in 3D (using triplet indices for the subvolumes) in a computational domain of volume $\Omega = 0.64^3$, divided into $K^3 = 64^3$ subvolumes of edge length $h = 0.01$. Again using periodic boundary conditions for the diffusion, the resulting reaction-diffusion system is:



where I_{all} is the set of all possible subvolume indices in 3D and (l', m', n') are the neighboring subvolumes of (l, m, n) taking into account the periodic boundary conditions, hence $l' \in \{[(l-1)\pm 1+2K] \bmod K+1\}$, $m' \in \{[(m-1)\pm 1+2K] \bmod K+1\}$, and $n' \in \{[(n-1)\pm 1+2K] \bmod K+1\}$. At $t = 0$, the initial population is:

$$\begin{aligned}
 n_1^{(l,m,n)} &= \begin{cases} \frac{uh^3}{2} + \lfloor 0.04(r-0.5)uh^3 + 0.5 \rfloor, & \text{for } 24 \leq l, m, n \leq 40 \\ uh^3, & \text{otherwise.} \end{cases} \\
 n_2^{(l,m,n)} &= \begin{cases} \frac{uh^3}{4} + \lfloor 0.02(r-0.5)uh^3 + 0.5 \rfloor, & \text{for } 24 \leq l, m, n \leq 40 \\ 0, & \text{otherwise,} \end{cases} \quad (6.29)
 \end{aligned}$$

where r is a uniform random number in $[0, 1]$ that acts as an initial perturbation. We use the same diffusion constants as in the 2D case above.

Figure 6.7 shows the 3D concentration distribution of species S_1 at time $t_f = 2000/(k_1 u^2)$, normalized with $u = 10^8$. For these parameters, the maximum number of molecules of species S_1 in any subvolume is $uh^3 = 100$ and hence the intrinsic noise breaks the symmetry of the Turing pattern. Approximately $36 \cdot 10^9$ reaction events are simulated until t_f with $f_r \approx 0.1$. The total runtime (CPU time) for PSRD is 77413 s, for NSM it is 100 636 s (extrapolated).

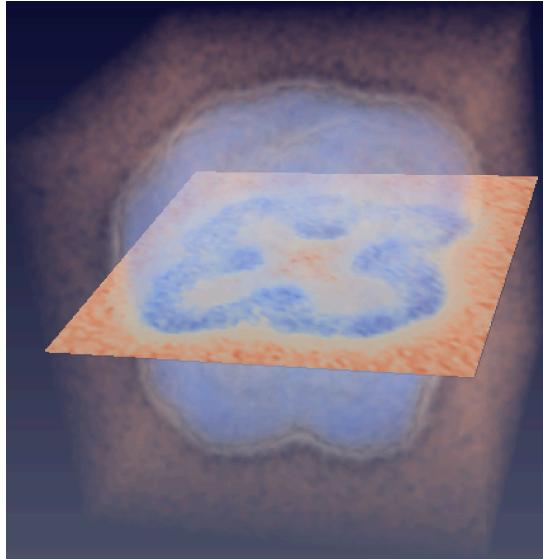


Figure 6.7: Normalized spatial concentration distribution of species S_1 in the three-dimensional Gray-Scott reaction-diffusion system (Eq. 6.28) for $F = 0.04$, $k = 0.06$, $k_1 = 1$, and $D_1 = 2D_2 = 2 \cdot 10^{-5}$ in a cubic computational domain of volume 0.64^3 , divided into $N_v = 64^3$ subvolumes of edge length $h = 0.01$. The concentration in each subvolume, normalized by $u = 10^8$, is shown as a color ranging from blue (concentration zero) to red (concentration one). The snapshot is taken at time $t = 2000/(k_1 u^2)$.

6.3 Conclusions and Summary

We have introduced the on-lattice partial-propensity stochastic reaction-diffusion (PSRD) method. PSRD proceeds by dividing the computational domain into N_v subvolumes. The chemical reaction system in each subvolume is assumed to be well mixed and it is imposed that molecules can only react with partners within the same subvolume. Diffusion is modeled by jump “reactions” between neighboring subvolumes. PSRD combines composition-rejection sampling (Devroye, 1986; Slepoy et al., 2008) (see Sections 2.3.7 and 3.4) with the concept of partial propensities (see Sec. 3.2). Computational efficiency is achieved by binning the subvolumes and using partial propensities to group the reactions within each subvolume.

PSRD samples trajectories from the exact solution of the reaction-diffusion master equation for on-lattice reaction-diffusion systems, provided the subvolume sizes are within admissible bounds (Erban and Chapman, 2009; Fange et al., 2010; Kuramoto and Yamada, 1976). This is done by first sampling the subvolume using composition-rejection sampling, and then sampling the index of the next reaction within that subvolume using linear search over the dynamically grouped partial propensities, analogous to the sorting partial-propensity direct method (SPDM) (see Sec. 3.3.2). The computational cost of PSRD to sample the next subvolume is $O(G_a)$, where the number of bins is $G_a = \log_2(a_{\max}/a_{\min}) + 1$, a_{\max} is the maximum total propensity in any subvolume, and a_{\min} is the smallest non-zero total propensity in any

subvolume. In any simulation, the number G_a scales at most as $O(\log_2 N_v)$. If the logarithmic span of the propensities can be *a priori* bounded by a constant, the cost of sampling the subvolume reduces to $O(1)$ (Slepoy et al., 2008). The computational cost to sample the index of the next reaction within a subvolume is $O(N)$, where N is the number of species in the reaction network. Thus, the overall computational cost of PSRD is $O(G_a + N)$, which is bounded in the worst case by $O(\log_2 N_v + N)$. This cost of PSRD is independent of whether the SRD simulation is dominated by “real” reactions or by “diffusion reactions”. We demonstrated this scaling of the computational cost using prototypical benchmark cases for both types of reaction networks: strongly coupled and weakly coupled.

PSRD inherits the limitations of partial-propensity methods. It is hence limited to reaction networks comprising only elementary reactions. For spatiotemporal reaction-diffusion simulations, however, including non-elementary reactions is of questionable value since no kinetic-theoretical framework exists for them (Fange et al., 2010; von Smoluchowski, 1917). It is hence unclear how the propensity functions of non-elementary reactions should be correctly formulated in a discretized space (Fange et al., 2010).

Due to the more complex data structures used in partial-propensity methods, we do not expect PSRD to offer significant speed-ups for small ($N \lesssim 10$) chemical reaction networks. In these cases, the next subvolume method (Elf and Ehrenberg, 2004; Hattne et al., 2005) can be as efficient or faster than PSRD. In addition, PSRD is restricted to chemical reaction networks that do not involve time delays. This could be overcome by using dPDM (delay PDM) (see Sec. 3.5.2) instead of SPDM inside each subvolume in PSRD. Our current software implementation of PSRD is moreover limited to rectangular computational domains. This limitation, however, is not inherent to the method as such and future developments will consider extending the method to computational domains of arbitrary shape (Sbalzarini et al., 2005, 2006), e.g., using unstructured meshes (Engblom et al., 2009).

While we have described the basic version of PSRD for simplicity and conciseness of the presentation, the algorithm can be further improved in efficiency using standard techniques. Using a binary tree search instead of linear search over subvolume bins (Gibson and Bruck, 2000), the computational cost of sampling the next subvolume can, for example, be reduced to $O(\log_2 G_a)$, rendering the overall computational cost of such a variant of PSRD $O(\log_2 G_a + N)$ and in the worst-case $O(\log_2 \log_2 N_v + N)$. Moreover, for weakly coupled reaction networks the computational cost of sampling the next reaction within a subvolume can be reduced to $O(G_r)$ using the partial-propensity method with composition-rejection sampling (PSSA-CR) within each subvolume. G_r is the logarithmic span of non-zero propensities within the subvolume. In summary, the computational cost of PSRD can be reduced to $O(\log_2 G_a + N)$ or even $O(\log_2 G_a + \log_2 G_r)$ for certain classes of reaction networks and when using a binary search tree also within PSSA-CR. These improvements can be realized at the expense of larger memory requirements, which is why we did not include them in the presentation here. Their implementation, however, is straightforward and they will be included in future versions of the PSRD software package.

PSRD uses dynamic bubble sort for the reactions within each subvolume. This is inspired by the sorting direct method (SDM) (McCollum et al., 2006) and its partial-propensity variant SPDM. Sorting SSAs have been shown to be particularly efficient on multi-scale (stiff) reaction networks where the propensities of different reactions are orders of magnitude apart. This means that a small fraction of reactions can potentially account for the majority of reaction

6 Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

events. The dynamic “bubbling up” of these reactions in the reaction list reduces the average search depth when sampling the next reaction as it accumulates the most frequent reactions at the top of the list. Using a sorting SSA inside each subvolume of an on-lattice SRD simulation is particularly advantageous since the propensities of different reaction types scale differently with subvolume size (see Eq. 6.2). While the propensities of bimolecular reactions scale as Ω_c^{-1} , those of source reactions scale as Ω_c , and the propensities of unimolecular reactions are independent of Ω_c . The propensities of “diffusion reactions” scale as h^{-2} . Reducing the grid spacing h thus renders the reaction network increasingly multi-scale with the propensity ratio between the fastest and slowest reactions scaling at most as h^6 in 3D subvolumes (h^4 in 2D subvolumes).

Taken together, PSRD offers an improved scaling of the computational cost for exact on-lattice SRD simulations. This can lead to significant performance improvements when simulating strongly coupled spatiotemporal processes, such as colloidal aggregation and scale-free biochemical networks (Strogatz, 2001; Albert and Barabási, 2002; Barabási and Oltvai, 2004; Albert, 2005).

A C++ implementation of PSRD is currently being implemented in pSSA simulation package (see Appendix A4).

7

Global parameter identification for stochastic reaction networks from single trajectories

"I have had my results for a long time; but I do not yet know how I am to arrive at them."

– *Carl F. Gauss*

We provide an example of an application that involves stochastic simulations of chemical kinetics as a forward model in solving an inverse problem. Specifically, we consider the problem of parameter identification in systems biology network models (Engl et al., 2009). This demonstrates a potential application of partial-propensity methods and of the fluctuation-relaxation kinetics in mesoscopic chemical systems.

Systems biology implies a holistic research paradigm, complementing the reductionist approach to biological organization (Kitano, 2002b,a). This frequently has the goal of mechanistically understanding the function of biological entities and processes in interaction with the other entities and processes they are linked to or communicate with. A formalism to express these links and connections is provided by network models of biological processes (Barabási and Oltvai, 2004; Albert, 2005). Using concepts from graph theory (Mason and Verwoerd, 2007) and dynamic systems theory (Wolkenhauer, 2001), the organization, dynamics, and plasticity of these networks can then be studied.

Systems biology models of molecular reaction networks contain a number of parameters. These are the rate constants of the involved reactions and, if spatiotemporal processes are

considered, the transport rates, e.g. diffusion constants, of the chemical species. In order for the models to be predictive, these parameters need to be inferred. The process of inferring them from experimental data is called *parameter identification*. If in addition also the network structure is to be inferred from data, the problem is called *systems identification*. Here, we consider the problem of identifying the parameters of a biochemical reaction network from a single, noisy measurement of the concentration time-course of some of the involved species. While this time series can be long, ensemble replicas are not possible, either because the measurements are destructive or one is interested in variations between different specimens or cells. This is particularly important in *molecular systems biology*, where cell-cell variations are of interest or large numbers of experimental replica are otherwise not feasible (Snijder et al., 2009; Zechner et al., 2011, submitted).

This problem is particularly challenging and traditional genomic and proteomic techniques do not provide single-cell resolution. Moreover, in individual cells the molecules and chemical reactions can only be observed indirectly. Frequently, fluorescence microscopy is used to observe biochemical processes in single cells. Fluorescently tagging some of the species in the network of interest allows measuring the spatiotemporal evolution of their concentrations from video microscopy and fluorescence photometry. In addition, fluorescence correlation spectroscopy (FCS) allows measuring fluctuation time-courses of molecule numbers (Lakowicz, 2006).

Using only a single trajectory of the mean concentrations would hardly allow identification of network parameters. There could be several combinations of network parameters that lead to the same mean dynamics. A stochastic network model, however, additionally provides information about the fluctuations of the molecular abundances. The hope is that there is then only a small region of parameter space that produces the correct behavior of the mean *and* the correct spectrum of fluctuations (Munsky et al., 2009). Experimentally, fluctuation spectra can be measured at single-cell resolution using FCS.

The stochastic behavior of biochemical reaction networks can be due to low copy numbers of the reacting molecules (Grima, 2009b). In addition, biochemical networks may exhibit stochasticity due to extrinsic noise (see Chapter 4 for an example). This can persist even at the continuum scale, leading to continuous-stochastic models. Extrinsic noise can, e.g., arise from environmental variations or variations in how the reactants are delivered into the system. Also measurement uncertainties can be accounted for in the model as extrinsic noise, modeling our inability to precisely quantify the experimental observables.

We model stochastic chemical kinetics using the chemical master equation (CME). Using a CME forward model in biological parameter identification amounts to tracking the evolution of a probability distribution, rather than just a single value. This prohibits predicting the state of the system and only allows statements about the probability for the system to be in a certain state, hence requiring sampling-based parameter identification methods. In the stochastic-discrete context, a number of different approaches have been suggested. Boys *et al.* proposed a fully Bayesian approach for parameter estimation using an explicit likelihood for data/model comparison and a Markov Chain Monte Carlo (MCMC) scheme for sampling (Boys et al., 2008). Zechner *et al.* developed a recursive Bayesian estimation technique (Zechner et al., 2011,,) to cope with cell-cell variability in experimental ensembles. Toni and co-workers used an approximate Bayesian computation (ABC) ansatz, as introduced by Marjoram and co-workers (Marjoram et al., 2003), that does not require an explicit likelihood (Toni et al.,

2009). Instead, sampling is done in a sequential Monte Carlo (or particle filter) framework. Reinker *et al.* used a hidden Markov model where the hidden states are the actual molecule abundances, and state transitions model chemical reactions (Reinker *et al.*, 2006). Inspired by Prediction Error Methods (Ljung, 2002), Cinquemani *et al.* identified the parameters of a hybrid deterministic–stochastic model of gene expression from multiple experimental time courses (Cinquemani *et al.*, 2008). A variational approach for stochastic two-state systems has been proposed by Stock and co-workers based on Maximum Caliber (Stock *et al.*, 2008), an extension of Jaynes’ Maximum Entropy principle (Jaynes, 1957) to nonequilibrium systems. Randomized optimization algorithms have been used, e.g., by Koutroumpas *et al.* who applied a Genetic Algorithm to a hybrid deterministic–stochastic network model (Koutroumpas *et al.*, 2008). More recently, Poovathingal and Gunawan used another global optimization heuristic, the Differential Evolution algorithm (Poovathingal and Gunawan, 2010). Gazzola *et al.* used the evolution strategy with covariance matrix adaptation (CMA-ES) (Hansen and Ostermeier, 2001; Hansen *et al.*, 2003) to estimate the parameters of a linear reaction network modeling stochastic microtubule dynamics (Gazzola *et al.*, 2009). CMA-ES has also been used for parameter identification in a variety of deterministic systems (Kern and Koumoutsakos, 2006; Hansen *et al.*, 2009; Gazzola *et al.*, 2011).

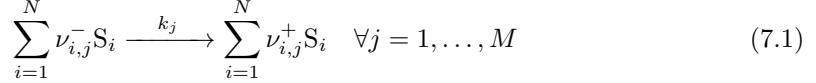
If estimates are to be made based on a single trajectory, the stochasticity of the measurements and of the model leads to very noisy similarity measures, requiring optimization and sampling schemes that are robust against noise in the data. Here, we propose a novel combination of exact stochastic simulations for a CME forward model and an adaptive Monte Carlo sampling technique, called Gaussian Adaptation, to address the single-trajectory parameter estimation problem for monostable stochastic biochemical reaction networks. Evaluations of the CME model are done using exact partial-propensity stochastic simulation algorithms (see Chapter 3). Parameter optimization uses Gaussian Adaptation (Müller and Sbalzarini, 2010b,a). The method iteratively samples model parameters from a multivariate normal distribution and evaluates a suitable objective function that measures the distance between the dynamics of the forward model output and the experimental measurements. In addition to estimates of the kinetic parameters in the network, the present method also provides an ellipsoidal volume estimate of the viable part of parameter space and is able to estimate the physical volume of the compartment in which nonlinear reactions take place.

We assume that quantitative experimental time series of either a transient or the steady state of the concentrations of some of the molecular species in the network are available. This can, for example, be obtained from single-cell fluorescence microscopy by translating fluorescence intensities to estimated chemical concentrations. Accurate methods that account for the microscope’s point-spread function and the camera noise model are available to this end (Helmuth *et al.*, 2009; Helmuth and Sbalzarini, 2009; Cardinale *et al.*, 2009). Additionally, FCS spectra can be analyzed in order to quantify the fluctuation relaxation kinetics of molecular populations (Lakowicz, 2006; Qian and Elson, 2004). The present approach requires only a *single* stochastic trajectory from each cell. Since the forward model is stochastic and only a single experimental trajectory is used, the objective function needs to robustly measure closeness between the experimental and the simulated trajectories. We review previously considered measures and present a new distance function in Sec. 7.4. First, however, we set out the formal problem description below. We then describe Gaussian Adaptation and its applicability to the current estimation task. The evaluation of the forward model is outlined

in Sec. 7.3. We consider a linear cyclic chain and a nonlinear colloidal aggregation model as benchmark test cases in Sec. 7.5 and conclude in Sec. 7.6.

7.1 Problem statement

We consider a network model of a biochemical system given by M coupled chemical reactions



between N species, where $\boldsymbol{\nu}^- = [\nu_{i,j}^-]$ and $\boldsymbol{\nu}^+ = [\nu_{i,j}^+]$ are the stoichiometry matrices of the reactants and products, respectively, and S_i is the i^{th} species in the reaction network. Let n_i be the population (molecular copy number) of species S_i . The reactions occur in a physical volume Ω and the macroscopic reaction rate of reaction j is k_j . This defines a dynamic system with integer-valued state $\mathbf{n}(t) = [n_i(t)]$ and $M + 1$ parameters $\boldsymbol{\theta} = [k_1, \dots, k_M, \Omega]$.

The problem considered here can then be formalized as follows: Given a forward model $\mathcal{M}(\boldsymbol{\theta})$ and a single noisy trajectory of the population of the chemical species $\hat{\mathbf{n}}(t_0 + (q-1)\Delta t_{\text{exp}})$ at K discrete time points $t = t_0 + (q-1)\Delta t_{\text{exp}}, q = 1, \dots, K$, we wish to infer $\boldsymbol{\theta} = [k_1, \dots, k_M, \Omega]$. The time between two consecutive measurements Δt_{exp} and the number of measurements K are given by the experimental technique used. As a forward model $\mathcal{M}(\boldsymbol{\theta})$ we use the CME as given in Eq. 1.19 and sample trajectories from it using the partial-propensity formulation of Gillespie's exact SSA as described in Sec. 7.3.

7.2 Gaussian Adaptation (GaA)

Gaussian Adaptation (GaA), introduced in the late 1960's by Gregor Kjellström (Kjellström, 1969; Kjellström and Taxen, 1981), is a Monte Carlo technique that has originally been developed to solve design-centering and optimization problems in analog electric circuit design. Design centering solves the problem of determining the nominal values (resistances, capacitances, etc.) of the components of a circuit such that the circuit output is within specified design bounds and is maximally robust against random variations in the circuit components with respect to a suitable criterion or objective function. This problem is a superset of general optimization, where one is interested in finding a parameter vector that minimizes (or maximizes) an objective function without any additional robustness criterion. GaA has been specifically designed for scenarios where the objective function $f(\boldsymbol{\theta})$ is only available in a black-box (or oracle) model that is defined on a real-valued domain $\mathcal{A} \subseteq \mathbb{R}^n$ and returns real-valued output. The black-box model assumes that gradients or higher-order derivatives of the objective function may not exist or may not be available, hence including the class of discontinuous and noisy functions. The specific objective function used here is presented in Sec. 7.4.

The principle idea behind GaA is the following: Starting from a user-defined point in parameter space, GaA explores the space by iteratively sampling single parameter vectors from a multivariate Gaussian distribution $\mathcal{N}(\mathbf{m}, \boldsymbol{\Sigma})$ whose mean $\mathbf{m} \in \mathbb{R}^n$ and covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$ are dynamically adapted based on the information from previously accepted samples. The acceptance criterion depends on the specific mode of operation, i.e., whether GaA is

used as an optimizer or as a sampler (Müller and Sbalzarini, 2010a; Müller, 2010). Adaptation is performed such as to maximize the entropy of the search distribution under the constraint that acceptable search points are found with a predefined, fixed hitting (success) probability $p < 1$ (Kjellström and Taxen, 1981). Using the definition of the entropy of a multivariate Gaussian distribution $\mathcal{H}(\mathcal{N}) = \log \left(\sqrt{(2\pi e)^n \det(\boldsymbol{\Sigma})} \right)$ shows that this is equivalent to maximizing the determinant of the covariance matrix $\boldsymbol{\Sigma}$. GaA thus follows Jaynes' Maximum Entropy principle (Jaynes, 1957).

GaA starts by setting the mean $\mathbf{m}^{(0)}$ of the multivariate Gaussian to an initial acceptable point $\boldsymbol{\theta}^{(0)}$ and the Cholesky factor $\mathbf{Q}^{(0)}$ of the covariance matrix to the identity matrix \mathbf{I} . At each iteration $g > 0$, the covariance $\boldsymbol{\Sigma}^{(g)}$ is decomposed as: $\boldsymbol{\Sigma}^{(g)} = (r \cdot \mathbf{Q}^{(g)}) (r \cdot \mathbf{Q}^{(g)})^T = r^2 (\mathbf{Q}^{(g)}) (\mathbf{Q}^{(g)})^T$, where r is the scalar step size that controls the scale of the search. The matrix $\mathbf{Q}^{(g)}$ is the normalized square root of $\boldsymbol{\Sigma}^{(g)}$, found by eigen- or Cholesky decomposition of $\boldsymbol{\Sigma}^{(g)}$. The candidate parameter vector in iteration $g + 1$ is sampled from a multivariate Gaussian according to $\boldsymbol{\theta}^{(g+1)} = \mathbf{m}^{(g)} + r^{(g)} \mathbf{Q}^{(g)} \boldsymbol{\eta}^{(g)}$, where $\boldsymbol{\eta}^{(g)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The parameter vector is then evaluated by the objective function $f(\boldsymbol{\theta}^{(g+1)})$.

Only if the parameter vector is accepted, the following adaptation rules are applied: The step size r is increased as $r^{(g+1)} = f_e \cdot r^{(g)}$, where $f_e > 1$ is termed the *expansion factor*. The mean of the proposal distribution is updated as

$$\mathbf{m}^{(g+1)} = \left(1 - \frac{1}{N_m} \right) \mathbf{m}^{(g)} + \frac{1}{N_m} \boldsymbol{\theta}^{(g+1)}. \quad (7.2)$$

N_m is a weighting factor that controls the learning rate of the method. The successful search direction $\mathbf{d}^{(g+1)} = (\boldsymbol{\theta}^{(g+1)} - \mathbf{m}^{(g)})$ is used to perform a rank-one update of the covariance matrix: $\boldsymbol{\Sigma}^{(g+1)} = \left(1 - \frac{1}{N_C} \right) \boldsymbol{\Sigma}^{(g)} + \frac{1}{N_C} \mathbf{d}^{(g+1)} \mathbf{d}^{(g+1)T}$. N_C weights the influence of the accepted parameter vector on the covariance matrix. In order to decouple the volume of the covariance (controlled by $r^{(g+1)}$) from its orientation, $\mathbf{Q}^{(g+1)}$ is normalized such that $\det(\mathbf{Q}^{(g+1)}) = 1$.

In case $\boldsymbol{\theta}^{(g+1)}$ is not accepted at the current iteration, only the step size is adapted as $r^{(g+1)} = f_c \cdot r^{(g)}$, where $f_c < 1$ is the *contraction factor*.

The behavior of GaA is controlled by several strategy parameters. Kjellström analyzed the information-theoretic optimality of the acceptance probability p for GaA in general regions (Kjellström and Taxen, 1981). He concluded that the efficiency E of the process and p are related as $E \propto -p \log p$, leading to an optimal $p = \frac{1}{e} \approx 0.3679$, where e is Euler's number. A proof is provided in (Kjellström, 1991). Maintaining this optimal hitting probability corresponds to leaving the volume of the distribution, measured by $\det(\boldsymbol{\Sigma})$, constant under stationary conditions. Since $\det(\boldsymbol{\Sigma}) = r^{2n} \det(\mathbf{Q} \mathbf{Q}^T)$, the expansion and contraction factors f_e and f_c expand or contract the volume by a factor of f_e^{2n} and f_c^{2n} , respectively. After S accepted and F rejected samples, a necessary condition for constant volume thus is: $\prod_{i=1}^S (f_e)^{2n} \prod_{i=1}^F (f_c)^{2n} = 1$. Using $p = \frac{S}{S+F}$, and introducing a small $\beta > 0$, the choice $f_e = 1 + \beta(1 - p)$ and $f_c = 1 - \beta p$ satisfies the constant-volume condition to first order. The scalar rate β is coupled to N_C . N_C influences the update of $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$, which contains n^2 entries. Hence, N_C should be related to n^2 . We suggested using $N_C = (n+1)^2 / \log(n+1)$ as a standard value, and coupling $\beta = \frac{1}{N_C}$ (Müller and Sbalzarini, 2010b). A similar reasoning is also applied to N_m . Since N_m influences the update of $\mathbf{m} \in \mathbb{R}^n$, it is reasonable to set

$N_m \propto n$. We propose $N_m = en$ as a standard value.

Depending on the specific acceptance rule used, GaA can be turned into a global optimizer (Müller and Sbalzarini, 2010b), an adaptive MCMC sampler (Müller and Sbalzarini, 2010a; Müller, 2010), or a volume estimation method (Müller and Sbalzarini, 2011), as described next.

7.2.1 GaA for global black-box optimization

In a minimization scenario, GaA uses an adaptive-threshold acceptance mechanism. Given an initial scalar cutoff threshold $c_T^{(0)}$, we accept a parameter vector $\theta^{(g+1)}$ at iteration $g+1$ if $f(\theta^{(g+1)}) < c_T^{(g)}$. Upon acceptance, the threshold c_T is lowered as $c_T^{(g+1)} = \left(1 - \frac{1}{N_T}\right)c_T^{(g)} + \frac{1}{N_T}f(\theta^{(g+1)})$, where N_T controls the weighting between the old threshold and the objective-function value of the *accepted* sample. This sample-dependent threshold update renders the algorithm invariant to linear transformations of the objective function. The standard strategy parameter value is $N_T = en$ (Müller and Sbalzarini, 2010a). We refer to (Müller and Sbalzarini, 2010a) for further information about convergence criteria and constraint handling techniques in GaA.

7.2.2 GaA for approximate Bayesian computation and viable volume estimation

Replacing the threshold acceptance-criterion by a probabilistic Metropolis criterion, and setting $N_m = 1$, turns GaA into an adaptive MCMC sampler with global adaptive scaling (Andrieu and Thoms, 2008). We termed this method *Metropolis-GaA* (Müller and Sbalzarini, 2010a; Müller, 2010). Its strength is that GaA can automatically adapt to the covariance of the target probability distribution while maintaining the fixed hitting probability. For standard MCMC, this cannot be achieved without fine-tuning the proposal using multiple MCMC runs. We hypothesize that GaA might also be an effective tool for approximate Bayesian computation (ABC) (Toni et al., 2009). In essence, the ABC ansatz is MCMC without an explicit likelihood function (Marjoram et al., 2003). The likelihood is replaced by a distance function — which plays the same role as our objective function — that measures closeness between a parameterized model simulation and empirical data \mathcal{D} , or summary statistics thereof. When a uniform prior over the parameters and a symmetric proposal are assumed, a parameter vector in ABC is unconditionally accepted if its corresponding distance function value $f(\theta^{(g+1)}) < c_T$ (Marjoram et al., 2003). The threshold c_T is a problem-dependent constant that is fixed prior to the actual computation. Marjoram and co-workers have shown that samples obtained in this manner are approximately drawn from the posterior parameter distribution given the data \mathcal{D} . While Pritchard *et al.* used a simple rejection sampler (Pritchard et al., 1999), Marjoram and co-workers proposed a standard MCMC scheme (Marjoram et al., 2003). Toni and co-workers used sequential MC for sample generation (Toni et al., 2009). To the best of our knowledge, however, the present work presents the first application of an adaptive MCMC scheme for ABC in biochemical network parameter inference. Finally, we emphasize that when GaA's mean, covariance matrix, and hitting probability p stabilize during ABC, they provide direct access to an ellipsoidal estimation of the volume of the viable parameter space as defined by

the threshold c_T (Müller and Sbalzarini, 2011). Hafner and co-workers have shown how to use such viable volume estimates for model discrimination (Hafner et al., 2009).

7.3 Evaluation of the forward model

In each iteration of the GaA algorithm, the forward model of the network needs to be evaluated for the proposed parameter vector $\boldsymbol{\theta}$. This requires an efficient and exact SSA for the chemical kinetics of the reaction network, used to generate trajectories $\mathbf{n}(t)$ from $\mathcal{M}(\boldsymbol{\theta})$. Since GaA could well propose parameter vectors that lead to low copy numbers for some species, it is important that the SSA be exact since approximate algorithms are not appropriate at low copy number.

In its original formulation, Gillespie's SSA has a computational cost that is linearly proportional to the total number M of reactions in the network (see Sec. 2.3). If many model evaluations are required, as in the present application, this computational cost quickly becomes prohibitive. Scale-free networks as seem to be characteristic for systems biology models (Albert, 2005; Strogatz, 2001) are by definition strongly coupled. This is due to the existence of *hubs* that have a higher connection probability than other nodes. These hubs frequently correspond to chemical reactions that produce or consume species that also participate in the majority of the other reactions, such as water, ATP, or CO₂ in metabolic networks. We hence use partial-propensity methods (see Chapter 3) to sample trajectories according to the exact solution of the CME.

7.4 Objective function

In the context of parameter identification of stochastic biochemical networks, a number of distance or objective functions have previously been suggested. Reinker *et al.* proposed an approximate maximum-likelihood measure under the assumption that only a small number of reactions fire between two experimental measurement points, and a likelihood based on singular value decomposition that works when many reactions occur per time interval (Reinker et al., 2006). Koutroumpas *et al.* compared objective functions based on least squares, normalized cross-correlations, and conditional probabilities using a Genetic Algorithm (Koutroumpas et al., 2008). Koepll and co-workers proposed the Kantorovich distance to compare experimental and model-based probability distributions (Koepll et al., 2010). Alternative distance measures include the Earth Mover's distance or the Kolomogorov-Smirnov distance (Poothathingal and Gunawan, 2010). These distance measures, however, can only be used when many experimental trajectories are available. In order to measure the distance between a *single* experimental trajectory $\hat{\mathbf{n}}(t)$ and a *single* model output $\mathbf{n}(t)$, we propose a novel cost function $f(\boldsymbol{\theta}) = f(\mathcal{M}(\boldsymbol{\theta}), \hat{\mathbf{n}})$ that reasonably captures the kinetics of a monostable system. We define a compound objective function $f(\boldsymbol{\theta}) = f_1(\boldsymbol{\theta}) + f_2(\boldsymbol{\theta})$ with

$$f_1(\boldsymbol{\theta}) = \sum_{i=1}^4 \gamma_i , \quad f_2(\boldsymbol{\theta}) = \sum_{i=1}^N \frac{\sum_{l=0}^{l^\times} |\text{ACF}_l(\hat{\mathbf{n}}_i) - \text{ACF}_l(\mathbf{n}_i)|}{\sum_{l=0}^{l^\times} \text{ACF}_l(\hat{\mathbf{n}}_i)} , \quad (7.3)$$

where

$$\gamma_i = \sum_{j=1}^N \sqrt{\left(\frac{\mu_i(n_j) - \mu_i(\hat{n}_j)}{\mu_i(\hat{n}_j)} \right)^2} \quad (7.4)$$

with the central moments given by

$$\mu_i(n_j) = \begin{cases} \sum_{p=1}^K n_j(t_0 + (p-1)\Delta t_{\text{exp}}) & \text{if } i = 1 \\ \left(\left| \sum_{q=1}^K (n_j(t_0 + (q-1)\Delta t_{\text{exp}}) - \mu_1(n_j))^q \right|^i \right)^{1/i} & \text{otherwise} \end{cases} \quad (7.5)$$

and the time autocorrelation function (ACF) at lag l given by

$$\text{ACF}_l(n_i) = \frac{n_i(t_0)n_i(t_0 + l \Delta t_{\text{exp}}) - (\mu_1(n_i))^2}{\mu_2(n_i)}.$$

The variable l^\times is the lag at which the experimental ACF crosses 0 for the first time. The function $f_1(\boldsymbol{\theta})$ measures the difference between the first four moments of \mathbf{n} and $\hat{\mathbf{n}}$. This function alone would, however, not be enough to capture the kinetics since it lacks information about correlations in time. This is taken into account by $f_2(\boldsymbol{\theta})$, measuring the difference in the lifetimes of all chemical species. These lifetimes are systematically modulated by the volume Ω (see Chapter 5), hence enabling volumetric measurements of reaction compartments along with the identification of the rate constants.

The present objective function allows inclusion of experimental readouts from image-based systems biology. The moment-matching part is a typical readout from fluorescence photometry, whereas the autocorrelation of the fluctuations can directly be measured using, e.g., FCS.

7.5 Results

We estimate the unknown parameters $\boldsymbol{\theta}$ for two reaction networks: a weakly coupled cyclic chain and a strongly coupled nonlinear colloidal aggregation network. For the cyclic chain we estimate $\boldsymbol{\theta}$ at steady state. For the aggregation model we estimate $\boldsymbol{\theta}$ both at steady state and in the transient phase. Every kinetic parameter is allowed to vary in the interval $[10^{-3}, 10^3]$ and the reaction volume Ω in $[1, 500]$. Each GaA run starts from a point selected uniformly at random in logarithmic parameter space.

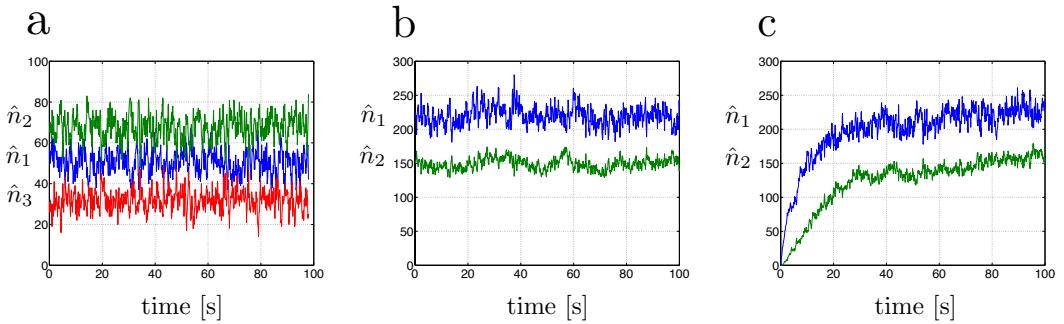
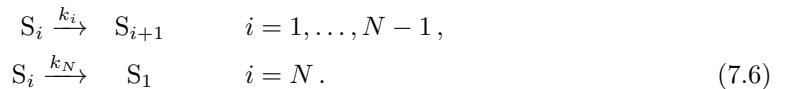


Figure 7.1: *In silico* data for all test cases. **a.** Time evolution of the populations of three species in the cyclic chain model at steady state (starting at $t_0 = 2000$). **b.** Time evolution of the populations of two species in the aggregation model at steady state (starting at $t_0 = 5000$) for $\Omega = 15$. **c.** Same as **b.**, but during the transient phase (starting at $t_0 = 0$).

7.5.1 Weakly coupled reaction network: cyclic chain

The cyclic chain network is given by:



In this linear network, the number of reactions M is equal to the number of species N . The maximum degree of coupling of this reaction network is 2, irrespective of the size of the system (length of the chain), rendering it weakly coupled. We hence use PSSA-CR to evaluate the forward model with a computational complexity of $O(1)$. In the present test case, we limit ourselves to 3 species and 3 reactions, i.e., $N = M = 3$. The parameter vector for this case is given by $\boldsymbol{\theta} = [k_1, k_2, k_3]$, since the population kinetics of linear reactions is independent of the volume Ω since the specific probability rates of unimolecular reactions is equal to the macroscopic reaction rate (see Eq. 1.3).

We simulate steady-state “experimental” data $\hat{\mathbf{n}}$ using PSSA-CR with ground truth $k_1 = 2$, $k_2 = 1.5$, $k_3 = 3.2$ (see Fig. 7.1a). We set the initial population of the species to $n_1(t = 0) = 50$, $n_2(t = 0) = 50$, and $n_3(t = 0) = 50$ and sample a single CME trajectory at equi-spaced time points with $\Delta t_{\text{exp}} = 0.1$ between $t = t_0$ and $t = t_0 + (K-1)\Delta t_{\text{exp}}$ with $t_0 = 2000$ and $K = 1001$ for each of the 3 species S_1 , S_2 , and S_3 . For the generated data we find $l^\times = 7$.

We generate trajectories from the forward model for every parameter vector $\boldsymbol{\theta}$ proposed by GaA using PSSA-CR between $t = 0$ and $t = (K-1)\Delta t_{\text{exp}} = 100$, starting from the initial population $n_i(t = 0) = \hat{n}_i(t = t_0)$.

Before turning to the actual parameter identification, we illustrate the topography of the objective function landscape for the present example. We fix $k_3 = 3.2$ to its optimal value and perform a two-dimensional grid sampling for k_1 and k_2 over the full search domain. We use 40 logarithmically spaced sample points per parameter, resulting in 40^2 parameter combinations. For each combination we evaluate the objective function. The resulting landscapes of $f_1(\boldsymbol{\theta})$, $f_2(\boldsymbol{\theta})$, and $f(\boldsymbol{\theta})$ are depicted in Fig. 7.2a. Figure 7.2b shows refined versions around the

global optimum. We see that the moment-matching term $f_1(\boldsymbol{\theta})$ is largely responsible for the

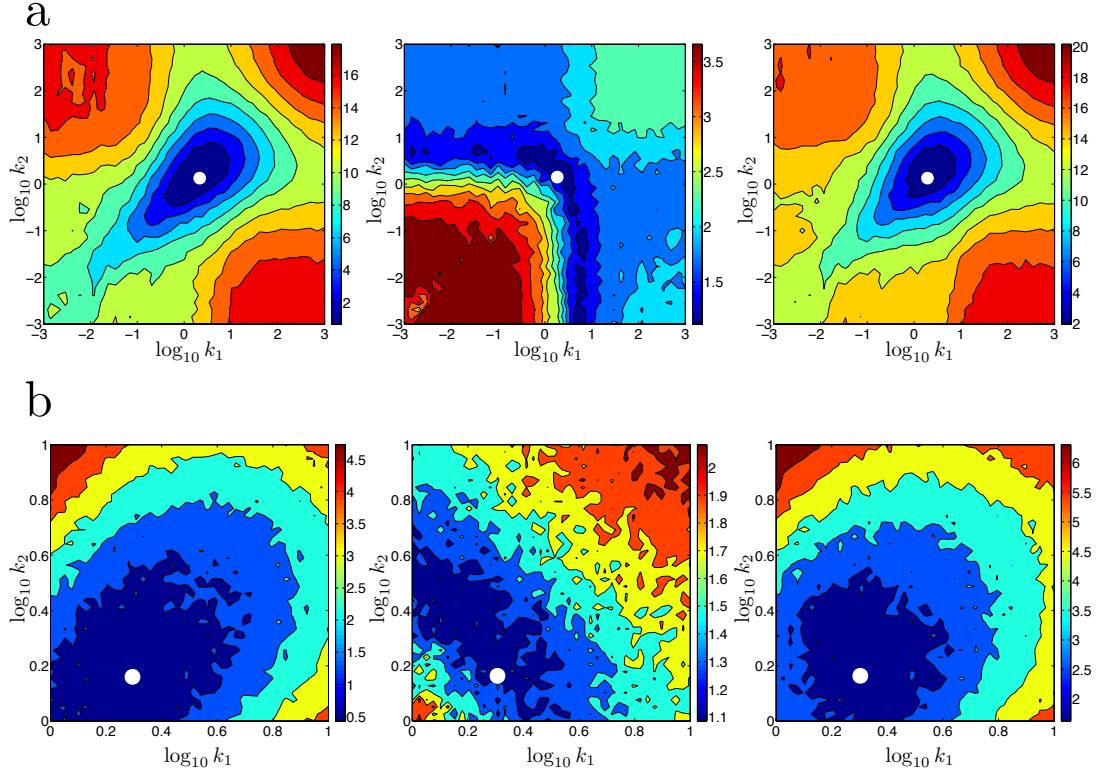


Figure 7.2: **a.** Global objective function landscape for the cyclic chain over the complete search domain for optimal $k_3 = 3.2$. The three panels from left to right show $f_1(\boldsymbol{\theta})$, $f_2(\boldsymbol{\theta})$, and $f(\boldsymbol{\theta})$, respectively. **b.** A refined view of the objective function landscape near the global optimum. The three panels from left to right show $f_1(\boldsymbol{\theta})$, $f_2(\boldsymbol{\theta})$, and $f(\boldsymbol{\theta})$, respectively. The white dots indicate the ground truth parameters.

global single-funnel topology of the landscape. The autocorrelation term $f_2(\boldsymbol{\theta})$ sharpens the objective function near the global optimum and renders it locally more isotropic.

We perform both global optimization and ABC runs using GaA. In each of the 15 independent optimization runs, the number of objective function evaluations (FES) is limited to MAX_FES = 1000M = 3000. We set the initial step size to $r^{(0)} = 1$ and perform all searches in logarithmic scale of the parameters. Independent restarts from uniformly random points are performed when the step size r drops below 10^{-4} (Müller and Sbalzarini, 2010b). For each of the 15 independent runs, the 30 parameter vectors with the smallest objective function values are collected and displayed in the box plot shown in the left panel of Fig. 7.3a. All 450 collected parameter vectors have objective function values smaller than 1.6. These results suggest that the present method is able to accurately determine the correct scale of the kinetic

parameters from a single experimental trajectory, although an overestimation of the rates is apparent.

We use the obtained optimization results for subsequent ABC runs. We conduct 15 independent ABC runs using $c_T = 2$. The starting points for the ABC runs are selected uniformly at random from the 450 collected parameter vectors in order to ensure stable initialization. For each run, we again set MAX_FES= 1000M = 3000. The initial step size $r^{(0)}$ is set to 0.1, and the parameters are again explored in logarithmic scale. For all runs we observe rapid convergence of the empirical hitting probability p_{emp} to the optimal $p = \frac{1}{e}$ (see Sec. 7.2). We collect the ABC samples along with the means and covariances of GaA as soon as $|p_{\text{emp}} - p| < 0.05$. As an example we show the histograms of the posterior samples for a randomly selected run in Fig. 7.3b. The means of the posterior distributions are again larger than the true kinetic parameters. Using GaA's means, covariance matrices, and the corresponding hitting probabilities that generated the posterior samples, we can construct an ellipsoidal volume estimation (Müller and Sbalzarini, 2011). This is done by multiplying each eigenvalue of the average of the collected covariance matrices with $c_{p_{\text{emp}}} = \text{inv } \chi_n^2(p_{\text{emp}})$, the n -dimensional inverse Chi-square distribution evaluated at the empirical hitting probability. The product of these scaled eigenvalues and the volume of the n -dimensional unit sphere, $|S(n)| = \frac{\pi^{\frac{n}{2}}}{\Gamma(\frac{n}{2}+1)}$, then yields the ellipsoid volume with respect to a uniform distribution (see (Müller and Sbalzarini, 2011) for details). The resulting ellipsoid contains the optimal kinetic parameter vector and is depicted in the right panel of Fig. 7.3a. It has a volume of 0.045 in log-parameter space. This constitutes only 0.0208% of the initial search space volume, indicating that GaA significantly narrows down the viable parameter space around the true optimal parameters, despite the noise in the forward model.

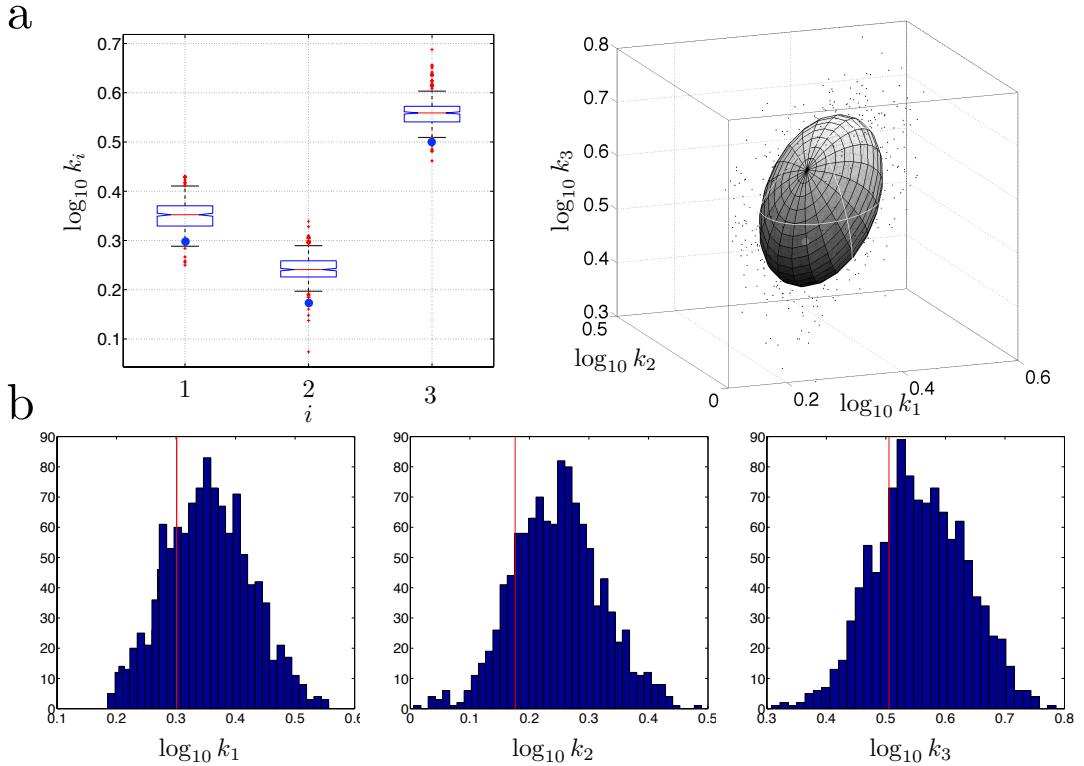
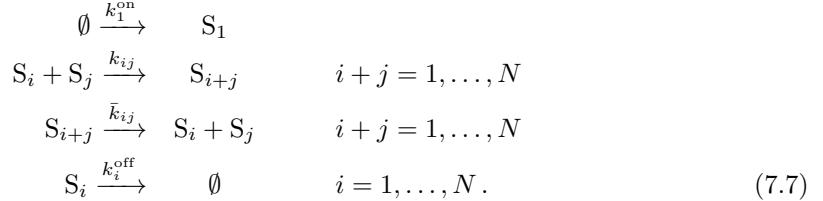


Figure 7.3: **a.** Left panel: Box plot of the 30 best parameter vectors from each of the 15 independent optimization runs. The blue dots indicate the true parameter values. Right panel: Ellipsoidal volume estimate of the parameter space below an objective-function threshold $c_T = 2$ from a single ABC run. **b.** Empirical posterior distributions of the kinetic parameters from the same single ABC run with $c_T = 2$. The red lines indicate the true parameters.

7.5.2 Strongly coupled reaction network: colloidal aggregation

The colloidal aggregation network is given by:



For this network of N species, the number of reactions is $M = \left\lfloor \frac{N^2}{2} \right\rfloor + N + 1$. The maximum degree of coupling of this reaction network is proportional to N , rendering the network strongly coupled. We hence use SPDM to evaluate the forward model with a computational complexity of $O(N)$. We use SPDM instead of PDM since the search path of GaA is unpredictable and could well generate parameters that lead to multi-scale networks. For this test case, we limit ourselves to two species, i.e., $N = 2$ and $M = 5$. The parameter vector for this case is $\theta = [k_{11}, \bar{k}_{11}, k_1^{\text{on}}, k_1^{\text{off}}, k_2^{\text{off}}, \Omega]$.

We perform GaA global optimization runs following the same protocol as for the cyclic chain network with MAX_FES = 1000($M + 1$) = 6000.

7.5.2.1 At steady state

We simulate “experimental” data \hat{n} using SPDM with ground truth $k_{11} = 0.1$, $\bar{k}_{11} = 1.0$, $k_1^{\text{on}} = 2.1$, $k_1^{\text{off}} = 0.01$, $k_2^{\text{off}} = 0.1$, and $\Omega = 15$ (see Fig. 7.1b). We set the initial population of the species to $n_1(t = 0) = 0$, $n_2(t = 0) = 0$, and $n_3(t = 0) = 0$ and sample $K = 1001$ equi-spaced data points between $t = t_0$ and $t = t_0 + (K - 1)\Delta t_{\text{exp}}$ with $t_0 = 5000$ and $\Delta t_{\text{exp}} = 0.1$.

We generate trajectories from the forward model for every parameter vector θ proposed by GaA using SPDM between $t = 0$ and $t = (K - 1)\Delta t_{\text{exp}} = 100$, starting from the initial population $n_i(t = 0) = \hat{n}_i(t = t_0)$.

The optimization results are summarized in the left panel of Fig. 7.4a. For each of 15 independent runs, the 30 lowest-objective parameter vectors are collected and shown in the box plot. We observe that the true parameters corresponding to $\theta_2 = \bar{k}_{11}$, $\theta_3 = k_1^{\text{on}}$, $\theta_4 = k_1^{\text{off}}$, and $\theta_5 = k_2^{\text{off}}$ are between the 25th and 75th percentiles of the identified parameters. Both the first parameter and the reaction volume are, on average, overestimated. Upon rescaling the kinetic rate constants with the estimated volume, we find $\theta^{\text{norm}} = [\theta_1/\theta_6, \theta_2, \theta_3 \theta_6, \theta_4, \theta_5]$, which are the specific probability rates of the reactions. The identified values are shown in the right panel of Fig. 7.4a. The median of the identified θ_3^{norm} coincides with the true specific probability rate. Likewise, θ_1^{norm} is closer to the 25th percentile of the parameter distribution. This suggests a better estimation performance of GaA in the space of specific probability rates, at the expense of not obtaining an estimate for the reactor volume.

7.5.2.2 In the transient phase

We simulate “experimental” data in the transient phase of the network dynamics using the same parameters as above between $t = t_0$ and $t = (K - 1)\Delta t_{\text{exp}}$ with $t_0 = 0$, $\Delta t_{\text{exp}} = 0.1$, and $K = 1001$ (see Fig. 7.1c). We evaluate the forward model with $n_i(t = 0) = \hat{n}_i(t = t_0)$ to obtain trajectories between $t = 0$ and $t = (K - 1)\Delta t_{\text{exp}}$ for every proposed parameter vector θ from GaA.

The optimization results for the transient case are summarized in Fig. 7.4b. We observe that the true parameters corresponding to $\theta_3 = k_1^{\text{on}}$, $\theta_5 = k_2^{\text{off}}$, and $\theta_6 = \Omega$ are between the 25th and 75th percentiles of the identified parameters. The remaining parameters are, on average, overestimated. In the space of rescaled parameters θ^{norm} we do not observe a significant improvement of the estimation.

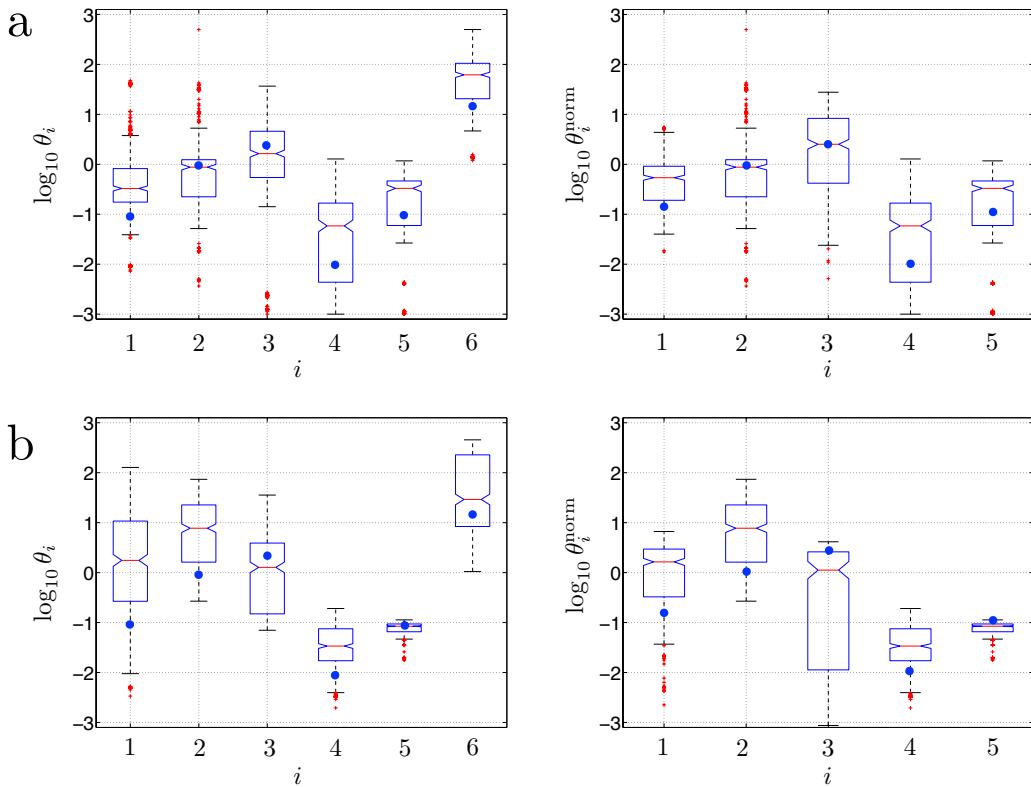


Figure 7.4: **a.** Left panel: Box plot of the 30 best parameter vectors from each of the 15 independent optimization runs for the steady-state data set. Right panel: Box plots of the normalized parameters (see main text for details). **b.** Left panel: Box plot of the 30 best parameter vectors from each of the 15 independent optimization runs for the transient data set. Right panel: Box plot of the normalized parameters (see main text for details). The blue dots indicate the true parameter values.

7.6 Conclusions and Summary

We have considered parameter estimation in monostable stochastic biochemical networks from single experimental trajectories. Parameter identification from single time series is desirable in image-based systems biology, where per-cell estimates of the fluorescence evolution and its fluctuations are available. This enables quantifying cell-cell variability on the level of network parameters. The histogram of the parameters identified for different cells provides a biologically meaningful way of assessing phenotypic variability beyond simple differences in the fluorescence levels.

We have proposed a novel combination of a flexible Monte Carlo method, the Gaussian Adaptation (GaA) algorithm, and efficient exact stochastic simulation algorithms, the partial-propensity methods. The presented method can be used for global parameter optimization, approximate Bayesian inference under uniform prior, and ellipsoidal volume estimation of the viable parameter space. We have introduced an objective function that measures closeness between a single experimental trajectory and a single trajectory generated by the forward model. The objective function comprises a moment-matching and a time-autocorrelation part. This allows including experimental readouts from, e.g., fluorescence photometry and fluorescence correlation spectroscopy.

We have applied the method to estimate the parameters of two monostable reaction networks from a single simulated temporal trajectory each, both at steady state and during transient phases. We considered the linear cyclic chain network and a nonlinear colloidal aggregation network. For the linear model we were able to robustly identify a small region of parameter space containing the true kinetic parameters. In the nonlinear aggregation model, we could identify several parameter vectors that fit the simulated experimental data well. There are two possible reasons for this reduced parameter identifiability: either GaA cannot find the globally optimal region of parameter space due to high ruggedness and noise in the objective function, or the nonlinearity of the aggregation network modulates the kinetics in a non-trivial way (Grima, 2009b) (see Chapters 4 and 5). Both cases are not accounted for in the current objective function, thus leading to reduced performance for nonlinear reaction networks.

We also used GaA as an adaptive MCMC method for approximate Bayesian inference of the posterior parameter distributions in the linear chain network. This enabled estimating the volume of the viable parameter space below a given objective-function value threshold. We found these volume estimates to be stable across independent runs. We thus believe that GaA might be a useful tool for exploring the parameter spaces of stochastic systems. This is corroborated by unpublished results indicating that GaA is able to robustly identify plausible parameters even in cases where CMA-ES is not. This might be due to CMA-ES being a non-elitist, population-based heuristic, which can be unfavorable on very noisy objective functions (Jin and Branke, 2005). Further studies are required to elucidate this hypothesis.

8

Conclusions, limitations and future work

“The pursuit of science has often been compared to the scaling of mountains, high and not so high. But who amongst us can hope, even in imagination, to scale the Everest and reach its summit when the sky is blue and the air is still, and in the stillness of the air survey the entire Himalayan range in the dazzling white of the snow stretching to infinity? ... But there is nothing mean or lowly in standing in the valley below and awaiting the sun to rise over Kangchenjunga.”

– *Subrahmanyan Chandrasekhar*

We have developed efficient stochastic simulation algorithms (SSA) for sampling trajectories from the exact solution of the chemical master equation (CME) and to better understand the role of fluctuations in mesoscopic chemical systems. We also applied the presented algorithms and results to stochastic simulations of spatiotemporal reaction-diffusion processes and to global parameter identification in stochastic chemical reaction networks. Below, we summarize the key conclusions, assess the limitations of the presented work and outline possible future research directions.

Stochastic chemical kinetics. We recapitulated the theoretical foundations of stochastic chemical kinetics in well-stirred reaction systems. We revisited the derivation of the CME from the Chapman-Kolmogorov equation and demonstrated which systematic approximations lead to the chemical Kramer-Moyal equation, the chemical nonlinear Fokker-Planck equation and the classical reaction rate equation (RRE). We also reviewed the derivation of Gillespie’s stochastic simulation algorithm (SSA) to sample trajectories from the exact solution of the CME.

8 Conclusions, limitations and future work

Assuming that the population of species in a chemical reaction system is proportional to the reactor volume, we demonstrated the effect of intrinsic noise due to low population in small reactor volumes on simple chemical reaction networks. We showed that the mean concentration of species as predicted by the CME in a monostable linear reaction network is equal to the concentration prediction from the RRE. For monostable nonlinear reaction networks, we demonstrated that the mean concentration predicted by the CME deviates from that predicted by the RRE. We also showed that intrinsic noise can induce oscillatory behavior and can cause switching between the fixed points of multistable reaction networks.

Computational cost of exact SSA formulations. We reviewed existing SSAs and analyzed their computational cost. We presented the topological properties of chemical reaction networks that can be inferred from their $N \times M$ stoichiometry matrices, where N is the number of chemical species and M the number of reactions. Using the dependency graph representation, we presented the cost-determining topological properties and used them to classify reaction networks into *strongly coupled* and *weakly coupled* networks. Based on the coupling class of a reaction network, we reformulated the computational cost of exact SSA formulations. We showed that for weakly coupled networks the computational cost of exact SSAs has been reduced to $O(\log_2 M)$ or even to $O(1)$. For strongly coupled networks, we showed that the computational cost of previous exact SSA formulations is at best linear in the number of reactions, i.e., $O(M)$.

Partial-propensity exact SSA formulations. We presented the first contribution of this thesis, a novel quantity called *partial propensity*, using which we devised a family of partial-propensity exact SSA formulations whose computational cost is at most linear in the number of chemical species, i.e., $O(N)$ irrespective of the coupling class of the reaction network. The concept of partial propensities was also proposed later by Indurkhy and Beal (Indurkhy and Beal, 2010). We further presented a family of SSAs and provided a comprehensive analysis of their computational cost.

The scaling of the computational cost of partial-propensity formulations is mediated by the use of partial propensities where sampling of a reaction can be interpreted as sampling a reactant followed by sampling its reaction partner. We presented the partial-propensity direct method (PDM) that has a computational cost of $O(N)$ irrespective of the coupling class of the network, rendering it especially efficient for strongly coupled reaction networks. The sorting variant of PDM (SPDM) incorporates dynamic “bubbling up” of reactions, rendering it especially efficient for multiscale reaction networks. Combining partial propensities and composition-rejection sampling, we devised the partial-propensity SSA with composition-rejection sampling (PSSA-CR) whose computational is reduced to $O(1)$ for weakly coupled networks. We further extended partial-propensity methods to reaction networks with time delays.

Finally, we illustrated that partial-propensity formulations form a family of SSAs with distinct algorithmic building blocks. Suitably modifying and combining these building blocks yields all possible partial-propensity formulations for reaction networks with and without delays. Using the modular representation of partial-propensity formulations, we developed the partial-propensity SSA (pSSA) software in C++, which conveniently simulates chemical reaction networks from their Systems Biology Markup Language (SBML) representation.

Partial-propensity formulations are limited to elementary reactions. Non-elementary reactions can, however, be broken down to elementary reactions. This reduction results in equivalent chemical kinetics in the large-volume quasi-steady-state approximation. The validity of the reduction, however, is questionable for mesoscopic systems in small reactor volumes. This limitation can be overcome by extending partial-propensity formulations to multi-dimensional instead of two-dimensional partial-propensity structures. In applications such as population ecology, partial propensities are only beneficial if the population of at least one species can be factored out from the propensity. The current implementation of partial-propensity formulations further requires that these reaction are at most binary. For small networks, partial-propensity formulations may not be efficient, as the overhead of creating the data structures may not be amortized. All of these limitations are specific to partial-propensity formulations and none of the existing exact SSAs are restricted by them. Using partial propensities will not change the scaling of the computational cost of approximate SSAs. Nevertheless, improved computational performance can be expected due to a smaller number of update operations owing to grouping of the partial propensities according to the index of the factored-out reactant.

A limitation of the analysis of the computational costs of exact SSAs is that we have only presented upper bounds. The favorable scaling of the upper bound for partial-propensity methods does not guarantee superior performance on any given reaction network. This limitation is due to lack of lower bounds for the previous methods. Nevertheless, we have demonstrated favorable performance on prototypical test cases. Deriving a lower bound for the computational cost requires much finer topological properties of chemical reaction networks. We believe that by restricting the topological properties of chemical reaction networks to those observed in real-world systems, one might be able to provide lower bounds. Future work will consider deriving such lower bounds for the scaling of the computational cost of previous SSAs and partial-propensity methods.

Discreteness-induced concentration inversion in mesoscopic chemical systems. We applied partial-propensity methods to study the effects of noise in mesoscopic chemical reaction networks. The results constitute the second key contribution of this thesis. We studied the effect of noise on static properties at steady state for mesoscopic monostable non-oscillating chemical reaction systems.

Chemical reaction systems with low population of reactants are strongly influenced by intrinsic noise. Assuming that the population of species is proportional to the reactor volume, intrinsic noise due to apparent molecular discreteness is expected to play an increasingly significant role in smaller reactor volumes. In addition, many reactors are subject to extrinsic noise originating from outside the reactor. We modeled extrinsic noise by the burst size of a reactant fed into the reactor. We demonstrated that the qualitative predictions of steady-state mean concentrations from RRE and from the linear-noise approximation (LNA) of the CME are only valid above a critical reactor volume. Below this critical volume, the ordering of the steady-state mean concentrations predicted by the CME is different due to noise. We call this novel effect *discreteness-induced concentration inversion*.

We showed that extrinsic noise increases the critical volumes, rendering the RRE and LNA incorrect over a larger range of reactor volumes. We also illustrated that stochastic burst, as observed for example in biological systems, further increases the critical reactor volume. We

8 Conclusions, limitations and future work

presented a theory that predicts the critical volume using van Kampen's system-size expansion. The results highlight that noise, intrinsic and extrinsic, not only deviates stochastic chemical kinetics from its macroscopic limit, but also renders the two qualitatively different. We found that the predictions of the presented theory are in good agreement with numerical simulations performed using exact SSAs.

The presented work, however, is limited to monostable systems. In multistable systems, noise can induce switching between stable fixed points of the RRE and steady-state mean concentrations of species are hence bad descriptors for studying the effect of noise. Even in mesoscopic oscillatory systems, oscillations due to noise render steady-state mean concentrations less relevant. The presented theory is also limited to monostable systems due to the limitations of the van Kampen ansatz in the system-size expansion. Furthermore, at small reactor volumes and large burst sizes, the presented theory predicts negative species concentrations. This is due to the truncation of van Kampen's asymptotic expansion to the least number of terms capable of predicting deviations in the mean concentration of stochastic chemical kinetics from that of the RRE. Future work include (i) using additional terms in van Kampen's asymptotic expansion to increase the domain of noise-magnitudes over which the resulting finite approximation of the expansion yields physical results, (ii) trying to observe *discreteness-induced concentration inversions* in experiments, and (iii) attempting to generalize the theory to non-monostable systems.

Relaxation kinetics of the fluctuations in mesoscopic chemical systems. Next, we studied the fingerprints of noise-induced modulations of dynamic properties of mesoscopic monostable reaction networks. We quantified the relaxation kinetics of the fluctuations around a nonequilibrium steady state of a reaction network. We studied the effect of increasing the magnitude of intrinsic noise by decreasing the reactor volume and increasing the magnitude of extrinsic noise by increasing burst size of a reactant fed into the reactor. We computed the lifetimes of concentration fluctuations, which we defined as the power spectral density of the fluctuations at zero frequency. We observed that the lifetimes of all species involved as a reactant in a nonlinear reaction increase with decreasing reactor volume. We also showed that the two sources of noise affect the derivative of the time-autocorrelation function in opposite ways, allowing us to differentiate between them. In addition, we showed that for mesoscopic oscillatory reaction systems, the bandwidth of the oscillations increases and the peak frequency decreases with decreasing reactor volume. This systematic change in the peak frequency and the bandwidth corroborates the increase in lifetime observed in non-oscillating mesoscopic systems. The decrease in the peak frequency is further exacerbated with decreasing reactor volume in reaction networks that are downstream in a cascade of self-similar networks. The bandwidth along the cascade decreases.

We believe that understanding the changes in the characteristics of the fluctuation spectrum at a nonequilibrium steady-state might be beneficial for estimating reaction rates from fluorescence correlation spectroscopy measurements. Understanding these changes might also advance the general understanding of the role of noise in stochastic chemical kinetics.

The changes in the fluctuation spectrum were demonstrated on specific model systems, namely the trimerization reaction network and a cascade of downstream-coupled Brusselators. This thesis does not provide evidence for the applicability of these results to other nonlinear reaction networks. Even in the case of general applicability of these results, we did not provide

a systematic methodology to incorporate the findings when estimating reaction rates from experimental data. Merely looking at the power spectral density of species fluctuations at a nonequilibrium steady state might not suffice for generally understanding the role of noise. A systematic analysis of the effect of noise on cross-correlation spectra might be beneficial, as their spectrum are directly related to reaction fluxes in stochastic chemical kinetics (Qian and Elson, 2004). We also did not provide a mechanistic description of how the dynamic properties affect the static properties of stochastic reaction networks. The analysis at this point remains purely mathematical without any intuitive interpretation of the role of fluctuations in stochastic chemical kinetics*. Future work will focus on understanding the role of noise in the cross-correlation spectrum, provide a systematic methodology to use the results for estimating reaction rates from experimental data and attempt to provide an intuitive understanding of the role of fluctuations in stochastic chemical kinetics.

Partial-propensity formulation of the on-lattice stochastic reaction-diffusion simulations. We used the algorithms and the results presented so far in two applications. The first one considered the simulation of stochastic spatiotemporal reaction-diffusion processes.

Spatial heterogeneity is relevant in cases where the diffusion of reactants is slower than the reactions between them. In such scenarios, the CME is not an appropriate description due to its implicit assumption of spatial homogeneity. Spatial heterogeneity can, however, be accounted for by the on-lattice reaction-diffusion master equation (RDME) which is analogous to the CME. In the RDME, the reactor is divided into subvolumes, in each of which the reaction system is well stirred. Diffusion is modeled as jump reactions between neighboring subvolumes. We presented an efficient algorithm for sampling trajectories from the exact solution of the RDME. We combined composition-rejection sampling to find the index of the next subvolume with a partial-propensity method to sample the index of the next reaction within the subvolume. The computational cost of the algorithm is at most logarithmic in the number of subvolumes and linear in the number of species.

The limitations of partial-propensity methods equally apply to the presented algorithm. Moreover, the lack of lower bounds of the computational cost hampers deciding between using the next subvolume method (NSM) (Elf and Ehrenberg, 2004), an existing on-lattice stochastic reaction-diffusion simulator, and the presented method. Future work will attempt to address this limitation. We will also render the software implementation more efficient by using elaborate algorithms as described in Sec. 6.3. We will further extend the method to make it compatible with Smoluchowski's microscopic foundation of reaction-diffusion as presented in a recent work (Fange et al., 2010). It will also be interesting to extend the presented method to adaptive subvolumes depending on the local population of reactants, as presented recently (Bayati et al., 2011). It remains to be seen, however, whether the RDME that is compatible with Smoluchowski's microscopic framework of reaction-diffusion can be extended to adaptive subvolumes.

Global parameter identification of stochastic chemical reaction networks using single trajectories The second application considered the problem of estimating the unknown reaction rates and the reactor volume of a stochastic chemical reaction network from a single, noisy

* “An equation for me has no meaning unless it expresses a thought of God.” – Srinivasa Ramanujan

8 Conclusions, limitations and future work

species population measurement. This constitutes an inverse problem. Assuming that the topology of the chemical reaction network is known, we attempted to address the problem of *parameter identification* from experimental data. We presented a novel methodology combining partial-propensity methods to sample from the exact solution of the CME with an adaptive Monte Carlo sampling technique called Gaussian adaptation (Kjellström and Taxen, 1992; Müller and Sbalzarini, 2010b,a; Müller, 2010) to search the parameter space. Gaussian adaptation iteratively proposes candidate parameters for which the partial-propensity method samples a realization from the solution of the CME. An objective function is then used to measure the distance between the experimental trajectory and the simulated trajectory. This distance function includes the fingerprints of the noise-induced modulations of stochastic chemical kinetics as presented in the previous chapters. We also used Gaussian adaptation to provide an ellipsoidal volume estimate of the viable parameter space as a means to quantify the uncertainty in the estimation of the parameters.

We found that the presented method provided reasonable parameter estimates for linear reaction networks. For nonlinear reaction networks, however, the estimation quality was less satisfactory. This could possibly be improved by using a more elaborate objective function. We used synthetic “experimental” trajectories generated *in silico* using an exact SSA. Therefore, we did not account for measurement noise, nor for sample variability. Future work will consider (i) alternative objective functions that include temporal cross-correlations between species and the derivatives of the autocorrelation; (ii) longer experimental trajectories; (iii) multi-stable and oscillatory systems; (iv) alternative global optimization schemes; and (v) real experimental data from single-cell fluorescence photometry and fluorescence correlation spectroscopy. Moreover, the applicability of the present method to large, nonlinear biochemical networks will be tested in future work.

On the eve of the Alan Turing year, I wish to conclude this thesis with the last sentence from his seminal work on artificial intelligence (Turing, 1950): “We can only see a short distance ahead, but we can see plenty there that needs to be done.”

Appendix

A1 Method of generating functions to compute moments from the CME

Consider the following reaction network in a reactor of volume Ω :



The CME is

$$\frac{\partial P(n_1, t)}{\partial t} = k_1 \Omega P(n_1 - 1, t) + (n_1 + 2)(n_1 + 1) \frac{k_2}{\Omega} P(n_1 + 2, t) \quad (\text{A1.2})$$

$$- \left[k_1 \Omega + n_1(n_1 - 1) \frac{k_2}{\Omega} \right] P(n_1, t), \quad (\text{A1.3})$$

where n_1 is the population of S_1 .

The moment-generating function is

$$F(z_1, t) = \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t). \quad (\text{A1.4})$$

The moments of $P(n_1, t)$ can be expressed in terms of the derivatives of $F(z_1, t)$ with respect to z_1 . For instance,

$$\left. \frac{\partial F(z_1, t)}{\partial z_1} \right|_{z_1=1} = \langle n_1 \rangle(t). \quad (\text{A1.5})$$

Other moments can be obtained similarly.

Multiplying the CME by $z_1^{n_1}$ and summing over n_1 we get

$$\begin{aligned} \frac{\partial \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t)}{\partial t} &= k_1 \Omega \left[\sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1 - 1, t) - \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t) \right] \\ &+ \frac{k_2}{\Omega} \left[\sum_{n_1=0}^{\infty} z_1^{n_1} (n_1 + 2)(n_1 + 1) P(n_1 + 2, t) - \sum_{n_1=0}^{\infty} z_1^{n_1} n_1(n_1 - 1) P(n_1, t) \right], \end{aligned} \quad (\text{A1.6})$$

Appendix

i.e.,

$$\begin{aligned}
\frac{\partial F(z_1, t)}{\partial t} &= k_1 \Omega \left[z_1 \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t) - \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t) \right] \\
&\quad + \frac{k_2}{\Omega} \left[\sum_{n_1=0}^{\infty} z_1^{n_1-2} n_1 (n_1-1) P(n_1, t) - \sum_{n_1=0}^{\infty} z_1^{n_1} n_1 (n_1-1) P(n_1, t) \right] \\
&= k_1 \Omega \left[z_1 \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t) - \sum_{n_1=0}^{\infty} z_1^{n_1} P(n_1, t) \right] \\
&\quad + \frac{k_2}{\Omega} \left[\sum_{n_1=0}^{\infty} z_1^{n_1-2} n_1 (n_1-1) P(n_1, t) - z_1^2 \sum_{n_1=0}^{\infty} z_1^{n_1-2} n_1 (n_1-1) P(n_1, t) \right] \\
&= k_1 \Omega (z_1 - 1) F(z_1, t) + \frac{k_2}{\Omega} (1 - z_1^2) \frac{\partial^2 F(z_1, t)}{\partial t^2}.
\end{aligned}$$

This equation has no obvious solution. Restricting ourselves to the steady-state solution $F_{ss}(z_1) = \sum_{n_1=0}^{\infty} z_1^{n_1} P_{ss}(n_1)$, the equation is simplified by setting the time derivative to zero:

$$\frac{\partial^2 F_{ss}(z_1)}{\partial z_1^2} = \frac{k_1 \Omega^2}{k_2(1+z_1)} F_{ss}(z_1). \quad (\text{A1.7})$$

The boundary conditions are $F_{ss}(1) = \sum_{n_1=0}^{\infty} P_{ss}(n_1) = 1$ and $F_{ss}(-1) = 0$. The solution is then given by

$$F_{ss}(z_1) = \frac{\sqrt{\frac{k_1 \Omega^2}{k_2}} (1+z_1) I_1 \left(2\sqrt{\frac{k_1 \Omega^2}{k_2}} (1+z_1) \right)}{\sqrt{2\frac{k_1 \Omega^2}{k_2}} I_1 \left(2\sqrt{2\frac{k_1 \Omega^2}{k_2}} \right)}, \quad (\text{A1.8})$$

where $I_a(x)$ is the modified Bessel function of the first kind.

The steady-state mean concentration $\langle \phi_1^* \rangle_{ss} = \Omega^{-1} \langle n \rangle_{ss} = \Omega^{-1} \left. \frac{\partial F_{ss}(z_1)}{\partial z_1} \right|_{z_1=1}$ is

$$\langle \phi_1^* \rangle_{ss} = \frac{1}{4\Omega} + \sqrt{\frac{k_1}{2k_2}} \frac{I'_1 \left(2\sqrt{2} \sqrt{\frac{k_1 \Omega^2}{k_2}} \right)}{I_1 \left(2\sqrt{2} \sqrt{\frac{k_1 \Omega^2}{k_2}} \right)}, \quad (\text{A1.9})$$

where $I'_a(b) = \left. \frac{dI_a(x)}{dx} \right|_{x=b}$. In terms of the RRE steady-state concentration $\phi_{1,ss} = \sqrt{\frac{k_1}{2k_2}}$

$$\langle \phi_1^* \rangle_{ss} = \frac{1}{4\Omega} + \phi_{1,ss} \frac{I'_1 (4\phi_{1,ss}\Omega)}{I_1 (4\phi_{1,ss}\Omega)}, \quad (\text{A1.10})$$

where the derivative $I'_1(a) = (1/a)I_1(a) + I_2(a)$.

The asymptotic expansion of the the steady-state mean concentration is

$$\langle \phi_1^* \rangle_{ss} = \phi_{1,ss} + \frac{\Omega^{-1}}{8} + \frac{3\Omega^{-2}}{128\phi_{1,ss}} + O(\Omega^{-3}). \quad (\text{A1.11})$$

A1 Method of generating functions to compute moments from the CME

We then see that

$$\lim_{\Omega \rightarrow \infty} \langle \phi_1^* \rangle_{ss} = \phi_{1,ss}. \quad (\text{A1.12})$$

Appendix

A2 Breakdown of a non-elementary reaction into elementary reactions

Consider the following reaction



The reaction is considered non-elementary if

$$\sum_{i=1}^N \nu_{i,\mu}^- > 2. \quad (\text{A2.2})$$

The simplest non-elementary reaction is the trimolecular reaction



It can be argued that such trimolecular reactions are not physical. This is because writing the reaction as in Eq. A2.3 requires the simultaneous collision of three molecules, which is highly improbable. The probability of such a collision to happen in an infinitesimal time interval dt scales as $O(dt^2)$ (Gillespie, 1991). This is particularly the case in dilute chemical systems, of which the CME is an exact mesoscopic description. This scaling is furthermore inconsistent with Eq. 1.2. Nevertheless, the reaction in Eq. A2.3 can be interpreted as a large-volume quasi-steady-state approximation of a system of 3 elementary reactions:



where

$$k = \frac{k_1 k_3}{k_2}, \quad (\text{A2.5})$$

such that k_2 is very large (Wilhelm, 2000; Schneider and Wilhelm, 2000; Gillespie, 1991, 1992).

In the above case, the trimolecular reaction is broken down into elementary reactions with 1 additional species and 2 additional reactions, assuming that the original non-elementary reaction is the large-volume, quasi-steady-state limit case of the elementary reactions. In general, following the same procedure, any reaction of the form described in Eq. A2.1 with $\sum_{i=1}^N \nu_{i,\mu}^- > 2$ can be broken down into elementary reactions with an additional $(\sum_{i=1}^N \nu_{i,\mu}^-) - 2$ species. The total number of resulting elementary reactions is $2(\sum_{i=1}^N \nu_{i,\mu}^-) - 3$.

It should, however, be noted that the stochastic kinetics of the non-elementary reaction is expected to be equivalent to the broken-down system of elementary reactions only in the quasi-steady-state approximation in large volumes. It has also been argued that the large-volume,

A2 Breakdown of a non-elementary reaction into elementary reactions

quasi-steady-state approximation is indeed necessary for the validity of any non-elementary, macroscopic chemical kinetic models used for enzyme kinetics, cooperativity, competitive inhibition, allosteric inhibition, and others. The condition required for these approximations to be justified, however, may not be met in all biochemical reactors, and hence the non-elementary macroscopic description may not be equivalent to the elementary, microscopic description (Grima, 2009b; Kholodenko and Westerhoff, 1995; Markevich et al., 2004; Flach and Schnell, 2006; Farrow and Edelson, 1974; Segel, 1988; Pedersen et al., 2008; Marquez-Lago and Stelling, 2010).

Appendix

A3 Equivalence of PDM's next reaction sampling to that of DM

In the direct method (DM), the next reaction index is sampled as

$$\mu = \min \left[\mu' : r_1 a < \sum_{m=1}^{\mu'} a_m \right], \quad (\text{A3.1})$$

where r_1 is a uniform random number $\in [0, 1]$ and a_m is the propensity of reaction m . Without loss of generality, we identify μ' by a unique pair of indices, I' and J' , such that $\mu' = L_{I', J'}$. Using this mapping to a group (row) index I' and an element (column) index J' , Eq. A3.1 becomes

$$\begin{pmatrix} I \\ J \end{pmatrix} = \min \left[\begin{pmatrix} I' \\ J' \end{pmatrix} : r_1 a < \sum_{i=0}^{I'-1} \sum_{\forall j} a_{L_{i,j}} + \sum_{j=1}^{J'} a_{L_{I',j}} \right], \quad (\text{A3.2})$$

such that $\mu = L_{I,J}$. This can be written for the group (row) index I alone

$$I = \min \left[I' : r_1 a < \sum_{i=0}^{I'} \sum_{\forall j} a_{L_{i,j}} \right] \quad (\text{A3.3})$$

and the element (column) index J alone

$$J = \min \left[J' : r_1 a < \sum_{i=0}^{I-1} \sum_{\forall j} a_{L_{i,j}} + \sum_{j=1}^{J'} a_{L_{I,j}} \right]. \quad (\text{A3.4})$$

Using the definitions for Σ_i and Π_i , Eqs. A3.3 and A3.4 are equivalent to Eqs. 3.2 and 3.3, respectively.

A4 The partial-propensity SSA (pSSA) software package

The pSSA software is designed to facilitate simulation of chemical reaction networks using partial-propensity formulations of the exact SSA. The software was developed by Oleksandr Ostrenko (Master student in the Computational Biology and Bioinformatics program at ETH Zurich) during his research assistantship at the MOSAIC group between November 2010 and June 2011. The text in this section is partially reproduced from his technical report.

pSSA is written in C++ using the guidelines and features of the ANSI/ISO C++ Standard, providing the following features:

- The whole application programming interface (API) is object-oriented, providing a higher level of abstraction over the underlying SSAs. This allows users to perform simulations at a high level of abstraction without needing to worry about low-level implementation details. All main functions are made available through a single class, which serves as the interface.
- All functional units of pSSA are written to facilitate extension of their functionality, for example, addition of new SSAs.

A4.1 Partial-propensity methods implemented in pSSA

pSSA implements the algorithmic building blocks of partial-propensity methods as presented in Sec. 3.6. This provides a simple interface to the following partial-propensity formulations:

- partial-propensity direct method (PDM)
- sorting partial-propensity direct method (SPDM)
- partial-propensity SSA with composition-rejection sampling (PSSA-CR)
- delay partial-propensity direct method (dPDM)
- delay sorting partial-propensity direct method (dSPDM)
- delay partial-propensity SSA with composition-rejection sampling (dPSSA-CR).

In addition, Gillespie's original direct method (DM) is also implemented for comparison.

A4.2 Platforms

pSSA has been tested on Linux 2.6 and Mac OS X 10.5 workstations using the GNU C++ compiler.

A4.3 Dependencies

Apart from the classes provided by the standard template library (STL), pSSA makes use of the following external libraries: the GNU Scientific Library (GSL) for random number generation, the Boost library for hash maps, the Systems Biology Markup Language (SBML) library (Hucka et al., 2003) for importing model data from SBML input files, and the Command Line Parser Library (TCLAP) for the command-line interface (CLI).

Appendix

A4.4 Functionality

The functionality of pSSA can be divided into two main categories:

- Sampling trajectories from the CME and then computing the average and standard deviation of the populations of user-specified species. We refer to this functionality as the “Traj” mode
- Computing the marginal population probability functions of user-specified species at a given time. In addition, pSSA also computes the multivariate population probability function. This functionality is referred to as the “Hist” mode

Both functionalities can be accessed through the CLI of pSSA or from within MATLAB.

A4.5 Command line interface (CLI) of pSSA

The CLI supports the two modes, “Traj” and “Hist”, as selected by a command line switch. In “Traj” mode we sample trajectories and compute average and standard deviation of the population. Simulations in “Traj” mode write output files containing the trajectories from each Monte Carlo run, trajectory of average population, and the trajectory of the standard deviation of the population. In “Hist” mode, the histogram of species at a specified time is computed for specified species, and written to an output file. In addition, the “Hist” mode also writes an output file with the average runtime per reaction event.

Detailed information on the command line arguments is given in the help page below:

USAGE:

```
./ build/pssa_cli -i </input/path/file.sbml>
                  -o </output/path/>
                  -m <‘‘0 1 2 3 4 5 6’’>
                  [-t <0.0>]
                  -f <1.0>
                  [-r]
                  [-b]
                  [-d <0.1>]
                  -n <‘‘10 20 30’’>
                  [-s <‘‘species_1 species_2 ... species_N’’>]
                  [-]
                  [--version]
                  [-h]
```

where :

```
-i </input/path/file.sbml>, --input-file </input/path/file.sbml>
  (required) String representing the SBML model file.
-o </output/path/>, --output-path </output/path/>
  (required) String representing the output path.
-m <‘‘0 1 2 3 4 5 6’’>, --method-label <‘‘0 1 2 3 4 5 6’’>
  (required) String of method labels separated by a space.
  Method labels are: 0 (Gillespie’s DM), 1 (PDM), 2 (PSSA-CR), 3 (SPDM),
  4 (dPDM), 5 (dPSSA-CR), 6 (dSPDM)
-t <0.0>, --start-time <0.0>
  Time to start output to file in ‘‘Traj’’ mode.
-f <1.0>, --final-time <1.0>
  (required) Final time of the simulation.
-r, --traj-mode
  If set, the program is run in ‘‘Traj’’ mode. Otherwise, ‘‘Hist’’ mode is
  assumed.
-b, --hist-info
  If run in ‘‘Traj’’ mode turns on statistical output. In ‘‘Hist’’ mode,
  causes the engine to switch between one- and multidimensional
  histograms
-d <0.1>, --dt <0.1>
  Time interval between outputs.
-n <‘‘10 20 30’’>, --ntrials <‘‘10 20 30’’>
  (required) If run in ‘‘Hist’’ mode: string consisting of numbers of
  trials to be simulated. Multiple instances are to be separated by a
  space. In ‘‘Traj’’ mode only first value will be considered.
-s <‘‘species_1 species_2 species_3 ... species_N’’>, --species-label <‘‘
  species_1 species_2 species_3 ... species_N’’>.
  If run in ‘‘Hist’’ mode: string of species labels for which the histogram
  at a given final time will be computed. Multiple labels are to be separated
  by a space. Ignored in ‘‘Traj’’ mode.
--, --ignore_rest
  Ignores the rest of the labeled arguments following this flag.
--version
  Displays version information and exits.
-h, --help
  Displays usage information and exits.
```

A4.6 Calling pSSA from MATLAB

pSSA can also be called from within MATLAB using the two functions:

Appendix

```

function [P] = pssalib_avg(input, method, ntrials, final_time, dt)
function [P] = pssalib_hist(input, method, ntrials, species, final_time, multi).

```

The first function samples individual trajectories and returns the average and the standard deviation of the population. The second function is provided to compute histograms for given species. Both functions call the CLI of pSSA. Arguments are defined as follows:

input	String representing the SBML model file
method	Array of method labels
ntrials	Array of numbers of trials to be simulated
final_time	Final time of the simulation
species	String of species labels for which the histogram at a given final time will be computed
dt	Time interval between outputs.
multi	If set to ‘true’, computes the multivariate histograms of species populations.

The intended usage of these functions is the same as of the CLI. Each function produces a cell array as its output:

- **pssalib_avg** produces a cell array containing arrays of average trajectories and respective standard deviations at every time point for each of the methods selected. The first cell array contains the species identifiers.
- **pssalib_hist** returns a cell array containing arrays of species histograms, one for each species simulated with each of the selected methods. Alternatively, if **multi** is set to true, a multidimensional species histogram over all of the chosen species is computed using each of the selected methods.

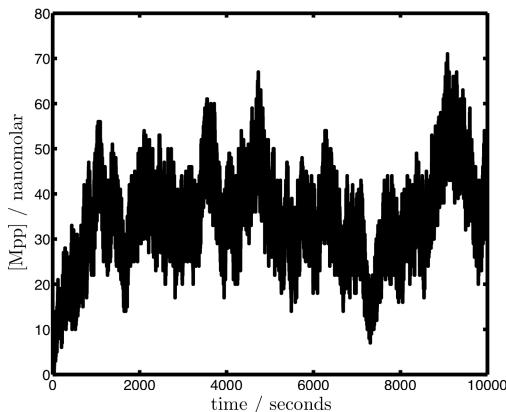
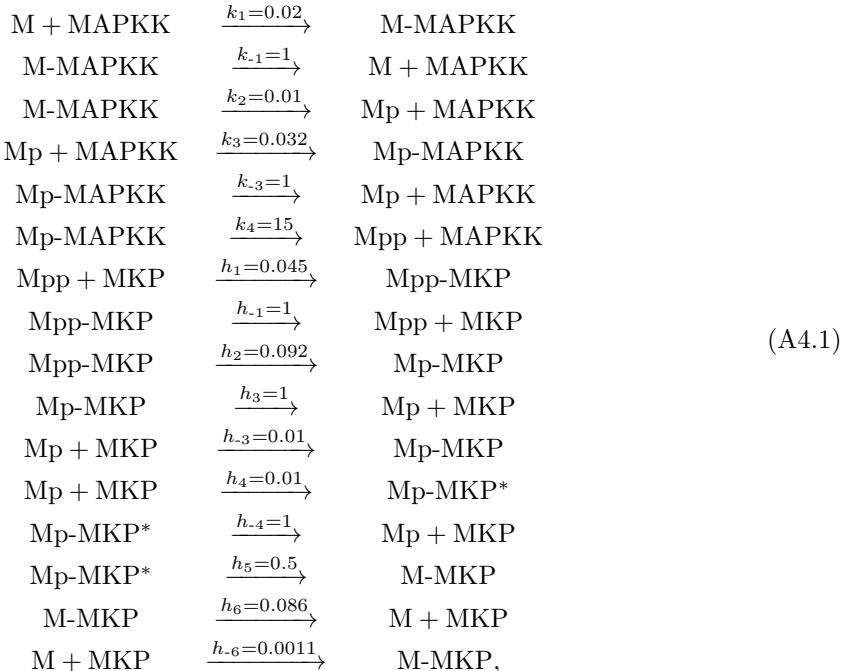


Figure A4.1: A single trajectory of the time evolution of the concentration of double-phosphorylated MAPK, obtained using pSSA.

A4.7 Simulation of the mitagen-activated protein kinase (MAPK) phosphorylation-dephosphorylation system using pSSA

The MAPK phosphorylation-dephosphorylation system is modeled using the following set of reactions (Markevich et al., 2004):



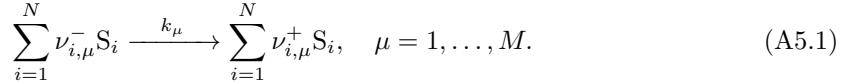
where the k_i and h_i are the macroscopic rate constants. The volume of the reactor is Ω . The unit for the rates of unimolecular reactions is s^{-1} , that for bimolecular reactions $\text{nanomolar}^{-1} s^{-1}$. M denotes dephosphorylated MAPK, MAPKK the MAPK kinase, Mp phosphorylated MAPK on the tyrosine or on the threonine residues, Mpp the double-phosphorylated MAPK on both residues, MKP the MAPK phosphatase, Mp-MKP the complex in which M is phosphorylated on the threonine residue and Mp-MKP* the complex in which M is phosphorylated on the tyrosine residue.

We simulate this reaction system using the CLI of pSSA. We choose SPDM as the simulation algorithm. The SBML file of the reaction system is generated manually and is used as input for pSSA. The initial concentration is $[M] = 500$ nanomolar, $[MAPKK] = 50$ nanomolar, $[MKP] = 100$ nanomolar. The concentrations of all other species are set to 0 nanomolar. We track the time evolution of concentration of Mpp with $\Omega = 1$ nanomolar^{-1} until a final time $t_f = 10000.0$ s. The command line call to execute the simulation is: `./build/pssa_cli -r -i 'YourPath/MAPK.sbml' -o 'YourPath/MAPK' -m '3' -s 'all species' -n '100' -f 10000.0 -d 0.1`. The result is shown in Fig. A4.1. On a Linux 2.6 workstation with a 2.4GHz Intel Core 2 Duo processor T8300, 4 GB of memory and 4 MB L2 cache, the total runtime for 100 Monte Carlo replica of the complete simulation is 104 seconds using SPDM and 127 seconds using DM.

Appendix

A5 The van Kampen system-size expansion

Consider the general reaction network with N species and M reactions as described in Eq. 1.1:



The RRE is given by (Eq. 1.46)

$$\frac{d\phi}{dt} = \boldsymbol{\nu} \mathbf{T}_0(\phi), \quad (\text{A5.2})$$

i.e.,

$$\frac{d\phi_i}{dt} = \sum_{\mu=1}^M \nu_{i,\mu} T_{\mu,0}(\phi) \quad i = 1, \dots, N. \quad (\text{A5.3})$$

The corresponding CME (Eqs. 1.19, 1.20 and 1.25) describing the mesoscopic system is

$$\frac{\partial P(\phi^*, t)}{\partial t} = \Omega \sum_{\mu=1}^M \left(\mathbb{E}^{-\Omega^{-1}\boldsymbol{\nu}_\mu} - 1 \right) T_\mu(\phi^*) P(\phi^*, t), \quad (\text{A5.4})$$

where \mathbb{E} is the step operator such that

$$\begin{aligned} \mathbb{E}^{-\Omega^{-1}\boldsymbol{\nu}_\mu} f(\phi^*) &= \left(\prod_{i=1}^N \mathbb{E}_i^{-\Omega^{-1}\nu_{i,\mu}} \right) f(\phi_1^*, \dots, \phi_N^*) \\ &= f(\phi_1^* - \Omega^{-1}\nu_{1,\mu}, \dots, \phi_N^* - \Omega^{-1}\nu_{N,\mu}). \end{aligned} \quad (\text{A5.5})$$

ϕ^* is the stochastic concentration vector described by the CME and ϕ is the deterministic concentration vector described by RRE.

The CME cannot be solved analytically, except in few special cases. Hence, a systematic approximation of the CME in a small parameter is beneficial. The small parameter needs to govern the size of the fluctuations and hence of the jumps in the Markov process describing stochastic chemical kinetics. For mesoscopic chemical reaction systems one such parameter is the reactor volume Ω ; when Ω is large the jumps are small. The resulting expansion is the van Kampen system-size expansion (van Kampen, 1961, 1976, 2001). It starts from the ansatz:

$$\phi^* = \phi + \Omega^{-1/2} \boldsymbol{\epsilon}, \quad (\text{A5.6})$$

where $\boldsymbol{\epsilon} = [\epsilon_1, \dots, \epsilon_N]^T$ is a continuous random variable that defines the fluctuations around ϕ . Using the above ansatz the probability distribution $P(\phi^*, t)$ is transformed to a new probability function $\Gamma(\boldsymbol{\epsilon}, t)$. The time derivative in Eq. A5.4 then becomes

$$\frac{\partial P(\phi^*, t)}{\partial t} = \frac{\partial \Gamma(\boldsymbol{\epsilon}, t)}{\partial t} + \sum_{i=1}^N \frac{d\epsilon_i}{dt} \frac{\partial \Gamma(\boldsymbol{\epsilon}, t)}{\partial \epsilon_i}. \quad (\text{A5.7})$$

This time derivative is taken at constant ϕ^* , therefore

$$\frac{d\epsilon_i}{dt} = -\Omega^{1/2} \frac{d\phi_i}{dt}. \quad (\text{A5.8})$$

Eq. A5.7 is then

$$\frac{\partial P(\phi^*, t)}{\partial t} = \frac{\partial \Gamma(\epsilon, t)}{\partial t} - \Omega^{1/2} \sum_{i=1}^N \frac{d\phi_i}{dt} \frac{\partial \Gamma(\epsilon, t)}{\partial t}. \quad (\text{A5.9})$$

From the definition of the step operator \mathbb{E} in Eq. A5.5 and the ansatz in Eq. A5.6

$$\begin{aligned} (\mathbb{E}^{-\Omega^{-1}\nu_\mu} - 1) &= -\Omega^{-1/2} \sum_{i=1}^N \nu_{i,\mu} \frac{\partial}{\partial \epsilon_i} + \frac{\Omega^{-1}}{2} \sum_{i,k=1}^N \nu_{i,\mu} \nu_{k,\mu} \frac{\partial^2}{\partial \epsilon_i \partial \epsilon_k} \\ &\quad - \frac{\Omega^{-3/2}}{6} \sum_{i,k,r=1}^N \nu_{i,\mu} \nu_{k,\mu} \nu_{r,\mu} \frac{\partial^3}{\partial \epsilon_i \partial \epsilon_k \partial \epsilon_r} + O(\Omega^{-2}). \end{aligned} \quad (\text{A5.10})$$

Under this ansatz, the function $T_\mu(\phi^*)$ hence is

$$\begin{aligned} T_\mu(\phi^*) &= T_\mu(\phi + \Omega^{-1/2}\epsilon) \\ &= T_\mu(\phi) + \Omega^{-1/2} \sum_{w=1}^N \epsilon_w \frac{\partial T_\mu(\phi)}{\partial \phi_w} + \frac{\Omega^{-1}}{2} \sum_{w,z=1}^N \epsilon_w \epsilon_z \frac{\partial^2 T_\mu(\phi)}{\partial \phi_w \partial \phi_z} + \dots \end{aligned} \quad (\text{A5.11})$$

Using Eqs. 1.37, 1.41 and 1.42, this becomes

$$\begin{aligned} T_\mu(\phi^*) &= T_{\mu,0}(\phi) + \Omega^{-1} T_{\mu,1}(\phi) + \Omega^{-1/2} \sum_{w=1}^N \epsilon_w \frac{\partial T_{\mu,0}(\phi)}{\partial \phi_w} \\ &\quad + \frac{\Omega^{-1}}{2} \sum_{w,z=1}^N \epsilon_w \epsilon_z \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_w \partial \phi_z} + O(\Omega^{-3/2}). \end{aligned} \quad (\text{A5.12})$$

Using the relationship given in Eq. 1.42,

$$T_{\mu,1}(\phi) = - \sum_{i=1}^N \frac{\phi_i}{2} \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_i^2}, \quad (\text{A5.13})$$

Eq. A5.12 is

$$T_\mu(\phi^*) = T_{\mu,0}(\phi) + \Omega^{-1/2} \sum_{w=1}^N \epsilon_w \frac{\partial T_{\mu,0}(\phi)}{\partial \phi_w} \quad (\text{A5.14})$$

$$+ \frac{\Omega^{-1}}{2} \left[\sum_{w,z=1}^N \epsilon_w \epsilon_z \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_w \partial \phi_z} - \sum_{w=1}^N \phi_w \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_w^2} \right] + O(\Omega^{-3/2}). \quad (\text{A5.15})$$

Appendix

Substituting Eqs. A5.9, A5.10 and A5.15 into Eq. A5.4 and noting that the coefficients of $\Omega^{1/2}$ get canceled upon substituting the RRE, we find the series expansion in $\Omega^{-1/2}$:

$$\begin{aligned} \frac{\partial \Gamma(\epsilon, t)}{\partial t} &= \sum_{\mu=1}^M \left\{ \Omega^0 \left[- \sum_{i,w=1}^N \nu_{i,\mu} \frac{\partial T_{\mu,0}(\phi)}{\partial \phi_w} \frac{\partial \epsilon_w \Gamma}{\partial \epsilon_i} + \frac{1}{2} \sum_{i,k=1}^N \nu_{i,\mu} \nu_{k,\mu} T_{\mu,0}(\phi) \frac{\partial^2 \Gamma}{\partial \epsilon_i \partial \epsilon_k} \right] \right. \\ &+ \frac{\Omega^{-1/2}}{2} \left[- \sum_{i,w,z=1}^N \nu_{i,\mu} \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_w \partial \phi_z} \frac{\partial \epsilon_w \epsilon_z \Gamma}{\partial \epsilon_i} + \sum_{i,w=1}^N \nu_{i,\mu} \phi_w \frac{\partial^2 T_{\mu,0}(\phi)}{\partial \phi_w^2} \frac{\partial \Gamma}{\partial \epsilon_i} \right] \\ &+ \frac{\Omega^{-1/2}}{2} \sum_{i,k,w=1}^N \nu_{i,\mu} \nu_{k,\mu} \frac{\partial T_{\mu,0}(\phi)}{\partial \phi_w} \frac{\partial^2 \epsilon_w \Gamma}{\partial \epsilon_i \partial \epsilon_k} \\ &- \frac{\Omega^{-1/2}}{6} \sum_{i,k,r=1}^N \nu_{i,\mu} \nu_{k,\mu} \nu_{r,\mu} T_{\mu,0}(\phi) \frac{\partial^3 \Gamma}{\partial \epsilon_i \partial \epsilon_k \partial \epsilon_r} \Big\} \\ &+ O(\Omega^{-1}). \end{aligned} \quad (\text{A5.16})$$

Equivalently,

$$\begin{aligned} \frac{\partial \Gamma(\epsilon, t)}{\partial t} &= \Omega^0 \left[- \sum_{i,w=1}^N J_{i,w}^{(0)}(\phi) \frac{\partial \epsilon_w \Gamma}{\partial \epsilon_i} + \frac{1}{2} \sum_{i,k=1}^N D_{i,k}^{(0)}(\phi) \frac{\partial^2 \Gamma}{\partial \epsilon_i \partial \epsilon_k} \right] \\ &+ \frac{\Omega^{-1/2}}{2} \left[- \sum_{i,w,z=1}^N \frac{\partial J_{i,w}^{(0)}(\phi)}{\partial \phi_z} \frac{\partial \epsilon_w \epsilon_z \Gamma}{\partial \epsilon_i} + \sum_{i,w=1}^N \phi_w \frac{\partial J_{i,w}^{(0)}(\phi)}{\partial \phi_w} \frac{\partial \Gamma}{\partial \epsilon_i} \right] \\ &+ \frac{\Omega^{-1/2}}{2} \sum_{i,k,w=1}^N J_{i,k,w}^{(0)}(\phi) \frac{\partial^2 \epsilon_w \Gamma}{\partial \epsilon_i \partial \epsilon_k} \\ &- \frac{\Omega^{-1/2}}{6} \sum_{i,k,r=1}^N D_{i,k,r}^{(0)}(\phi) \frac{\partial^3 \Gamma}{\partial \epsilon_i \partial \epsilon_k \partial \epsilon_r} \\ &+ O(\Omega^{-1}), \end{aligned} \quad (\text{A5.17})$$

where

$$J_{i,j,\dots,k,w}^{(0)}(\phi) = \sum_{\mu=1}^M \nu_{i,\mu} \nu_{j,\mu} \dots \nu_{k,\mu} \frac{\partial T_{\mu,0}(\phi)}{\partial \phi_w} \quad (\text{A5.18})$$

and

$$D_{i,j,\dots,k,w}^{(0)}(\phi) = \sum_{\mu=1}^M \nu_{i,\mu} \nu_{j,\mu} \dots \nu_{k,\mu} \nu_{w,\mu} T_{\mu,0}(\phi). \quad (\text{A5.19})$$

The $J^{(0)}$'s and $D^{(0)}$'s are related to the p^{th} jump moments $b_{p;q_1,\dots,q_N} = \sum_{i=0}^{\infty} b_{p;q_1,\dots,q_N}^{(i)} \Omega^{-i}$,

where $\sum_{i=1}^N q_i = p$ (Eqs. 1.31 and 1.38) such that

$$J_{i,j,\dots,k,w}^{(0)}(\phi) = \frac{\partial b_{p;q_1,\dots,q_N}^{(0)}}{\partial \phi_w} \quad \begin{aligned} q_i &= 1, q_j = 1, \dots, q_k = 1 \\ \text{and other } q\text{'s are zero,} \end{aligned} \quad (\text{A5.20})$$

and p is the sum of the q 's. In other words, it is one less than the number of indices of the $J^{(0)}$'s. Similarly, $D^{(0)}$'s are

$$D_{i,j,\dots,k,w}^{(0)}(\phi) = b_{p;q_1,\dots,q_N}^{(0)} \quad \begin{aligned} q_i &= 1, q_j = 1, \dots, q_k = 1, q_w = 1 \\ \text{and other } q\text{'s are zero,} \end{aligned} \quad (\text{A5.21})$$

where p is the sum of the q 's or the number of indices of the $D^{(0)}$'s. Using Eqs. 1.33 and 1.39 in the case when the $J^{(0)}$'s have only two indices, as in $J_{i,w}^{(0)}$, they can be written as the matrix:

$$\mathbf{J}^{(0)}(\phi) = [J_{i,w}^{(0)}] = \frac{\partial \mathbf{F}^{(0)}(\phi)}{\partial \phi} = \frac{\partial \boldsymbol{\nu} \mathbf{T}_0(\phi)}{\partial \phi}, \quad (\text{A5.22})$$

where $\mathbf{J}^{(0)}$ is the Jacobian of the right-hand side of the RRE (Eq. A5.2) with respect to the vector of concentrations. $\mathbf{F}^{(0)}$ is the first term in the asymptotic expansion of the drift \mathbf{F} in the nonlinear Fokker-Planck equation (see Eq. 1.39). Using Eqs. 1.34 and 1.40, the $D^{(0)}$'s with two indices as in $D_{i,j}^{(0)}$ can be written as the matrix (Elf and Ehrenberg, 2003)

$$\mathbf{D}^{(0)}(\phi) = \boldsymbol{\nu} \operatorname{diag}(\mathbf{T}_0(\phi)) \boldsymbol{\nu}^T. \quad (\text{A5.23})$$

$\mathbf{D}^{(0)}$ is the first term in the asymptotic expansion of the diffusion tensor \mathbf{D} in the nonlinear Fokker-Planck equation (see Eq. 1.40).

A5.1 The linear-noise approximation

Retaining only Ω^0 terms in Eq. A5.17 yields the *linear Fokker-Planck equation*, also known as the *linear-noise approximation* (LNA), for the fluctuations ϵ :

$$\frac{\partial \Gamma(\epsilon, t)}{\partial t} = \boldsymbol{\nabla}_\epsilon \cdot \left[2^{-1} \mathbf{D}^{(0)}(\phi) \boldsymbol{\nabla}_\epsilon \Gamma(\epsilon, t) - \mathbf{A}(\epsilon) \Gamma(\epsilon, t) \right] + O(\Omega^{-1/2}), \quad (\text{A5.24})$$

where $\boldsymbol{\nabla}_\epsilon = \left[\frac{\partial}{\partial \epsilon_1}, \dots, \frac{\partial}{\partial \epsilon_N} \right]^T$. The drift vector

$$\mathbf{A}(\epsilon) = \mathbf{J}^{(0)}(\phi) \epsilon \quad (\text{A5.25})$$

is hence linear in the fluctuations ϵ . The diffusion matrix $\mathbf{D}^{(0)}(\phi)$ is independent of ϵ . Eq. A5.24 is therefore a special case of the continuous Markov process described by the nonlinear Fokker-Planck equation, namely the *Ornstein-Uhlenbeck process*. The linear Fokker-Planck equation is a linear approximation of the nonlinear Fokker-Planck equation. The equation of motion of $\langle \epsilon \rangle$ according to the LNA is

$$\epsilon(t + dt) = \epsilon(t) + \boldsymbol{\Xi}(\Delta \epsilon \mid dt; \epsilon, t), \quad (\text{A5.26})$$

Appendix

where $\Xi(\Delta\epsilon \mid dt; \epsilon, t) \sim \Pi(\Delta\epsilon \mid dt; \epsilon, t)$ and $\Pi(\Delta\epsilon \mid dt; \epsilon, t)$ is the Gaussian $\mathcal{N}(\mathbf{A}(\epsilon)dt, \mathbf{D}^{(0)}(\phi)dt)$ (Gillespie, 1996a,b).

Multiplying Eq. A5.24 with ϵ and integration over all ϵ 's we get the time-evolution equation of the mean $\langle \epsilon \rangle$:

$$\frac{\partial \langle \epsilon \rangle(t)}{\partial t} = \mathbf{J}^{(0)}(\phi)\langle \epsilon \rangle(t) + O(\Omega^{-1/2}) \quad (\text{A5.27})$$

so that

$$\langle \epsilon \rangle(t) = e^{t\mathbf{J}^{(0)}}\langle \epsilon \rangle_0, \quad (\text{A5.28})$$

where $\langle \epsilon \rangle_0$ is the mean of the initial fluctuation. Eq. A5.27 can also be obtained by linearizing the RRE around a small perturbation $\delta\phi$, as in dynamical systems theory, where the linearized equation is usually evaluated at a fixed point of the RRE to ascertain the stability of the fixed point (see Appendix A10).

Considering that the initial condition of the CME (Eq. A5.4) is the unit impulse function

$$P(\phi^*, t = 0) = \delta(\phi^* - \phi_0), \quad (\text{A5.29})$$

$\phi^*(0)$ is a sure variable and is equal to the initial RRE concentration. Therefore, the initial fluctuation is zero. Consequently,

$$\langle \epsilon \rangle(t = 0) = \langle \epsilon \rangle_0 = \mathbf{0}. \quad (\text{A5.30})$$

The equation of motion for the covariance matrix (same as the matrix of second moments under Eqs. A5.28 and A5.30) $\mathbf{C} = [C_{i,j}] = [\langle \epsilon_i \epsilon_j \rangle]$ is obtained by multiplying Eq. A5.24 with \mathbf{C} and integrating over ϵ :

$$\frac{\partial \mathbf{C}(t)}{\partial t} = \mathbf{J}^{(0)}\mathbf{C} + \mathbf{C}\mathbf{J}^{(0)\top} + \mathbf{D}^{(0)} + O(\Omega^{-1/2}) \quad (\text{A5.31})$$

with initial condition $\mathbf{C}(t = 0) = \mathbf{0}$. The solution of this continuous-time Lyapunov equation is (van Kampen, 2001)

$$\mathbf{C}(t) = \int_0^t e^{(t-t')\mathbf{J}^{(0)}} \mathbf{D}^{(0)} e^{(t-t')\mathbf{J}^{(0)\top}} dt'. \quad (\text{A5.32})$$

Constructing a Gaussian distribution from the mean $\langle \epsilon \rangle$ and the covariance \mathbf{C} , and substituting it back into Eq. A5.24, we find that this Gaussian distribution is indeed the solution of the LNA with no initial fluctuations (van Kampen, 2001). Thus, the solution of the LNA is

$$\Gamma(\epsilon, t) = \mathcal{N}(\mathbf{0}, \mathbf{C}(t)), \quad (\text{A5.33})$$

where the covariance $\mathbf{C}(t)$ is given by Eq. A5.32. Using this result and the van Kampen ansatz (Eq. A5.6) the probability function $P(\phi^*, t)$ is

$$P(\phi^*, t) = \mathcal{N}(\phi, \Omega^{-1}\mathbf{C}(t)). \quad (\text{A5.34})$$

As for the nonlinear Fokker-Planck equation, the Markov propagator II of the linear Fokker-Planck equation (or the LNA) is a Gaussian. The solution of the LNA, however, is also a Gaussian, unlike the solution of the nonlinear Fokker-Planck equation.

Using Eq. A5.34, the mean concentration $\langle \phi^* \rangle$ described by the LNA is

$$\langle \phi^* \rangle(t) = \phi(t). \quad (\text{A5.35})$$

That is, at the level of the LNA, the mean concentration described by the CME is the same as the RRE concentration, for all times. Consequently, the LNA cannot explain any deviation of the mean concentration described by the CME from the RRE concentration.

In summary, the LNA amounts to an Ornstein-Uhlenbeck process. According to the LNA, $\phi^*(t)$ is a Gaussian random variable with the mean equal to the RRE concentration ϕ and the covariance scaling with Ω^{-1} . For linear reaction networks, the mean from the CME is the same as that from the RRE (Sec. 1.4.1). The LNA is therefore a suitable approximation for linear networks. The LNA, however, does not account for the deviations of the mean of ϕ^* from ϕ observed in nonlinear reaction networks (Sec. 1.4.2). To explain that we at least have to consider the $\Omega^{-1/2}$ terms in Eq. A5.17.

A5.2 Beyond the linear-noise approximation: the effective mesoscopic reaction rate equation (EMRE)

Equation A5.17 contains terms beyond the linear-noise approximation. We use this equation to obtain the time-evolution equation for the mean $\langle \epsilon \rangle$ of the fluctuations. Multiplying Eq. A5.17 by ϵ and integration over all ϵ 's we obtain

$$\frac{\partial \langle \epsilon \rangle}{\partial t} = \mathbf{J}^{(0)} \langle \epsilon \rangle + \Omega^{-1/2} \Delta(\mathbf{C}) + O(\Omega^{-1}), \quad (\text{A5.36})$$

where $\mathbf{C} = [C_{i,j}] = \langle \epsilon_i \epsilon_j \rangle$ and the vector $\Delta(\mathbf{C})$ is defined as

$$\Delta_l(\mathbf{C}) = \frac{1}{2} \left(\sum_{w,z=1}^N \frac{\partial J_{l,w}^{(0)}(\phi)}{\partial \phi_z} \langle \epsilon_w \epsilon_z \rangle - \sum_{w=1}^N \phi_w \frac{\partial J_{l,w}^{(0)}(\phi)}{\partial \phi_w} \right). \quad (\text{A5.37})$$

For Eq. A5.36 to have an $O(\Omega^{-1})$ error, we need to estimate \mathbf{C} with an error of at most $O(\Omega^{-1/2})$. This can be done using the LNA. Here we repeat the time-evolution equation of \mathbf{C} from the LNA is (see Eq. A5.31):

$$\frac{\partial \mathbf{C}(t)}{\partial t} = \mathbf{J}^{(0)} \mathbf{C} + \mathbf{C} \mathbf{J}^{(0)\text{T}} + \mathbf{D}^{(0)} + O(\Omega^{-1/2}). \quad (\text{A5.38})$$

Using this time evolution of \mathbf{C} from the LNA is indeed an approximation. Computing the time-evolution equation of \mathbf{C} from Eq. A5.17 would make it depend also on the third moments of the fluctuation ϵ , requiring additional closure for the moment equations. By using the time-evolution equation of \mathbf{C} from the LNA we avoid this problem at the expense of a reduced accuracy.

Appendix

Using Eqs. A5.6 and A5.36, the time evolution equation for the mean concentration $\langle \phi^* \rangle$ is

$$\begin{aligned}\frac{\partial \langle \phi^* \rangle}{\partial t} &= \frac{\partial \phi}{\partial t} + \Omega^{-1/2} \frac{\partial \langle \epsilon \rangle}{\partial t} \\ &= \frac{\partial \phi}{\partial t} + \mathbf{J}^{(0)} (\langle \phi^* \rangle - \phi) + \Omega^{-1} \Delta(\mathbf{C}) + O(\Omega^{-3/2}),\end{aligned}\quad (\text{A5.39})$$

where the time evolution of \mathbf{C} is given by Eq. A5.38 and the vector $\Delta(\mathbf{C})$ is defined in Eq. A5.37. Eq. A5.39, along with Eqs. A5.38 and A5.2 is called the *effective mesoscopic reaction rate equation* (EMRE) (Grima, 2010a). Unlike the LNA, this equation can be used to explain the deviation of the mean concentration predicted by the CME from that of the RRE (Grima, 2009b, 2010a).

A5.2.1 Example 1: Linear reaction network

We apply the EMRE to a monostable linear reaction network. We use the reaction network described in Eq. 1.53:



The mean steady-state concentration from the CME is

$$\langle \phi_1^* \rangle_{\text{ss}}^{\text{CME}} = \frac{k_1}{k_2}. \quad (\text{A5.41})$$

The RRE steady-state concentration is

$$\phi_{1,\text{ss}} = \frac{k_1}{k_2}, \quad (\text{A5.42})$$

confirming that the CME mean concentration is equal to the RRE concentration.

For evaluating the EMRE, we compute the Jacobian $\mathbf{J}^{(0)}$ and the diffusion tensor $\mathbf{D}^{(0)}$. Since the reaction network consists of only a single chemical species, both are scalar. The steady-state expressions for these quantities are (Eqs. A5.22 and A5.23)

$$J_{\text{ss}}^{(0)} = -k_2 \quad (\text{A5.43})$$

and

$$D_{\text{ss}}^{(0)} = k_1 + k_2 \phi_{1,\text{ss}}. \quad (\text{A5.44})$$

Using Eqs. A5.22, A5.23 and A5.38, the covariance matrix (scalar in this case) \mathbf{C} at steady state is

$$C_{\text{ss}} = \frac{k_1 + k_2 \phi_{1,\text{ss}}}{2k_2}. \quad (\text{A5.45})$$

Using Eq. A5.37, the quantity $\Delta(\mathbf{C})$ at steady state is

$$\Delta(\mathbf{C}_{\text{ss}}) = 0. \quad (\text{A5.46})$$

Substituting the above expressions in Eq. A5.39 evaluated at steady-state, we obtain

$$\langle \phi_1^* \rangle_{\text{ss}}^{\text{EMRE}} = \phi_{1,\text{ss}}. \quad (\text{A5.47})$$

This equality arises because $\Delta(\mathbf{C}_{\text{ss}}) = 0$, which is the result of the Jacobian being independent of the RRE concentrations. The Jacobian is independent of the RRE concentrations for all linear reaction networks and thus the mean concentration predicted by the EMRE is the same as the RRE concentration, which in turn is the same as the mean concentration predicted by the CME.

A5.2.2 Example 2: Nonlinear reaction network

We apply the EMRE to a monostable nonlinear reaction network. We use the reaction network described in Eq. 1.59:



in a reactor volume Ω . We test the effectiveness of the EMRE in approximating the steady-state mean concentration predicted by the CME. The mean concentration predicted by the CME is (see Eq. A1.10)

$$\langle \phi_1^* \rangle_{\text{ss}} = \frac{1}{4\Omega} + \phi_{1,\text{ss}} \frac{I'_1(4\phi_{1,\text{ss}}\Omega)}{I_1(4\phi_{1,\text{ss}}\Omega)}, \quad (\text{A5.49})$$

where $I_a(x)$ is the modified Bessel function of the first kind and $I'_a(b) = \frac{dI_a(x)}{dx}|_{x=b}$ denotes its first derivative. The steady-state RRE concentration $\phi_{1,\text{ss}}$ is

$$\phi_{1,\text{ss}} = \sqrt{\frac{k_1}{2k_2}}. \quad (\text{A5.50})$$

For evaluating the EMRE, we compute the Jacobian $\mathbf{J}^{(0)}$ and the diffusion tensor $\mathbf{D}^{(0)}$, which are (Eqs. A5.22 and A5.23)

$$J_{\text{ss}}^{(0)} = -4k_2\phi_{1,\text{ss}} \quad (\text{A5.51})$$

and

$$D_{\text{ss}}^{(0)} = k_1 + 4k_2\phi_{1,\text{ss}}^2. \quad (\text{A5.52})$$

Using Eqs. A5.22, A5.23 and A5.38, the covariance matrix (scalar in this case) \mathbf{C} at steady state is

$$C_{\text{ss}} = \frac{k_1 + 4k_2\phi_{1,\text{ss}}^2}{8k_2\phi_{1,\text{ss}}} = \frac{3k_1}{8k_2\phi_{1,\text{ss}}}. \quad (\text{A5.53})$$

The quantity $\Delta(\mathbf{C})$ at steady state (using Eq. A5.37) is:

$$\Delta(C_{\text{ss}}) = \frac{k_1}{4\phi_{1,\text{ss}}}. \quad (\text{A5.54})$$

Appendix

From Eq. A5.39, the steady-state concentration predicted by the EMRE is

$$\langle \phi_1^* \rangle_{ss}^{\text{EMRE}} = \phi_{1,ss} - \Omega^{-1} \frac{\Delta(C_{ss})}{J_{ss}^{(0)}} = \phi_{1,ss} + \frac{\Omega^{-1}}{8}. \quad (\text{A5.55})$$

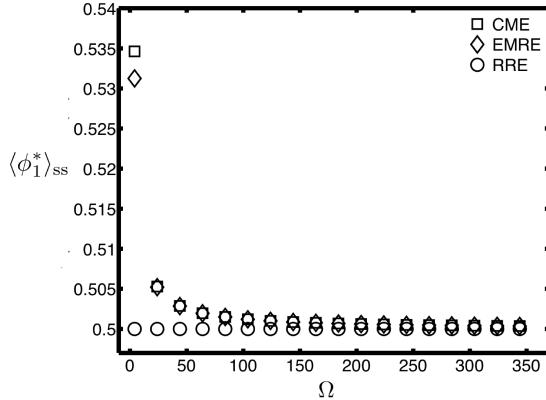


Figure A5.1: CME, EMRE and the RRE predictions of the steady-state mean concentration of species S_1 in the monostable nonlinear reaction network in Eq. A5.48. The predictions are shown as a function of the reactor volume Ω for the rates $k_1 = 5$ and $k_2 = 10$.

Fig. A5.1 shows the steady-state mean concentration predictions as a function of the reactor volume Ω for $k_1 = 5$ and $k_2 = 10$. We see that in contrast to the RRE, the EMRE does show a systematic modulation of the steady-state mean concentration as a function of Ω . At $\Omega = 10$, $\langle \phi_1^* \rangle_{ss}^{\text{CME}} = 0.5130$ and $\langle \phi_1^* \rangle_{ss}^{\text{EMRE}} = 0.5125$, whereas $\phi_{1,ss} = 0.5$. For this reactor volume, the difference between the CME and the EMRE prediction is about 1%, whereas that between CME and the RRE is 2.6%. The EMRE is therefore more accurate than the RRE.

Fig. A5.2 shows the error in the steady-state mean concentration predicted by the EMRE in comparison to the true concentration predicted by the CME. The error is quantified as the absolute difference between the CME and the EMRE predictions using Eqs. A5.49 and A5.55. We see that the error decreases as Ω^{-2} with increasing Ω . This can be proven analytically using the asymptotic expansion of $\langle \phi_1^* \rangle_{ss}^{\text{CME}}$ (see Eq. A1.11). Using Eq. A1.11 and A5.55, the error is given by

$$|\langle \phi_1^* \rangle_{ss}^{\text{CME}} - \langle \phi_1^* \rangle_{ss}^{\text{EMRE}}| = \frac{3\Omega^{-2}}{128\phi_{1,ss}} + O(\Omega^{-3}) \quad (\text{A5.56})$$

and is hence $O(\Omega^{-2})$ for this reaction network.

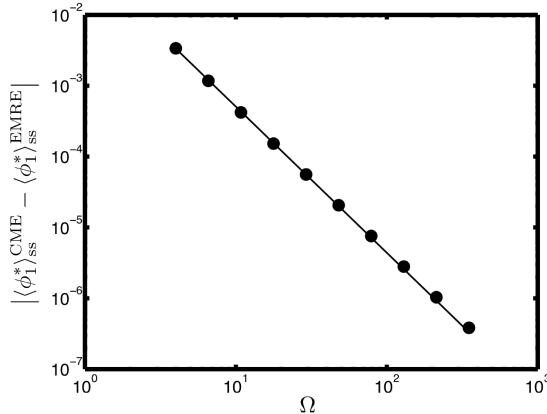


Figure A5.2: Difference between the CME and EMRE predictions of the steady-state mean concentration of species S_1 in the monostable nonlinear reaction network in Eq. A5.48. The error (circles) is plotted as a function of the reactor volume Ω using the EMRE and CME predictions from Eqs. A5.49 and A5.55, respectively. The solid line is the leading error term in Eq. A5.56 obtained by asymptotic expansion (in Ω) of the CME steady-state solution for the mean.

A5.3 Limitations of the van Kampen expansion and the EMRE

The examples in the previous section show that the EMRE, which is obtained using van Kampen system-size expansion, is competitive in predicting the CME steady-state concentration. The van Kampen expansion, and therefore the EMRE, however, also have limitations. All limitations of the van Kampen expansion arise from the ansatz in Eq. A5.6. This ansatz approximates the random jumps around the macroscopic state with Gaussian fluctuations, thereby introducing a small fluctuation variable that can be used to expand the CME. The ansatz stipulates that the variance of the concentration decreases as Ω^{-1} with increasing Ω . This, however, is not true for every chemical reaction network. As seen in Sec. 1.4, only monostable reaction networks obey the scaling of the variance prescribed by the van Kampen ansatz. Reaction networks whose RRE show limit cycle oscillations do not obey this scaling, neither do multistable networks. In such systems, the variance decrease much slower than Ω^{-1} (see Fig. 1.10) and the van Kampen ansatz, and hence the expansion therewith, is an unsubstantiated approximation (van Kampen, 2001). In addition, it has been shown that the van Kampen expansion fails to reproduce qualitative features of systems with absorbing states (McKane and Newman, 2005; Di Patti et al., 2011). This is expected since the fluctuations close to an absorbing state are not symmetric. The Gaussian approximation, therefore, does not necessarily capture certain qualitative features in such systems (McKane and Newman, 2005; Di Patti et al., 2011). In general, for systems for which the van Kampen expansion is not suitable, a generalized ansatz $\phi^* = \phi + \Omega^{-\alpha}\epsilon$, where $0 \leq \alpha < 1$, has been proposed (Di Patti et al., 2011). Such an ansatz can potentially account for non-Gaussian fluctuations around the macroscopic states and has been tested on birth-death processes with mixed success (Di Patti et al., 2011).

Appendix

The EMRE has further limitations owing to the truncation of the van Kampen expansion. The EMRE includes the minimum number of terms needed to explain deviations of the mean concentration predicted by the CME from that of the RRE. This results in a finite truncation error in the prediction of the mean. In addition, at small volumes Ω , where the number of molecules is few, the EMRE can predict negative concentrations. Such unphysical predictions can be avoided by including additional terms of the van Kampen expansion. Including additional terms, however, requires estimating an increasing number of moments of the fluctuations ϵ in order to close the evolution equation for the mean. For instance, if terms of order Ω^{-1} are to be included in Eq. A5.17, the second moment of ϵ needs to be estimated by including all terms up to order Ω^{-1} , the third moment up to order $\Omega^{-1/2}$ and the fourth moment up to order Ω^0 . These equations moreover tend to be coupled, except for special cases. The need to estimate an increasing number of moments makes the derivations laborious. In addition, analytical solutions that help provide a baseline picture may no longer be obtainable. Numerical simulations of such equations could, however, still prove quite useful.

A6 Discreteness-induced concentration inversion in an *in-silico* genetic network

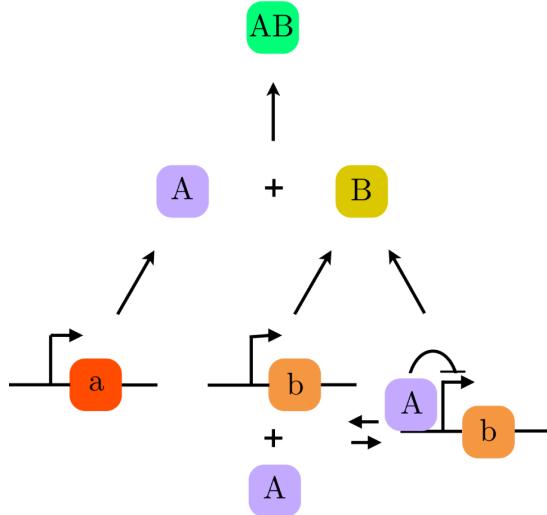
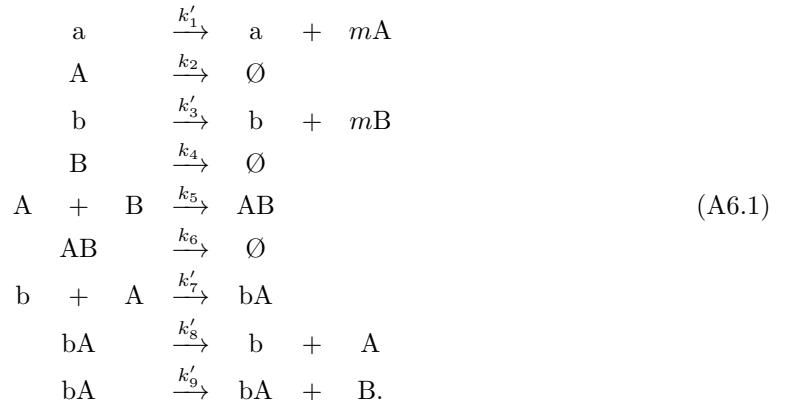


Figure A6.1: Illustration of the genetic reaction network in Eq. A6.1

We present the existence of the concentration inversion effect in an *in silico* genetic network (Francois and Hakim, 2004):



The k 's are the macroscopic reaction rates and the reactions occurs in a reactor of volume Ω . In this network, a and b are genes. The corresponding proteins encoded by these genes are A and B , respectively. The synthesis of these proteins occur in bursts of size m . Proteins A and B form a heterodimer AB . This network models a frequently found motif in gene networks (Francois and Hakim, 2004). Protein A represses gene b at the transcriptional level. Protein B , however, acts through heterodimerization with A , the heterodimer AB being

Appendix

unable to repress gene b. Thus, if A is high, b is repressed, but if B is high, all of the As are heterodimerized, and B remains high (Francois and Hakim, 2004) (see Fig. A6.1 for an illustration).

Note that since there is one gene of a and one gene of b, the concentration of gene a in the rate equation (RE) model is constant and equal to $[a] = 1/\Omega$, whereas the concentration of b and bA are $[b] = x/\Omega$ and $[bA] = y/\Omega$, respectively, where $x + y = 1$. This follows from the fact that gene b exists either in unbound or bound form. The term in the RE model for the rate of production of A due to transcription is $k'_1[a]m = k'_1m/\Omega$; other terms such as those leading to transcription of B and those modeling the reversible interaction of b and A also show this explicit volume dependence. This implies that the solution of the RE model for this system (and indeed for any genetic system) will depend on the volume of the compartment. This dependence stems from the fact that the copy number of genes is fixed, independent of the volume. Taking into account discreteness generates a volume dependence on top of this pre-existing volume dependence. To clearly distinguish the first from the second, we scale the rates such that we eliminate the volume dependence of the REs. This is achieved by setting $k'_1 = k_1\Omega/m$, $k'_3 = k_3\Omega/m$, $k'_7 = k_7\Omega$, $k'_8 = k_8\Omega$ and $k'_9 = k_9\Omega$. Note that here we have additionally scaled some rates by the burst size m in order to eliminate also dependence on the latter; this is convenient, but not essential.

We further set the parameters of this network so that the system is monostable: $k_1 = 2.0$, $k_2 = 0.34$, $k_3 = 0.74$, $k_4 = 0.1$, $k_5 = 0.72$, $k_6 = 0.53$, $k_7 = 2.0$, $k_8 = 2.0$ and $k_9 = 2.0$. We investigate the existence of the concentration inversion effect in the concentration levels of proteins A, B and their heterodimer AB. Initially, the number of molecules of gene a and b are set to 1, and the concentrations of all other chemical species are 0.

A6.1 Comparison of critical volumes from theory and simulation

Using burst size $m = 10$, the time evolution of the average concentrations of proteins A, B and their heterodimer AB predicted by the chemical master equation (CME) are shown in Fig. A6.2 for four different reactor volumes $\Omega = 100, 2.5, 1.4, 0.65$. At large Ω , the concentration predicted by the RE and the average concentration predicted by the CME are in good agreement (Fig. A6.2(a)). The theory using the effective mesoscopic reaction rate equation (EMRE) predicts the largest critical volume as $\Omega_{A,AB} = 3.35$. The simulations indeed show that at $\Omega = 2.5$ the average steady-state concentrations of A and AB are equal (Fig. A6.2(b)). Below this volume the ordering of the average steady-state concentration levels of A and AB is reversed with respect to their concentration levels at large Ω (Figs. A6.2(c) and (d)). The theory further predicts two addition critical volumes $\Omega_{B,AB} = 1.93$ and $\Omega_{A,B} = 0.41$. The values of these critical volumes from stochastic simulations are 1.4 and 0.65, respectively (Figs. A6.2(c) and (d)). Note that all possible concentration inversions are seen for this genetic network.

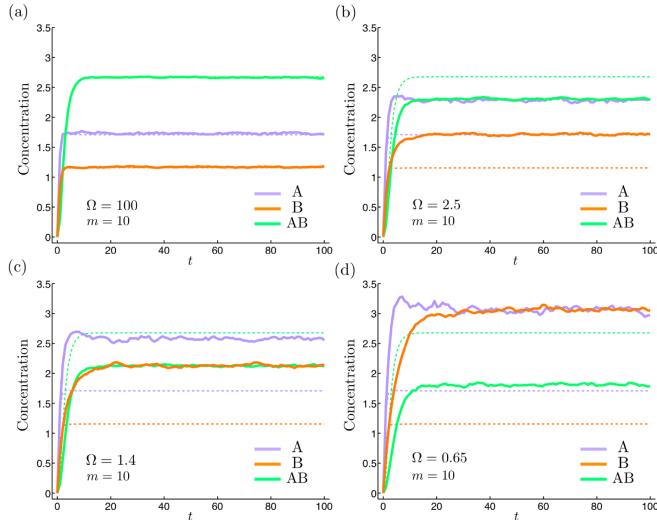


Figure A6.2: Ensemble-averaged concentrations of proteins A, B and their heterodimer AB as a function of time for a genetic reaction network confined in compartments of different volumes. The data (solid lines) are obtained from exact stochastic simulations using PDM SSA (Ramaswamy et al., 2009). The gene network is schematically shown in Fig. A6.1. The compartment volumes are: (a) $\Omega = 100$, (b) $\Omega = 2.5$, (c) $\Omega = 1.4$, (d) $\Omega = 0.65$. The rate constants are fixed to $k'_1 = 2.0\Omega/m$, $k_2 = 0.34$, $k'_3 = 0.74\Omega/m$, $k_4 = 0.1$, $k_5 = 0.72$, $k_6 = 0.53$, $k'_7 = 2.0\Omega$, $k'_8 = 2.0\Omega$, $k'_9 = 2.0\Omega$ and the burst size to $m = 10$. The dashed lines show the RE predictions for the same parameter values. The simulations confirm the theoretical prediction of a discreteness-induced inversion below the largest critical volume, $\Omega_{A,AB} = 3.35$, for species A and AB. Below this largest critical volume the ordering of steady-state concentrations predicted by the RE is incorrect. Note that time, concentration and volumes are in non-dimensional units.

For $m = 1$, the theoretical predictions of the critical volumes are $\Omega_{A,B} = 0.39$, $\Omega_{B,AB} = 0.32$ and $\Omega_{A,AB} = 0.27$. The corresponding values from stochastic simulations are 0.4, 0.28 and 0.18, respectively. For $m = 30$, the theoretical predictions of the critical volumes are $\Omega_{A,AB} = 10.38$, $\Omega_{B,AB} = 5.51$ and $\Omega_{A,B} = 0.43$. The corresponding values from simulations are 8, 4 and 1.2, respectively. From these results we observe that the critical volumes increase with increasing m , as indeed was also the case for the trimerization reaction studied in the main text. In addition, the largest critical volume predicted by the EMRE is the most accurate in comparison to the ones observed from exact stochastic simulations. This volume is also the most important, since it demarcates regions with inversion from regions with no inversion.

Appendix

A7 Effect of volume, burst and the mass-balance condition on the concentration variance

Consider the following chemical reaction



The CME is

$$\frac{\partial P(n_1, t)}{\partial t} = \Omega k_1 (\mathbb{E}_1^{-b} - 1) P + k_2 (\mathbb{E}_1 - 1) (n_1 P), \quad (\text{A7.2})$$

where Ω is the volume of the reactor, $P(n_1, t)$ is the probability distribution for having n_1 molecules of S_1 at time t and the step operator is defined as $\mathbb{E}^r g(n_1) = g(n_1 + r)$.

The deterministic RRE is given by

$$\frac{d\phi_1}{dt} = bk_1 - k_2\phi_1. \quad (\text{A7.3})$$

Multiplying Eq. A7.2 by n_1 and summing over all possible values of n_1 we get the evolution of the mean

$$\frac{d\langle n_1 \rangle}{dt} = \Omega k_1 b - k_2 \langle n_1 \rangle. \quad (\text{A7.4})$$

We obtain the steady-state mean by setting the time derivative to zero

$$\langle n_1 \rangle_{ss} = \frac{\Omega k_1 b}{k_2}. \quad (\text{A7.5})$$

Multiplying Eq. A7.2 by n_1^2 and summing up over all possible values of n_1 we get

$$\frac{d\langle n_1^2 \rangle}{dt} = \Omega k_1 b \left(b + 2\langle n_1 \rangle \right) - k_2 \left(2\langle n_1^2 \rangle - \langle n_1 \rangle \right). \quad (\text{A7.6})$$

By setting the time derivative to zero we see that at steady state

$$\langle n_1^2 \rangle_{ss} - \langle n_1 \rangle_{ss}^2 = \frac{k_1}{2k_2} b(b+1)\Omega, \quad (\text{A7.7})$$

which is the population variance. Hence, the variance of the concentration, $\phi_1^* = n_1/\Omega$, at steady state is

$$\sigma_{ss}^2 = \frac{k_1}{k_2} \frac{b(b+1)}{2\Omega}. \quad (\text{A7.8})$$

Note that $\sigma_{ss}^2 \propto b^2/\Omega$.

Imposing that the average occupied volume fraction $\Phi = v \langle \phi_1^* \rangle_{ss}$ is constant at steady state, where v is the volume of a single molecule of S_1 , leads to the mass-balance condition

$$v b k_1 / k_2 = \Phi.. \quad (\text{A7.9})$$

A7 Effect of volume, burst and the mass-balance condition on the concentration variance

Fixing k_2 , v , and Φ hence fixes the product bk_1 , which appears in the macroscopic RRE (Eq. A7.3). Fixing the product bk_1 renders the RRE invariant to the burst size b .

The condition in Eq. A7.9 leads to the concentration variance

$$\sigma_{ss}^2 = \frac{\phi}{v} \frac{(b+1)}{2\Omega} \quad (\text{A7.10})$$

and to the mean concentration

$$\langle \phi_1^* \rangle_{ss} = \Phi/v. \quad (\text{A7.11})$$

The scaling of the steady-state variance is thus b/Ω .

Having a non-linear reaction in Eq. A7.1 would leave this scaling unchanged at steady state.

Appendix

A8 CME, RRE and the stability of the fixed point of the trimerization system

The CME for the general driven colloidal aggregation reaction system described in Eq. 4.8 is:

$$\begin{aligned} \frac{\partial P(\mathbf{n}, t)}{\partial t} = & (\mathbb{E}_1^{-b} - 1) \Omega k_1^{\text{on}} P(\mathbf{n}, t) \\ & + \sum_{i=1}^{\lfloor N/2 \rfloor} (\mathbb{E}_i^2 \mathbb{E}_{2i}^{-1} - 1) \Omega^{-1} k_{i,i} n_i (n_i - 1) P(\mathbf{n}, t) \\ & + \sum_{i=1}^{\lfloor N/2 \rfloor} \sum_{j=i+1}^{N-i} (\mathbb{E}_i^1 \mathbb{E}_j^1 \mathbb{E}_{i+j}^{-1} - 1) \Omega^{-1} k_{i,j} n_i n_j P(\mathbf{n}, t) \\ & + \sum_{i=1}^N (\mathbb{E}_i^1 - 1) k_i^{\text{off}} n_i P(\mathbf{n}, t), \end{aligned} \quad (\text{A8.1})$$

where $\mathbf{n} = [n_1, \dots, n_N]^T$ is the population vector, N is the maximum size of a multimer and \mathbb{E}_i is the step operator defined as $\mathbb{E}_i^a g(n_i) = g(n_i + a)$. The corresponding RRE is

$$\begin{aligned} \frac{d\phi_i}{dt} = & k_1^{\text{on}} b \delta_{i,1} \\ & + \sum_{p=1}^{\lfloor N/2 \rfloor} \sum_{q=p}^{N-p} k_{p,q} \phi_p \phi_q (\delta_{p+q,i} - \delta_{p,i} - \delta_{q,i}) \\ & - k_i^{\text{off}} \phi_i, \end{aligned} \quad (\text{A8.2})$$

where ϕ_i is the macroscopic concentration of the trimers of size i . In general, the rates $k_{i,j}$'s depend on the sizes of the reacting trimers i and j (van Dongen, 1987; Axford, 1996; Leyvraz, 2003).

For the specific case of trimerization, i.e. $N = 3$, described in Eq. 4.9 the CME is

$$\begin{aligned} \frac{\partial P(\mathbf{n}, t)}{\partial t} = & (\mathbb{E}_1^{-b} - 1) \Omega k_0 P(\mathbf{n}, t) \\ & + (\mathbb{E}_1^2 \mathbb{E}_2^{-1} - 1) \Omega^{-1} k_1 n_1 (n_1 - 1) P(\mathbf{n}, t) \\ & + (\mathbb{E}_1^1 \mathbb{E}_2^1 \mathbb{E}_3^{-1} - 1) \Omega^{-1} 2 k_1 n_1 n_2 P(\mathbf{n}, t) \\ & + \sum_{i=1}^3 (\mathbb{E}_i^1 - 1) k_2 n_i P(\mathbf{n}, t). \end{aligned} \quad (\text{A8.3})$$

The corresponding RRE is

$$\begin{aligned} \frac{d\phi_1}{dt} = & k_0 b - 2k_1 \phi_1^2 - 2k_1 \phi_1 \phi_2 - k_2 \phi_1 \\ \frac{d\phi_2}{dt} = & k_1 \phi_1^2 - 2k_1 \phi_1 \phi_2 - k_2 \phi_2 \\ \frac{d\phi_3}{dt} = & 2k_1 \phi_1 \phi_2 - k_2 \phi_2. \end{aligned} \quad (\text{A8.4})$$

It is easy to see that when $k_0 b$ is constant, the RRE trajectory is invariant to the individual values of k_0 and b . In Chapters 4 and 5 we impose the mass-balance condition (Eq. 4.18) to render the RRE invariant to b . In addition, the RRE is also invariant to the reactor volume Ω .

Setting the time derivative to zero and solving for the steady-state concentrations $(\phi_{1,ss}, \phi_{2,ss}, \phi_{3,ss})$, we find that the trimerization system has a single fixed point. The Jacobian $\mathbf{J}_{ss}^{(0)}$ is

$$\mathbf{J}_{ss}^{(0)} = \begin{bmatrix} -4k_1\phi_{1,ss} - 2k_1\phi_{2,ss} - k_2 & -2k_1\phi_{1,ss} & 0 \\ 2k_1\phi_{1,ss} - 2k_1\phi_{2,ss} & -2k_1\phi_{1,ss} - k_2 & 0 \\ 2k_1\phi_{2,ss} & 2k_1\phi_{1,ss} & -k_2 \end{bmatrix}. \quad (\text{A8.5})$$

The eigenvalues $-\gamma_1$, $-\gamma_2$ and $-\gamma_3$ of $\mathbf{J}_{ss}^{(0)}$ are

$$\begin{aligned} \gamma_1 &= k_2 + 3k_1\phi_{1,ss} + k_1\phi_{2,ss} - k_1\sqrt{-3\phi_{1,ss}^2 + 6\phi_{1,ss}\phi_{2,ss} + \phi_{2,ss}^2}, \\ \gamma_2 &= k_2 + 3k_1\phi_{1,ss} + k_1\phi_{2,ss} + k_1\sqrt{-3\phi_{1,ss}^2 + 6\phi_{1,ss}\phi_{2,ss} + \phi_{2,ss}^2}, \\ \gamma_3 &= k_2. \end{aligned} \quad (\text{A8.6})$$

For some regions in the parameter space of the rates, the eigenvalues $-\gamma_1$ and $-\gamma_2$ are complex. The real parts of all three eigenvalues are, however, always negative throughout the parameter space since the rates and the steady-state concentrations are always non-negative. The fixed point $(\phi_{1,ss}, \phi_{2,ss}, \phi_{3,ss})$ of the trimerization system is, therefore, always stable. In regions of the parameter space where $-\gamma_1$ and $-\gamma_2$ are complex, the fixed point is a stable spiral. The trimerization system is hence monostable.

For the parameter values, $k_1 = 0.5$, $k_2 = 1.0$ and $k_0 = 10k_2/b$, used in Chapter 4 and Chapter 5

$$\begin{aligned} \gamma_1 &= 4.9803 - i 1.0387, \\ \gamma_2 &= 4.9803 + i 1.0387, \\ \gamma_3 &= 1.0. \end{aligned} \quad (\text{A8.7})$$

The real parts of γ_1 and γ_2 are approximately 4.8 times larger than the imaginary parts. In situations where the real part of a complex eigenvalue is a few times larger than the imaginary part, noise is not expected to make the mesoscopic system oscillate (Baxendale and Greenwood, 2011). In addition, fixing k_1 to 0.5 and varying $k_2 \in [0.01, 100]$, we observe that the real parts of γ_1 and γ_2 are always at least 4.3 times larger than the imaginary parts. The minimum ratio between the absolute values of the real and imaginary parts of γ_1 and γ_2 is observed at $k_2 = 3.2551$.

Appendix

A9 Stability of the fixed point and Hopf bifurcation of the Brusselator

The Brusselator in a reactor of volume Ω is given by



Here, X_0 is a buffer whose concentration is fixed to $x_0 = 1$, and the concentrations of species X_1 and Y_1 are x_1 and y_1 respectively. The RRE of the Brusselator is

$$\begin{aligned} \frac{dx_1}{dt} &= k_1 x_0 - k_2 x_1 - k_3 x_1 + k_4 x_1^2 y_1 \\ \frac{dy_1}{dt} &= k_3 x_1 - k_4 x_1^2 y_1. \end{aligned} \quad (\text{A9.2})$$

The fixed point of x_1 and y_1 is

$$\begin{aligned} x_{1,ss} &= \frac{k_1}{k_2} \\ y_{1,ss} &= \frac{k_3 k_2}{k_1 k_4}. \end{aligned} \quad (\text{A9.3})$$

The Jacobian $\mathbf{J}_{ss}^{(0)}$ evaluated at the RRE steady-state is

$$\mathbf{J}_{ss}^{(0)} = \begin{bmatrix} -k_2 - k_3 + 2k_4 x_{1,ss} y_{1,ss} & k_4 x_{1,ss}^2 \\ k_3 - 2k_4 x_{1,ss} y_{1,ss} & -k_4 x_{1,ss}^2 \end{bmatrix}. \quad (\text{A9.4})$$

The eigenvalues λ of $\mathbf{J}_{ss}^{(0)}$ are complex:

$$\lambda = \frac{k_3 - k_{3,c}}{2} \pm i \sqrt{\frac{k_1^2 k_4}{k_2} - \frac{k_3 - k_{3,c}}{2}}, \quad (\text{A9.5})$$

where

$$k_{3,c} = k_2 + \frac{k_1^2 k_4}{k_2}. \quad (\text{A9.6})$$

When $k_3 < k_{3,c}$, the real part of λ is negative and hence the fixed point of the Brusselator is stable. When $k_3 = k_{3,c}$, λ is purely imaginary and the Brusselator undergoes a Hopf bifurcation. For $k_3 > k_{3,c}$, the real part of λ is positive and the fixed point is unstable. However, a stable limit cycle exists (Gaspard, 2002).

A9 Stability of the fixed point and Hopf bifurcation of the Brusselator

For the parameters $k_1 = k_2 = k_3 = k_4 = 1$ that we use in Sec. 5.2

$$\lambda = -\frac{1}{2} \pm i \frac{\sqrt{3}}{2} \quad (\text{A9.7})$$

and hence the absolute value of the real part of λ is less than that of the imaginary part. It has been observed that under such conditions a mesoscopic system described by the CME can show oscillations (Qian et al., 2002; Baxendale and Greenwood, 2011). It can be verified that for every stage in the cascaded Brusselator system (Eq. 5.9), the steady state and stability of the fixed point are identical to those of a single stage (Shibata, 2004).

A10 Normalized power spectral density of a cascade of Brusselators at large volume

We analytically derive the normalized steady-state PSD $S_n(\omega)$ and its peak frequency ω_n^m for each cascade stage n in the large-volume regime. The derivation follows that of Shibata (Shibata, 2004).

The deterministic RRE for the reaction system given in Eq. 5.9 are

$$\begin{aligned}\frac{dx_1}{dt} &= k_1x_0 - k_3x_1 - k_2x_1 + k_4x_1^2y_1 \\ \frac{dx_n}{dt} &= k_2x_{n-1} - k_3x_n - k_2x_n + k_4x_n^2y_n \quad n = 2, \dots, N \\ \frac{dy_n}{dt} &= k_3x_n - k_4x_n^2y_n \quad n = 1, \dots, N,\end{aligned}\tag{A10.1}$$

where x_n and y_n are the concentrations of species X_n and Y_n , respectively. As in the numerical study for mesoscopic volumes, we set $k_2 = k_4 = x_0 = 1$. Let δx_n and δy_n be small perturbations around the steady state $(x_{n,ss}, y_{n,ss}) = (k_1, k_3/k_1)$. The linear equations for the perturbations around the steady state are

$$\begin{pmatrix} \frac{d\delta x_n}{dt} \\ \frac{d\delta y_n}{dt} \end{pmatrix} = \begin{pmatrix} k_3 - 1 & k_1^2 \\ -k_3 & -k_1^2 \end{pmatrix} \begin{pmatrix} \delta x_n \\ \delta y_n \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \delta x_{n-1} \\ \delta y_{n-1} \end{pmatrix}.\tag{A10.2}$$

This expression for the perturbations around the fixed point of the RRE is the same as the one that would be obtained from the linear-noise approximation using van Kampen expansion.

Taking the Fourier transform on both sides, we obtain

$$\begin{pmatrix} j\omega & 0 \\ 0 & j\omega \end{pmatrix} \begin{pmatrix} \mathcal{X}_n \\ \mathcal{Y}_n \end{pmatrix} = \begin{pmatrix} k_3 - 1 & k_1^2 \\ -k_3 & -k_1^2 \end{pmatrix} \begin{pmatrix} \mathcal{X}_n \\ \mathcal{Y}_n \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \mathcal{X}_{n-1} \\ \mathcal{Y}_{n-1} \end{pmatrix},\tag{A10.3}$$

where \mathcal{X}_n and \mathcal{Y}_n are the Fourier transforms of $x_n(t)$ and $y_n(t)$, respectively. Simplifying the above equation leads to

$$\mathcal{X}_n = F(\omega)\mathcal{X}_{n-1},\tag{A10.4}$$

where

$$F(\omega) = \frac{j\omega + k_1^2}{-\omega^2 + j(k_1^2 - k_3 + 1)\omega + k_1^2}.\tag{A10.5}$$

Since we consider a linearized version of the macroscopic system, the normalized PSD is invariant to the input noise excitation used to quantify the correlations introduced by the system. We here use the simple input noise $\delta x_0 = \epsilon_0(t)$, which is uncorrelated white noise with $\langle \epsilon_0(t) \rangle = 0$ and $\langle \epsilon_0(t)\epsilon_0(t+\tau) \rangle = \sigma_0^2\delta(\tau)$, where $\delta(\tau)$ is the unit impulse function. The PSD $P_0(\omega)$ of the input buffer to the first stage is then given by

$$\begin{aligned}P_0(\omega) &= \langle |\mathcal{X}_0(\omega)|^2 \rangle \\ &= \frac{\sigma_0^2}{2\pi}.\end{aligned}\tag{A10.6}$$

The PSD $P_n(\omega)$ of the output of cascade stage $n \geq 1$ is given by

$$P_n(\omega) = \langle |\mathcal{X}_n(\omega)|^2 \rangle. \quad (\text{A10.7})$$

Substituting Eqs. A10.4 and A10.6 into Eq. A10.7 we get

$$P_n(\omega) = |F(\omega)|^{2n} \frac{\sigma_0^2}{2\pi}.$$

The normalized steady-state PSD $S_n(\omega)$ is then

$$\begin{aligned} S_n(\omega) &= \frac{P_n(\omega)}{\int_0^\infty P_n(\omega') d\omega'} \\ &= \frac{|F(\omega)|^{2n}}{\int_0^\infty |F(\omega')|^{2n} d\omega'}. \end{aligned} \quad (\text{A10.8})$$

Given the expression for $F(\omega)$ (Eq. A10.5), the normalized steady-state PSD $S_n(\omega)$ has a Lorentzian-like form with peak frequency

$$\omega_n^m = \sqrt{-k_1^4 + k_1^2 \sqrt{2k_1^2 k_3 + 2k_3 - k_3^2}} \quad \forall n. \quad (\text{A10.9})$$

It is therefore evident that in the large-volume regime the peak frequency ω_n^m is independent of Ω and of the stage number n . The bandwidth ω_n^b in the large-volume limit is computed from Eq. A10.8.

Bibliography

- Akerfelt, M., R. I. Morimoto, and L. Sistonen. 2010. Heat shock factors: integrators of cell stress, development and lifespan. *Nature Rev. Mol. Cell Biol.* **11**:545–555.
- Albert, R. 2005. Scale-free networks in cell biology. *J. Cell Sci.* **118**:4947–4957.
- Albert, R., and A.-L. Barabási. 2002. Statistical mechanics of complex networks. *Rev. Mod. Phys.* **74**:47–97.
- Alberts, B., D. Bray, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. 1997. *Essential Cell Biology*. New York: Garland Publication, Inc.
- Alberts, B., A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. 1994. *Molecular biology of the cell*. Garland Science.
- Altschuler, S. J., S. B. Angenent, Y. Wang, and L. F. Wu. 2008. On the spontaneous emergence of cell polarity. *Nature* **454**:886–889.
- Anderson, D. F., A. Ganguly, and T. G. Kurtz. 2011. Error Analysis of tau-leap simulation methods. *Ann. Appl. Prob.* .
- Andrews, S. S., and D. Bray. 2004. Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.* **1**:137–151.
- Andrieu, C., and J. Thoms. 2008. A tutorial on adaptive MCMC. *Statistics and Computing* **18**:343–373.
- Angeli, D., J. E. Ferrell Jr., and E. D. Sontag. 2004. Detection of multistability, bifurcations, and hysteresis in a large class of biological positive-feedback systems. *Proc. Natl. Acad. Sci. USA* **101**:1822–1827.
- Anzenbacher, P., and M. A. Palacios. 2009. Polymer nanofibre junctions of attolitre volume serve as zeptomole-scale chemical reactors. *Nature Chem.* **1**:80–86.
- Arkin, A., J. Ross, and H. H. McAdams. 1998. Stochastic kinetic analysis of developmental pathway bifurcation in phage λ -infected *Escherichia coli* cells. *Genetics* **149**:1633–1648.
- Auger, A., P. Chatelain, and P. Koumoutsakos. 2006. *R*-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *J. Chem. Phys.* **125**:084103.
- Axford, S. D. T. 1996. Theoretical calculations on Smoluchowski kinetics: Perikinetic reactions in highly aggregated systems. *Proc. R. Soc. Lond. A* **452**:2355–2368.

Bibliography

- Bar-Even, A., J. Paulsson, N. Maheshri, M. Carmi, E. O'Shea, Y. Pilpel, and N. Barkai. 2006. Noise in protein expression scales with natural protein abundance. *Nat. Genet.* **38**:636–643.
- Barabási, A. L., and Z. N. Oltvai. 2004. Network biology: understanding the cell's functional organization. *Nat. Rev. Genet.* **5**:101–113.
- Barkai, N., and S. Leibler. 2000. Biological rhythms - Circadian clocks limited by noise. *Nature* **403**:267–268.
- Barlett, M. S. 1953. Stochastic Processes or the Statistics of Change. *J. R. Stat. Soc. C* **2**:44–64.
- Barrio, M., K. Burrage, A. Leier, and T. Tian. 2006. Oscillatory regulation of Hes1: Discrete stochastic delay modelling and simulation. *PLoS Comput. Biol.* **2**:1017–1030.
- Baxendale, P. H., and P. E. Greenwood. 2011. Sustained oscillations for density dependent Markov processes. *J. Math. Biol.* **63**:433–457.
- Bayati, B., P. Chatelain, and P. Koumoutsakos. 2009. D-leaping: Accelerating stochastic simulation algorithms for reactions with delays. *J. Comput. Phys.* **228**:5908–5916.
- . 2011. Adaptive mesh refinement for stochastic reaction–diffusion processes. *J. Chem. Phys.* **230**:13–26.
- Beard, D. A., and H. Qian. 2007. Relationship between Thermodynamic Driving Force and One-Way Fluxes in Reversible Processes. *PLoS ONE* **2**:e144.
- Bedeaux, D. 1977. Equivalence of the master equation and the Langevin equation. *Phys. Lett. A* **62**:10–12.
- Belousov, B. P. 1959. A periodic reaction and its mechanism. In: Abstracts on Radiation Medicine. Medl. Publ.
- . 1985. A periodic reaction and its mechanism. In: Field, R. J., and M. Burger, editors, *Oscillations and traveling waves in chemical systems*. Wiley, New York.
- Benguria, R. D., and M. C. Depassier. 1996. Speed of Fronts of the Reaction-Diffusion Equation. *Phys. Rev. Lett.* **77**:1171–1173.
- Berg, O. G., J. Paulsson, and M. Ehrenberg. 2000. Fluctuations and quality of control in biological cells: Zero-order ultrasensitivity reinvestigated. *Biophys. J.* **79**:1228–1236.
- Bergdorf, M., I. F. Sbalzarini, and P. Koumoutsakos. 2010. A Lagrangian particle method for reaction-diffusion systems on deforming surfaces. *J. Math. Biol.* **61**:649–663.
- Bortz, A. B., M. H. Kalos, and J. L. Lebowitz. 1975. New algorithm for Monte Carlo simulation of Ising spin systems. *J. Comput. Phys.* **17**:10–18.
- Boys, R. J., D. J. Wilkinson, and T. B. L. Kirkwood. 2008. Bayesian inference for a discretely observed stochastic kinetic model. *Statistics and Computing* **18**:125–135.

- Bratsun, D., D. Volfson, L. S. Tsimring, and J. Hasty. 2005. Delay-induced stochastic oscillations in gene regulation. *Proc. Natl. Acad. Sci. USA* **102**:14593–14598.
- Bruggeman, F., N. Blüthgen, and H. Westerhoff. 2009. Noise management by molecular networks. *PLoS Comp. Biol.* **5**:e1000506.
- Cai, L., C. Dalal, and M. Elowitz. 2008. Frequency-modulated nuclear localization bursts coordinate gene regulation. *Nature* **455**:485–490.
- Cai, L., N. Friedman, and X. S. Xie. 2006. Stochastic protein expression in individual cells at the single molecule level. *Nature* **440**:358–362.
- Cai, X. 2007. Exact stochastic simulation of coupled chemical reactions with delays. *J. Chem. Phys.* **126**:124108.
- Cai, X., and Z. Xu. 2007. K -leap method for accelerating stochastic simulation of coupled chemical reactions. *J. Chem. Phys.* **126**:074102.
- Cao, Y., D. T. Gillespie, and L. R. Petzold. 2005a. Avoiding negative populations in explicit Poisson tau-leaping. *J. Chem. Phys.* **123**:054104.
- . 2005b. The slow-scale stochastic simulation algorithm. *J. Chem. Phys.* **122**:014116.
- . 2006. Efficient step size selection for the tau-leaping simulation method. *J. Chem. Phys.* **124**:044109.
- Cao, Y., H. Li, and L. Petzold. 2004. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *J. Chem. Phys.* **121**:4059–4067.
- Cardinale, J., A. Rauch, Y. Barral, G. Székely, and I. F. Sbalzarini. 2009. Bayesian image analysis with on-line confidence estimates and its application to microtubule tracking. In: *Proc. IEEE Intl. Symposium Biomedical Imaging (ISBI)*. Boston, USA: IEEE, 1091–1094.
- Carrier, T., and J. Keasling. 1999. Investigating autocatalytic gene expression systems through mechanistic modeling. *J. Theor. Biol.* **201**:25–36.
- Chandra, F. A., G. Buzi, and J. C. Doyle. 2011. Glycolytic oscillations and limits on robust efficiency. *Science* **333**:187–192.
- Chandrasekhar, S. 1943. Stochastic problems in physics and astronomy. *Rev. Mod. Phys.* **15**:1–89.
- Chaplain, M. A. J., M. Ganesh, and I. G. Graham. 2001. Spatio-temporal pattern formation on spherical surfaces: numerical simulation and application to solid tumour growth. *J. Math. Biol.* **42**:387–423.
- Chapman, S. 1928. On the Brownian displacements and thermal diffusion of grains suspended in a non-uniform fluid. *Proc. R. Soc. Lond. A* **119**:34–53.
- Choudhary, C., and M. Mann. 2010. Decoding signalling networks by mass spectrometry-based proteomics. *Nature Rev. Mol. Cell Biol.* **11**:427–439.

Bibliography

- Cinquemani, E., A. Milias-Argeitis, S. Summers, and J. Lygeros. 2008. Stochastic dynamics of genetic networks: modelling and parameter identification. *Bioinformatics* **24**:2748–2754.
- Cohen, D., and J. Neu. 1979. Interacting oscillatory chemical reactors. *Annals of the New York Academy of Sciences* **316**:332–337.
- Devroye, L. 1986. Non-uniform random variate generation. Springer-Verlag New York.
- Di Patti, F., S. Azale, J. R. Banavar, and A. Maritan. 2011. System size expansion for systems with an absorbing state. *Phys. Rev. E* **83**:010102.
- Dobrzański, M., and F. J. Bruggeman. 2009. Elongation dynamics shape bursty transcription and translation. *Proc. Natl. Acad. Sci. USA* **106**:2583–2588.
- Doob, J. L. 1942. Topics in the theory of Markoff chains. *Trans. Amer. Math. Soc.* **52**:37–64.
- . 1945. Markoff chains – Denumerable case. *Trans. Amer. Math. Soc.* **58**:455–473.
- Eldar, A., and M. B. Elowitz. 2010. Functional roles for noise in genetic circuits. *Nature* **467**:167–173.
- Elf, J., and M. Ehrenberg. 2003. Fast Evaluation of Fluctuations in Biochemical Networks With the Linear Noise Approximation. *Genome Research* **13**:2475–2484.
- . 2004. Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases. *Systems Biology* **1**:230–236.
- Elf, J., J. Paulsson, O. G. Berg, and M. Ehrenberg. 2003. Near-critical phenomena in intracellular metabolite pools. *Biophys. J.* **84**:154–170.
- Elowitz, M. B., and S. Leibler. 2000. A synthetic oscillatory network of transcriptional regulators. *Nature* **403**:335–338.
- Elson, E. L. 1974. Fluorescence correlation spectroscopy. I. Conceptual basis and theory. *Biopolymers* **13**:1–27.
- Engblom, S. 2006. Computing the moments of high dimensional solutions of the master equation. *Appl. Math. Comput.* **180**:498–515.
- Engblom, S., L. Ferm, A. Hellander, and P. Lötstedt. 2009. Simulation of Stochastic Reaction-Diffusion Processes on Unstructured Meshes. *SIAM J. Sci. Comput.* **31**:1774–1797.
- Engl, H. W., C. Flamm, P. Kuegler, J. Lu, S. Mueller, and P. Schuster. 2009. Inverse problems in systems biology. *Inverse Problems* **25**:123014.
- Erban, R., and S. J. Chapman. 2007. Reactive boundary conditions for stochastic simulations of reaction-diffusion processes. *Phys. Biol.* **4**:16–28.
- . 2009. Stochastic modelling of reaction-diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.* **6**:046001.

- Fange, D., O. G. Berg, P. Sjoberg, and J. Elf. 2010. Stochastic reaction-diffusion kinetics in the microscopic limit. *Proc. Natl. Acad. Sci. USA* **107**:19820–19825.
- Fange, D., and J. Elf. 2006. Noise-induced Min phenotypes in *E. coli*. *PLoS Comput. Biol.* **2**:637–648.
- Farrow, L. A., and D. Edelson. 1974. The steady-state approximation: fact or fiction? *Int. J. Chem. Kinet.* **6**:787–800.
- Feller, W. 1940. On the integro-differential equations of purely discontinuous Markoff processes. *Trans. Amer. Math. Soc.* **48**:488–515.
- . 1957. On Boundaries and Lateral Conditions for the Kolmogorov Differential Equations. *Ann. Math.* **65**:527–570.
- Ferm, L., A. Hellander, and P. Lötstedt. 2010. An adaptive algorithm for simulation of stochastic reaction-diffusion processes. *J. Comput. Phys.* **229**:343–360.
- Fersht, A. 1998. Structure and mechanism in protein science. WH Freeman.
- Fisher, R. A. 1937. The wave of advance of advantageous genes. *Ann. Eugenics* **7**:355–369.
- Flach, E. H., and S. Schnell. 2006. Use and abuse of the quasi-steady-state approximation. *Systems Biology* **153**:187–191.
- Fox, B. L. 1990. Generating Markov-Chain Transitions Quickly: I. *ORSA J. Comput.* **2**:126–135.
- Francois, P., and V. Hakim. 2004. Design of genetic networks with specified functions by evolution *in silico*. *Proc. Natl. Acad. Sci. USA* **101**:580–585.
- Gardiner, C. W., K. J. McNeil, D. F. Walls, and I. S. Matheson. 1976. Correlations in stochastic theories of chemical-reactions. *J. Stat. Phys.* **14**:307–331.
- Gardner, T. S., C. R. Cantor, and J. J. Collins. 2000. Construction of a genetic toggle switch in *Escherichia coli*. *Nature* **403**:339–342.
- Gaspard, P. 2002. The correlation time of mesoscopic chemical clocks. *J. Chem. Phys.* **117**:8905–8916.
- Gazzola, M., C. J. Burckhardt, B. Bayati, M. Engelke, U. F. Greber, and P. Koumoutsakos. 2009. A Stochastic Model for Microtubule Motors Describes the In Vivo Cytoplasmic Transport of Human Adenovirus. *PLoS Comput. Biol.* **5**:e1000623.
- Gazzola, M., O. V. Vasilyev, and P. Koumoutsakos. 2011. Shape optimization for drag reduction in linked bodies using evolution strategies. *Computers and Structures* **89**:1224–1231.
- Ghaemmaghami, S., W.-K. Huh, K. Bower, R. W. Howson, A. Belle, N. Dephoure, E. K. O’Shea, and J. S. Weissman. 2003. Global analysis of protein expression in yeast. *Nature* **425**:737–741.

Bibliography

- Gibson, M. A., and J. Bruck. 2000. Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels. *J. Phys. Chem. A* **104**:1876–1889.
- Gierer, A., and H. Meinhardt. 1972. A theory of biological pattern formation. *Kybernetik* **12**:30–39.
- Gillespie, D. T. 1976. A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions. *J. Comput. Phys.* **22**:403–434.
- . 1977. Exact Stochastic Simulation of Coupled Chemical Reactions. *J. Phys. Chem.* **81**:2340–2361.
- . 1991. Markov Processes: An Introduction for Physical Scientists. Academic Press.
- . 1992. A rigorous derivation of the chemical master equation. *Physica A* **188**:404–425.
- . 1996a. The mathematics of Brownian motion and Johnson noise. *Am. J. Phys.* **64**:225–240.
- . 1996b. The multivariate Langevin and Fokker-Planck equations. *Am. J. Phys.* **64**:1246–1257.
- . 2000. The chemical Langevin equation. *J. Chem. Phys.* **113**:297–306.
- . 2001. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.* **115**:1716–1733.
- . 2007. Stochastic simulation of chemical kinetics. *Ann. Rev. Phys. Chem.* **58**:35–55.
- . 2009. A diffusional bimolecular propensity function. *J. Chem. Phys.* **131**:164109.
- Goldbeter, A. 1975. Mechanism for oscillatory synthesis of cyclic AMP in Dictyostelium discoideum. *Nature* **253**:540–542.
- Goodsell, D. S., and A. J. Olson. 2000. Structural symmetry and protein function. *Annu. Rev. Biophys. Biomol. Struct.* **29**:105–153.
- Gray, P., and S. K. Scott. 1983. Autocatalytic reactions in the isothermal, continuous stirred tank reactor - Isolas and other forms multistability. *Chem. Eng. Sci.* **38**:29–43.
- . 1984. Autocatalytic reactions in the isothermal, continuous stirred tank reactor : Oscillations and instabilities in the system $A + 2B \rightarrow 3B; B \rightarrow C$. *Chem. Eng. Sci.* **39**:1087–1097.
- . 1985. Sustained oscillations and other exotic patterns of behavior in isothermal reactions. *J. Phys. Chem.* **89**:22–32.
- Grima, R. 2009a. Investigating the robustness of the classical enzyme kinetic equations in small intracellular compartments. *BMC Sys. Biol.* **3**:101.
- . 2009b. Noise-Induced Breakdown of the Michaelis-Menten Equation in Steady-State Conditions. *Phys. Rev. Lett.* **102**:218103.

- _____. 2010a. An effective rate equation approach to reaction kinetics in small volumes: Theory and application to biochemical reactions in nonequilibrium steady-state conditions. *J. Chem. Phys.* **133**:035101.
- _____. 2010b. Intrinsic biochemical noise in crowded intracellular conditions. *J. Chem. Phys.* **132**:185102.
- Grima, R., and S. Schnell. 2008. Modelling reaction kinetics inside cells. *Essays Biochem.* **45**:41–56.
- Grönlund, A., P. Lötstedt, and J. Elf. 2010. Costs and constraints from time-delayed feedback in small gene regulatory motifs. *Proc. Natl. Acad. Sci. USA* **107**:8171–8176.
- Gunaratne, G. H., Q. Ouyang, and H. L. Swinney. 1994. Pattern formation in the presence of symmetries. *Phys. Rev. E* **50**:2802—2820.
- Hafner, M., H. Koepll, M. Hasler, and A. Wagner. 2009. ‘Glocal’ Robustness Analysis and Model Discrimination for Circadian Oscillators. *PLoS Comput. Biol.* **5**:e1000534.
- Hagerup, T., K. Mehlhorn, and I. Munro. 1993. Optimal Algorithms for Generating Time-Varying Discrete Random Variates. In: *Lect. Notes Comput. Sci.*, volume 700. Springer, New York, 253–264.
- Haken, H. 1975. Cooperative phenomena in systems far from thermal equilibrium and in nonphysical systems. *Rev. Mod. Phys.* **47**:67–121.
- Hansen, N., S. D. Muller, and P. Koumoutsakos. 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evol. Comput.* **11**:1–18.
- Hansen, N., A. S. P. Niederberger, L. Guzzella, and P. Koumoutsakos. 2009. A method for handling uncertainty in evolutionary optimization with an application to feedback control of combustion. In: *Proc. IEEE Transactions on Evolutionary Computation*, volume 13. 180–197.
- Hansen, N., and A. Ostermeier. 2001. Completely derandomized self-adaptation in evolution strategies. *Evol. Comput.* **9**:159–195.
- Hattne, J. 2006. The Algorithms and Implementation of MesoRD. Master thesis, Department of Information Technology, Uppsala University.
- Hattne, J., D. Fange, and J. Elf. 2005. Stochastic reaction-diffusion simulation with MesoRD. *Bioinformatics* **21**:2923–2924.
- Hayot, F., and C. Jayaprakash. 2004. The linear noise approximation for molecular fluctuations within cells. *Phys. Biol.* **1**:205–210.
- Hellander, S., and P. Lotstedt. 2011. Flexible single molecule simulation of reaction-diffusion processes. *J. Comput. Phys.* **230**:3948–3965.

Bibliography

- Helmut, J. A., C. J. Burckhardt, U. F. Greber, and I. F. Sbalzarini. 2009. Shape reconstruction of subcellular structures from live cell fluorescence microscopy images. *J. Struct. Biol.* **167**:1–10.
- Helmut, J. A., S. Reboux, and I. F. Sbalzarini. 2011. Exact stochastic simulations of intracellular transport by mechanically coupled molecular motors. *J. Comput. Sci.* **2**:324–334.
- Helmut, J. A., and I. F. Sbalzarini. 2009. Deconvolving active contours for fluorescence microscopy images. In: Proc. Intl. Symp. Visual Computing (ISVC), volume 5875 of *Lecture Notes in Computer Science*. Las Vegas, USA: Springer, 544–553.
- Henderson, I. R., P. Owen, and J. P. Nataro. 1999. Molecular switches — the ON and OFF of bacterial phase variation. *Mol. Microbiol.* **33**:919–932.
- Hess, B. 1979. The glycolytic oscillator. *J. Exp. Biol.* **81**:7–14.
- Heuett, W. J., and H. Qian. 2006. Grand canonical Markov model: a stochastic theory for open nonequilibrium biochemical networks. *J. Chem. Phys.* **124**:044110.
- Hilfinger, A., and J. Paulsson. 2011. Separating intrinsic from extrinsic fluctuations in dynamic biological systems. *Proc. Natl. Acad. Sci. USA* **108**:12167–12172.
- Hockney, R. W., and J. W. Eastwood. 1988. Computer Simulation using Particles. Institute of Physics Publishing.
- Holden, A. V. 1986. Chaos. Princeton University Press.
- Holloway, D. M., F. J. P. Lopes, L. d. F. Costa, B. A. N. Travencolo, N. Golyandina, K. Usevich, and A. V. Spirov. 2011. Gene Expression Noise in Spatial Patterning: hunchback Promoter Structure Affects Noise Amplitude and Distribution in *Drosophila* Segmentation. *PLoS Comput. Biol.* **7**:e1001069.
- Horsthemke, W., and L. Brenig. 1977. Non-linear Fokker-Planck equation as an asymptotic representation of the master equation. *Z. Physik B* **27**:341–348.
- Hu, Z., and J. Lutkenhaus. 1999. Topological regulation of cell division in *Escherichia coli* involves rapid pole to pole oscillation of the division inhibitor MinC under the control of MinD and MinE. *Mol. Microbiol.* **34**:82–90.
- Hucka, M., A. Finney, H. M. Sauro, et al. 2003. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19**:524–531.
- Indurkha, S., and J. Beal. 2010. Reaction Factoring and Bipartite Update Graphs Accelerate the Gillespie Algorithm for Large-Scale Biochemical Systems. *Plos One* **5**:e8125.
- Ishihama, Y., T. Schmidt, J. Rappaport, M. Mann, F. Hartl, M. Kerner, and D. Frishman. 2008. Protein abundance profiling of the *Escherichia coli* cytosol. *BMC genomics* **9**:102.
- Iyengar, K. A., L. A. Harris, and P. Clancy. 2010. Accurate implementation of leaping in space: The spatial partitioned-leaping algorithm. *J. Chem. Phys.* **132**:094101.

Bibliography

- Jaynes, E. T. 1957. Information Theory and Statistical Mechanics. *Phys. Rev.* **106**:620–630.
- Jeong, H., B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. 2000. The large-scale organization of metabolic networks. *Nature* **407**:651–654.
- Jeschke, M., R. Ewald, and A. M. Uhrmacher. 2011. Exploring the performance of spatial stochastic simulation algorithms. *J. Comp. Phys.* **230**:2562–2574.
- Jin, Y., and J. Branke. 2005. Evolutionary optimization in uncertain environments-A survey. *IEEE Trans. Evol. Comput.* **9**:303–317.
- Karlsson, M., M. Davidson, R. Karlsson, et al. 2004. Biomimetic nanoscale reactors and networks. *Annu. Rev. Phys. Chem.* **55**:613–649.
- Kashiwagi, A., I. Urabe, K. Kaneko, and T. Yomo. 2006. Adaptive Response of a Gene Network to Environmental Changes by Fitness-Induced Attractor Selection. *PLoS ONE* **1**:e49.
- Kendall, D. G. 1949. Stochastic Processes and Population Growth. *J. R. Stat. Soc. B* **11**:230–264.
- Kern, S., and P. Koumoutsakos. 2006. Simulations of optimized anguilliform swimming. *J. Exp. Biol.* **209**:4841–4857.
- Kholodenko, B. N. 2000. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *Eur. J. Biochem.* **267**:1583–1588.
- Kholodenko, B. N., and H. V. Westerhoff. 1995. The macroworld versus the microworld of biochemical regulation and control. *Trends Biochem. Sci.* **20**:52–54.
- Kitano, H. 2002a. Computational systems biology. *Nature* **420**:206–210.
- . 2002b. Systems biology: A brief overview. *Science* **295**:1662–1664.
- Kjellström, G. 1969. Network Optimization by Random Variation of Component Values. *Ericsson Technics* **25**:133–151.
- . 1991. On the Efficiency of Gaussian Adaptation. *J. Optim. Theory Appl.* **71**:589–597.
- Kjellström, G., and L. Taxen. 1981. Stochastic Optimization in System Design. *IEEE Trans. Circ. and Syst.* **28**:702–715.
- Kjellström, G., and L. Taxen. 1992. Gaussian Adaptation, an evolution-based efficient global optimizer. In: *Comp. Appl. Math.* Elsevier Science, 267–276.
- Klemm, J. D., S. L. Schreiber, and G. R. Crabtree. 1998. Dimerization as a regulatory mechanism in signal transduction. *Annu. Rev. Immunol.* **16**:569–592.
- Ko, C. H., Y. R. Yamada, D. K. Welsh, E. D. Buhr, A. C. Liu, E. E. Zhang, M. R. Ralph, S. A. Kay, D. B. Forger, and J. S. Takahashi. 2010. Emergence of Noise-Induced Oscillations in the Central Circadian Pacemaker. *PLoS Biol.* **8**:e1000513.

Bibliography

- Koch, A., and H. Meinhardt. 1994. Biological pattern formation: from basic mechanisms to complex structures. *Rev. Mod. Phys.* **66**:1481–1507.
- Koeppl, H., G. Setti, S. Pelet, M. Mangia, T. Petrov, and M. Peter. 2010. Probability metrics to calibrate stochastic chemical kinetics. In: Proc. IEEE Intl. Symp. Circuits and Systems. Paris, France, 541–544.
- Koh, W., and K. T. Blackwell. 2011. An accelerated algorithm for discrete stochastic simulation of reaction-diffusion systems using gradient-based diffusion and tau-leaping. *J. Chem. Phys.* **134**:154103.
- Kolmogorov, A. 1931. Über die analytischen Methoden in der Wahrscheinlichkeitsrechnung. *Math. Ann.* **104**:415.
- Kolmogorov, A. N., I. G. Petrovsky, and N. S. Piskunov. 1937. Étude de l'équation de la diffusion avec croissance de la quantité de matière et son application à un problème biologique. *Bulletin Université d'Etat à Moscou (Bjul. Moskowskogo Gos. Univ.) Série internationale A* **1**:1–26.
- Koutroumpas, K., E. Cinquemani, P. Kouretas, and J. Lygeros. 2008. Parameter identification for stochastic hybrid systems using randomized optimization: A case study on subtilin production by *Bacillus subtilis*. *Nonlinear Anal.: Hybrid Syst.* **2**:786–802.
- Kramers, H. A. 1940. Brownian motion in a field of force and the diffusion model of chemical reactions. *Physica* **7**:284–304.
- Kuramoto, Y. 2003. Chemical oscillations, waves, and turbulence. Dover Publications.
- Kuramoto, Y., and T. Yamada. 1976. Pattern formation in oscillatory chemical reactions. *Progr. Theoret. Phys.* **56**:724–740.
- Kurata, H., H. El-Samad, T.-M. Yi, M. Khammash, and J. Doyle. 2001. Feedback regulation of the heat shock response in *E. coli*. In: Proc. 40th IEEE conference on Decision and Control. 837–842.
- Kurtz, T. G. 1972. The Relationship between stochastic and deterministic models for chemical reactions. *J. Chem. Phys.* **57**:2976–2978.
- Lakowicz, J. R. 2006. Principles of Fluorescence Spectroscopy. Springer US.
- Lee, K. J., W. D. McCormick, Q. Ouyang, and H. L. Swinney. 1993. Pattern formation by interacting chemical fronts. *Science* **261**:192–194.
- Lewis, J. 2003. Autoinhibition with transcriptional delay: A simple mechanism for the zebrafish somitogenesis oscillator. *Curr. Biol.* **13**:1398–1408.
- Leyvraz, F. 2003. Scaling theory and exactly solved models in the kinetics of irreversible aggregation. *Physics Reports* **383**:95–212.
- Li, H., Y. Cao, L. R. Petzold, and D. T. Gillespie. 2008. Algorithms and software for stochastic simulation of biochemical reacting systems. *Biotechnol. Prog.* **24**:56–61.

Bibliography

- Li, H., and L. Petzold. 2006. Logarithmic Direct Method for Discrete Stochastic Simulation of Chemically Reacting Systems. Technical report, Department of Computer Science, University of California Santa Barbara.
- Li, Q., and X. Lang. 2008. Internal noise-sustained circadian rhythms in a *Drosophila* model. *Biophys. J.* **94**:1983–1994.
- Ljung, L. 2002. Prediction error estimation methods. *Circuits, systems, and signal processing* **21**:11–21.
- Luedeke, C., S. Buvelot Frei, I. Sbalzarini, H. Schwarz, A. Spang, and Y. Barral. 2005. Septin-dependent compartmentalization of the endoplasmic reticulum during yeast polarized growth. *J. Cell Biol.* **169**:897–908.
- Magde, E. E. L., D. 1974. Fluorescence correlation spectroscopy. II. An experimental realization. *Biopolymers* **13**:29–61.
- Marjoram, P., J. Molitor, V. Plagnol, and S. Tavaré. 2003. Markov chain Monte Carlo without likelihoods. *Proc. Natl. Acad. Sci. USA* **100**:15324–15328.
- Markevich, N. I., J. B. Hoek, and B. N. Kholodenko. 2004. Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades. *J. Cell Biol.* **164**:353–359.
- Marquez-Lago, T. T., and J. Stelling. 2010. Counter-Intuitive Stochastic Behavior of Simple Gene Circuits with Negative Feedback. *Biophys. J.* **98**:1742–1750.
- Mason, O., and M. Verwoerd. 2007. Graph theory and networks in biology. *Systems Biology, IET* **1**:89–119.
- McAdams, H. H., and A. Arkin. 1997. Stochastic mechanisms in gene expression. *Proc. Natl. Acad. Sci. USA* **94**:814–819.
- McClintock, P. V. E. 1999. Random fluctuations - Unsolved problems of noise. *Nature* **401**:23–25.
- McCollum, J. M., G. D. Peterson, C. D. Cox, M. L. Simpson, and N. F. Samatova. 2006. The sorting direct method for stochastic simulation of biochemical systems with varying reaction execution behavior. *Comput. Biol. Chem.* **30**:39–49.
- McKane, A. J., J. D. Nagy, T. J. Newman, and M. O. Stefanini. 2007. Amplified biochemical oscillations in cellular systems. *J. Stat. Phys.* **128**:165–191.
- McKane, A. J., and T. J. Newman. 2005. Predator-prey cycles from resonant amplification of demographic stochasticity. *Phys. Rev. Lett.* **94**:218102.
- McQuarrie, D. A. 1967. Stochastic approach to chemical kinetics. *J. Appl. Prob.* **4**:413–478.
- Meinhardt, H., and A. J. d. B. Piet. 2001. Pattern formation in *Escherichia coli* : A model for the pole-to-pole oscillations of Min proteins and the localization of the division site. *Proc. Natl. Acad. Sci. USA* **98**:14202–14207.

Bibliography

- Mertens, F., R. Imbihl, and A. Mikhailov. 1994. Turbulence and standing waves in oscillatory chemical reactions with global coupling. *J. Chem. Phys.* **101**:9903.
- Miller, P., A. M. Zhabotinsky, J. E. Lisman, and X. J. Wang. 2005. The stability of a stochastic CaMKII switch: Dependence on the number of enzyme molecules and protein turnover. *PLoS Biol.* **3**:705–717.
- Morelli, M. J., and P. R. ten Wolde. 2008. Reaction Brownian dynamics and the effect of spatial fluctuations on the gain of a push-pull network. *J. Chem. Phys.* **129**:054112.
- Mouri, K., J. C. Nacher, and T. Akutsu. 2009. A Mathematical Model for the Detection Mechanism of DNA Double-Strand Breaks Depending on Autophosphorylation of ATM. *PLoS ONE* **4**:e5131.
- Moyal, J. E. 1949. Stochastic Processes and Statistical Physics. *J. R. Stat. Soc. B* **11**:150–210.
- Müller, C. L. 2010. Black-box Landscapes: Characterization, Optimization, Sampling, and Application to Geometric Configuration Problems. PhD thesis, Diss. ETH No. 19438, ETH Zürich.
- Müller, C. L. 2010. Exploring the common concepts of adaptive MCMC and Covariance Matrix Adaptation schemes. In: Auger, A., J. L. Shapiro, D. Whitley, and C. Witt, editors, *Theory of Evolutionary Algorithms*, number 10361 in Dagstuhl Seminar Proceedings. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Germany.
- Müller, C. L., R. Ramaswamy, and I. F. Sbalzarini. 2012. Global Parameter Identification of Stochastic Reaction Networks from Single Trajectories. In: Goryanin, I. I., and A. B. Goryachev, editors, *Advances in Systems Biology*, volume 736 of *Advances in Experimental Medicine and Biology*, chapter 28. Springer, 477–498.
- Müller, C. L., and I. F. Sbalzarini. 2010a. Gaussian Adaptation as a unifying framework for continuous black-box optimization and adaptive Monte Carlo sampling. In: Proc. IEEE Congress on Evolutionary Computation (CEC). Barcelona, Spain, 2594–2601.
- . 2010b. Gaussian Adaptation revisited — an entropic view on Covariance Matrix Adaptation. In: Proc. EvoStar, volume 6024 of *Lect. Notes Comput. Sci.* Istanbul, Turkey: Springer, 432–441.
- . 2011. Gaussian Adaptation for Robust Design Centering. In: Poloni, C., D. Quagliarella, J. Périaux, N. Gauger, and K. Giannakoglou, editors, *Evolutionary and deterministic methods for design, optimization and control*, Proc. EuroGen. Capua, Italy: CIRA, ECCOMAS, ERCOFATAC, 736–742.
- Munsky, B., B. Trinh, and M. Khammash. 2009. Listening to the noise: random fluctuations reveal gene network parameters. *Mol. Sys. Biol.* **5**:318.
- Nakajima, M., K. Imai, H. Ito, T. Nishiwaki, Y. Murayama, H. Iwasaki, T. Oyarna, and T. Kondo. 2005. Reconstitution of circadian oscillation of cyanobacterial KaiC phosphorylation in vitro. *Science* **308**:414–415.

- Newman, J. R. S., S. Ghaemmaghami, J. Ihmels, D. K. Breslow, M. Noble, J. L. DeRisi, and J. S. Weissman. 2006. Single-cell proteomic analysis of *S. cerevisiae* reveals the architecture of biological noise. *Nature* **441**:840–846.
- Nicolis, G., and I. Prigogine. 1977. Self-Organization in Nonequilibrium Systems: From Dissipative Structures to Order through Fluctuations. John Wiley & Sons.
- Nishikawa, M., and T. Shibata. 2010. Nonadaptive Fluctuation in an Adaptive Sensory System: Bacterial Chemoreceptor. *PLoS ONE* **5**:e11224.
- Novak, B., and J. Tyson. 1993. Numerical analysis of a comprehensive model of M-phase control in *Xenopus* oocyte extracts and intact embryos. *J. Cell Sci.* **106**:1153–1168.
- O'Neill, J. S., G. van Ooijen, L. E. Dixon, C. Troein, F. Corellou, F.-Y. Bouget, A. B. Reddy, and A. J. Millar. 2011. Circadian rhythms persist without transcription in a eukaryote. *Nature* **469**:554–558.
- Ozbudak, E. M., M. Thattai, H. N. Lim, B. I. Shraiman, and A. van Oudenaarden. 2004. Multistability in the lactose utilization network of *Escherichia coli*. *Nature* **427**:737–740.
- Panja, D. 2004. Effects of fluctuations on propagating fronts. *Physics Reports* **393**:87–174.
- Paulsson, J. 2004. Summing up the noise in gene networks. *Nature* **427**:415–418.
- Pearson, J. E. 1993. Complex patterns in a simple system. *Science* **261**:189–192.
- Pedersen, M. G., A. M. Bersani, and E. Bersani. 2008. Quasi steady-state approximations in complex intracellular signal transduction networks - a word of caution. *J. Math. Chem.* **43**:1318–1344.
- Pedraza, J. M., and J. Paulsson. 2008. Effects of molecular memory and bursting on fluctuations in gene expression. *Science* **319**:339–343.
- Peng, X., and Y. Wang. 2007. *L*-leap: accelerating the stochastic simulation of chemically reacting systems. *Appl. Math. Mech.* **28**:1361–1371.
- Peng, X., W. Zhou, and Y. Wang. 2007. Efficient binomial leap method for simulating chemical kinetics. *J. Chem. Phys.* **126**:224109.
- Perc, M., M. Gosak, and M. Marhl. 2007. Periodic calcium waves in coupled cells induced by internal noise. *Chem. Phys. Lett.* **437**:143–147.
- Perc, M., A. K. Green, C. J. Dixon, and M. Marhl. 2008. Establishing the stochastic nature of intracellular calcium oscillations from experimental data. *Biophys. Chem.* **132**:33–38.
- Pike, L. J. 2009. The challenge of lipid rafts. *J. Lip. Res.* **50**:S323.
- Polak, M., and L. Rubinovich. 2008. Nanochemical Equilibrium Involving a Small Number of Molecules: A Prediction of a Distinct Confinement Effect. *Nano Letts.* **8**:3543–3547.

Bibliography

- Pomerening, J. R., E. D. Sontag, and J. E. Ferrell. 2003. Building a cell cycle oscillator: hysteresis and bistability in the activation of Cdc2. *Nat. Cell Biol.* **5**:346–351.
- Poovathingal, S. K., and R. Gunawan. 2010. Global parameter estimation methods for stochastic biochemical systems. *BMC Bioinformatics* **11**:414.
- Pourquié, O. 2003. The segmentation clock: converting embryonic time into spatial pattern. *Science* **301**:328.
- Prigogine, I. 1980. From Being to Becoming: Time and Complexity in the Physical Sciences. W. H. Freeman.
- Pritchard, J. K., M. T. Seielstad, A. Perez-Lezaun, and M. W. Feldman. 1999. Population growth of human Y chromosomes: A study of Y chromosome microsatellites. *Molecular Biology and Evolution* **16**:1791–1798.
- Provance, D. W., A. McDowall, M. Marko, and K. Luby-Phelps. 1993. Cytoarchitecture of size-excluding compartments in living cells. *J. Cell Sci.* **106**:565–577.
- Qian, H. 2006. Open-system nonequilibrium steady state: Statistical thermodynamics, fluctuations, and chemical oscillations. *J. Phys. Chem. B* **110**:15063–15074.
- Qian, H., and E. L. Elson. 2004. Fluorescence correlation spectroscopy with high-order and dual-color correlation to probe nonequilibrium steady states. *Proc. Natl. Acad. Sci. USA* **101**:2828–2833.
- Qian, H., S. Saffarian, and E. L. Elson. 2002. Concentration fluctuations in a mesoscopic oscillating chemical reaction system. *Proc. Natl. Acad. Sci. USA* **99**:10376–10381.
- Raj, A., C. S. Peskin, D. Tranchina, D. Y. Vargas, and S. Tyagi. 2006. Stochastic mRNA synthesis in mammalian cells. *PLoS Biol.* **4**:1707–1719.
- Rajasekaran, S., and K. W. Ross. 1993. Fast Algorithms for Generating Discrete Random Variates with Changing Distributions. *ACM Trans. Model. Comput. Simul.* **3**:1–19.
- Ramaswamy, R., N. González-Segredo, and I. F. Sbalzarini. 2009. A new class of highly efficient exact stochastic simulation algorithms for chemical reaction networks. *J. Chem. Phys.* **130**:244104.
- Ramaswamy, R., N. González-Segredo, I. F. Sbalzarini, and R. Grima. 2012. Discreteness-induced concentration inversion in mesoscopic chemical systems. *Nat. Commun.* **3**:779.
- Ramaswamy, R., and I. F. Sbalzarini. 2010a. Fast Exact Stochastic Simulation Algorithms Using Partial Propensities. In: Proc. ICNAAM, Numerical Analysis and Applied Mathematics, International Conference. AIP, 1338–1341.
- . 2010b. A partial-propensity variant of the composition-rejection stochastic simulation algorithm for chemical reaction networks. *J. Chem. Phys.* **132**:044102.
- . 2011a. Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods. *J. Chem. Phys.* **135**:244103.

- _____. 2011b. Intrinsic noise alters the frequency spectrum of mesoscopic oscillatory chemical reaction systems. *Sci. Rep.* **1**:154.
- _____. 2011c. A partial-propensity formulation of the stochastic simulation algorithm for chemical reaction networks with delays. *J. Chem. Phys.* **134**:014106.
- Ramaswamy, R., I. F. Sbalzarini, and N. González-Segredo. 2011. Noise-Induced Modulation of the Relaxation Kinetics around a Non-Equilibrium Steady State of Non-Linear Chemical Reaction Networks. *PLoS ONE* **6**:e16045.
- Rathinam, M., L. R. Petzold, Y. Cao, and D. T. Gillespie. 2003. Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method. *J. Chem. Phys.* **119**:12784–12794.
- _____. 2005. Consistency and stability of tau-leaping schemes for chemical reaction systems. *Multiscale Model. Simul.* **4**:867–895.
- Reinker, S., R. M. Altman, and J. Timmer. 2006. Parameter estimation in stochastic biochemical reactions. *IEE Proc.-Syst. Biol* **153**:168.
- Rigler, R., and E. S. Elson. 2001. Fluorescence Correlation Spectroscopy, Theory and Applications, volume 65 of *Springer Series in Chemical Physics*. Heidelberg, Germany: Springer-Verlag.
- Rossinelli, D., B. Bayati, and P. Koumoutsakos. 2008. Accelerated stochastic and hybrid methods for spatial simulations of reaction-diffusion systems. *Chem. Phys. Lett.* **451**:136–140.
- Samoilov, M., S. Plyasunov, and A. P. Arkin. 2005. Stochastic amplification and signaling in enzymatic futile cycles through noise-induced bistability with oscillations. *Proc. Natl. Acad. Sci. USA* **102**:2310–2315.
- Samoilov, M. S., and A. P. Arkin. 2006. Deviant effects in molecular reaction pathways. *Nature Biotechnology* **24**:1235–1240.
- Sbalzarini, I. F., A. Hayer, A. Helenius, and P. Koumoutsakos. 2006. Simulations of (An)Isotropic Diffusion on Curved Biological Surfaces. *Biophys. J.* **90**:878–885.
- Sbalzarini, I. F., A. Mezzacasa, A. Helenius, and P. Koumoutsakos. 2005. Effects of organelle shape on fluorescence recovery after photobleaching. *Biophys. J.* **89**:1482–1492.
- Schibler, U., and F. Naef. 2005. Cellular oscillators: rhythmic gene expression and metabolism. *Curr. Opin. Cell Biol.* **17**:223–229.
- Schlogl, F. 1972. Chemical reaction models for nonequilibrium phase-transitions. *Z. Physik* **253**:147–161.
- Schneider, K. R., and T. Wilhelm. 2000. Model reduction by extended quasi-steady-state approximation. *J. Math. Biol.* **40**:443–450.

Bibliography

- Schnell, S., and T. E. Turner. 2004. Reaction kinetics in intracellular environments with macromolecular crowding: simulations and rate laws. *Prog. Biophys. Mol. Biol.* **85**:235–260.
- Scott, M., T. Hwa, and B. Ingalls. 2007. Deterministic characterization of stochastic genetic circuits. *Proc. Natl. Acad. Sci. USA* **104**:7402–7407.
- Segel, L. A. 1988. On the validity of the steady state assumption of enzyme kinetics. *Bull. Math. Biol.* **50**:579–593.
- Seger, R., and E. Krebs. 1995. The MAPK signaling cascade. *The FASEB journal* **9**:726.
- Selkov, E. E. 1968. Self-oscillations in glycolysis .1. A simple kinetic model. *Eur. J. Biochem.* **4**:79–86.
- Shibata, T. 2004. Amplification of noise in a cascade chemical reaction. *Phys. Rev. E* **69**:056218.
- Shimoni, Y., S. Altuvia, H. Margalit, and O. Biham. 2009. Stochastic Analysis of the SOS Response in *Escherichia coli*. *PLoS ONE* **4**:e5363.
- Slepoy, A., A. P. Thompson, and S. J. Plimpton. 2008. A constant-time kinetic Monte Carlo algorithm for simulation of large biochemical reaction networks. *J. Chem. Phys.* **128**:205101.
- Smit, B., and T. Maesen. 2008. Towards a molecular understanding of shape selectivity. *Nature* **451**:671–678.
- Snijder, B., R. Sacher, P. Rämö, E.-M. Damm, P. Liberali, and L. Pelkmans. 2009. Population context determines cell-to-cell variability in endocytosis and virus infection. *Nature* **461**:520–523.
- Springer, M., and J. Paulsson. 2006. Biological physics - Harmonies from noise. *Nature* **439**:27–28.
- Stock, G., K. Ghosh, and K. A. Dill. 2008. Maximum Caliber: A variational approach applied to two-state dynamics. *J. Chem. Phys.* **128**:194102.
- Strogatz, S. H. 2001. Exploring complex networks. *Nature* **410**:268–276.
- Takahashi, K., S. Tanase-Nicola, and P. R. ten Wolde. 2010. Spatio-temporal correlations can drastically change the response of a MAPK pathway. *Proc. Natl. Acad. Sci. USA* **107**:2473–2478.
- Tam, Z. Y., Y. H. Cai, and R. Gunawan. 2010. Elucidating Cytochrome *c* Release from Mitochondria: Insights from an In Silico Three-Dimensional Model. *Biophys. J.* **99**:3155–3163.
- Thomas, P., A. V. Straube, and R. Grima. 2010. Stochastic theory of large-scale enzyme-reaction networks: Finite copy number corrections to rate equation models. *J. Chem. Phys.* **133**:195101.

- Tian, T., and K. Burrage. 2006. Stochastic models for regulatory networks of the genetic toggle switch. *Proc. Natl. Acad. Sci. USA* **103**:8372–8377.
- Tian, T., K. Burrage, P. M. Burrage, and M. Carletti. 2007. Stochastic delay differential equations for genetic regulatory networks. *J. Comput. Appl. Math.* **205**:696–707.
- Tomita, J., M. Nakajima, T. Kondo, and H. Iwasaki. 2005. No transcription-translation feedback in circadian rhythm of KaiC phosphorylation. *Science* **307**:251–254.
- Tomita, K., and H. Daido. 1980. Possibility of chaotic behaviour and multi-basins in forced glycolytic oscillations. *Phys. Lett. A* **79**:133–137.
- Toni, T., D. Welch, N. Strelkowa, A. Ipsen, and M. P. H. Stumpf. 2009. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface* **6**:187–202.
- Turing, A. M. 1950. I.—Computing Machinery and Intelligence. *Mind* **LIX**:433–460.
- . 1952. The chemical basis of morphogenesis. *Phil. Trans. R. Soc. London* **B237**:37–72.
- Turner, M. S., P. Sens, and N. D. Soccia. 2005. Nonequilibrium raftlike membrane domains under continuous recycling. *Phys. Rev. Lett.* **95**:168301.
- Tyson, J., B. Novak, G. Odell, K. Chen, and C. Dennis Thron. 1996. Chemical kinetic theory: understanding cell-cycle regulation. *Trends Biochem. Sci.* **21**:89–96.
- Tyson, J. J., C. I. Hong, C. D. Thron, and B. Novak. 1999. A simple model of circadian rhythms based on dimerization and proteolysis of PER and TIM. *Biophys. J.* **77**:2411–2417.
- Ugarte, D., A. Chatelain, and W. De Heer. 1996. Nanocapillarity and chemistry in carbon nanotubes. *Science* **274**:1897–1899.
- van Dongen, P. G. J. 1987. Fluctuations in coagulating systems. II. *J. Stat. Phys.* **49**:927–975.
- van Dongen, P. G. J., and M. H. Ernst. 1987. Fluctuations in coagulating systems. *J. Stat. Phys.* **49**:879–926.
- van Kampen, N. G. 1961. A power series expansion of the master equation. *Can. J. Phys.* **39**:551–567.
- . 1963. Thermal Fluctuations in Nonlinear Systems. *J. Math. Phys.* **4**:190.
- . 1976. The expansion of the master equation. *Adv. Chem. Phys.* **34**:245–309.
- . 2001. *Stochastic Processes in Physics and Chemistry*. North Holland, 2nd edition.
- van Zon, J. S., D. K. Lubensky, P. R. H. Altena, and P. R. ten Wolde. 2007. An allosteric model of circadian KaiC phosphorylation. *Proc. Natl. Acad. Sci. USA* **104**:7420–7425.
- van Zon, J. S., and P. R. ten Wolde. 2005. Green's-function reaction dynamics: A particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.* **123**:234910.

Bibliography

- Verlet, L. 1967. Computer experiments on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.* **159**:98–103.
- von Smoluchowski, M. 1917. Versuch einer mathematischen Theorie der Koagulationskinetik kolloider Lösungen. *Z. Phys. Chem.* **92**:129–168.
- Vriezema, D., M. Aragones, J. Elemans, J. Cornelissen, A. Rowan, and R. Nolte. 2005. Self-assembled nanoreactors. *Chem. rev.* **105**:1445–1490.
- Wattis, J. A. D. 2009. The modified Becker-Döring system with aggregation-dominated rate coefficients. *J. Phys. A: Math. Theor.* **42**:045002.
- Wilhelm, T. 2000. Chemical systems consisting only of elementary steps - a paradigm for nonlinear behavior. *J. Math. Chem.* **27**:71–88.
- Wolkenhauer, O. 2001. Systems biology: The reincarnation of systems theory applied in biology? *Briefings in Bioinformatics* **2**:258.
- Wu, X., M. Chee, and R. Kapral. 1991. Vortex dynamics in oscillatory chemical systems. *Chaos* **1**:421.
- Xiao, T., J. Ma, Z. Hou, and H. Xin. 2007. Effects of internal noise in mesoscopic chemical systems near Hopf bifurcation. *New J. Phys.* **9**:403.
- Xu, Z., and X. Cai. 2009. Stochastic Simulation of Delay-Induced Circadian Rhythms in *Drosophila*. *EURASIP J. Bioinform. Syst. Biol.* **2009**.
- Yang, L., M. Dolnik, A. M. Zhabotinsky, and I. R. Epstein. 2002. Spatial resonances and superposition patterns in a reaction-diffusion model with interacting turing modes. *Phys. Rev. Lett.* **88**:208303.
- Zechner, C., A. Ganguly, S. Pelet, M. Peter, and H. Koepll. 2011, submitted. Accounting for Extrinsic Variability in the Estimation of Stochastic Rate Constants. *Intl. J. Robust Nonlinear Control* .
- Zechner, C., S. Pelet, M. Peter, and H. Koepll. 2011. Recursive Bayesian Estimation of Stochastic Rate Constants from Heterogeneous Cell Populations. In: Proc. IEEE CDC, Conference on Decision and Control, volume in press.
- Zhabotinsky, A. M. 1964. Periodical oxidation of malonic acid in solution (a study of the Belousov reaction kinetics). *Biofizika* **9**:306–311.
- . 1991. A history of chemical oscillations and waves. *Chaos* **1**:379–386.
- Zhao, Y., and J. H. Keen. 2008. Gyrating Clathrin: Highly Dynamic Clathrin Structures Involved in Rapid Receptor Recycling. *Traffic* **9**:2253–2264.
- Zhou, C. X., H. Y. Guo, and Q. Ouyang. 2002. Experimental study of the dimensionality of black-eye patterns. *Phys. Rev. E* **65**:036118.

Bibliography

- Zhou, W., X. Peng, Z. Yan, and Y. Wang. 2008. Accelerated stochastic simulation algorithm for coupled chemical reactions with delays. *Comput. Biol. Chem.* **32**:240–242.
- Zwicker, D., D. K. Lubensky, and P. R. ten Wolde. 2010. Robust circadian clocks from coupled protein-modification and transcription-translation cycles. *Proc. Natl. Acad. Sci. USA* **107**:22540–22545.

Index

- approximate stochastic simulation algorithms, 34, 42
- Bachmann-Landau big-*O* symbol, 4
- Cauchy-Lorentz distribution, 134
- Chapman-Kolmogorov equation, 3
- chemical Kramer-Moyal equation, 6
- chemical Langevin equation, 8
- chemical master equation, xviii, 4
- chemical reaction network, 1
- composition-rejection method, 39
- computational cost, 34
- degree of coupling, 28
- delay partial-propensity direct method, 72
- dependency graph, 27
- direct method, 35
- discreteness-induced concentration inversion, 93
- effective mesoscopic reaction rate equation, 95, 206
- exact stochastic simulation algorithms, 34
- first reaction method, 34
- Fisher-KPP equation, 137
- Gaussian adaptation, 169, 170, 172
- generalized diffusion equation, 7
- Gray-Scott reaction, 161
- Hopf bifurcation, 218
- kinetic Monte Carlo, xviii, 10
- linear chemical Fokker-Planck equation, 203
- linear reaction networks, 10
- linear-noise approximation, 92, 203
- Liouville process, 9
- logarithmic direct method, 38
- Lorentzian, 134
- MAPK phosphorylation-dephosphorylation reaction, 199
- modules of partial-propensity algorithms, 86
- moment generating function, 189
- next reaction method, 36
- next subvolume method, 144
- nonlinear chemical Fokker-Planck equation, 7
- nonlinear reaction networks, 10
- normalized power spectral density, 129, 220
- objective function, 173
- off-lattice stochastic reaction-diffusion, 138
- on-lattice stochastic reaction-diffusion, 138, 140
- optimized direct method, 37
- Ornstein-Uhlenbeck process, 203
- parameter identification, 168
- partial propensity, 11, 46
- partial propensity stochastic simulation algorithms, 46
- partial-propensity direct method, 48
- partial-propensity stochastic reaction-diffusion, 147
- partial-propensity stochastic simulation algorithm with composition-rejection sampling, 64
- propensity, 3

Index

rate function, 9
reaction degeneracy, 2
reaction flux vector, 9, 26
reaction rate equation, xviii, 9
reaction-diffusion equation, 137
reaction-diffusion master equation, 138

sorting direct method, 37
sorting partial-propensity direct method,
 52
specific probability rate, 2
stochastic simulation algorithm, 10
strongly coupled reaction network, 25, 28
systems identification, 168

time autocorrelation function, 112, 129

van Kampen ansatz, 200
van Kampen system-size expansion, 200

weakly coupled reaction network, 25, 28

Publications

Refereed publications during PhD studies:

1. **Rajesh Ramaswamy**, Nérido González-Segredo and Ivo F. Sbalzarini, A new class of highly efficient exact stochastic simulation algorithm for chemical reaction networks, *Journal of Chemical Physics*, vol. 130, no. 24, 244104 (2009) (Swiss Institute of Bioinformatics best graduate award 2010)
2. **Rajesh Ramaswamy** and Ivo F. Sbalzarini, A partial-propensity variant of the composition-rejection stochastic simulation algorithm for chemical reaction networks, *Journal of Chemical Physics*, vol. 132, no. 4, 044102 (2010) (Swiss Institute of Bioinformatics best graduate award 2010)
3. **Rajesh Ramaswamy** and Ivo F. Sbalzarini, Fast exact stochastic simulation algorithms using partial propensities, in *Proc. ICNAAM, Numerical Analysis and Applied Mathematics, International Conference*, pages 1338–1341, American Institute of Physics (2010)
4. Anton A. Polyansky, **Rajesh Ramaswamy**, Pavel E. Volynsky, Ivo F. Sbalzarini, Siewert J. Marrink and Roman G. Efremov, Antimicrobial Peptides Induce Growth of Phosphatidylglycerol Domains in a Model Bacterial Membrane, *Journal of Physical Chemistry Letters*, vol. 1, pages 3108–3111 (2010)
5. **Rajesh Ramaswamy** and Ivo F. Sbalzarini, A partial-propensity formulation of the stochastic simulation algorithm for chemical reaction networks with delays, *Journal of Chemical Physics*, vol. 134, no. 1, 014106 (2011)
6. **Rajesh Ramaswamy**, Nérido González-Segredo and Ivo F. Sbalzarini, Noise-induced modulation of the relaxation kinetics around a non-equilibrium steady state of non-linear chemical reaction networks, *PLoS ONE*, vol. 6, no. 1, e16045 (2011)
7. **Rajesh Ramaswamy** and Ivo F. Sbalzarini, Intrinsic noise alters the frequency spectrum of mesoscopic chemical reaction systems, *Scientific Reports*, vol. 1, 154 (2011)
8. **Rajesh Ramaswamy** and Ivo F. Sbalzarini, Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods, *Journal of Chemical Physics*, vol. 135, no. 24, 244103 (2011)
9. Christian L. Müller, **Rajesh Ramaswamy*** and Ivo F. Sbalzarini, Global parameter identification for stochastic reaction networks from single trajectories, In: *Goryanin, I.*

*Equal contribution with Christian L. Müller

I., and A. B. Goryachev, editors, Advances in Systems Biology, vol. 736 of Advances in Experimental Medicine and Biology, chapter 28, pages 477–498, Springer (2012)

10. **Rajesh Ramaswamy**, Nélido González-Segredo, Ivo F. Sbalzarini and Ramon Grima, Discreteness-induced concentration inversion in mesoscopic chemical systems, *Nature Communications*, vol. 3, 779 (2012)

Earlier refereed publications:

1. N. Venkateswaran, **R. Rajesh**, N. Sudarshan, R. Rajasimhan, C. Chandramouli, R. Chidambareswaran, B. Harish, K. Arvind and M. Muilan, ψ NAM For Massive Neuronal Assembly Modeling: Part-I, Processing Elements, The Sixth International Conference on Computational Intelligence and Natural Computing (2003)
2. N. Venkateswaran, R. Chidambareswaran, B. Harish, K. Arvind, C. Chandramouli, **R. Rajesh**, R. Rajasimhan and N. Sudarshan, ψ NAM For Massive Neuronal Assembly Modeling: Part-II, The Array Architecture, The Sixth International Conference on Computational Intelligence and Natural Computing (2003)
3. N. Venkateswaran, **R. Rajesh**, R. Rajasimhan, N. Sudharshan and M. Muilan, Predicting the Morphology of Arbitrary Dendritic Trees through Simulated Annealing ,Engineering Intelligent Systems (EIS04), Portugal, sponsored by International Computing Sciences Conference (ICSC) (2004)
4. N. Venkateswaran and **R. Rajesh**, Partitioned Parallel Processing Approach for Predicting Multi-Million Neuron Interconnectivity in the Brain Involving Soma-Axon-Dendrites-Synapse : A Simulation Model, Brain Inspired Cognitive Systems (2004)

Curriculum Vitae

Rajesh Ramaswamy

Date of birth: April 10, 1983

Place of birth: Pondicherry, India

Citizenship: India

2000 – 2004 Bachelor studies at the University of Madras, Chennai, India

Major: Electronics and Communication

Degree: Bachelor of Engineering (B.Eng.)

First Class with Distinction, Aggregate: 82.64%

2004 – 2006 Master studies at the Max Planck Research School,

University of Tübingen, Tübingen, Germany

Major: Neural and Behavioral Sciences

Degree: Master of Science (M.Sc.)

CGPA: 0.9, overall grade: A (Excellent)

Placed on the Dean's list for excellent academic performance

2007 – 2011 PhD studies at the MOSAIC group

Department of Computer Science

ETH Zürich, Switzerland

Advisor: Prof. Dr. Ivo F. Sbalzarini

Degree: Doctor of Science (Dr. sc. ETH)

Nominated for an ETH medal