Analysis of Markovian Population Models

Michael Backenköhler

ANALYSIS OF MARKOVIAN POPULATION MODELS

MICHAEL BACKENKÖHLER

DISSERTATION

zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

Saarbrücken, Juni 2022

Tag des Kolloqiums:	
Dekan:	
Berichterstatter:	
Vorsitzender:	
Akademischer Mitarbeiter:	

DECLARATION OF ORIGINAL AUTHORSHIP

I hereby declare that this dissertation is my own original work except where otherwise indicated. All data or concepts drawn directly or indirectly from other sources have been correctly acknowledged. This dissertation has not been submitted in its present or similar form to any other academic institution either in Germany or abroad for the award of any other degree.

EIDESSTATTLICHE VERSICHERUNG

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in einem Verfahren zur Erlangung eines akademischen Grades vorgelegt.

Saarbrücken, June 2022

Michael Backenköhler

ABSTRACT

ffi
ffl
ff Short summary of the contents in English. . . a great guide by Kent Beck how to write good abstracts can be found here:

https://plg.uwaterloo.ca/~migod/research/beckOOPSLA.html

ZUSAMMENFASSUNG

Kurze Zusammenfassung des Inhaltes in deutscher Sprache. . .

ACKNOWLEDGMENTS

Put your acknowledgments here.

CONTENTS

Part I

PRELIMINARIES

1 INTRODUCTION

Test XP Modelling often is a delicate balance of adequate representation and abstraction. The former is the goal to capture all the relevant behaviours and effects in a model. Abstractions are made for the benefit of both, explainability and facilitating analysis.

Simplify representation for more efficiency per computation.

Markovian population models (MPMs) provide a widely used framework to capture stochastic interactions between groups of identical agents. This subclass of continuous-time Markov chains (CTMCs) is used to describe the stochastic dynamics of systems in various domains. Prominent applications are chemical reaction networks in quantitative biology (Ullah and Wolkenhauer, 2009), epidemic spreading (Porter and Gleeson, 2016), performance analysis of technical and information systems (Bortolussi et al., 2013; Gast, Bortolussi, and Tribastone, 2019) as well as the behavior of collective adaptive systems (Bernardo, De Nicola, and Hillston, 2016).

In many areas of science, stochastic models of interacting populations can describe systems in which the discrete population sizes evolve stochastically in continuous time. Such problems naturally occur in a wide range of areas such as chemistry (Gillespie, 1977), systems biology (Ullah and Wolkenhauer, 2011; Wilkinson, 2018), epidemiology (Mode and Sleeman, 2000) as well as queuing systems (Breuer, 2003) and finance (Pardoux, 2008).

Interactions between agents, commonly referred to as *reactions*, happen at exponentially distributed random times. Their rate depends on the current system state, i.e. the population sizes. This results in a continuous-time Markov chain semantics (Anderson, 2012).

MPMs are widely used to model the time evolution of complex discrete phenomena in continuous time. Such problems naturally occur in a wide range of areas such as chemistry (Gillespie, 1977), systems biology (Ullah and Wolkenhauer, 2011; Wilkinson, 2018), epidemiology (Mode and Sleeman, 2000) as well as queu-

ing systems (Breuer, 2003) and finance (Pardoux, 2008). In many applications, an MPM describes the stochastic interaction of populations of agents. The state variables are counts of individual entities of different populations.

Markovian Population Models that are used to describe cellular processes are often subject to inherent stochasticity. The dynamics of gene expression, for instance, is influenced by single random events (e.g. transcription factor binding) and hence, models that take this randomness into account must monitor discrete molecular counts and reaction events that change these counts. Discrete-state continuous-time Markov chains have successfully been used to describe networks of chemical reactions over time that correspond to the basic events of such processes. The time-evolution of the corresponding probability distribution is given by the chemical master equation, whose numerical solution is extremely challenging because of the enormous size of the underlying state-space.

1.1 ORGANIZATION

All contributions presented in this thesis are related to Markovian population models. Therefore the background chapter, i.e. Chapter 2 is relevant to all later parts of the thesis. The prerequisite for Chapter 6 and Chapter 7 is the aggregation technique presented in Chapter 5. Note that Chapter 3 and Chapter 4 share the temporal moment approach presented in both chapters. Figure 1.1 provides an overview of the dependencies between chapters.

1.2 PREVIOUS PUBLICATIONS

The ideas and much of the presented results have appeared previously in the following publications. As such, most chapters have been published in various conference proceedings. The publications and their respective sections are as indicated below.

- Chapter 3 has with minor differences been published as

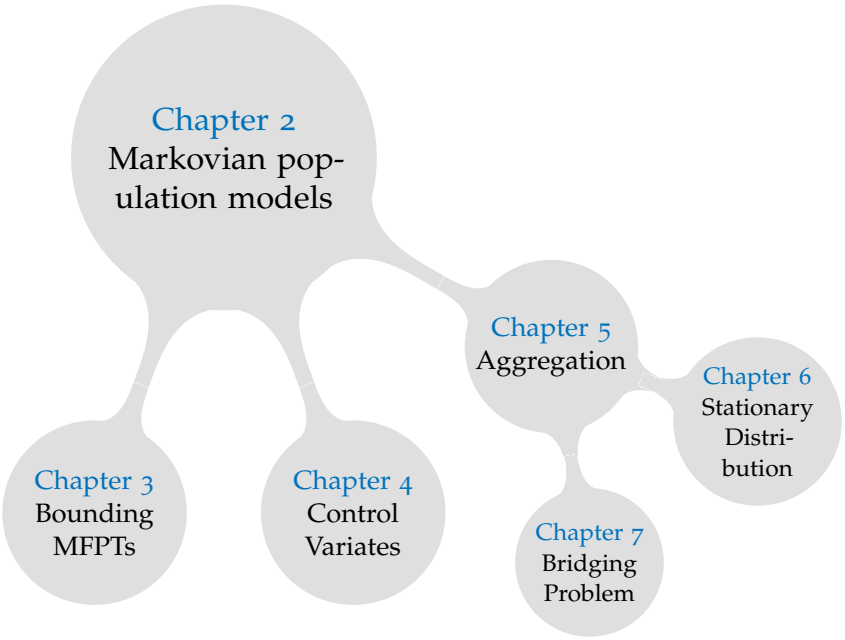


Figure 1.1: Chapter dependencies.

Michael Backenköhler, Luca Bortolussi, and Verena Wolf. "Bounding Mean First Passage Times in Population Continuous-Time Markov Chains." In: *17th International Conference on Quantitative Evaluation of SysTems (QEST)*. Vol. 12289. LNCS. Springer, 2020, pp. 155–174.

- Chapter 4 has with minor differences been published as

  Michael Backenköhler, Luca Bortolussi, and Verena Wolf. "Control Variates for Stochastic Simulation of Chemical Reaction Networks." In: *17th International Conference on Computational Methods in Systems Biology (CMSB)*. Vol. 11773. LNCS. Springer, 2019, pp. 42–59.

- Chapter 6 has with minor differences been published as

  Michael Backenköhler, Luca Bortolussi, Gerrit Großmann, and Verena Wolf. "Abstraction-Guid-

ed Truncations for Stationary Distributions of Markovian Population Models." In: *18th International Conference on Quantitative Evaluation of SysTems (QEST)*. (2021).

- Chapter 7 has with minor differences been published as

  Michael Backenköhler, Luca Bortolussi, Gerrit Großmann, and Verena Wolf. "Analysis of Markov Jump Processes under Terminal Constraints." In: *27th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*. Vol. 1265. LNCS. Springer, 2021, pp. 210–229.

Chapter 5 is in large part based on this and the preceeding publication. Chapter 2 contains introductory material and examples from all of the above publications.

BACKGROUND

2

2.1 CONTINUOUS-TIME MARKOV CHAINS

A *stochastic process* is a parameterized collection of random variables $\{X_t\}_{t\in T}$ defined on some probability space $(\Omega, \mathcal{F}, P)$ and takes values in complete metric space (S, r). In most contexts the index set T models *time*. In a discrete setting $T = \mathbb{N}$, while $T = [0, \infty)$ in the continuous setting, which we consider in this thesis. We observe the values $X(t, \omega)$ for some fixed, but unknown $\omega \in \Omega$. The information of the process up to time t is given by the σ-algebra $\mathcal{F}_t \subset \mathcal{F}$. The increasing family of sigma algebras, i.e. $\mathcal{F}_s \subseteq \mathcal{F}_t$ for $s \leqslant t$ is called a *filtration*.

See Feller (1971) for stoch. processes in general.

More specifically, we study with models having continuous-time Markov chain (CTMC) semantics — a type of *Markov process*. Such processes satisfy the *Markov property*: For all Borel-measurable functions f

$$\mathrm{E}\left(f(X_{t+s}) \mid \mathcal{F}_t\right) = \mathrm{E}\left(f(X_{t+s}) \mid X(t)\right). \tag{2.1}$$

Intuitively, this property expresses that the future of the process depends only on the latest condition, i.e. X_t and not earlier conditions ($\mathcal{F}_t$). A CTMC is a Markov process, that takes discrete values $S = \{s_0, s_1, \dots\}$ over continuous time $T = [0, \infty)$. If further

The time-discrete analogue is the DTMC.

$$\Pr\left(X_t = s_1 \mid X_0 = s_0\right) = \Pr\left(X_{t+h} = s_1 \mid X_h = s_0\right) \tag{2.2}$$

for all $t, h \geqslant 0$ and $s_0, s_1 \in S$ the chain is *time-homogenous*: The absolute time point is irrelevant, and the dynamics do not change if we shift in time. In this thesis we are only interested in timhomogenous CTMCs, but most techniques developed should carry over. We define *transition probabilities*

$$p_{ij}(h) = \Pr(X_h = s_j \mid X_0 = s_i). \tag{2.3}$$

Accordingly, the *transition matrix* is given by $P(h)_{ij} = p_{ij}(h)$ for all indices i and j. The Chapman-Kolmogorov equation

$$P(s + t) = P(s)P(t) \tag{2.4}$$

follows directly from the law of total probability and the Markov property (2.1). (2.4) directly tells us that the transition probabilities from a semigroup. Studying Markov processes from this direction is a popular approach (Ethier and Kurtz, 2009).

The standard method of specifying a Markov process, and CTMCs in particular, is is the *generator*. In general, this is an operator A on some class of functions and

$$\mathrm{E}\left(f(X_{t+h}) - f(X_t) \mid \mathcal{F}_t\right) = Af(X_t)h + o(h)\,, \tag{2.5}$$

where $\mathcal{F}_t$ is the filtration up to time t. This equation can be interpreted as the requirement, that

$$f(X_t) - f(X_0) - \int_0^t Af(x_s)\,ds$$

is a martingale (see Kurtz, 1981, p. 5). We will use this in Chapter 3 to derive bounds on mean first-passage times.

In CTMCs the generator is defined by giving the *intensities* or *rates* of transitions. Such a rate $q_{ij} > 0$ between state s_i and s_j implies

This is congruent with (2.5) under application the Markov property.

$$\Pr(X_h = s_j \mid X_0 = s_i) = q_{ij}h + o(h)\,. \tag{2.6}$$

Due to the discrete nature, the generator is a matrix, usually called the Q-matrix,

$$Q = \lim_{h \downarrow 0} \frac{1}{h}\left(P(h) - I\right)\,. \tag{2.7}$$

As such the change of the state probability distribution over time is fully characterized by the *Kolmogorov forward equation*

$$\frac{d}{dt}P(t) = P(t)Q\,. \tag{2.8}$$

Analogously, the *Kolmogorov backward equation* is

$$\frac{d}{dt}P(t) = QP(t)^{\mathsf{T}}\,. \tag{2.9}$$

Distributions in the context of Markov chains are typically in row-vector form

$$\pi(t) \coloneqq (\pi(x_1, t), \pi(x_2, t), \dots)\,,$$

where we define

$$\pi(x_i, t) \coloneqq \Pr(X_t = x_i), \quad \forall x_i \in \mathcal{S}, \forall t \geqslant 0\,.$$

Often, (2.8) and (2.9) are used in the context of an distributions. In this case it makes more sense to right-multiply the unit vector to these equations such that

$$\frac{\mathrm{d}}{\mathrm{d}t}\pi(t) = \pi(t)Q \tag{2.10}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}t}\pi(t) = Q\pi(t)^{\mathsf{T}}\,. \tag{2.11}$$

Given some initial distribution

$$\pi_0 \coloneqq \pi(0)\,,$$

the distribution $\pi(t)$ is given by give simple initial value problems (IVPs) of (2.10).

2.1.1 *Computing Transient Distributions*

Stewart (1994) provides a comprehensive overview of solution methods. Here, we only provide a basic overview and intuition relevant for the rest of this thesis.

MATRIX EXPONENTIAL The transition probability matrix $P(t)$ is the solution of (2.8)

$$P(t) = \exp(Qt)\,,$$

where the matrix exponential for a square matrix M

$$\exp(M) \coloneqq \sum_{k=0}^{\infty} \frac{1}{k!} M^k\,.$$

Due to the factor $h^k/k!$, the sum of the matrix exponential can be truncated to get an estimate of high quality. In practice, this method is unsuited to many problems, because the factor Q^k becomes incurs a prohibitive cost, especially if the state space is large.

The matrix exponential is the analytical solution.

NUMERICAL INTEGRATION The Kolmogorov equations (2.8), (2.9) provide us with an IVP that can be solved numerically. This method scales much better than the matrix exponential method, especially if the whole time-series is of interest. If an initial distribution π_0 is fixed, the ordinary differential equation (ODE) simplify further, such that we only have one equation per state. The drawback to this method is the error inherent to numerical integration schemes.

UNIFORMIZATION Uniformization is an elegant algorithm to compute transient solutions. Here, the CTMC is transformed into a DTMC. Using a uniformization rate

$$\lambda_0 \geqslant \max_i |q_{ii}|$$

the transition probabilities of the DTMC become

$$P_{ij} = \begin{cases} Q_{ij}/\lambda_0\,, & \text{if } i \neq j \\ 1 - \sum_k Q_{ik}/\lambda_0\,, & \text{otherwise} \end{cases}.$$

The transient distribution at t can be obtained, by weighting the k-step probabilities of the DTMC by a Poisson distribution with rate $\lambda_0 t$:

$$\pi(t) = \sum_{k=0}^{\infty} \pi_0 P^k \frac{(\lambda_0 t)^k}{k!} \exp(\lambda_0 t)\,.$$

Truncation of this series clearly gives an underapproximation.

MONTE CARLO SIMULATION A simple way is estimation using Monte Carlo methods. This entails stochastic simulation of many trajectories of the CTMC. Generating a trajectory is straightforward: Given that the process is in a particular state s_i the a transition has to be sampled along with the residence time in state s_i. The naive approach is to sample an exponential random variable for each q_{ij} and choose the one firing first. This algorithm can be improved by sampling a reaction directly and sampling the residence time separately. This algorithm will be shown later in the context of MPMs.

↪ *page 17*

2.2 MARKOVIAN POPULATION MODELS

An Markovian population model (MPM) among agents of n_S distinct types in a well-stirred system. Other names for this model class are population CTMC (pCTMC), chemical reaction network (CRN), and stochastic reaction network (SRN). The system is given by a continuous-time stochastic process $\{X_t\}_{t \geqslant 0}$. It models only the number of agents according to their type. Therefore the process takes n_S-dimensional vectors of natural numbers as values, i.e. the state-space is $\mathcal{S} \subseteq \mathbb{N}^{n_S}$. By only considering the number of agents, we are neglecting factors such as spatial variations in agent density or other factors influencing interactions. The assumption of all agents being equally distributed in space is called the *well-stirredness* assumptions.

"The secret to modelling is not being perfect."

— Karl Lagerfeld

These assumptions bring with them, the convenience of considering populations as a whole. The single agent is of no importance to the dynamics of the process. Just the overall size of each population determines the stochastic evolution of the process. Another consequence of this assumption is the limiting behaviour under this assumption. If all populations are proportionally scaled to infinity, their concentrations can be accurately described by deterministic ODEs. In fact, this is the most widely used paradigm to analyze this kind of reaction networks. However this methodology may by widely inaccurate. Consider, for example, an epidemic process. Typically individuals are not well-stirred in a societal context and the influence of specific contact structures is crucial to the process' dynamics (Großmann, Backenköhler, and Wolf, 2020). Furthermore such a process typically exhibits discrete stochastic effects, such as the epidemic dying out. While the latter effect is retained in an MPM, the former is already lost. Therefore, a great deal of care has to be taken by the modeller which kinds of abstractions are appropriate for the chosen abstraction.

Other assumptions, such as exponentially distributed firing times are discussed below.

Interactions between agents are expressed as *reactions*. These reactions have associated gains and losses of agents, given by non-negative integer vectors ν_j^- and ν_j^+ for reaction j, respectively. The overall change by a reaction is given by the vector $\nu_j = \nu_j^+ - \nu_j^-$.

A reaction between agents of types $S_1, \dots, S_{n_S}$ is specified in the following form:

$$\sum_{\ell=1}^{n_S} \nu_{j\ell}^{-} S_\ell \xrightarrow{\alpha_j(x)} \sum_{\ell=1}^{n_S} \nu_{j\ell}^{+} S_\ell \,. \tag{2.12}$$

The propensity function α_j gives the rate of the exponentially distributed firing time of the reaction as a function of the current system state $x \in \mathcal{S}$. Thus, for reaction j we have the intensity

$$\Pr\left(X_{t+h} = x + \nu_j \mid X_t = x\right) = \alpha_j(x) + o(ht)\,. \tag{2.13}$$

In most physical models, *mass-action* propensities are most common. These model combinatorial nature of well-mixed molecules moving randombly through space: In a reaction

$$A + B \xrightarrow{c x^{(A)} x^{(B)}} C$$

two molecules hit eachother with a probability, proportional to the product of their counts

$$c X_t^{(A)} X_t^{(B)}\,.$$

In general such rates are given by the product of the number of reactant combinations in x and a *rate constant* c_j, i.e.

$$\alpha_j(x) \coloneqq c_j \prod_{\ell=1}^{n_S} \binom{x^{(S_\ell)}}{\nu_{j\ell}^{-}}\,. \tag{2.14}$$

This notation avoids conflicts, when we use the subscript for time or as an index.

In this case, we give the rate constant in (2.12) instead of the function α_j. We use the superscript notation $x^{(A)}$ to denote the index corresponding to species A in some vector of length n_S.

According to (2.13) the stochastic process $\{X_t\}_{t \geqslant 0}$ describing the evolution of the population sizes over time t is a continuous-time Markov chain (CTMC). The infinitesimal generator matrix Q has the entries

$$Q_{x,y} = \begin{cases} \sum_{j : x + \nu_j = y} \alpha_j(x)\,, & \text{if } x \neq y, \\ -\sum_{j=1}^{n_R} \alpha_j(x)\,, & \text{otherwise.} \end{cases} \tag{2.15}$$

Note that in addition mild regularity assumptions are necessary for the existence of a unique CTMC X, such as non-explosiveness (Anderson, 2012). These assumptions are typically valid for realistic reaction networks. The probability distribution over time is given by an initial value problem. Given an initial state x_0, the probabilities

$$\pi(x_i, t) \coloneqq \Pr(X_t = x_i \mid X_0 = x_0), \quad t \geqslant 0, \quad x \in \mathcal{S} \tag{2.16}$$

evolve according to the Kolmogorov forward equation

$$\frac{d}{dt}\pi(t) = \pi(t)Q\,, \tag{2.17}$$

where $\pi(t)$ is an arbitrary vectorization

$$(\pi(x_1, t), \pi(x_2, t), \ldots, \pi(x_{|\mathcal{S}|}, t))$$

of the state probabilities. Often (2.17) is given for a single state. In this form – due to its usage in quantitative biology – it is commonly referred to as the *chemical master equation (CME)*

$$\frac{d\pi}{dt}(x, t) = \sum_{j=1}^{n_R} \left(\alpha_j(x - \nu_j)\pi(x - \nu_j, t) - \alpha_j(x)\pi(x, t)\right)\,. \tag{2.18}$$

EXAMPLE Consider the following simple MPM with non-linear propensities as an example.

Model 1 (Dimerization)**.** *We first examine a simple dimerization model on an unbounded state-space with reactions*

$$\varnothing \xrightarrow{\lambda} M, \quad 2M \xrightarrow{\delta} D$$

and initial condition $X_0^{(M)} = X_0^{(D)} = 0$.

The semantics is given by a CTMC $X_t = (X_t^{(M)}, X_t^{(D)})^T$, where $(S_1, S_2) = (M, D)$. The reaction propensities according to (2.14) are

$$\alpha_1(x) = \lambda \quad \text{and} \quad \alpha_2(x) = \delta\, x^{(M)}(x^{(M)} - 1)/2\,.$$

The change vectors for the first reaction are

$$\nu_1^- = (0,0)^T \quad \text{and} \quad \nu_1^+ = (1,0)^T .$$

For the second reaction the change vectors are

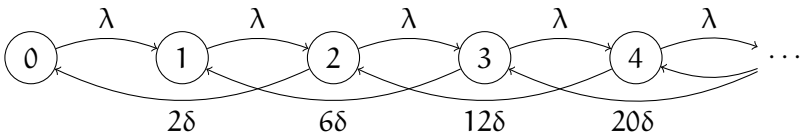
$$\nu_2^- = (2,0)^T \quad \text{and} \quad \nu_2^+ = (0,1)^T .$$

Consequently, $\nu_1 = (1,0)^T$ and $\nu_2 = (-2,1)^T$.

The generator matrix

$$Q = \begin{bmatrix} -\lambda & \lambda & 0 & & & \cdots \\ 0 & -\lambda & \lambda & 0 & & \cdots \\ 2\delta & 0 & -(\lambda+2\delta) & \lambda & 0 & \cdots \\ 0 & 6\delta & 0 & -(\lambda+6\delta) & \lambda & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \end{bmatrix} .$$

Often it is a good idea to visualize the state-space and transitions as a graph. This graph is constructed by interpreting the Q-matrix as an adjacency matrix for some subset of states.



For a state $(x^{(M)}, x^{(D)}) \in \mathbb{N}^2$, where $x^{(M)} \geqslant 2$, the CME (2.18) becomes

$$\begin{aligned} &\frac{d}{dt}\pi((x^{(M)}, x^{(D)}), t) \\ = &\frac{\delta}{2}(x^{(M)}+2)(x^{(M)}+1)\pi((x^{(M)}+2, x^{(D)}-1), t) \\ &-(\lambda + \frac{\delta}{2}x^{(M)}(x^{(M)}-1))\pi((x^{(M)}, x^{(D)}), t) \\ &+\lambda\pi((x^{(M)}-1, x^{(D)}), t) \end{aligned}$$

◇

The explicit representation of all state probabilities is often not possible, because there are infinitely many states. Usually the state-space is truncated to contain all relevant states (Andreychenko et al., 2011) or one switches to an approximation such as the mean-field (Bortolussi et al., 2013)

2.3 STATE-SPACE TRUNCATION

A complete solution of (2.18) is usually not possible. If the state-space with non-negligible probability is suitably small, a state space truncation can be performed. That is, (2.18) is integrated on a possibly time-dependent subset $\hat{\mathcal{S}}_t \subseteq \mathcal{S}$ (Henzinger, Mateescu, and Wolf, 2009; Munsky and Khammash, 2006; Spieler, 2014). Transitions to states, that are not part of this subset are typically re-directed to a introduced sink-state. This state captures the mass "lost" by the approximation and gives the error up to the numerical integration scheme. Munsky and Khammash (2006) coined the term of finite state projection (FSP) for such a method.

Uniformization can give a lower bound on the error.

To analyze the stationary distribution (Section 2.6) the redirection scheme needs to be altered (Kuntz et al., 2021b): Instead of a re-redirection into a sink-state, transitions are redirected in *some fashion* back into the truncation set.

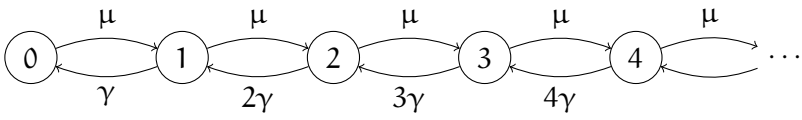
EXAMPLE Consider a birth-death process as a simple example. This model is used to describe a wide variety of phenomena and often constitutes a sub-module of larger models. For example, it represents an M/M/1 queue with service rates being linearly dependent on the queue length. Note that even for this simple model, the state-space is countably infinite.

Model 2 (Birth-Death Process)**.** *The model consists of exponentially distributed arrivals and service times proportional to queue length. It can be expressed using two mass-action reactions:*

$$\varnothing \xrightarrow{\mu} S \qquad \textit{and} \qquad S \xrightarrow{\gamma} \varnothing\,.$$

The initial condition $X_0 = 0$ holds with probability one.

The underlying CTMC has the following infinite structure.



The change of probability mass in a single state $x > 0$ is described by expanding (2.18) and

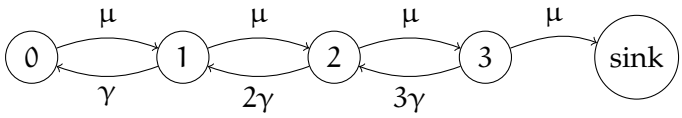
$$\frac{d}{dt}\pi_t(x) = \mu\pi(x-1,t) + \gamma(x+1)\pi(x+1,t) - (\mu+\gamma x)\pi(x,t)\,. \quad (2.19)$$

A typical truncation scheme to $[0, N]$ would use (2.19) for $x \in [0, \dots, N-1]$ and for $x = N$ the transition to $N+1$ via the first reaction is re-directed into a sink state. We can drop the sink state because of the invariant $\sum_x \pi_t(x) = 1, \forall t \geqslant 0$. The ODE for the boundary state would consequently read

In general, we can drop one equation in the CME system because of this invariant.

$$\frac{d}{dt}\pi_t(N) = \mu\pi(N-1,t) - (\gamma N + \mu)\pi(N,t)\,.$$

Visualizing the resulting CTMC and letting $N = 3$.

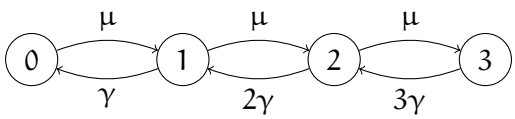


In a forward analysis, the sink state accumulates mass "lost" by the approximation. This provides a good practical error bound for the chosen truncation.

The accuracy of the bound is dependent on the specific method of the forward analysis. Numerical integration, for example, it is less strict.

If one is interested in the stationary probability distribution (Section 2.6), a sink state is not practical, since – in the long run – the process will usually enter such a state. Instead, the cut transitions are re-directed back into the truncation. A plethora of re-directions is possible (Gupta, Mikelson, and Khammash, 2017; Kuntz et al., 2021b; Spieler, 2014), but in general it is reasonable to re-direct to states with cut incoming transitions. In this example, this is equivalent to cutting the transition, because the truncation has only this border state.

Such re-directions are used in Chapter 6.



Thus, the ODE for state N here reads

$$\frac{d}{dt}\pi_t(N) = \mu\pi(N-1,t) - \gamma N\pi(N,t)\,.$$

◇

2.4 STOCHASTIC SIMULATIONS

We can generate trajectories of this model using the stochastic simulation algorithm (SSA) (Algorithm 1) (Gillespie, 1977). The simulation algorithm consists of repeatedly evaluating the race condition and jump times induced by (2.15) until some terminal criterion such as a maximum simulation time T is reached (line 2). In particular, the algorithm iteratively chooses a reaction, with a probability that is proportional to its rate given the current state s (line 4). The jump time $t_j - t_{j+1}$ is determined by sampling from an exponential distribution with rate $\sum_i \alpha_i(s)$ (line 5).

Algorithm 1: Sample a trajectory

1 $\tau \leftarrow$ empty list, $s \leftarrow$ sample from π_0, $t \leftarrow 0$;
2 **while** $t < T$ **do**
3 $\quad\tau \leftarrow \text{append}(\tau, (s,t))$;
4 $\quad k \leftarrow$ sample reaction i with probability $\alpha_i(s)/\sum_i \alpha_i(s)$;
5 $\quad\delta \sim \text{Exp}\left(\sum_i \alpha_i(s)\right)$;
6 $\quad s \leftarrow s + \nu_k$;
7 $\quad t \leftarrow t + \delta$;
8 **return** τ;

The output of Algorithm 1 is an alternating sequence of states and jump times

$$\tau = s_0 t_0 s_1 t_1 \ldots t_n s_n, \quad t \in [0, T]$$

called a *trajectory*.

Monte Carlo estimation requires a sufficiently large number of such trajectories to be generated. This collection of realizations facilitates a statistical estimate of a wide range of quantities such as expected values and probabilities. The main benefit of this

approach is its flexibility. It can solve essentially all relevant tasks. The main drawback is the cost associated with the generation of a sufficiently large trajectory ensemble. This problem becomes very pronounced in case of rare event probability estimation and stiff systems. Furthermore, the results only provide – by their very nature – statistical guarantees. Other methods such as explicit state-space representations can give stronger guarantees.

Bespoke techniques for these scenarios are available (Cao, Gillespie, and Petzold, 2005; Daigle Jr et al., 2011).

2.5 MOMENT DYNAMICS

Often times, the moments of a stochastic process provide sufficient information for its analysis. ODEs for the expected values of an MPM can be derived using its generator. We can apply Q to a polynomial f such that

This is called the drift *(see Section 2.6.1).*

$$Qf(x) = \sum_{j=1}^{n_R} \left(f(x+\nu_j) - f(x)\right) \alpha_j(x)\,. \tag{2.20}$$

By definition of the generator (2.5)

$$\frac{d}{dt}E\left(f(X_t)|X_t = x\right) = Qf(x)\,.$$

Left-multiplying the probability of a given state and summing up over all states, we obtain the time derivative of the expected value $E(f(X_t))$. Written as a matrix vector product

$$\frac{d}{dt}(\pi f) = \pi Q f\,.$$

For MPMs, in particular, we arrive at the functional form

$$\frac{d}{dt}E\left(f(X_t)\right) = \sum_{j=1}^{n_R} E\left(\left(f(X_t+\nu_j) - f(X_t)\right)\alpha_j(X_t)\right)\,. \tag{2.21}$$

Centered *moments (e.g. variance) are equivalent via the binomial transform.*

This equation is used to analyse (raw) *moments* of the process. A raw moment is

$$E\left(X^m\right) = E\left(\prod_{i=1}^{n_S} X_i^{m_i}\right)\,, \quad m \in \mathbb{N}^{n_S}$$

with respect to some probability measure. An ODE is given by (2.21), setting $f(x) = x^m$. The *order* of a moment $E(X^m)$ is given by the sum of its exponents, i.e. $\sum_i m_i$. Note that the notion of expected value can be generalized to any measure μ on a Borel-measurable space $(E, \mathcal{B}(E))$, where the m-th raw moment is $\int_E x^m \, d\mu(x)$. Throughout we assume that moments of arbitrary order remain finite over time, i.e. $E(|X_t^m|) < \infty$, $t \geqslant 0$. In Gupta, Briat, and Khammash (2014) the authors propose a framework to verify this property for a given model.

EXAMPLE Let us express the dynamics of the first two uncentered moments for Model 2 using (2.21).

$$\begin{aligned} \frac{d}{dt} E(X_t) &= \mu - \gamma E(X_t) \\ \frac{d}{dt} E\left(X_t^2\right) &= \mu(2E(X_t) + 1) - \gamma(2E\left(X_t^2\right) - E(X_t)) \end{aligned} \tag{2.22}$$

Setting initial moments these equations give as an IVP, we can solve (see Figure 2.1). This, however, is more an exception than the norm: Unless all ractions have linear or constant rate functions $\alpha_i(\cdot)$, $\forall i$, we would not end up with a closed system of ODEs as in (2.22). To illustrate, let us pretend the reaction ($S \xrightarrow{\gamma} \varnothing$) would become this non-linear reaction

The model is then the same as Model 1.

$$2S \xrightarrow{\gamma} \varnothing \, .$$

Accordingly, due to mass-action (2.14)

$$\alpha_2(x) = \gamma(x^2 - x) \, .$$

Therefore the first moment's derivative becomes

$$\frac{d}{dt} E(X_t) = \mu - \gamma \left(E\left(X_t^2\right) - E(X_t)\right) \, .$$

Note, that now the right-hand side of the derivative in the example depends on the value of the second moment $E\left(X_t^2\right)$. ◇

If we consider the general expression (2.21) for the moment of order k clearly a term of order $k+1$ occurs, that does (usually) not cancel out if a propensity function is at least quadratic.

Therefore, researches commonly rely on ad-hoc approximations to truncate this infinite system of ODEs (Hespanha, 2008; Schnoerr, Sanguinetti, and Grima, 2014, 2015). Unfortunately such schemes have typically no guarantees to converge – or even improve – with increasing truncation order (Schnoerr, Sanguinetti, and Grima, 2014) or increasing system size. Furthermore, fairly involved numerical schemes have to be employed to recover distributional approximations (Andreychenko et al., 2017). The only scheme with a convergence guarantee in the system size limit is the *mean-field* approximation (Bortolussi et al., 2013). Therein zero-covariances are assumed, i.e. the system is truncated at the first oder equations using the approximation $\mathrm{E}\left(X_t^2\right) = \mathrm{E}\left(X_t\right)^2$.

This finite moment problem could also be posed as a generalized moment problem. The resulting semi-definite program (SDP) can be solved numerically (Lasserre, 2010).

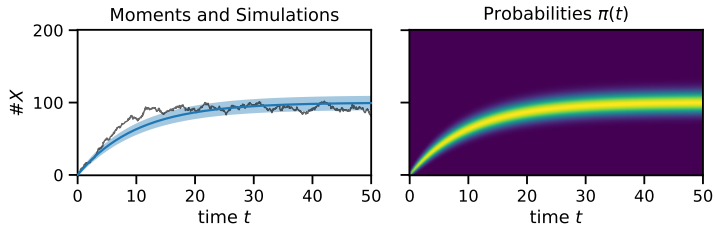


Figure 2.1: The expected value ± a standard deviation along with a sampled trajectory (left) and the probability distribution over time (right) of Model 2 with $\mu = 10$ and $\gamma = 0.1$.

2.5.1 *Hybrid Representations*

- Equivalent results are obtained by modifying the network to have 1 species per abundancy (basically rule based)

(Andreychenko, Mikeev, and Wolf, 2015; Hasenauer et al., 2014; Kazeroonian, Theis, and Hasenauer, 2014)

2.6 STATIONARY DISTRIBUTION

Assuming ergodicity of the underlying chain, a stationary distribution π_∞ is an invariant distribution, namely a fixed point of

the Kolmogorov forward equation (2.17). Let π_∞ be the vector description of a stationary distribution. It then satisfies

$$0 = \pi_\infty Q \tag{2.23}$$

as a fixed point of the Kolmogorov equation (2.17). Furthermore, the solution is constrained, to form a probability distribution, i.e. a measure with unit mass. Thus,

$$1 = \sum_{x \in \mathcal{S}} \pi_\infty(x) \,. \tag{2.24}$$

Stationary distributions are connected to the *long-run* behavior of an MPM (Dayar et al., 2011), as the system's distribution will converge to the (unique) stationary distribution. The connection of the stationary distribution to the long-run behavior becomes clear when considering the ergodic theorem. For some $A \subseteq \mathcal{S}$,

$$\lim_{T\to\infty} \frac{1}{T} \int_0^T 1_A(X_t)\, dt = \sum_{x \in A} \pi_\infty(x) \,. \tag{2.25}$$

Thus, the mean occupation time for set A over infinite trajectories is the stationary measure for A. Eq. (2.25) shows that we can assess long-run behavior using the stationary distribution and vice-versa.

EXAMPLE Returning to the example of Model 2 it is obvious that the state-space is irreducible. Further, we can easily show, that the stationary distribution is Poissonian with rate μ/γ:

$$\pi_\infty(x) = \frac{(\mu/\gamma)^x \exp(-\mu/\gamma)}{x!} \,.$$

◇

For simplicity, we assume throughout that the state-space is composed of a single communicating class. Checking ergodicity given a countably infinite number of states is achieved by providing a suitable Foster-Lyapunov function (Meyn and Tweedie, 2012). Some automated techniques have been proposed for this task (Dayar et al., 2011; Gupta, Briat, and Khammash, 2014; Milias-Argeitis and Khammash, 2014).

2.6.1 *Foster-Lyapunov Bounds*

It is well-known that for a CTMC X, ergodicity can be proven by a Foster-Lyapunov function $g : \mathcal{S} \to \mathbb{R}$ (Dayar et al., 2011; Meyn and Tweedie, 1993). This is the stochastic analogue of the Lyapunov functions, used to prove convergence of ODEs. The function g is required to have finite level sets:

$$|\{x \in \mathcal{S} \mid g(x) < l\}| < \infty, \quad \forall l > 0.$$

Typical choices for g are linear (Gupta, Briat, and Khammash, 2014; Milias-Argeitis and Khammash, 2014) or quadratic (Spieler, 2014). Given the g, we define its *drift* d as its average infinitesimal change, which is obtained applying the generator Q to g.

Intuitively, g is interpreted as a vector with the values $f(x_i)_i = f_i$ with the same ordering as in Q.

$$d(x) = Qg(x) = \sum_{j=1}^{n_R} \alpha_j(x)(g(x + \nu_j) - g(x)) \tag{2.26}$$

Note that we end up with (2.21) taking the expectation.

As such the drift can be interpreted as the expected local tendency of change of a scalar valued function g, i.e.

$$d(x) = \frac{d}{dt}\mathbb{E}\left(g(X_t) \mid X_t = x\right).$$

A Lyapunov function can be used to prove ergodicity of a CTMC: If there is a finite subset $C \subset \mathcal{S}$ such that

$$Qg(x) \leqslant -1, \ \forall x \in \mathcal{S} \setminus C, \tag{2.27}$$

$$Qg(x) < \infty, \ \forall x \in C, \text{ and} \tag{2.28}$$

$$\|x\| \to \infty \Rightarrow g(x) \to \infty, \quad \text{where} \quad \|x\| = \sum_i x_i, \tag{2.29}$$

then the chain is non-explosive and ergodic (Milias-Argeitis and Khammash, 2014; Tweedie, 1975). Intuitively, $g(X_t)$ should be a supermartingale outside C and a submartingale inside. Since the g is norm-like due to (2.29) this entails that the process tends towards C when outside, and out of from C when inside.

Given the above requirements

$$\mathcal{C}_{\epsilon_\ell} = \left\{x \in \mathcal{S} \mid \frac{\epsilon_\ell}{c} d(x) > \epsilon_\ell - 1\right\} \tag{2.30}$$

is finite, where

$$\infty > c \geqslant \sup_{x \in S} d(x) .$$

In this case, $\mathcal{C}_{\epsilon_\ell}$ contains at least $1 - \epsilon_\ell$ of stationary probability mass for any $\epsilon_\ell \in (0, 1)$ (Spieler, 2014, Thm. 8). Given that $\mathcal{C}_{\epsilon_\ell}$ is finite, the chain is ergodic and

$$\sum_{x \in \mathcal{C}_{\epsilon_\ell}} \pi(x) > 1 - \epsilon_\ell \tag{2.31}$$

bounding the stationary probability mass contained within $\mathcal{C}_{\epsilon_\ell}$.

EXAMPLE We return to Model 2 and choose $g(x) = x$. Then

$$d(x) = \mu - \gamma x .$$

The requirements (2.27), (2.28), and (2.29) are fulfilled for

$$C = \left\{ 0, \ldots, \frac{1+\mu}{\gamma} - 1 \right\}$$

and the underlying chain is ergodic. We can further bound the stationary distribution. Letting $c = \mu$,

$$\frac{\epsilon_\ell}{\mu} d(x) = \epsilon_\ell - \frac{\epsilon_\ell \gamma}{\mu} x$$

and following (2.30) the states

$$0 \leqslant x < \frac{\mu}{\epsilon_\ell \gamma}$$

have at least $1 - \epsilon_\ell$ stationary mass. ◇

2.7 A BRIEF TAXONOMY OF MULTIMODALITY

Multimodality is an overloaded term in the context of reaction network models. Specifically, it can be used to describe the following features.

OPERATIONAL MULTIMODALITY This kind of multimodality characterizes the model behavior directly. Consider, for example, a gene expression model: The gene state is digital, meaning it is either active or inactive. Depending on this state a protein is either synthesized or not. Therefore the system has distinct *operational modes* which dictate its dynamics. Non-biological examples can be found in the context of broadcasting systems, in which the dynamics change discretely due to messages shared between agents (Bortolussi, Hillston, and Loreti, 2020).

See Model 4 on page 52 for an example of this.

Naturally, nearly all models change their dynamics, given a change in their state vector. In this aspect this distinction is not wholly strict. It is mainly intended to indicate distinctive changes in the dynamics. In some instances, these can be as obvious as the examples mentioned above. In other cases they may not be as obvious: Consider an epidemics die-out for example, which is a significant change in the operating mode, which is not due to a switch-type reaction.

DISTRIBUTIONAL MULTIMODALITY Here the multimodality refers to the stationary distribution π_∞: It has multiple *modes*, i.e. local maxima. Given an ergodic underlying CTMC this entails, that the system spends most time in distinct regions of the state-space (cf. (2.25)). The *switching* between those distinct region is of interest for both, analysis and control, of such models.

While distributional multimodality implies operational multimodality, the reverse does not hold.

Literature:

- (Siegal-Gaskins et al., 2011)

Part II

MOMENT-BASED METHODS

We use moment properties to provide bounds on mean first-passage times and to improve statistical estimation of different quantities.

3 BOUNDING MEAN FIRST-PASSAGE TIMES

For the quantitative analysis of CTMCs, many approaches have been developed, where properties of interest are often expressed in terms of temporal logics such as CSL (Aziz et al., 1996; Baier et al., 2003, 2000), MTL (Chen et al., 2011), and specifications for timed-automata (Chen et al., 2009; Mikeev et al., 2013). In addition, there exist efficient software tools (Dehnert et al., 2017; Hinton et al., 2006; Kwiatkowska, Norman, and Parker, 2011) that can be used to analyze and verify system properties. The computation of reachability probabilities is a central problem in this context.

Popular exact methods for CTMCs rely on numerical approaches that explicitly consider each system state individually. A major problem is that these methods cannot scale in the context of population models with large copy numbers of agents. A popular alternative to tackle this problem is statistical model checking, which is based on stochastic simulation (David et al., 2015). For MPMs arising in the context of chemical reaction networks, trajectories of the process are usually generated using the SSA (Gillespie, 1977). However, since the number of possible interactions grows with the number of agents, stochastic simulations of MPMs are time-consuming. Moreover, they are subject to inherent statistical uncertainty and give only statistically estimated bounds.

As an alternative, recent work concentrates on numerical methods that approximate the statistical moments of the system without the need to compute the probability of each state. For groups of identically behaving agents, it is possible to derive systems of differential equations for the evolution of the statistical population moments (Bogomolov et al., 2015; Bortolussi and Lanciani, 2013; Engblom, 2006; Gast, Bortolussi, and Tribastone, 2019; Schnoerr, Sanguinetti, and Grima, 2015, 2017). However, as the system of exact moment equations is infinite-dimensional, approximation schemes typically rely on certain assumptions about the underlying probability distribution to truncate it. For example, one might employ a "low dispersion closure" which assumes ↪ *page 15*

that higher-order moments are the same as those of a normal distribution (Hespanha, 2008). Such approximations are, by nature, ad-hoc and do not come with any guarantees.

Moment-based methods often scale well in terms of population sizes. However, it is not possible to control the effects of the introduced approximations, which in some cases can lead to large errors (Schnoerr, Sanguinetti, and Grima, 2015). This issue reverberates on the application of these methods to compute reachability probabilities and mean first-passage times (Bortolussi and Lanciani, 2013, 2014; Hayden, Stefanek, and Bradley, 2012). Moreover, they can suffer from numerical instabilities, in particular, when the maximum order of the considered moments has to be increased to more appropriately describe the underlying distribution.

Here, we put forward a method based solely on moments that gives *exact bounds* for mean first-passage times (MFPTs) and reachability probabilities in MPMs. For a set of states B and a time-horizon T, the first-passage time (FPT)

$$\tau = \inf\{t \geqslant 0 \mid X_t \in B\} \wedge T\,.$$

This mean of this stopping time $E(\tau)$, i.e. the mean first-passage time, directly characterizes the probability of reaching set B within T time units. Thus, safe upper and lower bounds on MFPTs can constitute a core component for the verification of properties in MPMs. Our approach extends recent work on moment bounds (Dowdy and Barton, 2018b; Sakurai and Hori, 2017) and it is based on a martingale formulation of the stopped process that we derive from the exact moment equations. From this formalization, we deduce a set of linear moment constraints from which we derive upper and lower moment bounds using semidefinite program (SDP). Monotone sequences of both upper and lower bounds can be obtained by increasing the order of the relaxation. Crucially, no closure approximations are introduced. Therefore the bounds are exact up to the numerical accuracy of the SDP solver.

To experimentally validate our method in terms of accuracy and feasibility, we run some tests on examples from biology, leveraging an existing SDP solver and obtaining encouraging results.

Comparing with other moment-based methods, our approach is not based on approximations due to closure schemes, thus providing guarantees on the bounds up to the numerical accuracy of the computations. However, similarly to other moment-based methods, we also found the insurgence of numerical instabilities because moments of higher order tend to span over many orders of magnitude. We ameliorate this problem by considering scaling strategies that reduce such variability. We also extend our approach to deal with MPMs exhibiting strong multimodal behavior, due to the presence of populations having low copy numbers. This extension exploits some ideas from hybrid moment closures (Kazeroonian, Theis, and Hasenauer, 2014).
In summary, this chapter presents the following novel contributions:

- the derivation of moment constraints, based on a martingale formulation, for bounding mean first-passage times and reachability probabilities using a convex programming scheme;
- the extension of this scheme using hybrid moment conditions to systems exhibiting multimodal behavior;

3.1 RELATED WORK

TRUNCATIONS AND ANALYTIC SOLUTIONS Considerable effort has been directed at the analysis of first-passage time distributions in MPMs. Most works can either focus on an explicit state-space analysis (Barzel and Biham, 2008; Kuntz et al., 2019b, 2021a; Munsky, Nemenman, and Bel, 2009) or employ approximation techniques for which, in general, no error bounds can be given (Bortolussi and Lanciani, 2014; Hayden, Stefanek, and Bradley, 2012; Schnoerr et al., 2017). For some model classes such as kinetic proofreading, analytic solutions are possible (Bel, Munsky, and Nemenman, 2009; Iyer-Biswas and Zilman, 2016; Munsky, Nemenman, and Bel, 2009).

Barzel and Biham (2008) propose a recursive scheme that consists of one equation for each state, expressing the average time

the system needs to transition from that state to the target state. Kuntz et al. (2021a) propose to employ moment bounds in a linear programming approach to compute exit time distribution using state-space truncation schemes. In Kuntz et al. (2019b) the authors propose a finite state-space projection scheme to bound first-passage time distributions

MOMENTS APPROXIMATIONS In Hayden, Stefanek, and Bradley (2012), the authors use moment closure approximations and Chebychev's inequality to gain an understanding of first-passage time dynamics. Schnoerr et al. (2017) also employ a moment closure approximation and further approximate threshold functions to derive an approximate first-passage time distribution. Bortolussi and Lanciani (2014) use a mean-field approximation which is required to reach the target region.

MOMENT BOUNDS USING OPTIMIZATION Recently, several groups independently suggested the use of semi-definite optimization for the computation of moment bounds for the limiting distribution (Dowdy and Barton, 2018a; Ghusinga et al., 2017; Kuntz et al., 2019a; Sakurai and Hori, 2017). In this approach, the differential equations describing the moment dynamics are set to zero and form linear constraints (Backenköhler, Bortolussi, and Wolf, 2018). Alongside, semi-definite constraints can be placed on the *moment matrices*. These give a semi-definite program that can be solved efficiently.

This approach has been extended to the transient case (Dowdy and Barton, 2018b; Sakurai and Hori, 2019). The approach is similar in both works and is a cornerstone of the MFPT analysis presented here. They differ mainly by the fact that Sakurai and Hori (2019) apply a polynomial time-weighting, while Dowdy and Barton (2018b) use an exponential one. We adopt the former approach because it can be naturally adapted to the description of densities over time. The resulting forms can also be adapted to statistical estimation problems (Backenköhler, Bortolussi, and Wolf, 2019).

Semi-definite programming has been applied to a wide range of problems, including stochastic processes in the context of financial mathematics (Kashima and Kawai, 2009; Lasserre, Prieto-Rumeau, and Zervos, 2006). For good introductions and overviews of application areas, we refer the reader to Parrilo (2003) and, more recently, Lasserre (2010).

BOUNDING MFTPS USING OPTIMIZATION Particularly relevant for this work is the application of convex optimization to FPT. Helmes, Röhl, and Stockbridge (2001) formulated a linear program using the Hausdorff moment conditions to bound moments of the FPT distribution in Markovian processes. Semi-definite optimization has been successfully applied in financial mathematics by Kashima and Kawai (2009), as well as Lasserre, Prieto-Rumeau, and Zervos (2006) to bound prices of exotic options.

3.2 PRELIMINARIES

In this work, we are interested in first-passage times (FPTs) of such processes. That is the time, the process first enters a set of target states $B \subseteq S$. Naturally, the analysis of FPTs is equivalent to the analysis of times at which the process exits the complement $S \setminus B$. More formally, the first-passage time τ for some target set B is defined as the random variable

$$\tau = \inf\{t \geqslant 0 \mid X_t \in B\}. \tag{3.1}$$

In this example, we are interested in the time at which the number of type M agents exceed some threshold H. With the framework presented in the sequel, one can bound the expected value of this time using semi-definite programming. Further, it is possible to impose a time-horizon T, and find bounds on the probability of $X_t^{(M)} \geqslant H$ for some $0 \leqslant t \leqslant T$. The employed framework is centered around semi-definite relaxations of the generalized moment problem (GMP) (Lasserre, 2010). These require linear constraints on the moments of measures. In the following section, we derive such constraints.

3.3 MARTINGALE FORMULATION

Next, we will discuss the ordinary differential equations for the evolution of the statistical moments of the process. The moments over the state-space are then used to derive temporal moments, i.e. moments of measures over both the state-space and the time. This extended description results in a process with the martingale property. This property can be used to formulate linear constraints on the temporal moments and, as a special case, the mean first-passage time. In combination with semi-definite properties of moment matrices, we can formulate mathematical programs that yield upper and lower bounds on mean first-passage times.

↪ *page 13*

Let f be a polynomial function, $t \geqslant 0$. Using the CME (2.18), we can derive ODEs describing the dynamics of $\mathrm{E}(f(X_t))$ (Engblom, 2006). Specifically,

More details on the derivation of the moment ODEs is given in Section 2.5.

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{E}\left(f(X_t)\right) = \sum_{j=1}^{n_R} \mathrm{E}\left(\left(f(X_t+\nu_j)-f(X_t)\right)\alpha_j(X_t)\right)\,. \tag{3.2}$$

EXAMPLE Let us consider Model 1 and agent type M as an example. Further, let $X_t = X_t^{(M)}$ for ease of exposition. When choosing $f(X_t) = X_t^m$, $m = 1$ and $m = 2$ we obtain two differential equations describing the change of the first two moments of species M, $\mathrm{E}\left(X_t\right)$ and $\mathrm{E}\left(X_t^2\right)$, respectively.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{E}\left(X_t\right) &= \lambda\mathrm{E}\left(X_t^0\right) - 2\delta\left(\mathrm{E}\left(X_t^2\right) - \mathrm{E}\left(X_t\right)\right)\\ \frac{\mathrm{d}}{\mathrm{d}t}\mathrm{E}\left(X_t^2\right) &= \lambda(2\mathrm{E}\left(X_t\right)+1)\\ &\quad -4\delta\left(\mathrm{E}\left(X_t^3\right) - 2\mathrm{E}\left(X_t^2\right) + \mathrm{E}\left(X_t\right)\right)\,.\end{aligned} \tag{3.3}$$

Fixing initial moments, the ODE system describes the moments over time exactly. However, these ODEs cannot be integrated because the system is not closed. The right-hand side for moment $\mathrm{E}\left(X_t^m\right)$ always contains $\mathrm{E}(X_t^{m+1})$. ◇

To solve the IVP, one typically resorts to ad-hoc approximations of the highest order moments to close the system. Here we do *not* need such approximations because we do not numerically integrate the moment equations. Instead we adopt an approach

(Dowdy and Barton, 2018b; Sakurai and Hori, 2019) that extends the description of state-space moments to a temporal one.

This is achieved by the introduction of a time-dependent polynomial $w(t)$ that is multiplied to (3.2). An integration by parts on $[0, T]$ yields (Dowdy and Barton, 2018b; Sakurai and Hori, 2019)

$$\begin{aligned} & w(T)\mathrm{E}\left(f(X_T)\right) - w(t_0)\mathrm{E}\left(f(X_{t_0})\right) - \int_{t_0}^{T} \frac{\mathrm{d}w(t)}{\mathrm{d}t}\mathrm{E}\left(f(X_t)\right)\,\mathrm{d}t \\ & \quad = \sum_{j=1}^{n_R} \int_{t_0}^{T} w(t)\mathrm{E}\left(\left(f(X_t + \nu_j) - f(X_t)\right)\alpha_j(X_t)\right)\,\mathrm{d}t\,. \end{aligned} \tag{3.4}$$

Starting from this equation, it is possible to derive a martingale process, i.e. a process that has an expected value equal to 0, regardless of time.

We now want to interchange the order of integration and the summation due to the expected value. To this end, we have to assume the absolute convergence of the integrals. On finite time intervals $[0, T]$ this holds because w is polynomial and we assumed finite moments for all $t \geqslant 0$. Interchanging the summation and integral of a monomial x^m, i.e. pulling all expectation operators outside

$$\begin{aligned} & \int_0^T g(t)\mathrm{E}\left(X_t^m\right)\,\mathrm{d}t \\ & = \int_0^T \sum_{x\in\mathcal{S}} g(t)\Pr(X_s = x)x^m\,\mathrm{d}t \\ & = \int_0^T \int_\Omega g(t)X_s(\omega)^m\,\mathrm{d}P(\omega)\,\mathrm{d}t \\ & = \int_\Omega \int_0^T g(t)X_s(\omega)^m\,\mathrm{d}t\,\mathrm{d}P(\omega) \\ & = \mathrm{E}\left(\int_0^T g(t)X_t^m\,\mathrm{d}t\right). \end{aligned}$$

Hence, we are able to to pull out the expectation operator in (3.4).

$$0 = w(T)\,\mathrm{E}\left(f(X_T)\right) - w(0)\,\mathrm{E}\left(f(X_0)\right) - \mathrm{E}\left(\int_0^T \frac{\mathrm{d}w(t)}{\mathrm{d}t} f(X_t)\,\mathrm{d}t\right) - \sum_{j=1}^{n_R} \mathrm{E}\left(\int_0^T w(t)(f(X_t + \nu_j) - f(X_t))\alpha_j(X_t)\,\mathrm{d}t\right), \tag{3.5}$$

This gives us the expected value of a time-dependent function of the original process. The function can be viewed as a stochastic process of its own where the time-horizon T is the index variable. A key property of this process is also illustrated by (3.5): The process' expected value remains zero, regardless of the choice of T. This martingale property is particularly useful because it can be used to formulate linear constraints on stopping times of the process. Explicitly, we can define this process $\{Z_T\}_{T\geqslant 0}$ parameterized by the time-weighting w and polynomial f.

$$Z_T \coloneqq w(T)f(X_T) - w(0)f(X_0) - \int_0^T \frac{\mathrm{d}w(t)}{\mathrm{d}t} f(X_t)\,\mathrm{d}t - \sum_{j=1}^{n_R} \int_0^T w(t)(f(X_t + \nu_j) - f(X_t))\alpha_j(X_t)\,\mathrm{d}t\,. \tag{3.6}$$

In Chapter 4 we use an exponential weighting.

A useful choice for f and w are monomials. When choosing $w(t) = t^k$ with $k \in \mathbb{N}$ and $f(X) = X^m$ the process takes the form

$$Z_T^{(m,k)} = T^k X_T^m - 0^k X_0^m + \sum_i c_i \int_0^T t^{k_i} X_t^{m_i}\,\mathrm{d}t \tag{3.7}$$

where $(m_i)_i$, $(k_i)_i$, and $(c_i)_i$ are finite sequences resulting from the substitution of f and w and expansion of (3.6).

When choosing $w(t) = t^k$ with $k \in \mathbb{N}$ and $f(X) = X^m$ this process takes the form

$$Z_T^{(m,k)} = T^k X_T^m - 0^k X_0^m + \sum_i c_i \int_0^T t^{k_i} X_t^{m_i}\,\mathrm{d}t \tag{3.8}$$

where $(m_i)_i$, $(k_i)_i$, and $(c_i)_i$ are finite sequences resulting from the substitution of f and w. This choice allows to naturally characterize the behavior in time and state-space as moments, because the expected value of (3.8) then becomes a linear form of moments. We will use these as constraints in the semi-definite program used to bound MFPTs.

EXAMPLE If we apply this to our previous example (cf. (3.3)), letting $m = 1$ and $k = 1$ we obtain the following process for Model 1.

$$Z_T^{(1,1)} = TX_T - \int_0^T X_t \, dt - \lambda \int_0^T t \, dt - 2\delta \int_0^T tX_t \, dt + 2\delta \int_0^T tX_t^2 \, dt,$$

where the sequences according to (3.8) are $(m_i)_i = (1, 0, 1, 2)$, $(k_i)_i = (0, 1, 1, 1)$, and $(c_i)_i = (-1, -\lambda, -2\delta, 2\delta)$. ◇

3.4 BOUNDS FOR MEAN FIRST-PASSAGE TIMES

We now turn to the analysis of first passage times within some time-bound $T > 0$. Given some subset of the state-space $B \subseteq S$ the first passage time is given by the continuous random variable

$$\tau = \inf\{t \geqslant 0 \mid X_t \in B\} \wedge T$$

where $a \wedge b := \min\{a, b\}$. For this chapter, we only look at threshold hitting times, i.e. we set a threshold H for species S and thus

$$B = \{x \mid x^{(S)} \geqslant H\}.$$

Note, that this framework allows for a more general class of target sets, which are discussed in Section 3.5. In the sequel, we will use τ as a stopping time in our martingale formulation and consider $Z_\tau^{(m,k)}$ instead of $Z_T^{(m,k)}$. Since (3.8) defines a martingale, $Z_\tau^{(m,k)}$ remains a martingale by Doob's optional sampling

theorem (Gihman and Skorohod, 1975). In particular, this implies that

$$E(Z_\tau^{(m,k)}) = 0 \tag{3.9}$$

for all moment orders m and degrees k in the weighting function $w(t)$.

3.4.1 *Linear Moment Constraints*

To simplify our presentation, we fix an initial state x_0, i.e. $P(X_0 = x_0) = 1$. Expanding (3.9) using (3.8) for $Z_\tau^{(m,k)}$ yields the following linear constraint on expected values.

$$0 = E\left(\tau^k X_\tau^m\right) - 0^k x_0^m + \sum_i c_i E\left(\int_0^\tau t^{k_i} X_t^{m_i}\, dt\right), \tag{3.10}$$

where $0^0 = 1$. Hence, we have established a relationship between the process dynamics up to the hitting time via expected values of the time-integrals and the final process state at the hitting time via $E\left(\tau^k X_\tau^m\right)$.

For the ease of exposition, we now turn to the analysis of first passage times in one-dimensional processes w.r.t. an upper threshold H. In particular, we will consider moments $E(X^m)$, $m = 0, 1, 2, \ldots$, of a one-dimensional process. The approach proposed in the sequel, however, can be extended to multi-dimensional processes and more complex target sets B.

EXAMPLE Consider again Model 1 and assume that we are interested in the time at which species M exceeds threshold H while fixing the considered time-horizon to $T = 4$. That is, we are interested in the stopping time

$$\tau = \inf\{t \geqslant 0 \mid X_t \geqslant 10\} \wedge 4\,.$$

Since the abundance of D does not influence M, we can ignore species D and treat the process as one-dimensional. Figure 3.1 shows three example trajectories: Two reach an upper threshold $H = 10$, while one reaches the final time-horizon $T = 4$. The

figure also illustrates another aspect present in (3.10). It gives a connection between the terminal distribution, i.e. the distribution of X_τ, and the dynamic behavior up to τ. The statistics at τ are described by a distribution whose moments are represented by the $E(\tau^k X_\tau{}^m)$ term in (3.10). This distribution corresponding two moments encompasses both cases of how τ can be reached. In the first case threshold H is reached and the second case the process reaches the time-horizon T. In the following we will define the interplay between these measures more formally. ◇

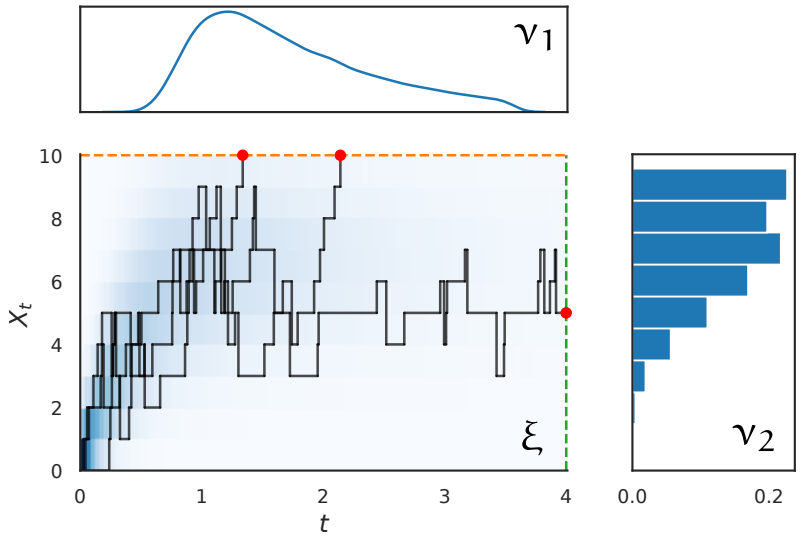


Figure 3.1: The relationship between the occupation measure ξ and the exit location probability measures ν_1 and ν_2. The shaded area indicates the structure of the occupation measure. Three example trajectories are additionally plotted with their exit location highlighted. The plots are based on 10,000 sample trajectories.

Eq. (3.10) describes a relationship between two measures (Lasserre, 2010, Chapter 9.2):

EXPECTED OCCUPATION MEASURE ξ describes the expected residence time inside a subset of the state-space and time. As such it is supported on $[0, H] \times [0, T]$:

$$\xi(A \times C) \coloneqq \mathrm{E}\left(\int_{[0,\tau]\cap C} 1_{\in A}(X_t)\,\mathrm{d}t\right), \tag{3.11}$$

EXIT LOCATION PROBABILITY ν gives the state probability associated with the stopping time τ. Therefore it is supported on $(\{H\} \times [0, T]) \cup ([0, H] \times \{T\})$:

$$\nu(A \times C) \coloneqq \Pr((X_\tau, \tau) \in A \times C), \tag{3.12}$$

where $A \times C$ is a measurable set, i.e. A and C are elements of the Borel σ-algebras on $[0, H]$ and $[0, T]$, respectively.

Using Figure 3.1, one can gain an intuition for these two measures. The expected occupation measure is shaded in blue. As the name implies $\xi(A \times C)$ tells us how much time the process spends in A up to τ restricting to the time instants belonging to C. In particular, $\xi([0, H] \times [0, T]) = \mathrm{E}(\tau)$. The exit location probability ν, while being a two-dimensional distribution, can be viewed as a composition of a density describing the time at which the process reaches H (if it does) and a probability mass function on the states of the process if the time-horizon is reached without exceeding H. We partition the measure ν into ν_1 and ν_2 by conditioning on $\tau = T$. Thus,

$$\nu_1(C) \coloneqq \Pr(\tau \in C, \tau < T)$$

and

$$\nu_2(A) \coloneqq \Pr(X_T \in A, \tau = T)$$

and hence $\nu(A \times C) = \nu_1(C) + \nu_2(A)$. To refer to the moments of these measures, we define *partial moments*

$$\mathrm{E}(g(X); f(Y) = y) \coloneqq \mathrm{E}(g(X) \mid f(Y) = y)\Pr(f(Y) = y),$$

for some polynomial g and some indicator function f. Then

$$\mathrm{E}\left(\tau^k X_\tau^m\right) = T^k \mathrm{E}(X_\tau^m; \tau = T) + H^m \mathrm{E}\left(\tau^k; \tau < T, X_\tau = H\right).$$

Therefore the linear moment constraints are

$$0 = T^k E\left(X_\tau^m; \tau = T\right) + H^m E\left(\tau^k; \tau < T, X_\tau = H\right) - 0^k x_0^m + \sum_i c_i E\left(\int_0^\tau t^{k_i} X_t^{m_i} \, dt\right) .$$

Next, we consider infinite sequences of partial moments $y_1 = (y_{1k})_{k\in\mathbb{N}}$, $y_2 = (y_{2m})_{m\in\mathbb{N}}$, and $z = (z_{mk})_{(m,k)^\intercal\in\mathbb{N}^2}$ of ν_1, ν_2, and ξ, respectively. In particular,

$$y_{1k} \coloneqq E\left(\tau^k; \tau < T\right) ,$$
$$y_{2m} \coloneqq E\left(X_\tau^m; \tau = T\right) ,$$

and

$$z_{km} \coloneqq E\left(\int_0^\tau t^k X_t^m \, dt\right) .$$

3.4.2 *Objective*

Given the above measures and their corresponding moments, we can now identify the moments we are particularly interested in. We formulate an optimization problem with variables corresponding to the moments defined above. The MFPT is exactly the zeroth moment of ξ,

$$z_{00} = E\left(\int_0^\tau 1_{\leqslant H}(X_t) \, dt\right) = E\left(\tau\right) .$$

Therefore z_{00} corresponds to the objective of the optimization problem that gives bounds for the MFPT. Furthermore, we can easily change the objective to the zeroth moment of ν_1,

$$y_{10} = E\left(\tau^0; \tau < T\right) = \Pr(\tau < T) .$$

This moment is the probability of reaching threshold H before reaching time-horizon T. Since the target set can be more complex, this formulation can be used to perform model checking on a wide variety of properties.

Moreover, it is possible to formulate objectives not directly corresponding to a raw moment such as the variance (Dowdy and Barton, 2018a; Sakurai and Hori, 2019).

3.4.3 *Hausdorff Constraints*

Before discussing the semi-definite constraints which we will use for the remainder of this study, we will briefly discuss the Hausdorff moment constraints. These constraints offer linear constraints on moments of bounded measures. The Hausdorff moment problem is the question whether an infinite sequence $(m_0, m_1, \dots)$ is a sequence of moments

$$m_k = \int_0^1 x^k \, d\mu(x), \quad \forall k \in \mathbb{N}$$

for some Borel-measure supported on the unit interval $[0, 1]$.

See also Feller (1971) for additional details.

Hausdorff (1921) came up with a necessary condition that

$$\int_0^1 x^\ell (1-x)^k \, d\mu(x) \geqslant 0, \quad \forall \ell, k \in \mathbb{N}\,. \tag{3.13}$$

The validity of this condition is easy to see since

$$x^\ell (1-x)^k \geqslant 0, \quad \forall x \in [0,1]\,. \tag{3.14}$$

Expanding this term yields a polynomial, which – by integration – provides a linear constraint on the moments. This moment condition has been used by Helmes, Röhl, and Stockbridge (2001) to bound MFPTs in a variety of stochastic processes. Since these conditions are linear in the moments they can solve linear programs instead of the semi-definite programs shown below.

EXAMPLE Let $\ell = 2$, $k = 2$. Then the (3.13) becomes

$$\int_0^1 x(1-x)^2 \, d\mu(x) = m_2 - 2m_3 + m_4 \geqslant 0$$

constraining the moments on $[0, 1]$. ◇

Since the Hausdorff moment problem is defined on $[0, 1]$, we need to adjust accordingly. Let the interval be $[0, H]$, $H > 0$ then we can simply change the variables. That means considering $y = x/H$ instead of x. Equivalently, we can apply this rescaling in (3.13) such that

$$\int_0^H x^\ell (H-x)^k \, d\mu(x) \geqslant 0, \quad \forall \ell, k \in \mathbb{N} \tag{3.15}$$

remains a valid constraint on $[0, H]$.

Generalizing (3.13) to multiple dimensions can be done by simply multiplying terms for each dimension in a similar manner. For n dimensions

$$\int_{[0,1]^n} \prod_{i=0}^{n} x_i^{\ell_i} (1 - x_i)^{k_i} \, d\mu(x) \geqslant 0$$

for all $\ell_i, k_i \in \mathbb{N}$ and $i \in \{0, \ldots, n\}$. With arbitrary positive upper bounds $H_1, H_2, \ldots, H_n$ this becomes

We treat the time horizon T exactly the same as the other bounds H_i here.

$$\int_{\times_{i=1}^{n} [0,H_i]} \prod_{i=0}^{n} x_i^{\ell_i} (H_i - x_i)^{k_i} \, d\mu(x) \geqslant 0 \tag{3.16}$$

This equation can simply be expanded using the multi-binomial theorem. This is just a vector version of the regular binomial theorem

$$(y + x)^k = \sum_{j=0}^{k} \binom{k}{j} y^j x^{k-j} .$$

That is, given n-dimensional vectors k, x, and y

$$\prod_{i=1}^{n} (y_i + x_i)^{k_i} = \sum_{j_1=0}^{k_1} \cdots \sum_{j_n=0}^{k_n} \prod_{i=1}^{n} \binom{k_i}{j_i} y_i^{j_i} x_i^{k_i - j_i} .$$

Thus, (3.16) becomes the linear moment constraint

We can interchange the integrals since all measures have finite support and mass.

$$\sum_{j_1=0}^{k_1} \cdots \sum_{j_n=0}^{k_n} \prod_{i=1}^{n} \binom{k_i}{j_i} H_i^{j_i} (-1)^{k_i - j_i} \int_{\times_{m=1}^{n} [0,H_m]} x^{k-j+\ell} \, d\mu(x) \geqslant 0 . \tag{3.17}$$

3.4.4 *Semi-Definite Constraints*

The linear constraints alone are not sufficient to identify moment bounds. We further leverage the fact that a necessary condition for a positive measure that the *moment matrices* are positive semi-definite. A matrix $M \in \mathbb{R}^{n \times n}$ is positive semi-definite, denoted by $M \succeq 0$ if and only if

$$v^T M v \geqslant 0 \quad \forall v \in \mathbb{R}^n .$$

EXAMPLE As an example, let us consider a one-dimensional random variable Z with moment sequence z. For moment order r, the entries of the $(r+1) \times (r+1)$ moment matrix $M_r(x)$ are given by the raw moments. In particular,

$$(M_r(z))_{ij} = z_{i+j-2} = \mathrm{E}\left(Z^{i+j-2}\right)$$

for $i, j \in \mathbb{N}_r$ where $\mathbb{N}_r = \{0, 1, \ldots, r\}$ and the maximum order in the matrix is $2r$. For instance,

$$M_1(x) = \begin{bmatrix} x_0 & x_1 \\ x_1 & x_2 \end{bmatrix} \tag{3.18}$$

needs to be positive semi-definite. By Sylvester's criterion this means $\det M_1 \geqslant 0$ and $x_0 \geqslant 0$. We can easily see that in this case this entails

$$\det M_1 = x_0 x_2 - x_1^2 = \mathrm{E}\left(X^2\right) - \mathrm{E}\left(X\right)^2 = \mathrm{Var}(X) \geqslant 0\,.$$

This restriction is natural since the variance cannot be negative. ◇

The restriction of the non-negative variance we saw in the example generalizes to moment matrices in form of a positive semi-definite constraint (Parrilo, 2003). This gives us the following restrictions on the moment matrices.

This constraint is valid for general positive measures — even if they do not model probabilities.

$$M_r(z) \succeq 0, \quad M_r(y_1) \succeq 0, \quad \text{and} \quad M_r(y_2) \succeq 0 \tag{3.19}$$

for arbitrary orders r, providing a first tranche of moment constraints.

Furthermore, we need to enforce the restriction of the measures ξ, ν_1, and ν_2 to their supports. This can be done, by defining non-negative polynomials on the intended support of the measure.

EXAMPLE The exit location probability ν_2 has support $[0, H]$. We can now define

$$u_H(t, x) = Hx - x^2, \quad x \in \mathbb{R}$$

as a polynomial that is non-negative on $[0, H]$. Using such polynomials, we can construct *localizing matrices*, which have to be

positive semi-definite (Lasserre, 2010). Applying u_H to the moment matrix of measure ν_2, i.e. $M_1(y_2)$

$$M_1(u_H, y_2) = \begin{bmatrix} Hy_{21} - y_{22} & Hy_{22} - y_{23} \\ Hy_{22} - y_{23} & Hy_{23} - y_{24} \end{bmatrix}$$

with the constraint $M_1(u_H, y_2) \succeq 0$, where the application of a polynomial such as u_H to a moment matrix is formally defined for the multidimensional case in Section 3.5. Similarly, let $u_T(t, x) = Tt - t^2$ to restrict ν_1 to $[0, T)$. The expected occupation measure ξ is constrained similarly to its domain $[0, H] \times [0, T]$. This gives us the following restrictions on the moment matrices.

$$\begin{aligned} &M_r(u_T, z) \succeq 0\,, \quad M_r(u_T, y_1) \succeq 0\,, \\ &M_r(u_H, z) \succeq 0\,, \quad M_r(u_H, y_2) \succeq 0\,. \end{aligned} \tag{3.20}$$

◇

3.5 MULTI-DIMENSIONAL GENERALIZATION

For a general multi-dimensional moment sequence

$$y = (E(X^m))_{m \in \mathbb{N}^{n_s}}\,,$$

the moment matrix is (Lasserre, 2010)

$$M_r(y)(\alpha, \beta) = y_{\alpha+\beta}, \quad \forall \alpha, \beta \in \mathbb{N}^n_r \tag{3.21}$$

where row and column indices, α and β, are ordered according to the canonical basis

$$\begin{aligned} v_r(x) = (1, x_1, x_2, \ldots, x_n, x_1^2, x_1x_2, \ldots \\ \ldots, x_1x_n, \ldots, x_1^r, \ldots, x_n^r)^T\,. \end{aligned} \tag{3.22}$$

Equivalently to (3.21),

$$M_r(y) = E\left(v_r(x)v_r(x)^T\right)\,.$$

For a moment sequence the semi-definite restriction $M_r(y) \succeq 0$ must hold.

Measures can be restricted to semi-algebraic sets

$$\{x \in \mathbb{R}^n \mid u_j(x) \geqslant 0, j = 1, \ldots, m\},$$

where u_j, $j = 1, \ldots, m$ are polynomials (Lasserre, 2010). This is done by placing restrictions on the localizing matrices. For each polynomial $u_i \in \mathbb{R}[x]$ with coefficient vector $u = \{u_\gamma\}$, i.e.

$$u(x) = \sum_{\gamma \in \mathbb{N}^n} u_\gamma x^\gamma,$$

the localizing matrix is

$$M_r(u, y)(\alpha, \beta) = \sum_{\gamma \in \mathbb{N}^n} u_\gamma y_{\gamma+\alpha+\beta}, \quad \forall \alpha, \beta \in \mathbb{N}^n_r.$$

Requiring that this matrix is positive semi-definite restricts the measure to $\{x \mid u_i(x) \geqslant 0\}$. This way we can, for example, restrict the moment sequence y to measures that are positive w.r.t. dimension j. Simply letting $u(x) = x_j$ and requiring $M_1(u, y) \succeq 0$ for $i = 1, \ldots, n_S$ gives us this restriction.

3.5.1 *A Semi-Definite Program*

With the linear constraints given in (3.10) and the semi-definite constraints (3.19) and (3.20) discussed in the previous sections, we can now formulate a semi-definite program (SDP) for any relaxation order $0 < r < \infty$. An SDP is a convex optimization problem over the set of positive semi-definite $n \times n$-matrices $\mathcal{X}$ under linear constraints:

$$\begin{aligned} \min_{X \in \mathcal{X}} \quad & \sum_{i,j} A^{(0)}_{ij} X_{ij} \\ \text{such that} \quad & X \succeq 0 \\ & \sum_{i,j} A^{(k)}_{ij} X_{ij} \leqslant b_k, \quad k = 1, \ldots, m \end{aligned} \tag{3.23}$$

with constant matrices $A^{(i)} \in \mathbb{R}^{n \times n}$, $i = 0, \ldots, m$ and constants $b_k \in \mathbb{R}$, $k = 1, \ldots, m$ to define a set of m linear constraints.

Such a problem is convex and can be solved efficiently using off-the-shelf solvers (Vandenberghe, 2010).

With each moment sequence x we associate a sequence proxy variables x' used in the optimization problem.

$$\begin{aligned}
\text{min/max} \quad & z'_{00} \\
\text{such that} \quad & M_r(z') \succeq 0, M_r(u_T, z') \succeq 0, M_r(u_H, z') \succeq 0 \\
& M_r(y'_1) \succeq 0, M_r(u_T, y'_1) \succeq 0 \\
& M_r(y'_2) \succeq 0, M_r(u_H, y'_2) \succeq 0 \\
& 0 = y'_{1k} H^m - y'_{2m} T^k - 0^k x_0^m \\
& \qquad + \sum_i c_i z'_{k_i m_i}, \quad \forall m, k
\end{aligned} \tag{3.24}$$

This SDP can be compiled to the canonical form. To this end, the moment matrices can be arranged in a block-diagonal form and the localizing constraints (3.20) can be encoded by the introduction of new variables and appropriate equality constraints. This transformation can be done automatically using modeling frameworks such as CVXPY (Diamond and Boyd, 2016). We therefore only give the SDP in the more intuitive format. This problem can be solved using off-the-shelf SDP solvers such as MOSEK (MOSEK ApS, 2018), CVXOPT (Vandenberghe, 2010), or SCS (O'Donoghue et al., 2017).

In principle, we can choose an arbitrarily large order r for the moment matrices and their corresponding constraints, because there are infinitely many moments. In practice, however, the order is bounded by practical issues such as the program size (number of constraints and variables) and numerical issues. These issues are discussed in Section 3.6 in more detail. Choosing a finite r is a relaxation of the problem since it removes constraints regarding higher-order moments.

3.6 IMPLEMENTATION AND EVALUATION

The implementation of the SDP (3.24) is straightforward using modeling frameworks and off-the-shelf solvers. However, as noted in previous work (Dowdy and Barton, 2018a,b; Sakurai

and Hori, 2017, 2019) on moment-based SDPs the direct implementation of the problem may lead to difficulties for the solver. A source of these is that moments of various orders by nature may differ by many orders of magnitude. A re-scaling of the moments (Dowdy and Barton, 2018a; Sakurai and Hori, 2019) such that moments only vary by few orders of magnitude may alleviate this problem. In other scenarios such as the bounding of general transient or steady-state moments, the scaling can be particularly difficult, because the magnitude of moments is generally not known a priori. In the context of MFPTs with a finite time-horizon moments are trivially bounded.

3.6.1 *Moment Scaling*

Using the fact that $S \setminus B$ is often finite, it is possible to derive trivial bounds, which can be used to scale moments. If, for example, we have a one-dimensional process X_t with $X_0 = 0$ a.s. and are interested in the hitting time of an upper threshold $H > 0$ until time $T > 0$ for $i, k \in \mathbb{N}$

$$z_{ik} = E\left(\int_0^\tau t^i X_t^k \, dt\right) \leqslant E\left(\int_0^T t^i X_t^k \, dt\right) \leqslant H^k \int_0^T t^i \, dt = \frac{T^{i+1} H^k}{i+1}.$$

Thus, we fix a scaling vector d with entries $d_{ik} = T^{i+1} H^k$ in the same order as the canonical base vector (3.22). Using this scaling vector, we can define a scaling matrix $D = dd^T$. Clearly, $D \succeq 0$. Now we can formulate the optimization (3.24) over a scaled version $D^{-1} M(z')$ instead of $M(z')$. The moment matrices of the exit location probabilities are scaled in the same way. Alternatively, one could use approximations such as moment closures or bounds obtained by lower-order relaxations or solve a sequence of problems, incrementally increasing the time-horizon, and adjust the scaling accordingly (suggested in (Dowdy and Barton, 2018b)).

These scaling strategies have not been evluated so far.

In Figure 3.2 we illustrate the influence the scaling has on the optimization variables. While the unscaled version shows large

differences between values, these differences become significantly smaller in the scaled version of the problem.

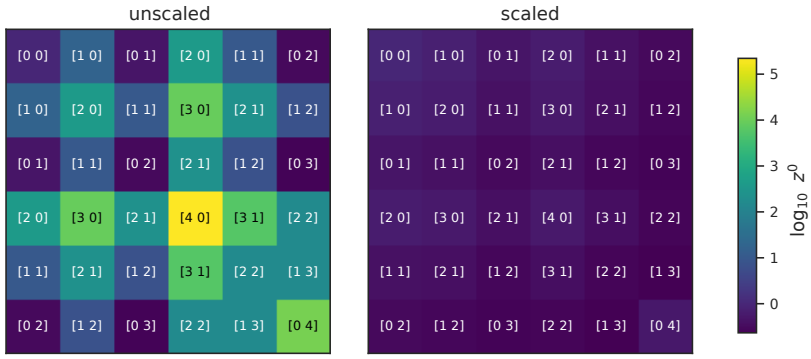


Figure 3.2: The unscaled and scaled value of the moment matrix proxy variable $M(z')$ after optimization using MOSEK. The indices are given along the logarithmic (base 10) values. The unscaled version (left) shows large differences in magnitudes, while on the scaling suppresses these large variations (right). The case study used here is Model 1, with a threshold $H = 25$ for species M and a time-horizon $T = 1$. The relaxation order $r = 2$. Therefore moments of orders up to $2r = 4$ appear.

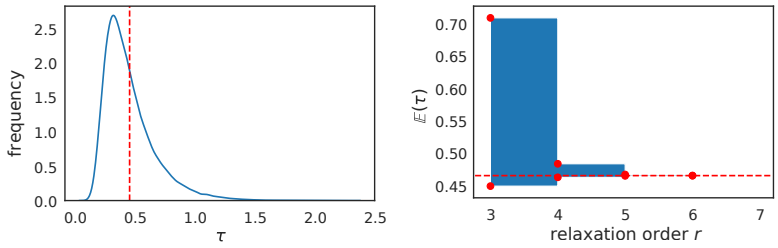
3.6.2 *Case Studies*

We implemented and solved the SDP programs described above using optimization suite MOSEK (MOSEK ApS, 2018) (version 9.1.2) via the CVXPY interface (Diamond and Boyd, 2016) (version 1.0.24).

3.6.2.1 *Dimerization*

As a first case study, we use Model 1 with parameters $\lambda = 100$ and $\delta = 0.2$. In this model, we are interested in the time at which the number of agents of type M surpasses a threshold of 25 before some time-horizon T, i.e.

$$\tau = \inf\{t \geqslant 0 \mid X_t \geqslant 25\} \wedge T\,.$$



(a) FPT distribution (b) MFPT bounds.

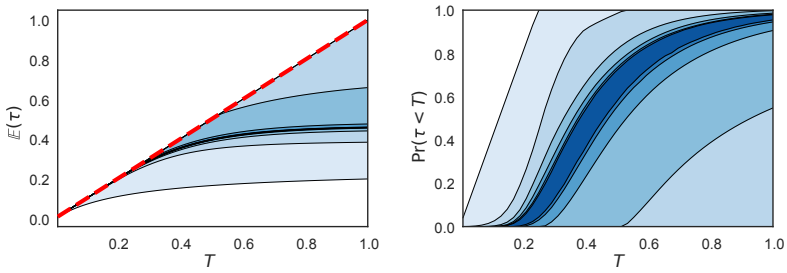
Figure 3.3: First-passage time characterisitics for Model 1 with $\tau = \inf\{t \geqslant 0 \mid X_t \geqslant 10\} \wedge \infty$. The dashed red line denotes the sampled MFPT based on 100,000 SSA samples. Bounds are based on the SDP (3.24) with different moment orders

We can let $T \to \infty$ because this system is ergodic and therefore $\tau < \infty$ a.s.

First, we set no finite time-horizon T, i.e. $T = \infty$. This is achieved by dropping the moments y_2 of measure ν_2 in the linear constraints (3.24). This can be done because the threshold on M makes the state-space finite and therefore the first passage time distribution is a phase-type distribution which possesses finite moments (Stewart, 2009, Chapter 7.6).

The empirical FPT distribution based on 100,000 SSA simulations is given in Figure 3.3a and the bounds, given different moment orders, are given in Figure 3.3b. As we can see in Figure 3.3b, the bounds capture the MFPT precisely for orders 5 and 6. The difference between upper and lower bound decreases roughly exponentially with increasing relaxation order r. We found that this trend was consistent among the case studies presented here (cf. Figure 3.5).

Next, we look at first passage times within a finite time-horizon T. In Figure 3.4a we summarize the bounds obtained for the MFPT over T. While low-order relaxations (light) give rather loose bounds, the bounds are already fairly tight when using $r = 4$. In many cases, hitting probabilities, that is, the probability of reaching the threshold before time T, are of particular interest. This is done by switching the optimization objective in (3.24) from the mass of the expected occupation measure ξ to the mass of ν_1. In terms of moments, the objective changes from z_{00} to



(a) MFPT bounds over T. (b) Reaching probabilities up to T.

Figure 3.4: MFPTs and reaching probability bounds for the dimerization model with $\tau = \inf\{t \geqslant 0 \mid X_t \geqslant 25\} \wedge T$ and varying T. The results for SDP relaxations of orders 1 (light) to 6 (dark) are shown.

y_{10}. The need for such a scenario often arises in the context of model checking, where one might be interested in the probability of a population exceeding a critical threshold. By varying the time-horizon, we are able to recover bounds on the cumulative density

$$F(t) = \Pr(X_s = H \mid s < t)$$

of the first passage time (Figure 3.4b).

Finally, we look at turn to the dimer species D that is synthesized by the combination of two monomers M. Here, we look at the time until the agents of type D exceed a threshold of five with a time-horizon $T = 1$. Note that we do not limit the number of M agents. Therefore the analyzed state-space is countably infinite. As in the previous two examples, we observe a roughly exponential decrease in interval size with increasing relaxation order r (cf. Figure 3.5 and Table 3.1).

3.6.2.2 *Parallel Dimerizations*

As a second study, we consider a two-dimensional model by combining two independent dimerizations.

Model 3 (Parallel independent dimerizations). *This model consists of two independent versions of Model 1.*
↪ page 13

$$\varnothing \xrightarrow{10^4} M_1 \quad 2M_1 \xrightarrow{0.1} D_1 \quad \varnothing \xrightarrow{10^4} M_2 \quad 2M_2 \xrightarrow{0.1} D_2$$

Initially,

$$0 = X_0^{(M_1)} = X_0^{(M_2)} = X_0^{(D_1)} = X_0^{(D_2)}.$$

As a FPT we consider the time at which either M_1 or M_2 surpasses a threshold of 200 or a time-horizon of $T = 10$ is reached, i.e.

$$\tau = \inf\{t \geqslant 0 \mid X_t^{(M_1)} \geqslant 200\} \wedge \inf\{t \geqslant 0 \mid X_t^{(M_2)} \geqslant 200\} \wedge 10\,.$$

As before, we ignore the product species D_1 and D_2 since they do not influence τ. The SSA (using $n = 10{,}000$ runs) gives the estimate $E(\tau) \approx 0.028378$ which is captured tightly by the SDP bounds (cf. Table 3.1). For higher relaxation orders $r \geqslant 5$ numerical issues prevented the solution of the corresponding SDPs.

3.6.3 *Hybrid Models and Multi-Modality*

The analysis of switching times is a particularly interesting case of FPTs that arises in many contexts. Often mode switching in such systems can be described a modulating Markov process whose switching rates may depend on the system state (e.g. the population sizes). In biological applications, mode switching often describes a change of the DNA state (Hasenauer et al., 2014; Stekel and Jenkins, 2008) and the analysis of switching time distribution is of particular interest (Barzel and Biham, 2008; Spieler, Hahn, and Zhang, 2011).

In the context of MPMs, typically the state-space $\mathcal{S} = \mathbb{N}^{\tilde{n}_S} \times \{0,1\}^{\hat{n}_S}$. This state is modeled by $\hat{n}_S$ population variables with binary domains. Therefore, at each time point, the state of these modulator variables is given by a set of Bernoulli random variables. When considering the moments of such a variable X, clearly

$$E(X^m) = E(X) = \Pr(X = 1)\,, \quad \forall m \geqslant 1$$

We can use this fact two ways: We could use the same moment constraints as above and impose additional equality constraints on the moments matrices to ensure $\mathrm{E}(X^m) = \mathrm{E}(X)$, $m \geqslant 1$. Alternatively, we can apply this simplification to the moment equation, which we choose here.

We apply a split of variables X_t into the high count part $\tilde{X}_t$ and the binary part $\hat{X}_t$ to the expectations in (3.2). Similarly, we split ν_j and with a case distinction over the mode variable, we arrive at a similar result as in (Hasenauer et al., 2014):

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{E}\left(\tilde{X}_t^m 1_{=y}(\hat{X}_t)\right) \\
&= \sum_{j=1}^{n_R} \mathrm{E}\left(\left(\left(\tilde{X}_t + \tilde{\nu}_j\right)^m 1_{=y}(\hat{X}_t + \hat{\nu}_j) - \tilde{X}_t^m 1_{=y}(\hat{X}_t)\right)\alpha(X_t)\right) \\
&= \sum_{j=1}^{n_R} \mathrm{E}\left(\left(\tilde{X}_t + \tilde{\nu}_j\right)^m \alpha_j(\tilde{X}_t, y - \hat{\nu}_j) 1_{=y-\hat{\nu}_j}(\hat{X}_t)\right) \\
&\quad - \sum_{j=1}^{n_R} \mathrm{E}\left(\tilde{X}_t^m \alpha_j(\tilde{X}_t, y) 1_{=y}(\hat{X}_t)\right) .
\end{aligned} \tag{3.25}$$

Similarly to the general moment case, we can derive a constraint, by multiplying with a time-weighting factor and integrating.

For simplicity, here we assume $\tilde{n}_S = \hat{n}_S = 1$. Fixing appropriate sequences $(c_i)_i$, $(m_i)_i$, $(k_i)_i$, and $(y_i)_i$ the constraint has the following form.

We fix $0^0 = 1$.

$$\begin{aligned}
&\sum_{y\in\{0,1\}} H^m \mathrm{E}\left(\tau^k; \hat{X}_\tau = y, \tau < T\right) \\
&\qquad + T^k \mathrm{E}\left(\tilde{X}_T^m; \hat{X}_T = y, \tau = T\right) \\
&= 0^k \tilde{x}_0^m 1_{=y}(\hat{x}_0) + \sum_i c_i \mathrm{E}\left(\int_0^\tau t^{k_i} \tilde{X}_t^{m_i}\,\mathrm{d}t; \hat{X}_t = y_i\right)
\end{aligned} \tag{3.26}$$

This way we can decompose the moment matrices such that for each mode $y \in \{0,1\}$, we have moment matrices composed of the respective partial moments. To this end, let $z_m^{(y)}$ be the partial moment w.r.t. $\hat{X} = y$. The moment constraint over the partial moments has a linear structure:

$$0 = y_{1k} H^m - y_{2m} T^k - 0^k x_0^m + \sum_i c_i z_{k_i m_i}^{(y_i)} . \tag{3.27}$$

Table 3.1: MFPT bounds on Model 1, Model 3, Model 4.

		relaxation order r				
		1	2	3	4	5
Model 1	lower	0.0909	0.2661	0.2845	0.2867	0.2871
	upper	1.0000	0.3068	0.2932	0.2886	0.2875
Model 3	lower	0.0010	0.0250	0.0275	0.0280	0.0280
	upper	10.0000	0.0575	0.0323	0.0299	0.0290
Model 4	lower	4.0000	6.0028	6.2207	6.3377	6.3772
	upper	10.7179	6.4619	6.4079	6.4004	6.3835

3.6.3.1 *Gene Expression with Negative Feedback*

As an instance of a multi-modal system, we consider a simple gene expression with self-regulating negative feedback which is a common pattern in many genetic circuits (Stekel and Jenkins, 2008).

Model 4 (Negative self-regulated gene expression)**.** *This model consists of a gene state that is either on or off, i.e.* $X_t^{D_{on}} + X_t^{D_{off}} = 1$, $\forall t \geqslant 0$. *Therefore the system has two* modes.

$$D_{on} \xrightarrow{\tau_0} D_{off} \quad D_{off} \xrightarrow{\tau_1} D_{on} \quad D_{on} \xrightarrow{\rho} D_{on} + P$$

$$P \xrightarrow{\delta} \varnothing \quad P + D_{on} \xrightarrow{\gamma} D_{off}$$

The model parameters are $(\tau_0, \tau_1, \rho, \delta, \gamma) = (10, 10, 2, 0.1, 0.1)$ *and* $X_0^{(D_{off})} = 1$, $X_0^{(P)} = 0$ *a.s.*

As a first passage time we consider

$$\tau = \inf\{t \geqslant 0 \mid X_t^{(P)} \geqslant 5\} \wedge 20\,.$$

The results are summarized in Table 3.1. The estimated MFPT based on 100,000 SSA samples is $\mathbb{E}(\tau) \approx 6.37795 \pm 0.02847$ at 99% confidence level. Note that our SDP solution for $r = 5$ yields tighter moment bounds than the statistical estimation.

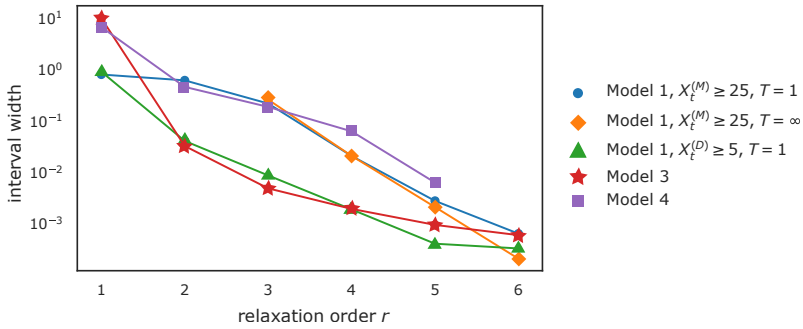


Figure 3.5: The interval width, i.e. the difference between upper and lower bound, for different case studies and targeted first passage times against the order r of the SDP relaxation.

In Figure 3.5 we summarize our results about the decrease of the interval widths for increasing relaxation order r by plotting them on a log-scale. We see an approximately exponential decrease with increasing r. The SDPs above were all solved within at most a few seconds.

3.7 CONCLUSION

Numerical methods to compute reachability probabilities and first passage times for continuous-time Markov chains that are based on an exhaustive exploration of the state-space are exact up to numerical precision. Such methods, however, do not scale and cannot be efficiently applied to models with large or infinite state-spaces, an issue exacerbated in population models. Moment-based methods offer an alternative analysis approach for MPMs, which scales with the number of different populations in the system but are approximations with little or no control of the error. In this work, we bridge this gap by proposing a rigorous approach to derive bounds on first passage times and reachability probabilities, leveraging a semi-definite programming formulation based on appropriate moment constraints.

The method we propose is shown to be accurate in several examples. It does, however, suffer, like all moment-based meth-

ods, from numerical instabilities in the SDP solver, caused by the fact that moments typically span several orders of magnitude. We proposed a scaling of moments to mitigate this effect. However, the scaling only addresses the moment matrices but not the linear constraints which still contain values with varying orders of magnitudes. Therefore, we plan as future work to investigate an appropriate scaling for the linear constraints or to redefine the moment constraints (e.g. using an exponential time weighting (Dowdy and Barton, 2018b)). Based on this investigation, we expect to make this approach applicable to more problems including, for example, the computation of bounds of rare event probabilities. We also expect that the development of more sophisticated scaling techniques will improve approximate moment-based methods.

Furthermore, moment-based analysis approaches have shown to be successful in a wide range of applications such as optimal control problems or the estimation of densities (Lasserre, 2010). We expect that our proposed ideas can be adapted to a wider range of stochastic models such as stochastic hybrid systems, exhibiting partly deterministic dynamics.

4 CONTROL VARIATES FOR MONTE-CARLO ESTIMATION

Analysis approaches based on sampling, such as the stochastic simulation algorithm (SSA) (Gillespie, 1977), can be applied independent of the size of the model's state-space. However, statistical approaches are costly since a large number of simulation runs is necessary to reduce the statistical inaccuracy of estimators. This problem is particularly severe if reactions occur on multiple time scales or if the event of interest is rare. A particularly popular technique to speed up simulations is τ-leaping which applies multiple reactions in one step of the simulation. However, such multi-step simulations rely on certain assumptions about the number of reactions in a certain time interval. These assumptions are typically only approximately fulfilled and therefore introduce approximation errors on top of the statistical uncertainty of the considered point estimators.

↪ *page 17*

Moment-based techniques offer a fast approximation of the statistical moments of the model. The exact moment dynamics can be expressed as an infinite-dimensional system of ODEs, which cannot be directly integrated for a transient analysis. Hence, ad-hoc approximations need to be introduced, expressing higher order moments as functions of lower-order ones (Ale, Kirk, and Stumpf, 2013; Engblom, 2006). However, moment-based approaches rely on assumptions about the dynamics that are often not even approximately fulfilled and may lead to high approximation errors. Recently, equations expressing the moment dynamics have also been used as constraints for parameter estimation (Backenköhler, Bortolussi, and Wolf, 2018) and for computing moment bounds using semi-definite programming (Dowdy and Barton, 2018b; Ghusinga et al., 2017).

In this work, we propose a combination of such moment constraints with the SSA approach. Specifically, we interpret these constraints as random variables that are correlated with the estimators of interest usually given as functions of chemical population variables. These constraints can be used as (linear) control

variates (CVs) in order to improve the final estimate and reduce its variance (Lavenberg, Moeller, and Welch, 1982; Szechtman, 2003). The method is easy on an intuitive level: If a control variate is positively correlated with the function to be estimated then we can use the estimate of the variate to adjust the target estimate.

The incorporation of control variates into the SSA introduces additional simulation costs for the calculation of the constraint values. These values are integrals over time, which we accumulate based on the piece-wise constant trajectories. This introduces a trade-off between the variance reduction that is achieved by using control variates versus the increased simulation cost. This trade-off is expressed as the product of the variance reduction ratio and the cost increase ratio.

For a good trade-off, it is crucial to find an appropriate set of control variates. Here we propose a class of constraints which is parameterized by a moment vector and a weighting parameter, resulting in infinitely many choices. We present an algorithm that samples from the set of all constraints and proceeds to remove constraints that are either only weakly correlated with the target function or are redundant in combination with other constraints.

In a case study, we explore different variants of this algorithm both in terms of generating the initial constraint set and of removing weak or redundant constraints. We find that the algorithm's efficiency is superior to a standard estimation procedure using stochastic simulation alone in almost all cases.

Although in this work we focus on estimating first order moments at fixed time points, the proposed approach can in principle deal with any property that can be expressed in terms of expected values such as probabilities of complex path properties. Another advantage of our technique is that an increased efficiency is achieved without the price of an additional approximation error as it is the case for methods based on moment approximations or multi-step simulations.

4.1 RELATED WORK

STATE-SPACE TRUNCATIONS If the state-space is finite and small enough one can deal with the underlying Markov chain directly. But there are also cases where the transient distribution has an infinitely large support and one can still deal with explicit state probabilities. To this end, one can fix a finite state-space, that should contain most of the probability (Munsky and Khammash, 2006). Refinements of the method work dynamically and adjust the state-space according to the transient distributions (Andreychenko et al., 2011; Henzinger, Mateescu, and Wolf, 2009; Mateescu et al., 2010).

MOMENT APPROXIMATIONS On the other end of the spectrum there are mean-field approximations, which model the mean densities faithfully in the system size limit (Bortolussi et al., 2013). In between there are techniques such as moment closure (Singh and Hespanha, 2006), that not only consider the mean, but also the variance and other higher order moments. These methods depend on ad-hoc approximations of higher order moments to close the ODE system given by the moment equations. Yet another class of methods approximate molecular counts continuously and approximate the dynamics in such a continuous space, e.g. the system size expansion (Van Kampen, 1992) and the chemical Langevin equation (Gillespie, 2000).

While the moment closure method uses ad-hoc approximations for high order moments to facilitate numerical integration, they can be avoided in some contexts. For the equilibrium distribution, for example, the time-derivative of all moments is equal to zero. This directly yields constraints that have been used for parameter estimation at steady-state (Backenköhler, Bortolussi, and Wolf, 2018) and bounding moments of the equilibrium distribution using semi-definite programming (Ghusinga, Lamperski, and Singh, 2018; Ghusinga et al., 2017; Kuntz et al., 2019a). The latter technique of bounding moments has been successfully adapted in the context of transient analysis (Dowdy and Barton, 2018b; Sakurai and Hori, 2017, 2019). We adapt the constraints proposed

in these works to improve statistical estimations via stochastic simulation (cf. Section 4.2).

MONTE-CARLO SIMULATION While the above techniques give a deterministic output, stochastic simulation generates single executions of the stochastic process (Gillespie, 1977). This necessitates accumulating large numbers of simulation runs to estimate quantities. This adds a significant computational burden. Consequently, considerable effort has been directed at lowering this cost. A prominent technique is τ-leaping (Gillespie, 2001), which in one step performs multiple instead of only a single reaction. Another approach is to find approximations that are specific to the problem at hand, such as approximations based on time-scale separations (Bortolussi, Milios, and Sanguinetti, 2015; Cao, Gillespie, and Petzold, 2005).

Recently, multilevel Monte Carlo methods have been applied in to time-inhomogenous MPM (Anderson and Yuan, 2018). In this techniques estimates are combined using estimates of different approximation levels.

The most prominent application of a variance reduction technique in the context of stochastic reaction networks is importance sampling (Kuwahara and Mura, 2008). This technique relies on an alteration of the process and then weighting samples using the likelihood-ratio between the original and the altered process.

4.2 MOMENT CONSTRAINTS

The time-evolution of $\mathbb{E}(f(X_t))$ for some function f is given by (2.21). The integration of (2.21) with such functions f is well-known in the context of moment approximations of MPM. For most models the arising ODE system is infinitely large, because the time-derivative of low order moments usually depends on the values of higher order moments. To close this system, *moment closures*, i.e. ad-hoc approximations of higher order moments are applied (Schnoerr, Sanguinetti, and Grima, 2015). The main drawback of this kind of analysis is that it is not known whether the chosen closure gives an accurate approximation for the case

↪ *page 18*

at hand. Here, such approximations are not necessary, since we will apply the moment dynamics in the context of stochastic sampling instead of trying to integrate (2.21).

Apart from integration strategies, setting (2.21) to zero has been used as a constraint for parameter estimation at steady-state (Backenköhler, Bortolussi, and Wolf, 2018) and bounding moments at steady-state (Dowdy and Barton, 2018a; Ghusinga et al., 2017; Kuntz et al., 2019a). The extension of the latter has recently lead to the adaption of these constraints to a transient setting (Dowdy and Barton, 2018b; Sakurai and Hori, 2019). These two transient constraint variants are analogously derived by multiplying (2.21) by a time-dependent, differentiable weighting function $w(t)$ and integrating:

As in the previous chapter, multiplication with $w(t)$ and integration on $[t_0, T]$ yields (Dowdy and Barton, 2018b; Sakurai and Hori, 2019)

↪ page 33

$$\begin{aligned} &w(T)\mathbb{E}\left(f(X_T)\right) - w(t_0)\mathbb{E}\left(f(X_{t_0})\right) - \int_{t_0}^{T} \frac{dw(t)}{dt}\mathbb{E}\left(f(X_t)\right)\, dt \\ &\quad = \sum_{j=1}^{n_R} \int_{t_0}^{T} w(t)\mathbb{E}\left(\left(f(X_t + \nu_j) - f(X_t)\right)\alpha_j(X_t)\right)\, dt \end{aligned} \tag{4.1}$$

In the context of computing moment bounds via semi-definite programming the polynomial $w(t) = t^s$ (Sakurai and Hori, 2019) and the exponential $w(t) = e^{\lambda(T-t)}$ (Dowdy and Barton, 2018b) have been proposed. While both choices proved to be effective in different case studies, relying solely on the latter choice, i.e.

$$w(t) = e^{\lambda(T-t)}$$

was sufficient.

In Chapter 3 we chose the monomial, such that (4.1) would admit the interpretation of temporal moments. Here the exponential is a good choice, because it can model both increasing and decreasing weighting.

By expanding the rate functions and f in (4.1) and substituting the exponential weight function we can re-write (4.1) as

$$\begin{aligned} 0 = \ &\mathbb{E}\left(f(X_T)\right) - e^{\lambda T}\mathbb{E}\left(f(X_{t_0})\right) \\ &+ \sum_k c_k \int_{t_0}^{T} e^{\lambda(T-t)}\mathbb{E}\left(X_t^{m_k}\right)\, dt \end{aligned} \tag{4.2}$$

with coefficients c_k and vectors m_k defined accordingly. Assuming the moments remain finite on $[0, T]$, we can define the random variable

$$Z = f(X_T) - e^{\lambda T} f(X_{t_0}) + \sum_k c_k \int_{t_0}^{T} e^{\lambda(T-t)} X_t^{m_k} \, dt \tag{4.3}$$

with $E(Z) = 0$.

Note, that a realization of Z depends on the whole trajectory $\tau = x_0 t_1 x_1 t_2 \dots t_n x_n$ over $[t_0, T]$. Thus, for the integral terms in (4.3) we have to compute sums

$$\frac{1}{\lambda} \sum_{i=1}^{n} \left(e^{\lambda(T-t_{i+1})} - e^{\lambda(T-t_i)} \right) x_i^{m_k} \tag{4.4}$$

over a given trajectory. This accumulation is best done during the simulation to avoid storing the whole trajectory. Still, the cost of a simulation run increases. For the method to be efficient, the variance reduction (Section 4.3) needs to overcompensate for this increased cost of a simulation run.

EXAMPLE For Model 2 the moment equation for $f(x) = x$ becomes

$$\frac{d}{dt} E(X_t) = \gamma - \delta E(X_t) \, .$$

The corresponding constraint (4.2) with $\lambda = 0$ gives

$$0 = E(X_T) - E(X_0) - \gamma T + \delta \int_0^T E(X_t) \, dt \, .$$

In this instance the constraint leads to an explicit function of the moment over time. If $X_0 = 0$ w.p. 1, then (4.2) becomes

$$E(X_T) = \frac{\gamma}{\delta} \left(1 - e^{-\delta T} \right) \tag{4.5}$$

when choosing $\lambda = -\delta$. ◇

4.3 CONTROL VARIATES

Now, we are interested in the estimation of some quantity $\mathsf{E}(V)$ by stochastic simulation. Let $V_1, \ldots, V_n$ be independent samples of V. Then the sample mean

$$\hat{V}_n = \frac{1}{n} \sum_{i=1}^{n} V_k$$

is an estimate of $\mathsf{E}(V)$. By the central limit theorem

$$\sqrt{n}\hat{V}_n \xrightarrow{d} N(\mathsf{E}(V), \sigma_V^2).$$

Now suppose, we know of a random variable Z with $0 = \mathsf{E}(Z)$. The variable Z is called a *control variate (CV)*. If a control variate Z is correlated with V, we can use it to reduce the variance of $\hat{V}_n$ (Glasserman and Yu, 2005; Nelson, 1990; Szechtman, 2003; Wilson, 1984). For example, consider we are running a set of simulations and consider a single constraint. If the estimated value of this constraint is larger than zero and we estimate a positive correlation between the constraint Z and V, we would, intuitively, like to decrease our estimate $\hat{V}_n$ accordingly. This results in an estimation of the mean of the random variable

$$Y_\beta = V - \beta Z$$

instead of V. The variance

$$\sigma_{Y_\beta}^2 = \sigma_V^2 - 2\beta \operatorname{Cov}(V, Z) + \beta^2 \sigma_Z^2.$$

The optimal choice β can be computed by considering the minimum of $\sigma_{Y_\beta}^2$. Then

$$\beta^* = \operatorname{Cov}(V, Z)/\sigma_Z^2.$$

Therefore

$$\sigma_{Y_{\beta^*}} = \sigma_Z^2(1 - \rho_{VZ}^2),$$

where ρ_{VZ} is the correlation of Z and V.

If we have multiple control variates, we can proceed in a similar fashion. Now, let Z denote a vector of d control variates and let

$$\Sigma = \begin{bmatrix} \Sigma_Z & \Sigma_{VZ} \\ \Sigma_{ZV} & \sigma_V^2 \end{bmatrix}$$

be the covariance matrix of (Z, V). As above, we estimate the mean of

$$Y_\beta = V - \beta^T Z .$$

The ideal choice of β is the result of an ordinary least squares regression between V and Z_i, $i = 1, \ldots, n$. Specifically,

$$\beta^* = {\Sigma_Z}^{-1} \Sigma_{ZV} .$$

Then, asymptotically the variance of the estimator

$$\hat{Y}_{\beta^*} = \hat{V} - {\beta^*}^\top \hat{Z} \tag{4.6}$$

is (Szechtman, 2003),

$$\sigma^2_{\hat{Y}_{\beta^*}} = (1 - R^2_{ZV})\sigma^2_{\hat{V}} , \tag{4.7}$$

where

$$R^2_{ZV} = \Sigma_{ZV} \Sigma_Z^{-1} \Sigma_{ZV} / \sigma^2_V .$$

In practice, however, β^* is unknown and needs to be replaced by an estimate $\hat{\beta}$. Then the estimator

$$\hat{Y}_{\hat{\beta}} = \hat{V} - \hat{\beta}^\top \hat{Z} . \tag{4.8}$$

This leads to an increase in the estimator's variance. Under the assumption of Z and V having a multivariate normal distribution (Cheng, 1978; Lavenberg, Moeller, and Welch, 1982), the variance of the estimator is

$$\sigma^2_{\hat{Y}_{\hat{\beta}}} = \frac{n-2}{n-2-d}(1 - R^2_{ZV})\sigma^2_{\hat{V}} . \tag{4.9}$$

Clearly, a control variate is "good" if it is highly correlated with V. The constraint in (4.5) is an example of the extreme case. When we use this constraint as a control variate for the estimation of the mean at some time point t, it has a correlation of ± 1 since it describes the mean at that time precisely. Therefore the variance is reduced to zero. We thus aim to pick control variates that are highly correlated with V.

EXAMPLE Consider, for example, the above case of the birth-death process. If we choose (4.5) as a constraint, it would always yield the exact difference of the exact mean to the sample mean and therefore have a perfect correlation. Clearly, $\hat{\beta}$ reduces to 1 and $\hat{Y}_1 = \mathbb{E}(X_t)$. ◇

4.4 MOMENT-BASED VARIANCE REDUCTION

We propose an adaptive estimation algorithm (Algorithm 2) that starts out with an initial set of control variates and periodically removes potentially inefficient variates. The "accumulator set" A represents the time-integral terms (4.4). The size of A has the most significant impact on the overall speed of the algorithm since it represents the only factor incurring a direct cost increase in the SSA itself (line 4).

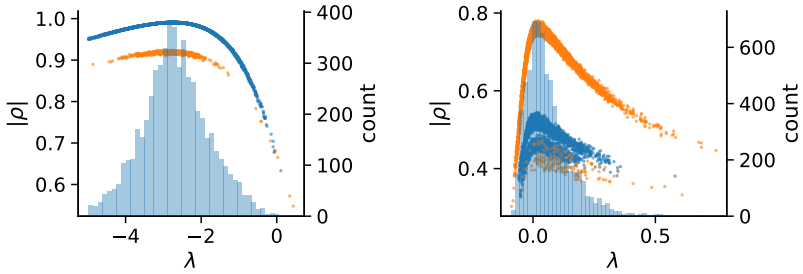
The algorithm consists of a main loop which performs n simulation runs (line 3). Between each run the mean and covariance estimates of $[Z, V]$ are updated (line 5). Every $d < n$ iterations, the control variates are checked for *efficiency* and *redundancy* (lines 6–11).

Checking both conditions is based on the correlation ρ_{ij} between the i-th and j-th control variate and the correlation ρ_{iv} of a control variate i to V. These are elements of the correlation matrix

$$C = \begin{bmatrix} 1 & \dots & \rho_{1k} & \rho_{1v} \\ \vdots & \ddots & \vdots & \vdots \\ \rho_{k1} & \dots & 1 & \rho_{kv} \\ \rho_{v1} & \dots & \rho_{vk} & 1 \end{bmatrix}.$$

The first condition is a simple lower threshold $\rho_{\min}$ for a correlation ρ_{iv}. This condition aims to remove those variates from the control variate set that are only weakly correlated to V (line 8). The rationale is that, if variate i has a low correlation with the variable of interest V, its computation may not be worth the costs. Here, we propose to set $\rho_{\min}$ heuristically as

$$\rho_{\min} = \min\left(0.1, \frac{\max_i \rho_{iv}}{k_{\min}}\right),$$



(a) $\mathrm{E}\left(X_2^A\right)$ in Model 1 (b) $\mathrm{E}\left(X_{50}^X\right)$ in Model 5

Figure 4.1: The absolute correlation of different constraints to V arising from different choices of λ. The blue dots represent constraints based on first order moments, while the orange refers to control variates derived from second order moments. In both cases 10,000 samples were used with 30 initial samples for λ from $N(0,1)$ and $k_{\min} = 2$. A quadratic decision bound was used for the redundancy removal. Furthermore, a histogram of control variates selected by Algorithm 2 is given. In (a) $\mathrm{E}\left(X_2^A\right)$ in the dimerization model was estimated. In (b) $\mathrm{E}\left(X_{50}^X\right)$ in the distributive modification model was estimated.

where $k_{\min} > 1$ is an algorithm parameter.

The second condition aims to remove redundant conditions. This is not only beneficial for the efficiency of the estimator, but also necessary for the matrix inversion (4.7) because perfectly and highly correlated constraints will make the covariance matrix estimate $\hat{\Sigma}_Z$ (quasi-) singular. For all considered criteria we iterate over all tuples $(i,j) \in \{1,\dots,k\}^2$, $i \neq j$, removing the weaker of the two, i.e. $\arg\min_{k\in\{i,j\}} \rho_{kv}$, if the two control variates are considered redundant (line 9).

There are many ways to define such a redundancy criterion. Here, we focus on criteria that are defined in terms of the average correlation

$$\bar{\rho}_{ij} = (\rho_{iv} + \rho_{jv})/2\,.$$

For two variates i and j we then check if their mutual correlation ρ_{ij} exceeds a some function ϕ of $\bar{\rho}_{ij}$, i.e. we check the inequality

$$\phi(\bar{\rho}_{ij}) \leqslant \rho_{ij} .$$

If this inequality holds, constraint $\arg\min_{k\in\{i,j\}} \rho_{kv}$ is removed. Naturally, there are many possible choices for the above decision boundary ϕ (cf. Figure 4.2).

The simplest choice is to ignore $\bar{\rho}_{ij}$ and just fix a constant close to 1 as a threshold, e.g.

$$\phi_c(\bar{\rho}_{ij}) = 0.99 .$$

While this often leads to the strongest variance reduction and avoids numerical issues in the control variate computation, it turns out that the computational overhead is not as well-compensated as by other choices of ϕ (see Section 4.5).

Another option is to fix a simple linear function, i.e.

$$\phi_\ell(\bar{\rho}_{ij}) = \bar{\rho}_{ij} .$$

For this choice the intuition is, that one of two constraints is removed if their mutual correlation exceeds their average correlation with V.

Here, we also assess two quadratic choices for ϕ. The first choice of

$$\phi_q(\bar{\rho}) = 1 - (1 - \bar{\rho})^2$$

is more tolerant than the linear function and more strict than a threshold function, except for highly correlated control variates. Another variant of ϕ is given by including the lower bound $\rho_{\min}$ and scaling the quadratic function accordingly:

$$\phi_{sq}(\bar{\rho}) = 1 - ((1 - \bar{\rho})/(1 - \rho_{\min}))^2 .$$

The different choices of ϕ considered here are plotted in Figure 4.2.

Now, we discuss the choice of the initial control variates. We identify control variate k by a tuple (m_k, λ_k) of a moment vector

Algorithm 2: Estimate the mean of species i at time T

input : $n, d, n_{max}, n_\lambda, k_{min}$
output: An estimate using linear control variates

1 $L \leftarrow \{\lambda_i \sim \pi_\lambda \mid 1 \leqslant i < n_\lambda\} \cup \{0\}$;
2 $P \leftarrow \{(m, \lambda) | 1 \leqslant |m| \leqslant n_{max}, \lambda \in L\}$;
3 **for** $i = 1, \ldots, n$ **do**
4 $\quad \tau \leftarrow SSA(\pi_0, T, A)$;
5 $\quad$ compute constraints using A and update $\hat{\Sigma}$ and $\hat{V}_i$;
6 $\quad$ **if** $i \mod d = 0$ **then**
7 $\quad\quad \rho_{min} \leftarrow \min(0.1, \max_i \rho_{iv}/k_{min})$;
8 $\quad\quad P \leftarrow P \setminus \{(m_k, \lambda_k) \mid \rho_{kv} < \rho_{min}\}$;
9 $\quad\quad P \leftarrow P \setminus \{(m_k, \lambda_k) \mid \exists i, j . i \neq j, \phi(\bar{\rho}_{ij}) < \rho_{ij},$
10 $\quad\quad\quad\quad\quad\quad\quad\quad k = \arg\min_{k \in \{i,j\}} \rho_{kv}\}$;
11 $\quad$ remove unnecessary accumulators from A;
12 **return** $\hat{V}_n - (\hat{\Sigma}_Z^{-1} \hat{\Sigma}_{ZV})^\top \hat{Z}_n$;

m_k and a time-weighting parameter λ_k. That is, we use $w(t) = e^{\lambda_k(T-t)}$ and $f(x) = x^{m_k}$ in (4.1). For a given set of parameters L, we use all moments up to some fixed order n_{max} (line 2). The ideal set of parameters L is generally not known. For certain choices the correlation of the control variates and the variable of interest is higher then for others. To illustrate this, consider the above example of the birth-death process. Choosing $\lambda = -\delta$ leads to a control variate that has a correlation of ± 1 with V. Therefore, the ideal choice of initial values for would be $L = \{-\delta\}$. This, however, is generally not known. Therefore, we sample a set of λ's from some fixed distribution π_λ (line 1).

4.5 CASE STUDIES

We first define a criterion of *efficiency* in order to estimate whether the reduction in variance is worth the increased cost. A natural baseline of a variance reduction is, that it is more efficient to pay for the computational overhead of the reduction than to generate more samples to achieve a similar reduction of variance. Let σ_Y^2

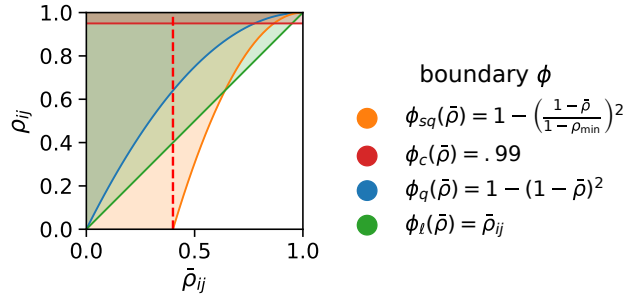


Figure 4.2: Different decision functions used in the redundant control variate removal. The weaker of any two control variates is removed if the pair $(\bar{\rho}_{ij}, \rho_{ij})$ belongs to the shaded area of the considered function. The vertical dashed line indicates $\rho_{\min}$.

be the variance of Y. The *efficiency* of the method is the ratio of the necessary cost to achieve a similar reduction with the CV estimate Y_{CV} compared to the standard estimate Y (L'Ecuyer, 1994), i.e.

$$E = \frac{c_0 \sigma_Y^2}{c_1 \sigma_{Y_{CV}}^2}. \tag{4.10}$$

This is the ratio between *slowdown* c_0/c_1 and variance reduction $\sigma_Y^2/\sigma_{Y_{CV}}^2$. That ratio c_0/c_1 depends heavily on both the specific implementation and the technical setup. The cost increase is mainly due to the computation of the integrals in (4.2). But the repeated checking of control variates for efficiency also increases the cost. The accumulation over the trajectory directly increases the cost of a single simulation which is the critical part of the estimation. To estimate the base-line cost c_0, 2000 estimations were performed without considering any control variates.

The simulation is implemented in the Rust programming language. The model description is parsed from a high level specification. Rate functions are compiled to stack programs for fast evaluation. Code is made available online (Backenköhler, 2019). www.rust-lang.org

We consider four non-trivial case studies. Three models exhibit complex multi-modal behaviour. We now describe the models and the estimated quantities in detail.

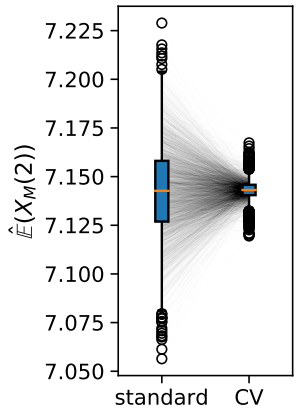


Figure 4.3: The effect of including CVs on the mean estimates $\hat{E}(X_2^M)$ in the dimerization case study. Parameters were $\pi_\lambda = N(0,1)$, $n_\lambda = 30$, $k_{min} = 4$, $\phi(\bar{\rho}) = 1-(1-\bar{\rho})^2$.

This model appeared earlier as Model 1 on page 13.

The first model is a simple dimerization on a countably infinite state-space.

Model (Dimerization)**.** *We first examine a simple dimerization model on an unbounded state-space.*

$$\varnothing \xrightarrow{10} M \qquad 2M \xrightarrow{0.1} D$$

with initial condition $X_0^M = 0$.

Despite the models simplicity, the moment equations are not closed for this system due to the second reaction which is non-linear. Therefore a direct analysis of the expected value would require a closure. For this model we will estimate $E\left(X_2^M\right)$.

The following two models are bimodal, i.e. they each posses two stable regimes among which they can switch stochastically. For both models we choose the initial conditions such that the process will move towards either attracting region with equal probability.

Model 5 (Distributive Modification)**.** *The distributive modification model was introduced in Cardelli and Csikász-Nagy (2012). It consists of the reactions*

$$X + Y \xrightarrow{0.001} B + Y \qquad B + Y \xrightarrow{0.001} 2Y$$
$$Y + X \xrightarrow{0.001} B + X \qquad B + X \xrightarrow{0.001} 2X$$

with initial conditions $X_0^X = X_0^Y = X_0^B = 100$.

Model 6 (Exclusive Switch)**.** *The exclusive switch model consists of 5 species, 3 of which are typically binary (activity states of the genes) (Loinger et al., 2007).*

$$D + P_1 \to D.P_1 \qquad D.P1 \to D + P_1 \qquad D.P_1 \to D.P_1 + P_1$$
$$D + P_2 \to D.P_2 \qquad D.P2 \to D + P_2 \qquad D.P_2 \to D.P_2 + P_2$$
$$D \to D + P_1 \qquad D \to D + P_2$$
$$P_1 \to \varnothing \qquad P_2 \to \varnothing$$

with initial conditions

$$X_0^D = 1 \text{ and } X_0^{D.P1} = X_0^{D.P_2} = X_0^{P_1} = X_0^{P_2} = 0\,.$$

We evaluate the influence of algorithm parameters, choices of distributions to sample λ from, and the influence of the sample size on the efficiency of the proposed method. Note that the implementation does not simplify the constraint representations or the state space according to stoichiometric invariants or limited state spaces. Model 5, for example has the invariant

$$X_t^X + X_t^Y + X_t^B = \text{const.}, \forall t \geqslant 0\,,$$

which could be used to reduce the state-space dimensionality to two. In Model 6 the invariant

$$\forall t \geqslant 0. X_t^D, X_t^{D.P_1}, X_t^{D.P_2} \in \{0, 1\}$$

could be used to optimize the algorithm by eliminating redundant moments, e.g. $\mathrm{E}((X^D)^2) = \mathrm{E}\left(X^D\right)$. Such an optimization would further increase the efficiency of the algorithm.

We first turn to the choice of the λ sampling distribution. Here we consider two choices:

1. a standard normal distribution $N(0,1)$,
2. a uniform distribution on $[-5,5]$.

We deterministically include $\lambda = 0$ in the constraint set, as this parameter corresponds to a uniform weighting function. We performed estimations on the case studies using different valuations of the algorithm parameters of the minimum threshold k_{min} and the number of λ-samples n_λ. We used samples size $n = 10{,}000$ and checked the control variates every $d = 100$ iterations for the defined criteria. For each valuation 1000 estimations were performed. In Figure 4.4, we summarize the efficiencies for the arising parameter combinations on the three case studies. Most strikingly, we can note that the efficiency was consistently larger than one in all cases. Generally, the normal sampling distribution out-performed the alternative uniform distribution, except in case of the dimerization. The reason for this becomes apparent, when examining Figure 4.1: In case of the dimerization model the most efficient constraints are found for $\lambda \approx -3$, while in case of the distributive modification they are located just above 0 (we observe a similar pattern for the exclusive switch case study). Therefore the sampling of efficient λ values is more likely using a uniform distribution for the dimerization case study, than it is for the others. Given that larger absolute values for λ seem unreasonable due their exponential influence on the weighting function and the problem of fixing a suitable interval for a uniform sampling scheme, the choice of a standard normal distribution for π_λ seems superior.

In Figure 4.5 we compare efficiencies for different maximum orders of constraints n_{max}. This comparison is performed for different choices of the redundancy rule and initial λ sample sizes n_λ. Again, for each parameter valuation 1000 estimations were performed. With respect to the maximum constraints order n_{max} we see a clear tendency, that the inclusion of second order constraints lessens the efficiency of the method. In case of a constant redundancy threshold it even dips below break-even for the distributive modification case study. This is not surprising, since the inclusion of second order moments increases the number of

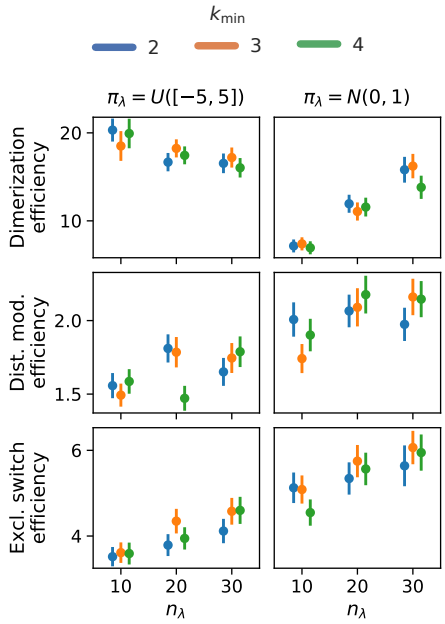


Figure 4.4: The efficiencies for different valuations of n_λ and k_{min} and choices of π_λ. The sample size was $n = 10{,}000$ in all cases with $d = 100$. The bars give the bootstrapped (1000 iterations) standard deviations.

initial constraints quadratically and the incurred cost, especially of the first iterations, lessens efficiency.

Figure 4.7 depicts the trade-off between the variance reduction σ_0^2/σ_1^2 versus the cost ratio c_0/c_1. Comparing the redundancy criterion based on a constant threshold ϕ_c to the others, we observe both a larger variance reduction and an increased cost. This is due to the fact, that more control variates are included throughout the simulations (**??**, Table A.3). Depending on the sample distribution π_λ and the case study, this permissive strategy may pay off. In case of the dimerization, for example, it pays off, while in case of the distributive modification it leads to a lower efficiency ratio. In the latter case the model is more complex, and therefore the set of initial control variates is larger. With a more permissive redundancy strategy, more control variates are kept (ca. 10 when using ϕ_c vs. ca. 2 to 3 for the others). The other

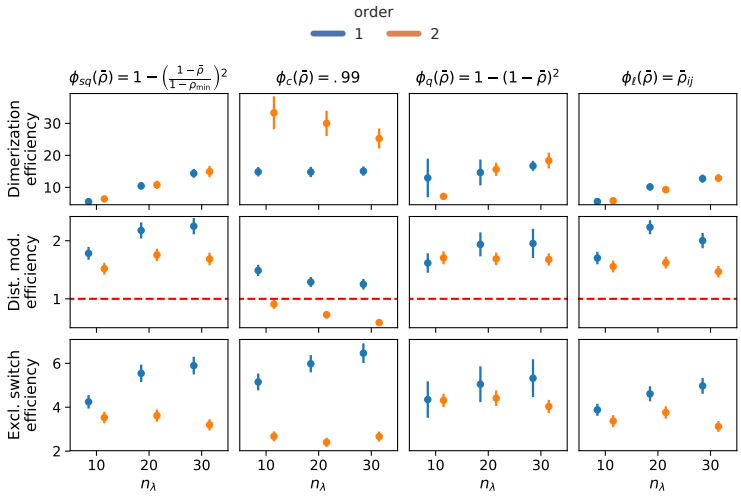


Figure 4.5: The efficiency for different redundancy policies ϕ and maximal moment orders n_{max}. The sample size was $n = 10{,}000$ in all cases with $d = 100$. Furthermore, $k_{min} = 3$, $\pi_\lambda = N(0, 1)$, and $n_{max} = 1$. The bars give the bootstrapped (1000 iterations) standard deviations.

redundancy boundaries move the results further in the direction of less variance reduction while keeping the cost increase low. On the opposite end is the linear ϕ_ℓ. The quadratic versions ϕ_q and ϕ_{sq} can be found in the middle of this spectrum.

We also observe, that an increase of n_λ is particularly beneficial, if the sampling distribution π_λ does not capture the parameter region of the highest correlations well. This can be seen for the Dimerization case study, where the variance reduction increases strongly with increasing sample size (Figure 4.6, **??**, Table A.3). Since $\pi_\lambda = N(0, 1)$, more samples are needed to sample efficient λ-values (cf. Figure 4.1).

In Figure 4.6 we give detailed information on the influence of algorithm parameters k_{min}, the number of initial λ values, and different redundancy rules. The λ sampling distribution π_λ is a standard normal.

Finally, we discuss the effect of the sample size n on the efficiency E. In Figure 4.8 we give both the efficiencies and the

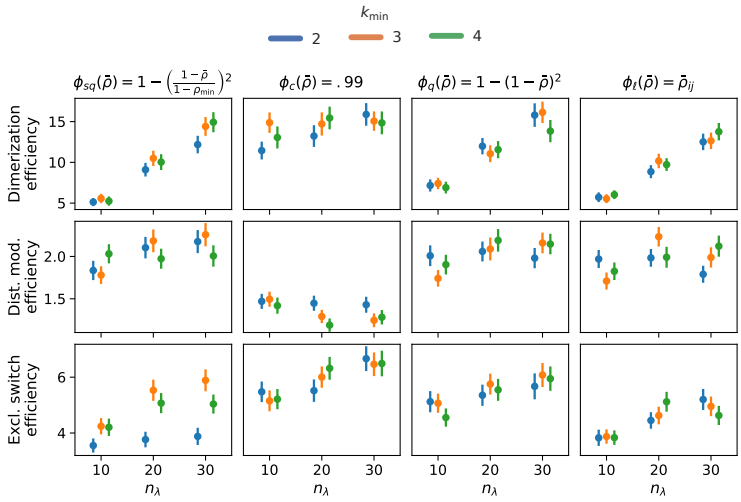


Figure 4.6: The empirical efficiencies for different n_λ and k_{min}. On the considered case studies. The sample size was $n = 10{,}000$ in all cases with $d = 100$. 1000 estimations were performed for each case. The bars give the bootstrapped (1000 iterations) standard deviations. The break-even $E = 1$ is marked by the dotted red line.

slowdown for different sample sizes. As a redundancy rule we used the unscaled quadratic function, 30 initial values of λ, and $k_{min} = 3$. With increasing sample size, the efficiency usually approaches an upper limit. This is due to the fact that most control variates are dropped early on and the control variates often remain the same for the rest of the simulations. If we assume there are no helpful control variates in the initial set and all would be removed at iteration 100, the efficiency would converge to 1 for $n \to \infty$.

4.6 CONCLUSION

In this work we have shown that known constraints on the moment dynamics can be successfully leveraged in simulation-based estimation of expected values. The empirical results indicate that the supplementing a standard SSA estimation with mom-

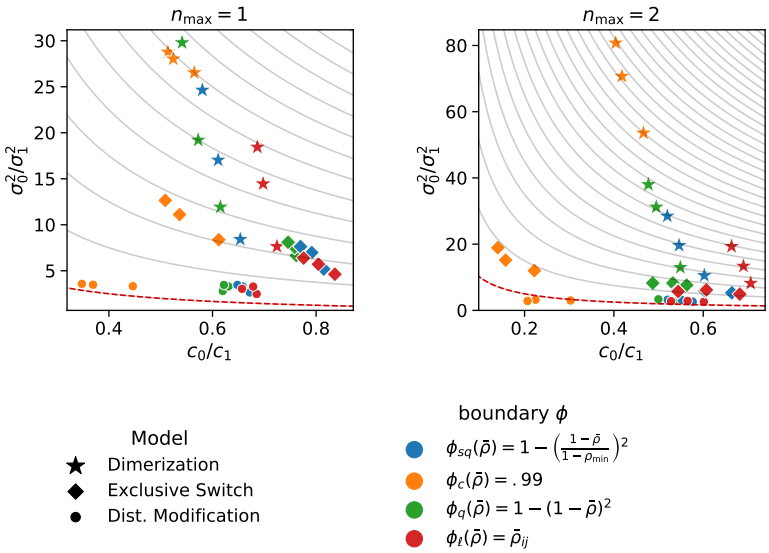
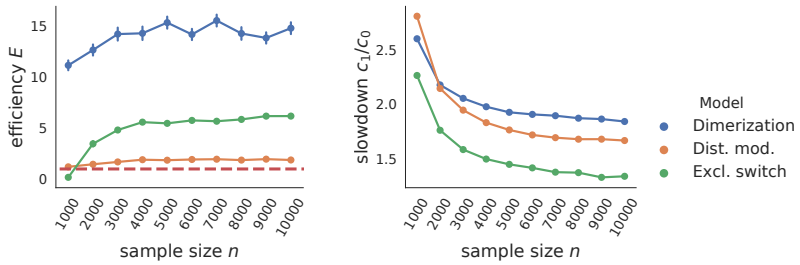


Figure 4.7: A visualisation of the trade-off between variance reduction σ_0^2/σ_1^2 and cost ratio c_0/c_1. Isolines for efficiencies are given in grey. The break-even is marked by the dashed red line. Markers of the same kind differ in n_λ and shift with increasing value upwards in variance reduction and lower in c_0/c_1, i.e. the shift is to the left and upwards with increasing n_λ. The sample size was $n = 10{,}000$ in all cases with $d = 100$. Furthermore, $k_{min} = 3$ and $\pi_\lambda = N(0, 1)$.

ent information can drastically reduce the estimators' variance. This reduction is paid for by accumulating information on the trajectory during simulation. However, the reduction is able to compensate for this increase. This means that for fixed costs, using estimates with control variates is more beneficial than using estimates without control variates.

While a variety of algorithmic variants was evaluated, many aspects remain subject to further study. In particular different choices of f and w in (4.1) may improve efficiency further. These choices become particularly interesting when moving from the estimation of simple first order moments to more complex queries such as behavioural probabilities of trajectories. In such cases,



(a) Efficiency v. sample size

(b) Slowdown v. sample size

Figure 4.8: The effect of sample size on the efficiency E and slowdown in the different case studies. The break-even $E = 1$ is marked by the dashed red line. The cost increase due to the variance reduction over different sample sizes.

one might even attempt to find efficient control variate functions using machine learning methods.

Another open question regarding this work is its performance when multiple quantities instead of a single quantity are to be estimated. In such a case, constraints would be particularly beneficial, if they lead to improvements as many estimation targets as possible.

Furthermore the identification of the best weighting parameters λ could be done in a more adaptive fashion. The presented scheme of a sampling from π_λ could be extended into a Bayesian-like procedure, wherein the values for λ are periodically re-sampled from a distribution that is adjusted according to the best-performing constraints up to that point.

Part III

AGGREGATION & REFINEMENT

We present a state-space lumping scheme that aggregates states in a grid structure. Approximations based on this lumping are used to iteratively refine relevant and truncate irrelevant parts of the state-space. This way, the algorithm learns a well-justified finite-state projection for different scenarios.

5 STATE-SPACE AGGREGATION

In this part, we propose a method to identify a truncation that optimizes the trade-off between the size of the considered state-space and the approximation error due to the finite state projection (FSP). To this end, we start with a very coarse-grained model abstraction that we refine iteratively. The coarse-grained model is based on an grid-shaped aggregation (i.e. lumping) scheme that identifies a set of macro-states. These macro-states can be used to compute an interim model solution that guides the refinement in the next step. We perform refinements until the approximation arrives at the resolution of the original model (i.e. each macro-state has only one constituent) such that the aggregation introduces no approximation error.

↪ *page 15*

5.1 MACRO-STATES

A macro-state is a collection of micro-states (or simply states) treated as one state in the aggregated model, which can be seen as an abstraction of the original model. The aggregation scheme defines a partitioning of the state-space. We choose a scheme based on a grid structure. That is, each macro-state is a hypercube in $\mathbb{Z}^{n_S}_{\geqslant 0}$.

Hence, each macro-state $\bar{x}_i(\ell^{(i)}, u^{(i)})$ (denoted by $\bar{x}_i$ for notational ease) can be identified using two vectors $\ell^{(i)}$ and $u^{(i)}$. The vector $\ell^{(i)}$ gives the corner closest to the origin, while $u^{(i)}$ gives the corner farthest from the origin. Formally,

$$\bar{x}_i = \bar{x}_i(\ell^{(i)}, u^{(i)}) = \{x \in \mathbb{N}^{n_S} \mid \ell^{(i)} \leqslant x \leqslant u^{(i)}\}, \tag{5.1}$$

where '$\leqslant$' denotes element-wise comparison.

In order to solve the aggregated model, we need to define transition rates between macro-states. Therefore, we assume that, given that the system is in a particular macro-state, all constituent states are equally likely (uniformity assumption). This assumption is the reason why the aggregated model provides only a coarse-grained approximation.

The uniformity assumption is a modeling choice yielding significant advantages. Firstly, it eases the computation of the rates between macro-states and, therefore, makes a fast solution of the aggregated model possible. Secondly, even though it induces an approximation error, it provides suitable guidance as uniformity assumption spreads out the probability mass conservatively. Hence, it becomes less likely that regions of interest are disregarded. Lastly, the uniformity assumption is theoretically well-founded, as it stems from the maximum entropy principle: In the absence of concrete knowledge about the probability distribution inside a macro-state, we assume the distribution with the highest uncertainty, i.e., the uniform distribution.

5.2 CONSTRUCTION

The grid structure makes the computation of transition rates between macro-states particularly convenient and computationally simple. Mass-action reaction rates can be given in a closed-form, due to the Faulhaber formulae (Knuth, 1993) and more complicated rate functions such as Hill-functions can often be handled as well by taking appropriate integrals (see Section 7.5.2.2).

Suppose, we are interested in the transition rate from macro-state $\bar{x}_i$ to macro-state $\bar{x}_k$ according to reaction j. Using the uniformity assumption, this is simply the mean rate of the states in $\bar{x}_i$ that go to $\bar{x}_k$ using j. However, only a small subset of constituents in $\bar{x}_i$ are actually relevant for this transition. Hence, we identify the subset of states of $\bar{x}_i$ that lie at the border to $\bar{x}_k$ and in such a way that applying reaction j shifts them to a state in $\bar{x}_k$. Then, we sum up the corresponding rates of these states. Lastly, we normalize according to the number of states inside of $\bar{x}_i$.

It is easy to see that the relevant set of border states is itself an interval-defined macro-state $\bar{x}_{i \xrightarrow{j} k}$. To compute this macro-state we can simply shift $\bar{x}_i$ by ν_j, take the intersection with $\bar{x}_k$ and project this set back. Formally,

$$\bar{x}_{i \xrightarrow{j} k} = ((\bar{x}_i + \nu_j) \cap \bar{x}_k) - \nu_j \,, \tag{5.2}$$

where the additions are applied element-wise to all states making up the macro-states. For ease of notation, we also define a general exit state

$$\bar{x}_{i \xrightarrow{j}} = ((\bar{x}_i + \nu_j) \setminus \bar{x}_i) - \nu_j. \tag{5.3}$$

This state captures all micro-states inside $\bar{x}_i$ that can leave the state via reaction j.

A particularly convenient feature of the transition states is, that they also are macro-states. That means, they also are specified by independent intervals in each dimension as in (5.1). This holds because all operations in both (5.2) and (5.3) preserve this structure.

EXAMPLE In Figure 5.1 we give an example of two adjacent macro states and the transition state from the left to the right via one reaction. As such it illustrates the result of the computation given in (5.2): The left state is shifted along the reaction vector, intersected with the right macro-state, and finally shifted back. ◇

Model 7 on page 96 gives this structure.

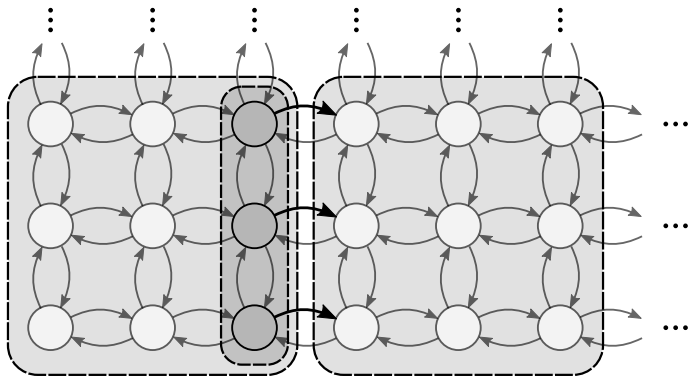


Figure 5.1: Two macro-states and a transition state from the left to the right.

This uniformity assumption gives rise to the following Q-matrix of the aggregated model:

$$\bar{Q}_{\bar{x}_i, \bar{x}_k} = \begin{cases} \sum_{j=1}^{n_R} \bar{\alpha}_j \left(\bar{x}_{i \xrightarrow{j} k} \right) / |\bar{x}_i| \,, & \text{if } \bar{x}_i \neq \bar{x}_k \\ -\sum_{j=1}^{n_R} \bar{\alpha}_j \left(\bar{x}_{i \xrightarrow{j}} \right) / |\bar{x}_i| \,, & \text{otherwise} \end{cases} \tag{5.4}$$

where

$$\bar{\alpha}_j(\bar{x}) = \sum_{x \in \bar{x}} \alpha_j(x). \tag{5.5}$$

is the sum of all rates belonging to reaction j in $\bar{x}$. In particular, the division by $|\bar{x}_i|$ in (5.4) is due to the uniformity assumption: According to the assumption, given the process is in $\bar{x}_i$ it is in each constituent micro-state of the transition state $\bar{x}_{i \xrightarrow{j} k}$ with probability $1/|\bar{x}_i|$. Therefore, each of the added micro-state rates in (5.5) is divided by the state volume $|\bar{x}_i|$.

Under the assumption of polynomial rates, as is the case for mass-action systems, we can compute the sum of rates over this transition set efficiently using Faulhaber's formula.

EXAMPLE Consider the following mass-action reaction $2X \xrightarrow{c} \varnothing$. For macro-state $\bar{x} = \{0, \dots, n\}$ we can compute the corresponding lumped transition rate

$$\begin{aligned}\bar{\alpha}(\bar{x}) &= \frac{c}{2} \sum_{i=1}^{n} i(i-1) = \frac{c}{2} \left(\sum_{i=1}^{n} i^2 - \sum_{i=1}^{n} i \right) \\ &= \frac{c}{2} \left(\frac{2n^3 + 3n^2 + n}{6} - \frac{n^2 + n}{2} \right)\end{aligned}$$

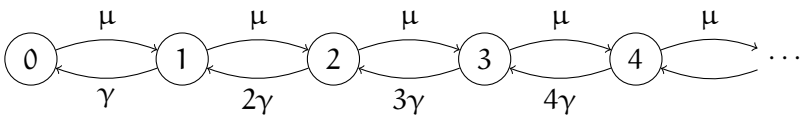
eliminating the explicit summation in the lumped propensity function. ◇

Interestingly, the lumped distribution tends to be less concentrated. This is due to the assumption of a uniform distribution inside macro-states. This effect is illustrated by the example of a birth-death process below. Due to this effect, an iterative refinement typically keeps an over-approximation in terms of state-space area. This is a desirable feature since relevant regions are less likely to be pruned due to lumping approximations.

EXAMPLE We illustrate the scheme on the birth-death process, i.e. Model 2. Its CTMC has the following generator matrix

$$Q = \begin{bmatrix} -\mu & \mu & 0 & & & \cdots \\ \gamma & -(\mu+\gamma) & \mu & 0 & & \cdots \\ 0 & 2\gamma & -(\mu+2\gamma) & \mu & 0 & \cdots \\ 0 & 0 & 3\gamma & -(\mu+3\gamma) & \mu & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

The structure is more obvious in the graph visualization:



Now we lump states in groups of 5 states. The states are constructed as

We omit the vector notation here for clarity because the process has a single dimension.

$$\bar{x}_k(5k, (5+1)k-1)$$

The transition states for the birth reaction $\varnothing \to S$ is

$$\bar{x}_{k \xrightarrow{1} k+1} = \{5(k+1)-1\}.$$

The lumped transition rate

$$\bar{\alpha}_1\left(\bar{x}_{k \xrightarrow{1} k+1}\right) = \mu.$$

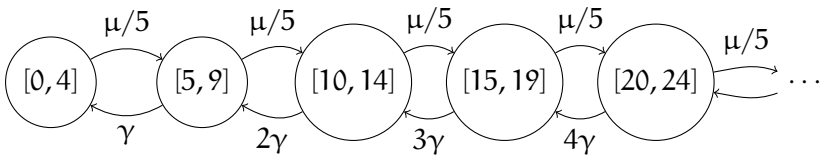
Similarly, the transition states for the death reaction $S \to \varnothing$ is

$$\bar{x}_{k \xrightarrow{2} k+1} = \{5(k+1)\}.$$

The lumped transition rate

$$\bar{\alpha}_2\left(\bar{x}_{k+1 \xrightarrow{1} k}\right) = 5k\gamma.$$

Using (5.4) the aggregated transitions are as follows.



In this example the rates remain in effect the same. But the same number of macro-states covers more micro-states.

The integration is done using an ad-hoc FSP on $[0, 200]$.

A forward integration of both, the model at original granularity and the aggregated version is shown in Figure 5.2. ◇

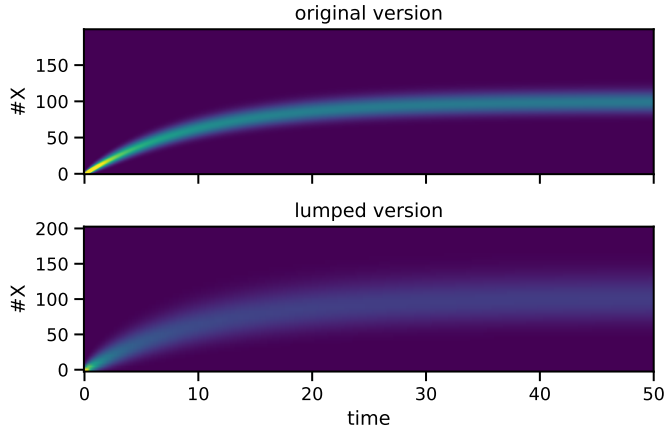


Figure 5.2: A lumping approximation of Model 2 on the state-space truncation to $[0, 200]$ on $t \in [0, 50]$. On the left-hand side solutions of a regular truncation approximation and a lumped truncation (macro-state size is 5) are given.

5.3 APPROXIMATION FEATURES

Considering the previous example, we observe that the lumped version shows a similar temporal dynamic, but the distributions are spread out more. This is a desirable feature because it indicates the location of the main probability mass with significantly less states.

These features are not valid in general and the aggregation scheme remains a heuristic approach. This is partly due to the MPM formalism allowing for arbitrary propensity functions. One

can construct a process such that a significant change in dynamics inside a macro-state would be missed. Mostly we encounter this phenomenon in models that exhibit a significant change near the zero-boundary for some species. Examples include the toggle switch with Hill-functions (Model 12) and die-out in epidemics models (Model 13). Other cases are rather rare in the standard model repertoire, but some awareness is necessary.

↪ *page 117*
↪ *page 120*

Fortunately, by refining the aggregation down to "full resolution", we can gain the guarantees inherent to FSP.

6 TRUNCATIONS FOR STATIONARY DISTRIBUTIONS

An important part of the analysis of such models concerns their long-run behavior. Given an ergodic underlying Markov chain, the chain's stationary distribution characterizes this behavior. For some special model classes, such as zero-deficiency networks (Anderson and Kurtz, 2011), analytical solutions for the stationary distribution are known. However, most models require numerical approaches, often based on some form of approximation to guarantee tractability. Those approaches can be based on stochastic simulation (Gillespie, 1977) (which for steady-state analysis tends to be slow and inaccurate) or moment-bounds via mathematical programming (Kuntz et al., 2019a). Here, we draw on numerical approaches based on state-space truncation, which represent a viable option to approximate stationary distributions (Kuntz et al., 2021a). Truncation-based approaches have the benefit of describing the complete dynamics within a finite subset of the typically very large or infinite state-space. As such, they enable the approximation of complex distributions that are not well-described by low-order moments.

The main step in the computation of such an approximation is the identification of a suitable truncation, i.e. a subset of the state-space encompassing most of the stationary probability mass. Existing methods typically rely on Foster-Lyapunov drift conditions to define such subsets (Dayar et al., 2011). While these truncations come with bounds on the contained stationary probability mass, they typically are far larger than necessary. The truncation is usually strongly constrained by the form of the chosen Lyapunov function (Dayar et al., 2011; Gupta, Briat, and Khammash, 2014). Optimizing over possible functions to identify efficient truncations is technically challenging and, to our knowledge, has not been demonstrated for general reaction networks (Milias-Argeitis and Khammash, 2014).

In this work, we address the identification of suitable truncations by using an aggregation-refinement scheme. Initially, a

Lyapunov analysis yields a set containing at least $1 - \epsilon$ of the stationary probability mass. On this subset of the state-space, we apply an aggregation scheme that groups together states in hypercube macro-states. Throughout each of these macro-states, we assume a uniform distribution among its constituent micro-states. This allows us to roughly analyze large portions of the state-space with exponentially fewer variables. We then iteratively truncate and refine the approximation based on the stationary distribution of this aggregated Markov chain. We keep only the most relevant macro-states and continue this scheme until the macro-states contain a single original state. In this way, we arrive at an effective truncation to compute an approximation of the stationary distribution.

We investigate the approximation results on case studies with known stationary distributions and complex models with intricate stationary distributions. We evaluate the truncation quality by assessing the stationary probability mass captured. To this end, we use analytical solutions and bounds given by a Lyapunov analysis. Further, we explore the control of the truncation size through the truncation parameter. Finally, we demonstrate the method on the p53 oscillator model exhibiting a complex stationary distribution.

6.1 RELATED WORK

ANALYTICAL SOLUTIONS For some specific models, analytical solutions for the stationary distribution have been found (Kurasov et al., 2018; Mélykúti, Hespanha, and Khammash, 2014). For the class of zero-deficiency networks, the stationary distribution is known to have a Poisson product form (Anderson, Craciun, and Kurtz, 2010). Monomolecular reaction networks can be solved explicitly, as well (Jahnke and Huisinga, 2007).

TRUNCATION-BASED ANALYSIS The analysis of countably infinite-sized state-spaces is often handled by pre-defined truncations (Kwiatkowska, Norman, and Parker, 2011). Sophisticated state-space truncations for the (unconditioned) forward analy-

sis have been developed that give lower bounds. They typically provide a trade-off between computational load and tightness of the bound (Andreychenko et al., 2011; Henzinger, Mateescu, and Wolf, 2009; Lapin, Mikeev, and Wolf, 2011; Mikeev et al., 2013; Munsky and Khammash, 2006). Such methods cannot be directly applied to the estimation of stationary distributions because the approximation usually introduces a sink-state.

Truncations for stationary distributions often involve re-direction schemes for transitions leaving and entering the subset. A comprehensive survey of such state-space truncation methods can be found in (Kuntz et al., 2021b). A popular method of identifying truncations is the construction of a suitable Lyapunov function. Beyond their use for establishing ergodicity (Dayar et al., 2011; Gupta, Briat, and Khammash, 2014; Meyn and Tweedie, 1993), these functions can be used to obtain truncations, guaranteed to contain a certain amount of stationary probability mass (Dayar et al., 2011). Using Lyapunov functions for the construction of truncations often leads to very conservative sets (Milias-Argeitis and Khammash, 2014). Different approaches have been employed to find truncations: In Gupta, Mikelson, and Khammash (2017) SSA estimates are used to set up an increasing family of truncations.

MOMENT-BASED APPROXIMATION Apart from approaches based on state-space truncations, moment-based approaches have been particularly popular recently (Dowdy and Barton, 2018a; Ghusinga et al., 2017; Kuntz et al., 2019a; Sakurai and Hori, 2017). Such approaches are based on the fact that particular matrices of distributional moments such as mean and variance are positive semi-definite. Along with linear constraints stemming from the Kolmogorov equations (Backenköhler, Bortolussi, and Wolf, 2016), a semi-definite program can be formulated and solved using existing tools. While this method is suited to compute bounds on both moments and subsets of the state-space, its application is limited, due to numerical issues inherent in the formulation (Dowdy and Barton, 2018a).

An approach where quantities are only described in terms of their magnitude has been proposed in Ceska and Kretínský (2019). This allows for an efficient qualitative analysis of both dynamic and transient behavior.

AGGREGATION An aggregation scheme similar to the one used here has been previously proposed in Backenköhler et al. (2021) to analyze the bridging problem on Markov population models. This is the problem of analyzing process dynamics under both initial and terminal constraints.

Aggregation-based numerical methods for computing the stationary distribution of discrete or continuous-time Markov chains have been studied in previous work. Popular approaches rely on an alternation of aggregation and disaggregation of the state-space (Schweitzer, 1991; Stewart, 1994). In the case of stiff chains, such aggregations are typically based on a separation of time-scales (Cao and Stewart, 1985). However, these methods have been developed for finite chains with arbitrary structure and are motivated by numerical issues of standard methods such as the power method or Jacobi iteration (Stewart, 1994). They do not consider a truncation of irrelevant states, while here our aggregation approach is used to determine the most relevant states under stationary conditions in large or infinite chains with population structure.

6.2 TRUNCATION-BASED APPROXIMATION

In many relevant cases, the state-space is huge or infinite and therefore the stationary solution cannot be computed directly. To make such a computation possible we have to restrict ourselves to a finite manageable subset of the state-space and assume the majority of the probability mass is concentrated within that finite subset. The main problem is to deal with the transitions leading to and from the truncated set (cf. Figure 6.1). In forward analysis, the outgoing transitions are simply redirected into a sink-state. This way, a forward analysis provides lower bounds since mass leaving the truncation does not re-enter. This approach, however,

is unsuitable for the computation of stationary distributions because mass would accumulate in the sink-state leading to a distribution assigning all mass to it. Therefore, transitions leaving the truncation need to be redirected back into the truncation.

The process' dynamics outside the truncation are defined by the *stochastic complement* (Spieler, 2014). If its behavior was known, one could redirect outgoing to incoming transitions optimally and preserve the correct stationary distribution. However, this reentry distribution is typically unknown in most relevant cases. Many different reentry distributions have been used, such as redirecting to some internal state or states with incoming transition from outside the truncation. Kuntz et al. (2021a) provides a comprehensive review of such methods.

The most natural choice is to pick a reentry distribution that redirects mass to states with incoming transitions from truncated states (cf. Figure 6.1 (center)).

Using varying redirections, we can compute bounds on the stationary probability conditioned on a truncation (Spieler, 2014, Thm. 14). To do this, one has to compute the stationary distribution for every possible way of connecting all outgoing to a single incoming transition. Naturally, such an algorithm is rather expensive since one has to solve a linear system for each combination. Therefore this method of computing bounds is costly on very large truncations, often given by Lyapunov functions.

When computing an approximation instead of bounds, we employ a uniform redirection scheme: Outgoing transitions are split uniformly among incoming transitions. Due to the threshold-based truncation scheme, we are likely to end up with a somewhat uniform distribution over in-boundary states (see Section 6.2.2).

The identification of good truncations remains a major task in such approximations. Using approaches such as Lyapunov functions (Section 2.6.1) (Dayar et al., 2011) or moment-bounds (Kuntz et al., 2021a) can provide a good initial estimate, but typically the resulting truncations are far larger than necessary. This leads to dramatically increased computational costs, especially when bounding methods mentioned above are performed. Until

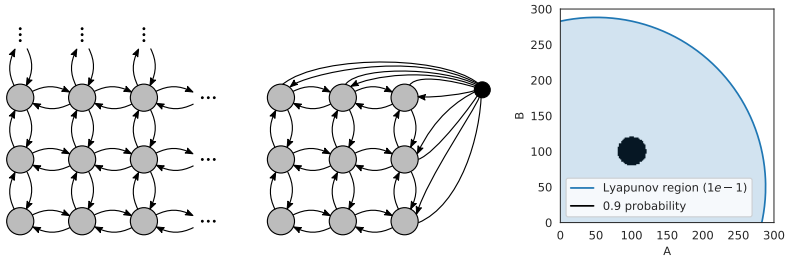


Figure 6.1: (left) A countably infinite state-space. (center) Outgoing transitions are re-directed (according to the reentry distribution) to states that have incoming transitions from outside the truncation. (right) A comparison of the area prescribed by a Lyapunov analysis using Geobound and threshold 0.1 and the minimal area containing 0.9 stationary probability mass. The model is a parallel birth-death process (Model 7).

a system for a larger truncation is solved, the precise location of most of the probability mass is often unknown. Instead of solving the full system for such a large space, we employ an aggregation scheme to cover large areas of the state-space with exponentially fewer variables.

Error bounds have been derived for increasing truncation sets in the case of linear Lyapunov functions (Gupta, Mikelson, and Khammash, 2017). However, until now it has not been shown that these bounds are applicable in practice (Meyn, Tweedie, et al., 1994). Alternatively, one can monitor the product of the probability-ouflow rate and the maximum L1-norm, which bounds the approximation error up to a constant $M > 0$, assuming a linear Lyapunov function exists (Gupta, Mikelson, and Khammash, 2017).

6.2.1 *Initial Aggregation*

The initial aggregated space $\hat{S}^{(0)}$ should encompass all regions of the state-space that could contain significant mass because states outside this initial area will not be refined. In principle, multiple approaches could be used to identify such a region.

One possibility is the computation of moment bounds for the stationary distribution (Dowdy and Barton, 2018a; Ghusinga et al., 2017). Based on these bounds on expectations and covariances, an initial truncation could be fixed. The approach we use here is to identify such a region by a Lyapunov analysis (Dayar et al., 2011). This way, we obtain a polynomial describing a semi-algebraic subset of the entire state-space containing $1 - \epsilon_\ell$ of the mass, where $\epsilon_\ell > 0$ can be fixed arbitrarily. These sets usually are far larger than a minimal set containing $1 - \epsilon_\ell$ of stationary probability mass would be. As an initial aggregation, we build an aggregation on a subset $[0..n]^{n_S} \subset \mathcal{S}$ containing the set prescribed by the Lyapunov analysis. We also employ this approach to estimate errors in the evaluation. Specifically, we employ the tool Geobound (Spieler, 2010) with L2-norm as function g implementing techniques presented in Dayar et al. (2011).

In many cases, simple choices of g such as the L1- or L2- norm are sufficient. However, the sets resulting from such functions are often very conservative. Consider Figure 6.1 (right) as an example, where the Lyapunov truncation with $\epsilon_\ell = 0.1$ for two parallel birth-death processes (Model 7) is compared to the smallest set containing 0.9 of stationary probability. Clearly, the area given by the Lyapunov function is magnitudes larger than necessary to capture probability mass consistent with ϵ_ℓ.

6.2.2 *Iterative Refinement Algorithm*

The refinement algorithm (Algorithm 3) starts with a set of large macro-states that are iteratively refined, based on approximate stationary distributions. We start by constructing square macro-states of size 2^m in each dimension for some $m \in \mathbb{N}$ such that they form a large-scale grid $\mathcal{S}^{(0)}$. Hence, each initial macro-state has a volume of $(2^m)^{n_S}$. This choice of grid size is convenient because we can halve states in each dimension. Moreover, this choice ensures that all states have an equal volume and we end up with unit-sized macro-states, equivalent to a truncation of the original non-lumped state-space.

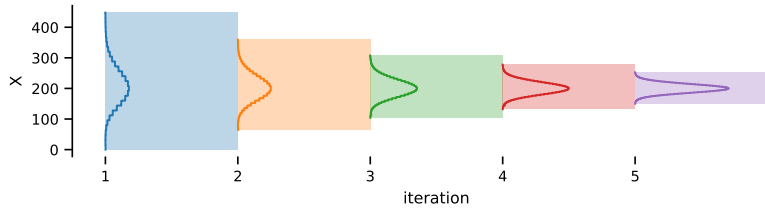


Figure 6.2: The state-space refinement algorithm on a birth-death process. From left to right the state size is halved and states with low probability are removed from the truncation. The final truncation is a typical truncation with states of size 1 and the initial states are of size 2^4.

Algorithm 3: Approximating the stationary distribution

input : Initial partitioning $\mathcal{S}^{(0)}$, truncation threshold ϵ
output: approximate stationary distribution $\hat{\pi}_\infty$

1 **for** $i = 1, \ldots, m$ **do**
2 $\quad \hat{\pi}_\infty^{(i)} \leftarrow$ approximate stationary distribution on $\mathcal{S}^{(i)}$;
3 $\quad \mathcal{R} \leftarrow$ smallest $\mathcal{R}' \subseteq \mathcal{S}^{(i)}$ s. t. $\sum_{\bar{x} \in \mathcal{R}'} \hat{\pi}_\infty^{(i)}(\bar{x}) \geqslant 1 - \epsilon$;
4 $\quad \mathcal{S}^{(i+1)} \leftarrow \bigcup_{\bar{x} \in \mathcal{R}} \text{split}(\bar{x})$;
5 $\quad$ update $\hat{Q}$-matrix;
6 **return** $\hat{\pi}_\infty^{(m)}$;

An iteration of the state-space refinement starts by computing the stationary distribution, using the lumped $\hat{Q}$-matrix. Based on a threshold parameter $\epsilon > 0$ states are either removed or split (line 4), depending on the mass assigned to them by the approximate stationary probabilities $\hat{\pi}_\infty^{(i)}$. Thus, each macro-state is either split into 2^{n_S} new states or removed entirely. The result forms the next lumped state-space $\mathcal{S}^{(i+1)}$. The $\hat{Q}$-matrix is updated (line 5) using (5.4) to calculate the transition rates of the next aggregated truncation $\mathcal{S}^{(i+1)}$. Entries of truncated states are removed from the updated transition matrix. Transitions leading to them are re-directed according to the re-entry matrix (Section 6.2). After m iterations (we started with states of side lengths 2^m) we have a

standard FSP scheme on the original model tailored to computing an approximation of the stationary distribution.

This way, the refinement algorithm focuses only on those parts of the state-space contributing most to the stationary distribution. For instance, in Figure 6.2 the stationary probability mass mostly concentrates around $\#S = 200$. Therefore, states that are further away from this area can be dropped in further refinement. This filtering (line 3 in Algorithm 3) ensures that states contributing significantly to $\hat{\pi}_\infty^{(i)}$ will be kept and refined in the next iteration. The selection of states is done by sorting states in descending order according to their approximate probability mass. This ensures the construction of the smallest possible subset chosen for refinement according to the approximation. Then states are collected until their overall approximate mass is above $1 - \epsilon$.

An interesting feature of the aggregation scheme is that the distribution tends to spread out more. This is due to the assumption of a uniform distribution inside macro-states. To gain an intuition, consider a macro-state that encompasses a peak of the stationary distribution. If we re-distribute the actual probability mass inside this macro-state uniformly, a higher probability is assigned to states at the macro-state's border. When plugging such macro-states together, this increased mass away from the peak will increase the mass assigned to adjacent macro-states. This effect is illustrated by the example of a birth-death process in Figure 6.2. Due to this effect, an iterative refinement typically keeps an over-approximation in terms of state-space area. This is a desirable feature since relevant regions are less likely to be pruned due to lumping approximations.

6.3 RESULTS

A prototype was implemented in Rust 1.50 and Python 3.8. The linear systems were solved either using Numpy (Harris et al., 2020) for up to 5000 states, or the sparse linear solver as available through Scipy (Virtanen et al., 2020), or the iterative biconjugate gradient stabilized algorithm (Vorst, 1992) (up to 10,000 iterations and absolute tolerance 10^{-16}).

The examples that we consider in the sequel are typical benchmarks for the analysis of MPMs. For most of them, appropriate Lyapunov functions have been determined using Geobound (Spieler, 2010, 2014). However, the corresponding Lyapunov sets containing at least $1 - \epsilon_\ell$ of the stationary probability mass are very large for typical choices of ϵ_ℓ (e.g. $\epsilon_\ell \in \{0.1, 0.05, 0.001\}$). Even for extremely large ϵ_ℓ, say $\epsilon_\ell = 0.8$, the remaining state-space may still be huge (e.g, 15,198 states).

6.3.1 *Parallel Birth-Death Process*

We first examine the algorithm on the simple example of two parallel, uncoupled birth-death processes.

Model 7 (Parallel Birth-Death Process)**.** *Two uncoupled parallel birth-death processes result in a simple stationary distribution that is given by a product of two Poisson distributions.*

$$\varnothing \xrightarrow{\rho} A \qquad A \xrightarrow{\delta} \varnothing \qquad \varnothing \xrightarrow{\rho} B \qquad B \xrightarrow{\delta} \varnothing$$

As a parameterization we choose $\rho = 100$ *and* $\delta = 1$.

For this model, the stationary distribution is known to be the product of two Poisson distributions with rate ρ/δ.

According to the Lyapunov analysis with a 1×10^{-4} bound, we fix the initial truncation to a 70×70 grid of macro-states with size 2^7 in each dimension. This implies 8 iterations of the algorithm to arrive at a truncation with the original granularity. In Figure 6.3, we illustrate the truncations of different iterations. Over the iterations, the covered area decreases, while the aggregation granularity increases. The final truncation distribution approximation is also depicted and covers $1 - 1.27 \times 10^{-2}$ of the true stationary distribution (cf. Table 6.1).

For this case study, we also compute state-wise bounds on the probabilities conditioned on the truncation as discussed in Section 6.2. In Figure 6.4, we present the difference between upper and lower bound for $\epsilon = 0.1$. We observe intervals that are narrowest in the truncation's interior near the distribution's mode. The largest intervals or the largest absolute uncertainty is present

in the boundary states. This indicates, that the specific reentry distribution has little effect on the main approximate stationary mass. More detailed results on the intervals' magnitudes are given in Table 6.1.

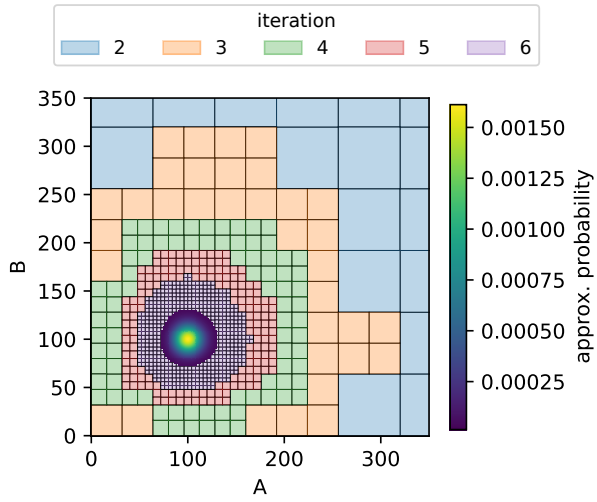


Figure 6.3: Truncations of different iterations are layered on top of each other. At higher iterations, truncations cover less area but increase in detail, due to the refinement of macro-states. The final approximation is indicated by its approximate probabilities.

6.3.2 *Exclusive Switch*

The exclusive switch (Barzel and Biham, 2008) has three different modes of operation, depending on the DNA state, i.e. on whether a protein of type one or two is bound to the DNA.

Model 8 (Exclusive Switch)**.** *The exclusive switch model consists of a promoter region that can express both proteins* P_1 *and* P_2*. Both can bind to the region, suppressing the expression of the other protein. For*

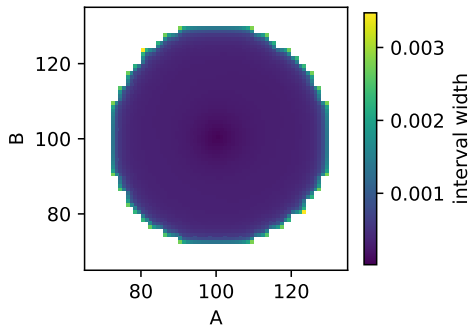


Figure 6.4: Results for Model 7 with truncation threshold $\epsilon = 0.1$. The difference between the upper and lower bounds on the probability conditioned on the truncation.

	threshold parameter ϵ			
	1×10^{-1}	1×10^{-2}	1×10^{-3}	1×10^{-4}
total width	1.2336	3.09×10^{-2}	5.39×10^{-4}	8.12×10^{-6}
max. width	3.47×10^{-3}	9.29×10^{-5}	4.04×10^{-7}	4.65×10^{-9}
outside mass	1.27×10^{-2}	1.05×10^{-4}	1.05×10^{-6}	1.06×10^{-8}

Table 6.1: Results for Model 7 : The characteristics of the lower-upper bound intervals on the conditional probability and the mass not contained in the truncation are given.

certain parameterizations, this leads to a bi-modal or even tri-modal behavior.

$$D \xrightarrow{\rho_1} D + P_1 \qquad D \xrightarrow{\rho_2} D + P_2 \qquad P_1 \xrightarrow{\lambda} \varnothing \qquad P_2 \xrightarrow{\lambda} \varnothing$$

$$D + P_1 \xrightarrow{\beta} D.P_1 \qquad D.P_1 \xrightarrow{\gamma_1} D + P_1 \qquad D.P_1 \xrightarrow{\rho_1} D.P_1 + P_1$$

$$D + P_2 \xrightarrow{\beta} D.P_2 \qquad D.P_2 \xrightarrow{\gamma_2} D + P_2 \qquad D.P_2 \xrightarrow{\rho_2} D.P_2 + P_2$$

We choose parameter values $\rho_1 = 0.7$, $\rho_2 = 0.6$, $\lambda = 0.02$, $\beta = 0.005$, $\gamma_1 = 0.06$, *and* $\gamma_2 = 0.05$.

Since the exclusive switch models mutually exclusive binding of proteins to a single genetic locus, we know a priori that there

are exactly three distinct operating modes. In particular are D, $D.P_1$, and $D.P_2$ mutually exclusive such that

$$X_D(t) + X_{D.P_1}(t) + X_{D.P_2}(t) = 1, \quad \forall t \geqslant 0.$$

This model characteristic often leads to bi-modal stationary distributions, where one or the other protein is more abundant depending on the genetic state.

Accordingly, we adjust the initial truncation: The state-space for the DNA states is not lumped. Instead we "stack" lumped approximations of the P_1-P_2 plane upon each other. Such special treatment of DNA states is common for such models (Lapin, Mikeev, and Wolf, 2011). Using Lyapunov analysis for threshold 0.001, we fix an initial state-space of 63×63 macro-states with size 2^7. Detailed results for different parameters ϵ are presented Table A.5. We compute error bounds using a worst-case analysis based on reference solutions provided by Geobound with $\epsilon_\ell = 0.01$. We observe a strong decrease in both upper bounds on the total absolute and maximal absolute error in the final iteration. Interestingly, the errors between different thresholds are very close in earlier iterations. This is mainly due to the usage of absolute errors which causes probabilities close to the mode dominate.

↪ page 146

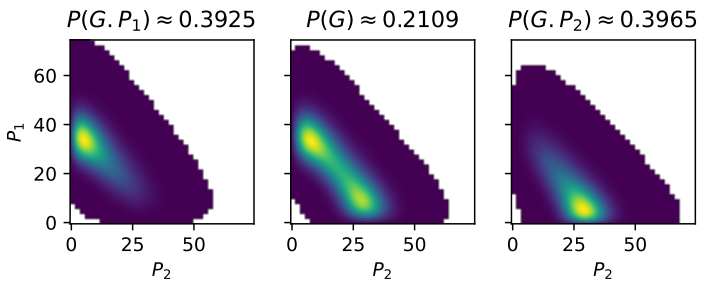


Figure 6.5: The approximate stationary distribution of the exclusive switch (Model 8) obtained with $\epsilon = 1 \times 10^{-4}$.

Using Geobound we observe that our final truncation captures the stationary mass very well (cf. Table 6.2 and Figure 6.5). We

use the Geobound's lower bounds with $\epsilon_\ell = 1\times10^{-2}$ and find that the uncovered mass by the aggregation-based truncation is magnitudes lower than ϵ or close to it (for $\epsilon = 0.1$). While they capture the mass well, they are much smaller than the Geobound truncation ($\epsilon_\ell = 0.1$) with 16,780 states, regardless of the threshold parameter ϵ.

	threshold parameter ϵ			
	1×10^{-1}	1×10^{-2}	1×10^{-3}	1×10^{-4}
total width	5.5171	1.5559	2.89×10^{-2}	3.71×10^{-4}
max. width	1.58×10^{-1}	3.30×10^{-3}	3.47×10^{-5}	3.84×10^{-7}
outside mass $\leqslant$	1.52×10^{-1}	1.29×10^{-3}	2.02×10^{-5}	2.72×10^{-7}

Table 6.2: Results for Model 8: The characteristics of the lower-upper bound intervals on the conditional probability and the upper bound on mass not contained in the truncation are given.

In Figure 6.6, we show the effect of the threshold parameter ϵ on the size of the final truncation. We observe a roughly linear increase in size with an exponential decrease of ϵ.

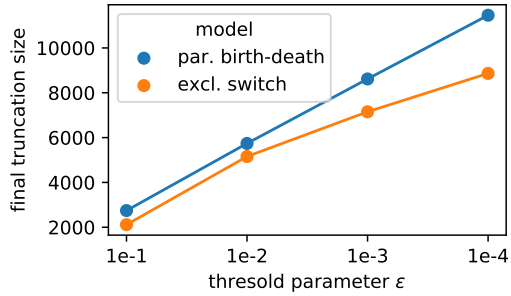


Figure 6.6: The sizes of the final truncation vs. the threshold parameter ϵ (Model 2 and Model 8).

6.3.3 *p53 Oscillator*

We now consider a model of the interactions of the tumor suppressor p53 (Geva-Zatorsky et al., 2006). The system describes the negative feedback loop between p53 and the oncogene Mdm2. Species pMdm2 models a precursor to Mdm2. This model is particularly interesting due to its complex three-dimeansional oscillatory behavior. The model is ergodic with a unique stationary distribution (Gupta, Briat, and Khammash, 2014).

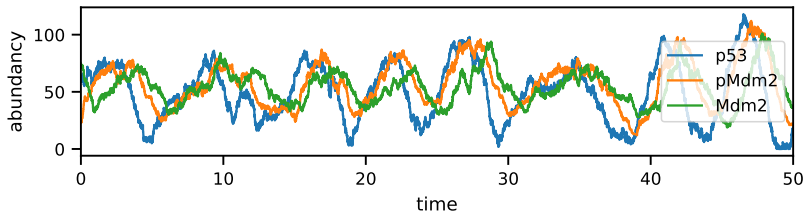


Figure 6.7: A sample trajectory illustrating the oscillatory long-run behavior.

Model 9 (p53 Oscillator)**.**

$$\varnothing \xrightarrow{k_1} p53 \qquad p53 \xrightarrow{k_2} \varnothing \qquad p53 \xrightarrow{k_4} p53 + pMdm2$$
$$p53 \xrightarrow{\alpha_4(\cdot)} \varnothing \qquad pMdm2 \xrightarrow{k_5} Mdm2 \qquad Mdm2 \xrightarrow{k_6} \varnothing$$

The non-polynomial degradation reaction rate

$$\alpha_4(x) = k_3 x_{Mdm2} \frac{x_{p53}}{x_{p53} + k_7}\,.$$

The parameterization based on (Ale, Kirk, and Stumpf, 2013) is $k_1 = 90$, $k_2 = 0.002$, $k_3 = 1.7$, $k_4 = 1.1$, $k_5 = 0.93$, $k_6 = 0.96$, *and* $k_7 = 0.01$.

With the exception of propensity function α_4, we can compute the transition rates $\bar{\alpha}_i$ using the Faulhaber formulae, as discussed in Chapter 5. We consider α_4 separately, because it is non-polynomial and therefore, we have to make an approximation. The fraction occurring in the non-linear propensity function

α_4 can roughly be characterized as an activation function: Due to the low value of parameter $k_7 = 0.01$ we can approximate

Note, that $\sum_{i=0}^{n} i/(i+k_7)$ can be solved analytically. However, the approximation presented above is much simpler to compute.

$$\frac{x_{p53}}{x_{p53}+k_7} \approx \begin{cases} 0 & \text{if } x_{p53} = 0 \\ 1 & \text{otherwise} \end{cases}$$

We use this approximation at the coarser levels of aggregation to efficiently compute the approximate transition rate $\bar{\alpha}_4$. At the finest granularity we switch back to exact propensity function α_4.

We now derive Lyapunov-sets for the p53 oscillator case study (Model 9). Let the Lyapunov function

$$g(x) = 120x_{p53} + 0.2x_{pMdm2} + 0.1x_{Mdm2}\,. \tag{6.1}$$

Then the drift

$$\begin{aligned} d(x) =& -\frac{k_3 x_{Mdm2} x_{p53}}{x_{p53}+k_7} - 0.1k_6 x_{Mdm2} + 120k_1 \\ & - 120k_2 x_{p53} + 0.2k_4 x_{p53} - 0.1k_5 x_{pMdm2} \\ =& -\frac{204 x_{Mdm2} x_{p53}}{x_{p53}+0.01} - 0.096 x_{Mdm2} - 0.02 x_{p53} \\ & - 0.0093 x_{pMdm2} + 10{,}800\,. \end{aligned} \tag{6.2}$$

Clearly, $c = \sup_{x\in S} d(x) = 10{,}800$. In particular, the supremum c is at the origin since all non-constant terms are negative. The slowest rate of decrease for (6.2) is x_{p53} with $x_{Mdm2} = x_{pMdm2} = 0$. We are content with a superset of a Lyapunov set (2.30) for some threshold ϵ_ℓ. Therefore taking (2.30), we can solve the inequality

$$\frac{\epsilon_\ell}{c}(c - 0.02 x_{p53}) > \epsilon_\ell - 1$$

for x_{p53} and

$$\frac{c}{0.02\epsilon_\ell} < x_{p53}\,.$$

Therefore

$$\pi_\infty\left(\left\{x \in S \mid \frac{c}{0.2\epsilon_\ell} < \|x\|\right\}\right) > 1 - \epsilon_\ell\,.$$

Due to the exponential increase stemming from the three-dimensional nature of this model, we only evaluated with parameter $\epsilon = 0.1$. According to the Lyapunov analysis shown above, the area covered by an $6 \times 6 \times 6$ macro-states with size 2^{20}, covers 0.9 of stationary mass. A truncation of this same area would consist of 226,492,416 states instead of the 216 macro-states. The model has a striking oscillatory behavior (cf. Figure 6.7) that is reflected in its stationary distribution. This feature is well-captured in the approximate distribution, where the oscillatory behavior leads to a complex stationary distribution (cf. Figure 6.9). This distribution leads to a non-trivial truncation (357,488 states) which is tailored to the main stationary mass (Figure 6.8).

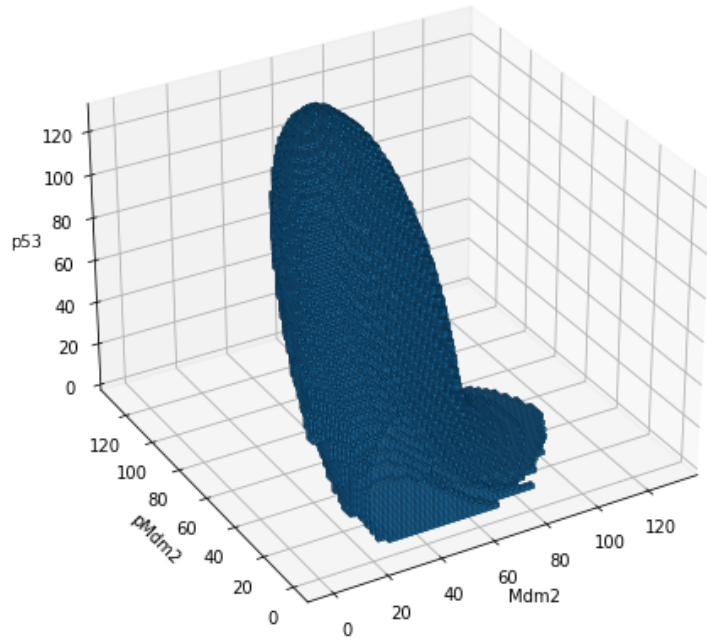


Figure 6.8: The final truncation at original granularity derived for the p53 oscillator.

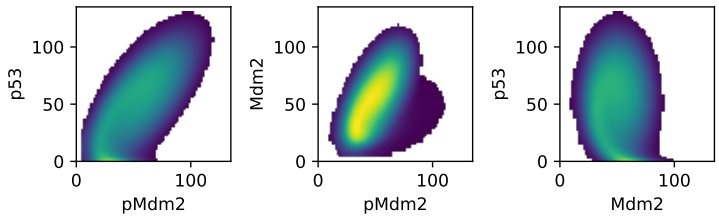


Figure 6.9: The approximate marginal distributions of the stationary distribution based on the truncation derived with $\epsilon = 0.1$.

6.4 CONCLUSION

State-of-the-art methods for numerically calculating the stationary distribution of Markovian population models rely on coarse truncations of irrelevant parts of large or infinite discrete state-spaces. These truncations are either obtained from the stationary statistical moments of the process or from Lyapunov theory. They are limited in shape because these methods do not take into account the detailed steady-state flow within the truncated state-space but only consider the average drift or stationary moments.

Here, we propose a method to find a tight truncation that is not limited in its shape and iteratively optimizes the set based on numerically cheap solutions of abstract intermediate models. It captures the main portion of probability mass even in the case of complex behaviors efficiently. In particular, the method represents another option, where Lyapunov analysis leads to forbiddingly large truncations.

7 ANALYSIS UNDER TERMINAL CONSTRAINTS

Many tasks, such as the analysis of rare events or the inference of agent counts under partial observations naturally introduce terminal constraints on the system. In these cases, the system's initial state is known, as well as the system's (partial) state at a later time-point. The probabilities corresponding to this so-called *bridging problem* are often referred to as *bridging probabilities* (Golightly and Sherlock, 2019; Golightly and Wilkinson, 2011). For instance, if the exact, full state of the process X_t has been observed at time $t = 0$ and $t = T$, the bridging distribution is given by

$$\Pr(X_t = x \mid X_0 = x_0, X_T = x_g)$$

for all states x and times $t \in [0, T]$. Often, the condition is more complex, such that in addition to an initial distribution, a terminal distribution is present. Such problems typically arise in a Bayesian setting, where the a priori behavior of a system is filtered such that the posterior behavior is compatible with noisy, partial observations (Broemeling, 2017; Huang et al., 2016). For example, time-series data of protein levels is available while the mRNA concentration is not (Adan et al., 2017; Huang et al., 2016). In such a scenario our method can be used to identify a good truncation to analyze the probabilities of mRNA levels.

Bridging probabilities also appear in the context of rare events. Here, the rare event is the terminal constraint because we are only interested in paths containing the event. Typically researchers have to resort to Monte-carlo simulations in combination with variance reduction techniques in such cases (Daigle Jr et al., 2011; Kuwahara and Mura, 2008).

Efficient numerical approaches that are not based on sampling or ad-hoc approximations have rarely been developed.

Here, we combine state-of-the-art truncation strategies based on a forward analysis (Andreychenko et al., 2011; Lapin, Mikeev, and Wolf, 2011) with a refinement approach that starts from an abstract MPM with lumped states. We base this lumping on a grid-like partitioning of the state-space. Throughout a lumped state,

we assume a uniform distribution that gives an efficient and convenient abstraction of the original MPM. Note that the lumping does not follow the classical paradigm of Markov chain lumpability (Buchholz, 1994) or its variants (Dayar and Stewart, 1997). Instead of an approximate block structure of the transition-matrix used in that context, we base our partitioning on a segmentation of the molecule counts. Moreover, during the iterative refinement of our abstraction, we identify those regions of the state-space that contribute most to the bridging distribution. In particular, we refine those lumped states that have a bridging probability above a certain threshold δ and truncate all other macro-states. This way, the algorithm learns a truncation capturing most of the bridging probabilities. This truncation provides guaranteed lower bounds because it is at the granularity of the original model.

7.1 RELATED WORK

TERMINAL CONSTRAINTS The problem of endpoint constrained analysis occurs in the context of Bayesian estimation (Särkkä, 2013). For MPMs, this problem has been addressed by Huang et al. (2016) using moment closure approximations and by Wildner and Koeppl (2019) further employing variational inference. Golightly and Sherlock modified stochastic simulation algorithms to approximatively augment generated trajectories (Golightly and Sherlock, 2019). Since a statistically exact augmentation is only possible for few simple cases, diffusion approximations (Golightly and Wilkinson, 2005) and moment approximations (Milner, Gillespie, and Wilkinson, 2013) have been employed. Such approximations, however, do not give any guarantees on the approximation error and may suffer from numerical instabilities (Schnoerr, Sanguinetti, and Grima, 2014).

RARE EVENT PROBABILITIES The bridging problem also arises during the estimation of first passage times and rare event analysis. Approaches for first-passage times are often of heuristic nature (Bortolussi and Lanciani, 2014; Hayden, Stefanek, and Bradley, 2012; Schnoerr et al., 2017). Rigorous approaches yield-

ing guaranteed bounds are currently limited by the performance of state-of-the-art optimization software (Backenköhler, Bortolussi, and Wolf, 2020). In biological applications, rare events of interest are typically related to the reachability of certain thresholds on molecule counts or mode switching (Strasser, Theis, and Marr, 2012). Most methods for the estimation of rare event probabilities rely on importance sampling (Daigle Jr et al., 2011; Kuwahara and Mura, 2008). For other queries, alternative variance reduction techniques such as control variates are available (Backenköhler, Bortolussi, and Wolf, 2019). Apart from sampling-based approaches, dynamic finite-state projections have been employed by Mikeev, Sandmann, and Wolf (2013), but are lacking automated truncation schemes.

STATE-SPACE TRUNCATION The analysis of countably infinite state-spaces is often handled by a pre-defined truncation (Kwiatkowska, Norman, and Parker, 2011). Sophisticated state-space truncations for the (unconditioned) forward analysis have been developed to give lower bounds and rely on a trade-off between computational load and tightness of the bound (Andreychenko et al., 2011; Henzinger, Mateescu, and Wolf, 2009; Lapin, Mikeev, and Wolf, 2011; Mikeev et al., 2013; Munsky and Khammash, 2006).

REACHABILITY IN VERIFICATION Reachability – relevant in the context of probabilistic verification (Bortolussi and Lanciani, 2014; Neupane et al., 2019) – is a bridging problem where the endpoint constraint is the visit of a set of goal states. Backward probabilities are commonly used to compute reachability likelihoods (Amparore and Donatelli, 2013; Zapreev and Katoen, 2006). Approximate techniques for reachability, based on moment closure and stochastic approximation, have also been developed in (Bortolussi and Lanciani, 2014; Bortolussi, Lanciani, and Nenzi, 2018), but lack error guarantees.

HIDDEN MARKOV MODELS There is also a conceptual similarity between computing bridging probabilities and the forward-

backward algorithm for computing state-wise posterior marginals in hidden Markov models (HMMs) (Rabiner and Juang, 1986). Like MPMs, HMMs are a generative model that can be conditioned on observations. We only consider two observations (initial and terminal state) that are not necessarily noisy but the forward and backward probabilities admit the same meaning.

7.2 BACKWARDS PROBABILITIES

Let $x_g \in \mathcal{S}$ be a fixed goal state. Given the terminal constraint

$$\Pr(X_T = x_g) = 1 \text{ for some } T \geqslant 0\,,$$

we are interested in the so-called backward probabilities

$$\beta(x_i, t) = \Pr(X_T = x_g \mid X_t = x_i), \quad t \leqslant T\,. \tag{7.1}$$

Note that $\beta(\cdot, t)$ is a function of the conditional event and thus is no probability distribution over the state-space. Instead $\beta(\cdot, t)$ gives the reaching probabilities for all states over the time span of $[t, T]$. To compute these probabilities, we can employ the Kolmogorov backward equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\beta(t) = Q\beta(t)\,, \tag{7.2}$$

where we use the same vectorization to construct $\beta(t)$ as we used for $\pi(t)$. The above equation is integrated backwards in time and yields the reachability probability for each state x_i and time $t < T$ of ending up in x_g at time T. Similar to the CME (2.18), we can state a backward chemical master equation

$$\frac{\mathrm{d}\beta}{\mathrm{d}t}(x, t) = \sum_{j=1}^{n_R} \big(\beta(x, t) - \beta(x + \nu_j, t)\big)\, \alpha_j(x)\,. \tag{7.3}$$

The state-space of many MPMs, even simple ones, is countably infinite. In this case, we have to truncate the state-space to a *reasonable* finite subset. The choice of this truncation heavily depends on the goal of the analysis. If one is interested in the most

"common" behavior, for example, a dynamic mass-based truncation scheme is most appropriate (Mikeev and Sandmann, 2019). Such a scheme truncates states with small probability during the numerical integration. However, common mass-based truncation schemes are not as useful for the bridging problem. This is because trajectories that meet the specific terminal constraints can be far off the main bulk of the probability mass. We solve this problem by a state-space lumping in connection with an iterative refinement scheme.

7.3 BRIDGING DISTRIBUTION

The process' probability distribution given both initial and terminal constraints is formally described by the conditional probabilities

$$\gamma(x_i, t) = \Pr(X_t = x_i \mid X_0 = x_0, X_T = x_g)\,, \tag{7.4}$$

for $0 \leqslant t \leqslant T$. for fixed initial state x_0 and terminal state x_g. We call these probabilities the *bridging probabilities*. It is straightforward to see that γ admits the factorization

$$\gamma(x_i, t) = \pi(x_i, t)\beta(x_i, t)/\pi(x_g, T) \tag{7.5}$$

due to the Markov property. The normalization factor, given by the reachability probability

$$\pi(x_g, T) = \beta(x_0, 0)\,,$$

ensures that $\gamma(\cdot, t)$ is a distribution for all time points $t \in [0, T]$. We call each $\gamma(\cdot, t)$ a *bridging distribution*. From the Kolmogorov equations (2.17) and (7.2) we can obtain both the forward probabilities $\pi(\cdot, t)$ and the backward probabilities $\beta(\cdot, t)$ for $t < T$. ↪ page 13

We can easily extend this procedure to deal with hitting times constrained by a finite time-horizon by making the goal state x_g absorbing.

EXAMPLE In Figure 7.1 we plot the forward, backward, and bridging probabilities for Model 2. The probabilities are com- ↪ page 15

puted on a $[0, 100]$ state-space truncation. The approximate forward solution $\hat{\pi}$ shows how the probability mass drifts upwards towards the stationary distribution Poisson(100). The backward probabilities are highest for states below the goal state $x_g = 40$. This is expected because upwards drift makes reaching x_g more probable for "lower" states. Finally, the approximate bridging distribution $\hat{\gamma}$ can be recognized to be proportional to the product of forward $\hat{\pi}$ and backward probabilities $\hat{\beta}$. ◇

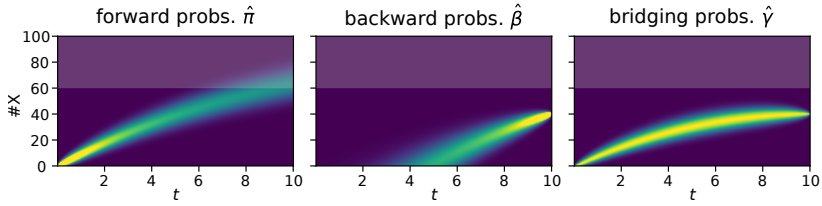


Figure 7.1: Forward, backward, and bridging probabilities for Model 2 with initial constraint $X_0 = 0$ and terminal constraint $X_{10} = 40$ on a truncated state-space. Probabilities over 0.1 in $\hat{\pi}$ and $\hat{\beta}$ are given full intensity for visual clarity. The lightly shaded area ($\geqslant 60$) indicates a region being more relevant for the forward than for the bridging probabilities.

7.4 BRIDGE TRUNCATION VIA LUMPING APPROXIMATION

We first discuss the truncation of countably infinite state-spaces to analyze backward and forward probabilities (Section 7.4.1). To identify effective truncations we employ a lumping scheme. Finally, in Section 7.4.2 we present an iterative refinement algorithm yielding a suitable truncation for the bridging problem.

7.4.1 *Finite State Projection*

Even in simple models such as a birth-death Process (Model 2), the reachable state-space is countably infinite. Direct analyzes of backward (7.1) and forward equations (2.16) are often infeasible. Instead, the integration of these differential equations requires working with a finite subset of the infinite state-space (Munsky

↪ *page 13*

and Khammash, 2006). If states are truncated, their incoming transitions from states that are not truncated can be re-directed to a *sink state*. The accumulated probability in this sink state is then used as an error estimate for the forward integration scheme. Consequently, many truncation schemes, such as dynamic truncations (Andreychenko et al., 2011), aim to minimize the amount of "lost mass" of the forward probability. We use the same truncation method but base the truncation on bridging probabilities rather than the forward probabilities.

7.4.2 *Iterative Refinement Algorithm*

The iterative refinement algorithm (Algorithm 4) starts with a set of large macro-states that are iteratively refined, based on approximate solutions to the bridging problem. We start by constructing square macro-states of size 2^m in each dimension for some $m \in \mathbb{N}$ such that they form a large-scale grid $\mathcal{S}^{(0)}$. Hence, each initial macro-state has a volume of $(2^m)^{n_s}$. This choice of grid size is convenient because we can halve states in each dimension. Moreover, this choice ensures that all states have equal volume and we end up with states of volume $2^0 = 1$ which is equivalent to a truncation of the original non-lumped state-space.

An iteration of the state-space refinement starts by computing both the forward and backward probabilities (line 2 and line 3) via integration of (2.17) and (7.2), respectively, using the lumped $\hat{Q}$-matrix. Based on the resulting approximate forward and backward probabilities, we compute an approximation of the bridging distributions (line 4). This is done for each time-point in an equispaced grid on $[0, T]$. The time grid granularity is a hyper-parameter of the algorithm. If the grid is too fine, the memory overhead of storing backward $\hat{\beta}^{(i)}$ and forward solutions $\hat{\pi}^{(i)}$ increases. We denote the approximations with a hat (e.g. $\hat{\pi}$) rather than a bar (e.g. $\bar{\pi}$) to indicate that not only the lumping approximation but also a truncation is applied and similarly for the Q-matrix. If, on the other hand, the granularity is too low, too much of the state-space might be truncated.

↪ page 13

Based on a threshold parameter $\delta > 0$ states are either removed or split (line 7), depending on the mass assigned to them by the approximate bridging probabilities $\hat{\gamma}_t^{(i)}$. A state can be split by the `split`-function which halves the state in each dimension. Otherwise, it is removed. Thus, each macro-state is either split into 2^{n_S} new states or removed entirely. The result forms the next lumped state-space $\mathcal{S}^{(i)}$. The Q-matrix is adjusted (line 10) such that transition rates for $\mathcal{S}^{(i)}$ are calculated according to (5.4). Entries of truncated states are removed from the transition matrix. Transitions leading to them are re-directed to a sink state (Section 7.4.1). After m iterations (we started with states of side lengths 2^m) we have a standard FSP scheme on the original model tailored to computing an approximation of the bridging distribution.

Algorithm 4: Iterative refinement for the bridging problem

input : Initial partitioning $\mathcal{S}^{(0)}$, truncation threshold δ
output: approximate bridging distribution $\hat{\gamma}$

1 **for** $i = 1, \dots, m$ **do**
2 $\hat{\pi}_t^{(i-1)} \leftarrow$ approximate forward equation on $\mathcal{S}^{(i)}$;
3 $\hat{\beta}_t^{(i-1)} \leftarrow$ approximate backward equation on $\mathcal{S}^{(i)}$;
4 $\hat{\gamma}_t^{(i)} \leftarrow \hat{\beta}^{(i)}\hat{\pi}^{(i)}/\hat{\pi}(x_g, T)$; `/* approx. bridging */`
5 $\mathcal{S}^{(i)} \leftarrow \emptyset$;
6 **foreach** $\bar{x} \in \mathcal{S}^{(i)}$ **do**
7 **if** $\exists t. \hat{\gamma}_t^{(i)}(\bar{x}) \geqslant \delta$; `/* refinement */`
8 **then**
9 $\mathcal{S}^{(i)} \leftarrow \mathcal{S}^{(i)} \cup$ `split`$(\bar{x})$;
10 update $\hat{Q}$-matrix;
11 **return** $\hat{\gamma}^{(i)}$;

In Figure 7.2 we give a demonstration of how Algorithm 4 works to refine the state-space iteratively. Starting with an initial lumped state-space $\mathcal{S}^{(0)}$ covering a large area of the state-space, repeated evaluations of the bridging distributions are performed. After five iterations the remaining truncation includes all states

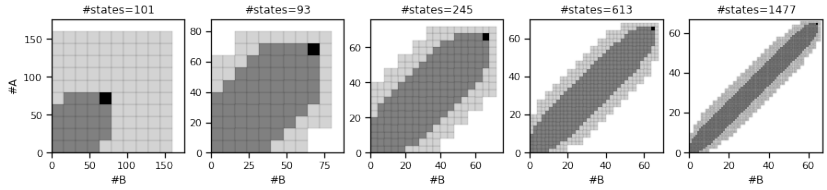


Figure 7.2: The state-space refinement algorithm on two parallel unit-rate arrival processes. The bridging problem from $(0, 0)$ to $(64, 64)$ and $T = 10$ and truncation threshold $\delta = 5 \times 10^{-3}$. States with a bridging probability below δ are light grey. The macro-state containing the goal state is marked in black. The initial macro-states are of size 16×16.

that significantly contribute to the bridging probabilities over the times $[0, T]$.

It is important to realize that determining the most relevant states is *the* main challenge. The above algorithm solves this problem by considering only those parts of the state-space that contribute most to the bridging probabilities. The truncation is tailored to this condition and might ignore regions that are likely in the unconditioned case. For instance, in Figure 7.1 the bridging probabilities mostly remain below a population threshold of $\#X = 60$ (as indicated by the lighter/darker coloring), while the forward probabilities mostly exceed this bound. Hence, in this example a significant portion of the forward probabilities $\hat{\pi}_t^{(i)}$ is captured by the sink state. However, the condition in line 7 of Algorithm 4 ensures that states contributing significantly to $\hat{\gamma}_t^{(i)}$ will be kept and refined in the next iteration.

7.5 RESULTS

We present four examples in this section to evaluate our proposed method. A prototype was implemented in Python 3.8. For numerical integration we used the Scipy implementation (Virtanen et al., 2020) of the implicit method based on backward-differentiation formulas (Byrne and Hindmarsh, 1975). The analysis as a Jupyter notebook is made available online (Backenköhler, 2020).

7.5.1 *Bounding Rare Event Probabilities*

We consider a simple model of two parallel Poisson processes describing the production of two types of agents. The corresponding probability distribution has Poisson product form at all time points $t \geqslant 0$ and hence we can compare the accuracy of our numerical results with the exact analytic solution. We use the proposed approach to compute lower bounds for rare event probabilities. These bounds are rigorous up to the approximation error of the numerical integration scheme. However, the forward solution could be replaced by an uniformization approach for a more rigorous error control.

↪ *page 10*

Model 10 (Parallel Poisson Processes)**.** *The model consists of two parallel independent Poisson processes with unit rates.*

$$\varnothing \xrightarrow{1} A \qquad \textit{and} \qquad \varnothing \xrightarrow{1} B\,.$$

The initial condition $X_0 = (0, 0)$ *holds with probability one. After* t *time units each species abundance is Poisson distributed with rate* $\lambda = t$*.*

We consider the final constraint of reaching a state where both processes exceed a threshold of 64 at time 20. Without prior knowledge, a reasonable truncation would have been 160×160. But our analysis shows that just 20% of the states are necessary to capture over 99.6% of the probability mass reaching the target event (cf. Table 7.1). Decreasing the threshold δ leads to a larger set of states retained after truncation as more of the bridging distribution is included (cf. Figure 7.3). We observe an increase in truncation size that is approximately logarithmic in δ, which, in this example, indicates robustness of the method with respect to the choice of δ.

COMPARISON TO OTHER METHODS The truncation approach that we apply is similar to the one used by Mikeev, Sandmann, and Wolf (2013) for rare event estimation. However, they used a given linearly biased MPM model to obtain a truncation. A general strategy to compute an appropriate biasing was not

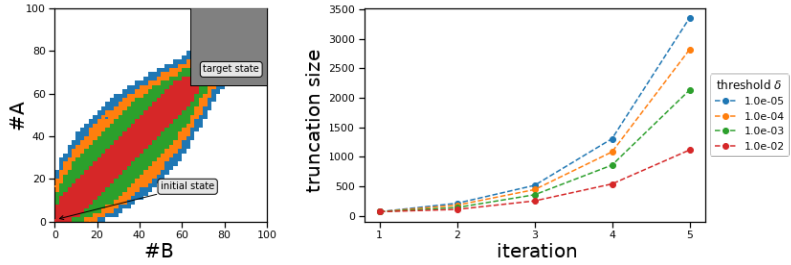


Figure 7.3: State-space truncation for varying values of the threshold parameter δ: Two parallel Poisson processes under terminal constraints $X_{20}^{(A)} \geqslant 64$ and $X_{20}^{(B)} \geqslant 64$. The initial macro-states are 16×16 such that the final states are regular micro states.

	threshold δ			
	1×10^{-2}	1×10^{-3}	1×10^{-4}	1×10^{-5}
$\lvert\mathcal{S}^{(m)}\rvert$	1154	2354	3170	3898
$\sum_m \lvert\mathcal{S}^{(m)}\rvert$	2074	3546	4586	5450
estimate	8.88×10^{-30}	1.85×10^{-29}	1.86×10^{-29}	1.86×10^{-29}
rel. error	5.22×10^{-1}	3.66×10^{-3}	3.74×10^{-5}	9.52×10^{-8}

Table 7.1: Estimated reachability probabilities based on varying truncation thresholds δ: The true probability is 1.8625×10^{-29}. We also report the size of the final truncation $\lvert\mathcal{S}^{(m)}\rvert$ and the accumulated size of all truncations during refinement iterations (overall states) $\sum_m \lvert\mathcal{S}^{(m)}\rvert$.

proposed. It is possible to adapt our truncation approach to the dynamic scheme in Mikeev, Sandmann, and Wolf (2013) where states are removed in an on-the-fly fashion during numerical integration.

A finite state-space truncation covering the same area as the initial lumping approximation would contain 25,600 states. The standard approach would be to build up the entire state-space for such a model (Kwiatkowska, Norman, and Parker, 2011). Even using a conservative truncation threshold $\delta = 1\times10^{-5}$, our method

The goal is not treated as a single state. Otherwise, it consisted of 24,130 *states.*

yields an accurate estimate using only about a fifth (5450) of this accumulated over all intermediate lumped approximations.

7.5.2 *Mode Switching*

This often is an instance of rare-event analysis.

Mode switching occurs in models exhibiting *multi-modal* behavior (see Section 2.7 for details) when a trajectory traverses a potential barrier from one mode to another. Often, mode switching is a rare event and occurs in the context of gene regulatory networks where a mode is characterized by the set of genes being currently active (Loinger et al., 2007). Similar dynamics also commonly occur in queuing models where a system may for example switch its operating behavior stochastically if traffic increases above or decreases below certain thresholds. Using the presented method, we can get both a qualitative and quantitative understanding of switching behavior without resorting to Monte-Carlo methods such as importance sampling.

7.5.2.1 *Exclusive Switch*

The exclusive switch (Barzel and Biham, 2008) has three different modes of operation, depending on the deoxyribonucleic acid (DNA) state, i.e. on whether a protein of type one or two is bound to the DNA.

Model 11 (Exclusive Switch)**.** *The exclusive switch model consists of a promoter region that can express both proteins* P_1 *and* P_2*. Both can bind to the region, suppressing the expression of the other protein. For certain parameterizations, this leads to a bi-modal or even tri-modal behavior.*

$$D \xrightarrow{\rho} D + P_1 \qquad D \xrightarrow{\rho} D + P_2 \qquad P_1 \xrightarrow{\lambda} \varnothing \qquad P_2 \xrightarrow{\lambda} \varnothing$$

$$D + P_1 \xrightarrow{\beta} D.P_1 \qquad D.P1 \xrightarrow{\gamma} D + P_1 \qquad D.P_1 \xrightarrow{\alpha} D.P_1 + P_1$$

$$D + P_2 \xrightarrow{\beta} D.P_2 \qquad D.P2 \xrightarrow{\gamma} D + P_2 \qquad D.P_2 \xrightarrow{\alpha} D.P_2 + P_2$$

The parameter values are $\rho = 1\times10^{-1}$, $\lambda = 1\times10^{-3}$, $\beta = 1\times10^{-2}$, $\gamma = 8\times10^{-3}$, *and* $\alpha = 1\times10^{-1}$.

Since we know a priori of the three distinct operating modes, we adjust the method slightly: The state-space for the DNA states is not lumped. Instead we "stack" lumped approximations of the P_1-P_2 phase space upon each other. Special treatment of DNA states is common for such models (Lapin, Mikeev, and Wolf, 2011).

To analyze the switching, we choose the transition from

Variable order: P_1, P_2, D, D.P_1, D.P_2.

$$x_1 = (32, 0, 0, 0, 1)$$

to

$$x_2 = (0, 32, 0, 1, 0)$$

over the time interval $t \in [0, 10]$. The initial lumping scheme covers up to 80 molecules of P_1 and P_2 for each mode. Macro-states have size 8×8 and the truncation threshold is $\delta = 1 \times 10^{-4}$.

In the analysis of biological switches, not only the switching probability but also the switching dynamics is a central part of understanding the underlying biological mechanisms. In Figure 7.4, we therefore plot the time-varying probabilities of the gene state conditioned on the mode. We observe a rapid unbinding of P_2, followed by a slow increase of the binding probability for P_1. These dynamics are already qualitatively captured by the first lumped approximation (dashed lines).

7.5.2.2 *Toggle Switch*

Next, we apply our method to a toggle switch model exhibiting non-polynomial rate functions. This well-known model considers two proteins A and B inhibiting the production of the respective other protein (Lipshtat et al., 2006).

Model 12 (Toggle Switch using Hill functions)**.** *We have population types* A *and* B *with the following reactions and reaction rates.*

$$\varnothing \xrightarrow{\alpha_1(\cdot)} A\,, \quad \textit{where} \quad \alpha_1(x) = \frac{\rho}{1 + x_B}, \qquad A \xrightarrow{\lambda} \varnothing$$

$$\varnothing \xrightarrow{\alpha_1(\cdot)} B\,, \quad \textit{where} \quad \alpha_1(x) = \frac{\rho}{1 + x_A}, \qquad B \xrightarrow{\lambda} \varnothing$$

The parameterization is $\rho = 10$, $\lambda = 0.1$.

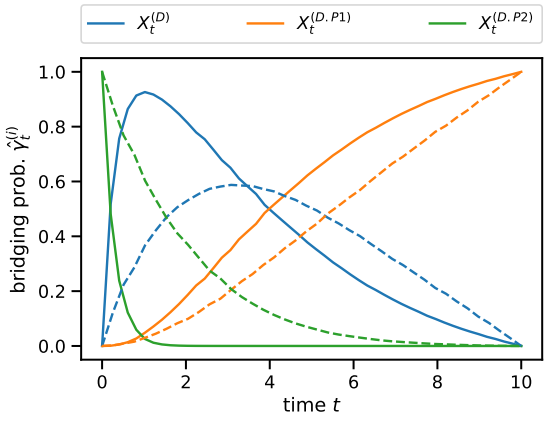


Figure 7.4: Mode probabilities of the exclusive switch bridging problem over time for the first lumped approximation (dashed lines) and the final approximation (solid lines) with constraints $X_0 = (32, 0, 0, 1, 0)$ and $X_{10} = (0, 32, 0, 0, 1)$.

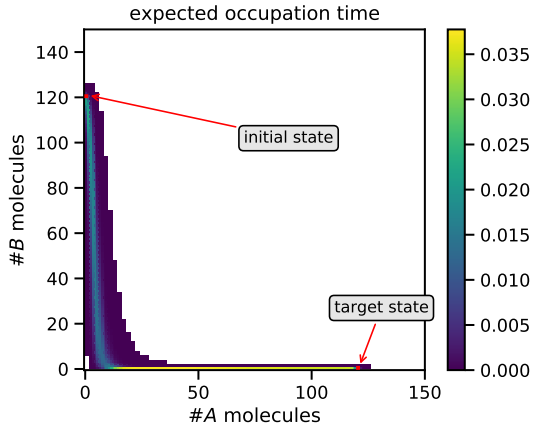


Figure 7.5: The expected occupation time (excluding initial and terminal states) for the switching problem of the toggle switch using Hill-type functions. The bridging problem is from initial $(0, 120)$ to a first passage of $(120, 0)$ in $t \in [0, 10]$.

Due to the non-polynomial rate functions α_1 and α_2, the transition rates between macro-states are approximated by using the continuous integral

$$\bar{\alpha}_1(\bar{x}) \approx \int_{a-0.5}^{b+0.5} \frac{\rho}{1+x}\,dx = \rho\left(\log(b+1.5) - \log(a+0.5)\right)$$

for a macro-state $\bar{x} = \{a, \ldots, b\}$.

We analyze the switching scenario from $(0, 120)$ to the first visit of state $(120, 0)$ up to time $T = 10$. The initial lumping scheme covers up to 352 molecules of A and B and macro-states have size 32×32. The truncation threshold is $\delta = 1 \times 10^{-4}$. The resulting truncation is shown in Figure 7.5. It also illustrates the kind of insights that can be obtained from the bridging distributions. For an overview of the switching dynamics, we look at the expected occupation time under the terminal constraint of having entered state $(120, 0)$. Letting the corresponding hitting time be

$$\tau = \inf\{t \geqslant 0 \mid X_t = (120, 0)\},$$

the expected occupation time for some state x is

$$\mathrm{E}\left(\int_0^\tau 1_{=x}(X_t)\,dt \mid \tau \leqslant 10\right).$$

We observe that in this example the switching behavior seems to be asymmetrical. The main mass seems to pass through an area where initially a small number of A molecules is produced followed by a total decay of B molecules.

7.5.3 *Recursive Bayesian Estimation*

We now turn to the method's application in recursive Bayesian estimation. This is the problem of estimating the system's past, present, and future behavior under given observations. Thus, the MPM becomes a hidden Markov model (HMM). The observations in such models are usually noisy, meaning that we cannot infer the system state with certainty.

This estimation problem entails more general distributional constraints on terminal $\beta(\cdot, T)$ and initial $\pi(\cdot, 0)$ distributions than

the point mass distributions considered up until now. We can easily extend the forward and backward probabilities to more general initial distributions and terminal distributions $\beta(T)$. For the forward probabilities we get

$$\pi(x_i, t) = \sum_j \Pr(X_t = x_i \mid X_0 = x_j)\pi(x_j, 0), \tag{7.6}$$

and similarly the backward probabilities are given by

$$\beta(x_i, t) = \sum_j \Pr(X_T = x_j \mid X_t = x_i)\beta_T(x_j)\,. \tag{7.7}$$

We apply our method to an susceptible-exposed-infected-removed (SEIR) model. This is widely used to describe the spreading of an epidemic such as the current COVID-19 outbreak (Großmann, Backenköhler, and Wolf, 2020; He, Peng, and Sun, 2020). Temporal snapshots of the epidemic spread are mostly only available for a subset of the population and suffer from inaccuracies of diagnostic tests. Bayesian estimation can then be used to infer the spreading dynamics given uncertain temporal snapshots.

Model 13 (Epidemics Model). *A population of susceptible individuals can contract a disease from infected agents. In this case, they are exposed, meaning they will become infected but cannot yet infect others. After being infected, individuals change to the removed state. The mass-action reactions are as follows.*

$$S + I \xrightarrow{\lambda} E + I \qquad E \xrightarrow{\mu} I \qquad I \xrightarrow{\rho} R$$

The parameter values are $\lambda = 0.5$, $\mu = 3$, $\rho = 3$. *Due to the stoichiometric invariant* $X_t^{(S)} + X_t^{(E)} + X_t^{(I)} + X_t^{(R)} = \text{const.}$, *we can eliminate* R *from the system.*

We consider the following scenario: There are N individuals. We know that initially ($t = 0$) one individual is infected and the rest is susceptible. At time $t = 0.3$ all individuals are tested for the disease. The test, however, only identifies infected individuals with probability $p_{tp} = 0.99$. Moreover, the probability of a false positive is $p_{fp} = 0.05$. The random variable Y_t models the

The false positive probability is the same for all non-infected invididuals.

measurement likelihood at time t. Based on the description above

$$\Pr(Y_t = \hat{n}_I \mid X_t^{(I)} = n_I) = \sum_{k=0}^{\hat{n}_I} B(k; n_I, p_{tp}) B(\hat{n}_I - k; N - n_I, p_{fp})$$

where B is the binomial probability mass function and

$$B(k; n, p) = \binom{n}{k} p^k (1-p)^{n-k}.$$

We like to identify the distribution given both the initial state and the measurement at time $t = 0.3$. In particular, we want to infer the distribution over the latent counts of S and E by *recursive Bayesian estimation*.

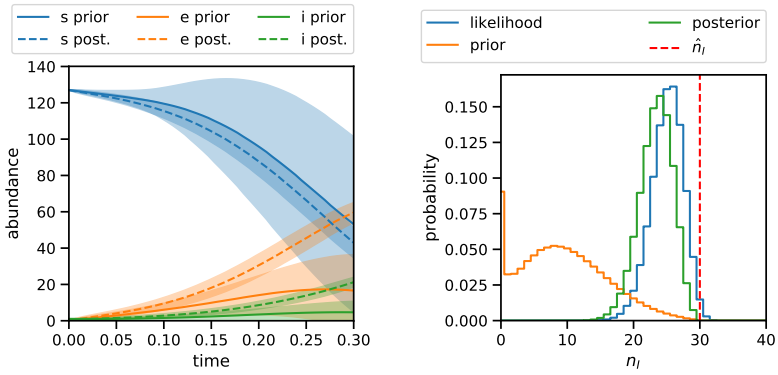
The posterior for n_I infected individuals at time t, given measurement $Y_t = \hat{n}_I$ can be computed using Bayes' rule

$$\Pr(X_t^{(I)} = n_I \mid Y_t = \hat{n}_I, X_0 = x_0) \propto \Pr(Y_t = \hat{n}_I \mid X_t^{(I)} = n_I) \Pr(X_t^{(I)} = n_I \mid X_0 = x_0). \quad (7.8)$$

This problem is an extension of the bridging problem discussed up until now. The difference is that the terminal posterior is estimated it using the result of the lumped forward equation and the measurement distribution using (7.8). Based on this estimated terminal posterior, we compute the bridging probabilities and refine the truncation tailored to the location of the posterior distribution. In Figure 7.6a, we illustrate the bridging distribution between the terminal posterior and initial distribution. In the context of filtering problems this is commonly referred to as smoothing. Using the learned truncation, we can obtain the posterior distribution for the number of infected individuals at $t = 0.3$ (Figure 7.6b). Moreover, can we infer a distribution over the unknown number of susceptible and exposed individuals (Figure 7.7).

7.6 CONCLUSION

The analysis of MPMs with constraints on the initial and terminal behavior is an important part of many probabilistic inference



(a) Prior and posterior dynamics (b) Beliefs and likelihood at $t = 0.3$

Figure 7.6: (a) A comparison of the prior dynamics and the posterior smoothing (bridging) dynamics. (b) The prior, likelihood, and posterior of the number of infected individuals n_I at time $t = 0.3$ given the measurement $\hat{n}_I = 30$.

tasks such as parameter estimation using Bayesian or maximum likelihood estimation, inference of latent system behavior, the estimation of rare event probabilities, and reachability analysis for the verification of temporal properties. If endpoint constraints correspond to atypical system behaviors, standard analysis methods fail as they have no strategy to identify those parts of the state-space relevant for meeting the terminal constraint.

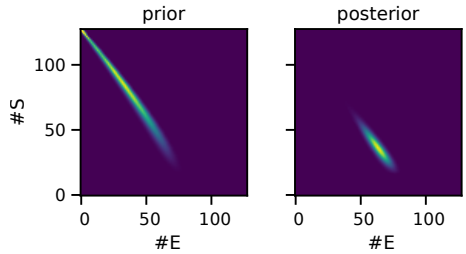


Figure 7.7: The prior and posterior distribution over the latent types E and S at time $t = 0.3$.

Here, we proposed a method that is not based on stochastic sampling and statistical estimation but provides a direct numerical approach. It starts with an abstract lumped model, which is iteratively refined such that only those parts of the model are considered that contribute to the probabilities of interest. In the final step of the iteration, we operate at the granularity of the original model and compute lower bounds for these bridging probabilities that are rigorous up to the error of the numerical integration scheme.

Our method exploits the population structure of the model, which is present in many important application fields of MPMs. Based on experience with other work based on truncation, the approach can be expected to scale up to at least a few million states (Mikeev, Sandmann, and Wolf, 2011). Compared to previous work, our method neither relies on approximations of unknown accuracy nor additional information such as a suitable change of measure in the case of importance sampling. It only requires a truncation threshold and an initial choice for the macro-state sizes.

In future work, we plan to extend our method to hybrid approaches, in which a moment representation is employed for large populations while discrete counts are maintained for small populations. Moreover, we will apply our method to model checking where constraints are described by some temporal logic (Hajnal et al., 2019).

Part IV

FOSTER-LYAPUNOV FUNCTIONS

Existing Foster-Lyapunov functions often give sets that are far larger necessary for a given probability bound. Here, we explore the possibility to improve upon such a function by *local* alterations. This leaves the global properties – and thereby its guarantees – intact, while we obtain much tighter sets for the same probability bounds.

AUGMENTED FOSTER-LYAPUNOV BOUNDS

8

In many cases, researchers are interested in the *long-run* behavior of MPMs. This behavior is characterized by the stationary distribution of the underlying CTMC. Rigorous analyses of the stationary distribution either hinge on bounds on the moments of this distribution or a Foster-Lyapunov analysis (Kuntz et al., 2021b). Using the former, i.e. moment bounds, entails solving a semi-definite program involving linear constraints dictated by the moment dynamics. These moment problems are prone to numerical problems because moments increase by several orders of magnitudes with their orders. Since – in most cases – constraints up to high moment orders are necessary, solving such convex problems is often difficult in practice. This problem becomes particularly problematic, if some species tend to have high abundances at stationarity. Reasoning over the drift is often less plagued by numerical issues than the moment bounding approach. The use of Foster-Lyapunov approaches, is often two-fold: They are used to prove ergodicity and to provide sets with a lower bound $1 - \epsilon$ on their stationary probability mass. Often practical methods focus on proving *global* properties of their associated drift (Gupta, Briat, and Khammash, 2014; Spieler, 2014). In many models, simple affine functions (Gupta, Mikelson, and Khammash, 2017) and even the identity (Spieler, 2014) is sufficient. While such simple forms provide some of the necessary ease of a global analysis, the task of optimizing them with performance in mind, is much more difficult (Milias-Argeitis and Khammash, 2014).

↪ *page 18*

Performance *in this context means least states to cover most stationary probability mass.*

Ideally one would have the best of both worlds: The ease of working with simple forms for the global guarantees of the Foster-Lyapunov criteria and the freedom of choice to fit efficient functions and get sets that are as small as possible. The approach presented in this chapter achieves this by *locally* altering a proposal Foster-Lyapunov function. This proposal function is essentially assumed as an input for this method. Such a function can be identified by convex analysis, for example (Gupta, Briat, and Khammash, 2014). Using a set of probability $> 1 - \epsilon$

is identified using such a functional. On this set, the original function is then replaced by any other lower bounded function. At the set boundary this supplementary function is *phased out* and we switch back to the original function using some smooth step function. The choice of such a supplementary function offers much room for experimentation since all the necessary global criteria. For this function we can, for example, use any polynomial or even more variable models such as neural networks.

The procedure to identify an efficient function is thereby reduced to a simple machine learning problem. The objective exponentially rewards negative drift yielding tight sets. Its computation needs, in principle, all points in a sufficiently large set. For optimization however, we switch to an approach of sampling uniformly from the augmentation set.

This method yields sets that are smaller on the order of 10^3 for a fixed probability bound over a simple linear functional. We study neural networks and polynomial templates as candidates for local augmentation.

8.1 THE DRIFT AND ITS PROPERTIES

We explain the interpretation and give some basic use of the drift in Section 2.6.1 on page 22.

The drift (2.26) plays a central role in this chapter.

$$d(x; g) = Qg(x) = \sum_{j=1}^{n_R} \alpha_j(x)(g(x + \nu_j) - g(x)) \tag{8.1}$$

An interesting fact about the drift is, that it is invariant to linear transforms to g. That is

$$Q(g + b)(x) = Qg(x) \tag{8.2}$$

for some constant $b \in \mathbb{R}$. Clearly a positive linear factor m to g factors out, i.e. $Q(f \circ g)(x) = mQg(x)$ for $f(x) = mx$, $m > 0$. Consequently, if the drift is scaled by its maximum value, the scaled version is invariant to linear transforms of g:

$$\frac{Q(f \circ g)(x)}{\max_{x \in \mathcal{S}} Q(f \circ g)(x)} = \frac{Qg(x)}{\max_{x \in \mathcal{S}} Qg(x)} \tag{8.3}$$

Since the probability bounded sets C_{ϵ_ℓ} depend on the scaled drift. Therefore invariance under linear transformation implies that we cannot change – especially improve – the tightness of the sets by such a transform.

EXAMPLE What makes the perfect Foster-Lyapunov function? The confidence interval for level $1-\alpha$ of a Poisson with rate μ is

The beauty standard may vary for other applications.

$$\frac{1}{2}P^{-1}(\alpha/2;2k) \leqslant \mu \leqslant \frac{1}{2}P^{-1}(1-\alpha/2;2k+2)\,.$$

where $P^{-1}(\,\cdot\,;l)$ is the inverse cumulative density function (CDF) of a χ^2_l distribution. Thus, its the inverse of the regularized gamma function

$$\frac{1}{\Gamma(k/2)}\int_0^{x/2} t^{k/2-1}e^{-t}\,dt$$

w.r.t. x. This gives us the "ideal" intervals $[l_\epsilon, h_\epsilon]$, $(l_\epsilon, h_\epsilon) \in \mathbb{N}^2$ such that $\pi_\infty([l_\epsilon, h_\epsilon]) = 1-\epsilon$. These sets are given as the dark area in Figure 8.2. Therefore, the perfect Lyapunov function would coincide with those sets. That is, a function g such that

$$C_\epsilon = [l_\epsilon, h_\epsilon]\,, \quad \forall \epsilon > 0\,.$$

◇

8.2 AUGMENTATION VIA LOCAL SUBSTITUTION

We use a proved Foster-Lyapunov function as a starting point. For many relevant reaction networks, simple choices such as L1 or L2 norms are sufficient choices (Spieler, 2014). The resulting sets, however, are typically very large. Tasks such as computing approximate stationary distributions on truncations set up according to these sets can be very costly. This cost is exacerbated when a system has to be solved for a lot of different reentry matrices, which is necessary when state-wise bounds on the probability conditioned on a truncation are desired (Dayar et al., 2011).

We propose to augment the proposal function by a function, that is limited to local influence guided by the initial set. This supplementary function is phased out asymptotically using a simple sigmoid threshold function

$$\gamma_{k,z}(x) = \frac{1}{1 + k\exp(-x - y)}\,. \tag{8.4}$$

Thus, in a one-dimensional model the augmented Lyapunov g' function becomes

$$g'(x) = \gamma_{k,z}(x)g(x) + (1 - \gamma_{k,z}(x))g^*(x)\,. \tag{8.5}$$

The threshold function γ guarantees that g^* vanishes asymptotically. The drifts d' and d^* are defined accordingly.

EXAMPLE Consider the example of Model 2. Using a simple L2 norm as an initially, i.e. $g(x) = x^2$, we obtain a fairly large set. In Figure 8.1, we contrast this result to the solution given by the choice of $g^*(x) = (x - 250)^2$, which gives a much tighter subset with the same guarantees. In this case, the guarantee is that the sets contain at least 0.9 stationary probability mass. We further demonstrate how the incorporation g^* into g significantly tightens the set proposed initially. ◇

The benefit of the threshold-based construction (8.5) is that we only require g^* to be non-negative. All other properties are inherited from the proposal function. This freedom enables the use of flexible machine learning models to search an efficient g^*.

8.3 POLYNOMIAL AUGMENTATION

8.4 NEURAL AUGMENTATION

The characteristics of the augmentation function g^* are typically not known beforehand. The formulation of augmented Foster-Lyapunov functions only places basic constraints on the function used: The function needs to be non-negative and an upper bound of the drift has to be known. Neural networks lend themselves naturally as an extremely flexible functional family.

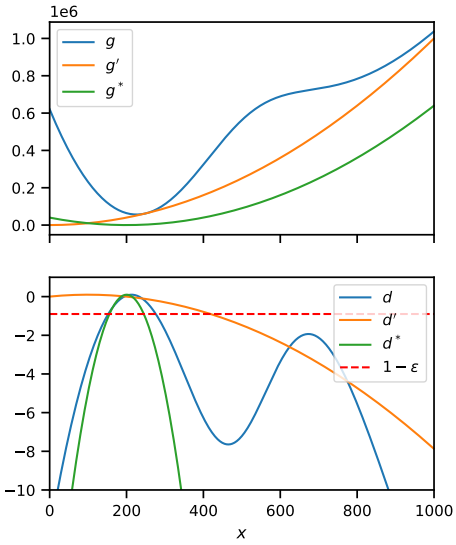


Figure 8.1: Different example Lyapunov functions for Model 2. The drifts d, d^*, and d' are scaled and the appropriate threshold for $\epsilon = 0.1$ is given.

The central piece of fitting g^* is an objective function. Since the actual sets, bounded in probability, are defined in terms of their drift, this objective needs to be a function of this drift. A desirable augmented drift has tight level sets with more emphasis placed on its peak regions. A natural way to express this prioritization is the objective

$$\sum_{x \in \mathcal{S}} \int_{-\infty}^{d^*(x)} \exp(y)\, dy = \sum_{x \in \mathcal{S}} \exp(d'(x))$$

based on the scaled augmented drift.

Model 14 (Competitive Spread)**.** *Two uncoupled parallel birth-death processes (cf. Model 7) supplemented with two Gray-Scott type reactions.* ↪ page 96

$$\varnothing \xrightarrow{\rho} A \qquad A \xrightarrow{\delta} \varnothing \qquad \varnothing \xrightarrow{\rho} B \qquad B \xrightarrow{\delta} \varnothing$$

$$2A + B \xrightarrow{\nu_1} 3A \qquad 2A + A \xrightarrow{\nu_2} 3B$$

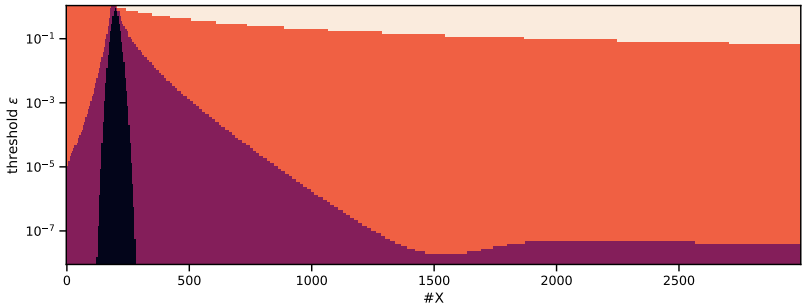
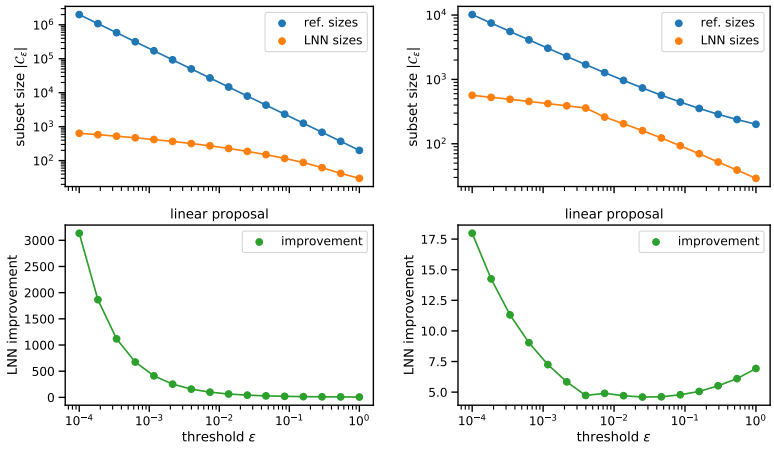


Figure 8.2: Lyapunov sets for the birth-death process for different probability thresholds ϵ for the augmented function (red) and the proposal (orange). The "perfect" sets are computed using the confidence interval (dark).

As a parameterization we choose $\rho = 5$, $\delta = 0.1$, $\nu_1 = 6\times10^{-5}$, *and* $\nu_2 = 6.2\times10^{-5}$.



(a) Linear proposal (b) Quadratic proposal

Figure 8.3

Part V

CONCLUDING REMARKS

CONCLUSION

9

Part VI

APPENDIX

ADDITIONAL RESULTS & DATA

A.1 DETAILED RESULTS FOR LINEAR CONTROL VARIATES

n_{max}	n_λ	ϕ	$1 - \frac{\sigma_1^2}{\sigma_0^2}$	slowdown	efficiency	$\lvert P \rvert$
1	10	ϕ_{sq}	0.807184	1.227471	4.239255	2.536479
		ϕ_c	0.880285	1.633530	5.135205	7.411732
		ϕ_q	0.849082	1.312416	5.067770	3.639250
		ϕ_ℓ	0.783459	1.195821	3.874778	2.090101
	20	ϕ_{sq}	0.856593	1.263340	5.539683	2.206154
		ϕ_c	0.910480	1.864405	6.011256	9.441336
		ϕ_q	0.867987	1.317958	5.765884	3.140806
		ϕ_ℓ	0.825518	1.243075	4.627662	1.981143
	30	ϕ_{sq}	0.869165	1.298893	5.905196	2.059415
		ϕ_c	0.921019	1.966191	6.461331	9.928998
		ϕ_q	0.876822	1.340409	6.079876	2.762449
		ϕ_ℓ	0.843288	1.288925	4.968796	1.983174
2	10	ϕ_{sq}	0.811956	1.505521	3.544783	2.323999
		ϕ_c	0.916866	4.507566	2.681363	21.692390
		ϕ_q	0.868874	1.776190	4.309354	4.739893
		ϕ_ℓ	0.795802	1.466579	3.353046	2.016196
	20	ϕ_{sq}	0.832562	1.657484	3.617313	2.085711
		ϕ_c	0.934280	6.348223	2.406431	29.976320
		ϕ_q	0.878944	1.879341	4.416281	3.990881
		ϕ_ℓ	0.837922	1.647329	3.759896	1.978017
	30	ϕ_{sq}	0.829427	1.844766	3.190308	2.043201
		ϕ_c	0.947324	7.130628	2.673225	32.513670
		ϕ_q	0.878830	2.053317	4.034987	3.611746
		ϕ_ℓ	0.824936	1.838879	3.118728	1.978836

Table A.1: Variance reduction results for up to second order moments with parameters $n_{max} = 2$, $n = 10{,}000$, $d = 100$, $k_{min} = 3$. Exclusive switch.

n_{max}	n_λ	ϕ	$1 - \frac{\sigma_1^2}{\sigma_0^2}$	slowdown	efficiency	$\lvert P \rvert$
1	10	ϕ_{sq}	0.619560	1.488483	1.770218	3.232575
		ϕ_c	0.700255	2.241695	1.492171	8.008607
		ϕ_q	0.643550	1.613001	1.743500	3.817641
		ϕ_ℓ	0.596650	1.459405	1.703170	2.657000
	20	ϕ_{sq}	0.697414	1.519425	2.181687	2.631677
		ϕ_c	0.713445	2.706546	1.292838	10.295856
		ϕ_q	0.697654	1.585313	2.092817	3.398235
		ϕ_ℓ	0.695846	1.473976	2.235418	2.226530
	30	ϕ_{sq}	0.712941	1.543068	2.263644	2.378037
		ϕ_c	0.721354	2.874249	1.252541	10.910880
		ϕ_q	0.711877	1.607712	2.164485	2.979704
		ϕ_ℓ	0.669963	1.522184	1.996300	2.085473
2	10	ϕ_{sq}	0.619450	1.734737	1.519168	3.148184
		ϕ_c	0.665361	3.301159	0.909443	13.456259
		ϕ_q	0.680592	1.840457	1.705876	3.864240
		ϕ_ℓ	0.612674	1.662962	1.556868	2.659592
	20	ϕ_{sq}	0.684789	1.811408	1.755652	2.687379
		ϕ_c	0.689835	4.455005	0.726640	17.609554
		ϕ_q	0.687665	1.901258	1.688449	3.413595
		ϕ_ℓ	0.651262	1.770238	1.623924	2.266729
	30	ϕ_{sq}	0.690602	1.922217	1.686011	2.375455
		ϕ_c	0.649191	4.837419	0.591701	19.145054
		ϕ_q	0.701253	2.001179	1.677062	3.007525
		ϕ_ℓ	0.639123	1.894074	1.467403	2.086275

Table A.2: Variance reduction results for up to second order moments with parameters $n_{max} = 2$, $n = 10{,}000$, $d = 100$, $k_{min} = 3$. Distributive modification.

n_{max}	n_λ	ϕ	$1 - \frac{\sigma_1^2}{\sigma_0^2}$	slowdown	efficiency	$\lvert P \rvert$
1	10	ϕ_{sq}	0.881641	1.530137	5.558692	1.621917
		ϕ_c	0.965224	1.945588	14.859417	3.338501
		ϕ_q	0.916445	1.625232	7.409904	1.997045
		ϕ_ℓ	0.868288	1.380344	5.529745	1.081152
	20	ϕ_{sq}	0.941153	1.637272	10.437978	1.842971
		ϕ_c	0.964204	1.907999	14.747328	2.915082
		ϕ_q	0.947984	1.747519	11.072422	2.227250
		ϕ_ℓ	0.931030	1.433401	10.169570	1.088572
	30	ϕ_{sq}	0.959517	1.723449	14.404936	1.972426
		ϕ_c	0.962514	1.770936	15.142156	2.216103
		ϕ_q	0.966216	1.847441	16.117387	2.446661
		ϕ_ℓ	0.945724	1.456432	12.710196	1.084188
2	10	ϕ_{sq}	0.905254	1.659799	6.402491	2.388472
		ϕ_c	0.987526	2.474939	33.074955	6.501180
		ϕ_q	0.923063	1.822654	7.195544	3.179257
		ϕ_ℓ	0.878232	1.415909	5.830248	1.092264
	20	ϕ_{sq}	0.949038	1.831995	10.797164	2.890898
		ϕ_c	0.985710	2.391457	29.704344	5.450299
		ϕ_q	0.968076	2.021487	15.662368	3.681229
		ϕ_ℓ	0.925413	1.449386	9.298961	1.072761
	30	ϕ_{sq}	0.964855	1.924268	14.911787	3.026275
		ϕ_c	0.981507	2.144089	25.520987	4.179125
		ϕ_q	0.973902	2.095985	18.507746	3.685851
		ϕ_ℓ	0.948349	1.507425	12.904707	1.074538

Table A.3: Variance reduction results for up to second order moments with parameters $n_{max} = 2$, $n = 10{,}000$, $d = 100$, $k_{min} = 3$. Dimerization.

ϵ		iteration i							
		1	2	3	4	5	6	7	8
1×10^{-1}	$\lvert S^{(i)}\rvert$	4900	28	52	112	232	472	960	1932
	tot. error	1.91	1.84	1.73	1.55	1.29	9.35×10^{-1}	4.88×10^{-1}	3.54×10^{-2}
	max. error	3.15×10^{-3}	3.13×10^{-3}	3.08×10^{-3}	2.98×10^{-3}	2.77×10^{-3}	2.38×10^{-3}	1.57×10^{-3}	6.04×10^{-5}
1×10^{-2}	$\lvert S^{(i)}\rvert$	4900	52	104	208	464	988	2008	4052
	tot. error	1.91	1.84	1.73	1.56	1.30	9.46×10^{-1}	5.01×10^{-1}	6.22×10^{-4}
	max. error	3.15×10^{-3}	3.13×10^{-3}	3.08×10^{-3}	2.98×10^{-3}	2.78×10^{-3}	2.39×10^{-3}	1.59×10^{-3}	8.33×10^{-7}
1×10^{-3}	$\lvert S^{(i)}\rvert$	4900	84	152	300	652	1440	2996	6068
	tot. error	1.91	1.83	1.73	1.56	1.30	9.46×10^{-1}	5.01×10^{-1}	9.83×10^{-6}
	max. error	3.15×10^{-3}	3.13×10^{-3}	3.08×10^{-3}	2.98×10^{-3}	2.78×10^{-3}	2.39×10^{-3}	1.59×10^{-3}	1.14×10^{-8}
1×10^{-4}	$\lvert S^{(i)}\rvert$	4900	116	212	400	848	1872	3960	8060
	tot. error	1.91	1.83	1.73	1.56	1.30	9.46×10^{-1}	5.01×10^{-1}	9.83×10^{-6}
	max. error	3.15×10^{-3}	3.13×10^{-3}	3.08×10^{-3}	2.98×10^{-3}	2.78×10^{-3}	2.39×10^{-3}	1.59×10^{-3}	1.83×10^{-10}

Table A.4: Detailed results for Model 7. The errors are computed wrt. the reference Poissonian product. The total absolute error and the maximum absolute errors are given.

ϵ		iteration i							
		1	2	3	4	5	6	7	8
1×10^{-1}	$\lvert S^{(i)}\rvert$	11,907	20	32	60	140	340	840	2116
	tot. error $\leqslant$	1.86×10^{0}	1.85×10^{0}	1.45×10^{0}	1.18×10^{0}	9.31×10^{-1}	6.41×10^{-1}	4.67×10^{-1}	4.89×10^{-1}
	max. error $\leqslant$	1.63×10^{-3}	1.63e-3	1.55×10^{-3}	1.40×10^{-3}	1.22×10^{-3}	9.36×10^{-4}	8.40×10^{-4}	1.40×10^{-3}
1×10^{-2}	$\lvert S^{(i)}\rvert$	11,907	48	112	148	300	720	1892	5156
	tot. error $\leqslant$	1.86×10^{0}	1.84×10^{0}	1.44×10^{0}	1.21×10^{0}	9.56×10^{-1}	6.65×10^{-1}	3.41×10^{-1}	3.31×10^{-2}
	max. error $\leqslant$	1.63×10^{-3}	1.62×10^{-3}	1.53×10^{-3}	1.39×10^{-3}	1.20×10^{-3}	9.59×10^{-4}	5.86×10^{-4}	5.37×10^{-5}
1×10^{-3}	$\lvert S^{(i)}\rvert$	11,907	84	192	244	488	1084	2692	7152
	tot. error $\leqslant$	1.86×10^{0}	1.83×10^{0}	1.46×10^{0}	1.22×10^{0}	9.63×10^{-1}	6.67×10^{-1}	3.37×10^{-1}	8.01×10^{-4}
	max. error $\leqslant$	1.63×10^{-3}	2.95×10^{-2}	1.54×10^{-3}	1.39×10^{-3}	1.20×10^{-3}	9.51×10^{-4}	5.79×10^{-4}	1.09×10^{-6}
1×10^{-4}	$\lvert S^{(i)}\rvert$	11,907	124	324	352	672	1436	3408	8864
	tot. error $\leqslant$	1.86×10^{0}	1.83×10^{0}	1.46×10^{0}	1.22×10^{0}	9.63×10^{-1}	6.67×10^{-1}	3.37×10^{-1}	1.12×10^{-5}
	max. error $\leqslant$	1.63×10^{-3}	3.19×10^{-2}	1.54×10^{-3}	1.39×10^{-3}	1.20×10^{-3}	9.51×10^{-4}	5.79×10^{-4}	1.28×10^{-8}

Table A.5: Detailed results for Model 8. Upper bounds on the total absolute error and the maximum absolute error are given. The worst-case errors are computed wrt. the reference Geobound solution with $\epsilon_\ell = 1\times10^{-2}$.

Adan, Aysun, Günel Alizada, Yağmur Kiraz, Yusuf Baran, and Ayten Nalbant (2017). "Flow cytometry: basic principles and applications." In: *Critical reviews in biotechnology* 37.2, pp. 163–176.

Ale, Angelique, Paul Kirk, and Michael PH Stumpf (2013). "A general moment expansion method for stochastic kinetic models." In: *The Journal of chemical physics* 138.17, p. 174101.

Amparore, Elvio Gilberto and Susanna Donatelli (2013). "Backward Solution of Markov Chains and Markov Regenerative Processes: Formalization and Applications." In: *Electron. Notes Theor. Comput. Sci.* 296, pp. 7–26.

Anderson, David F, Gheorghe Craciun, and Thomas G Kurtz (2010). "Product-form stationary distributions for deficiency zero chemical reaction networks." In: *Bulletin of mathematical biology* 72.8, pp. 1947–1970.

Anderson, David F and Thomas G Kurtz (2011). "Continuous time Markov chain models for chemical reaction networks." In: *Design and analysis of biomolecular circuits*. Springer, pp. 3–42.

Anderson, David F and Chaojie Yuan (2018). "Low variance couplings for stochastic models of intracellular processes with time-dependent rate functions." In: *Bulletin of mathematical biology*, pp. 1–29.

Anderson, William J (2012). *Continuous-time Markov chains: An applications-oriented approach*. Springer Science & Business Media.

Andreychenko, Aleksandr, Linar Mikeev, David Spieler, and Verena Wolf (2011). "Parameter identification for Markov models of biochemical reactions." In: *International Conference on Computer Aided Verification*. Springer, pp. 83–98.

Andreychenko, Alexander, Luca Bortolussi, Ramon Grima, Philipp Thomas, and Verena Wolf (2017). "Distribution approximations for the chemical master equation: comparison of the method of moments and the system size expansion." In: *Modeling Cellular Systems*. Springer, pp. 39–66.

Andreychenko, Alexander, Linar Mikeev, and Verena Wolf (2015). "Reconstruction of multimodal distributions for hybrid moment-based chemical kinetics." In: *Journal of Coupled Systems and Multiscale Dynamics* 3.2, pp. 156–163.

Aziz, Adnan, Kumud Sanwal, Vigyan Singhal, and Robert Brayton (1996). "Verifying continuous time Markov chains." In: *International Conference on Computer Aided Verification*. Springer, pp. 269–276.

Backenköhler, Michael (2019). *Stochastic Simulation and LCV Estimation Code*. URL: https://github.com/mbackenkoehler/cme-simulation.

Backenköhler, Michael (2020). *MJP Aggregation and Bridging Code*. URL: https://www.github.com/mbackenkoehler/mjp_bridging.

Backenköhler, Michael, Luca Bortolussi, Gerrit Großmann, and Verena Wolf (2021). "Analysis of Markov Jump Processes under Terminal Constraints." In: *27th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*. Vol. 1265. Lecture Notes in Computer Science. Springer, pp. 210–229.

Backenköhler, Michael, Luca Bortolussi, and Verena Wolf (2016). "Generalized method of moments for stochastic reaction networks in equilibrium." In: *International Conference on Computational Methods in Systems Biology*. Springer, pp. 15–29.

Backenköhler, Michael, Luca Bortolussi, and Verena Wolf (2018). "Moment-Based Parameter Estimation for Stochastic Reaction Networks in Equilibrium." In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)* 15.4, pp. 1180–1192.

Backenköhler, Michael, Luca Bortolussi, and Verena Wolf (2019). "Control Variates for Stochastic Simulation of Chemical Reaction Networks." In: *17th International Conference on Computational Methods in Systems Biology (CMSB)*. Vol. 11773. Lecture Notes in Computer Science. Springer, pp. 42–59.

Backenköhler, Michael, Luca Bortolussi, and Verena Wolf (2020). "Bounding Mean First Passage Times in Population Continuous-Time Markov Chains." In: *17th International Conference on Quantitative Evaluation of SysTems (QEST)*. Vol. 12289. Lecture Notes in Computer Science. Springer, pp. 155–174.

Baier, Christel, Boudewijn Haverkort, Holger Hermanns, and J-P Katoen (2003). "Model-checking algorithms for continuous-time Markov chains." In: *IEEE Transactions on software engineering* 29.6, pp. 524–541.

Baier, Christel, Boudewijn Haverkort, Holger Hermanns, and Joost-Pieter Katoen (2000). "Model checking continuous-time Markov chains by transient analysis." In: *International Conference on Computer Aided Verification*. Springer, pp. 358–372.

Barzel, Baruch and Ofer Biham (2008). "Calculation of switching times in the genetic toggle switch and other bistable systems." In: *Physical Review E* 78.4, p. 041919.

Bel, Golan, Brian Munsky, and Ilya Nemenman (2009). "The simplicity of completion time distributions for common complex biochemical processes." In: *Physical biology* 7.1, p. 016003.

Bernardo, Marco, Rocco De Nicola, and Jane Hillston, eds. (2016). *Formal Methods for the Quantitative Evaluation of Collective Adaptive Systems*. Vol. 9700. Lecture Notes in Computer Science. Springer.

Bogomolov, Sergiy, Thomas A Henzinger, Andreas Podelski, Jakob Ruess, and Christian Schilling (2015). "Adaptive moment closure for parameter inference of biochemical reaction networks." In: *International Conference on Computational Methods in Systems Biology*. Springer, pp. 77–89.

Bortolussi, Luca, Jane Hillston, Diego Latella, and Mieke Massink (May 2013). "Continuous approximation of collective system behaviour: A tutorial." In: *Performance Evaluation* 70.5, pp. 317–349.

Bortolussi, Luca, Jane Hillston, and Michele Loreti (2020). "Fluid approximation of broadcasting systems." In: *Theoretical Computer Science* 816, pp. 221–248.

Bortolussi, Luca and Roberta Lanciani (2013). "Model checking Markov population models by central limit approximation." In: *International Conference on Quantitative Evaluation of Systems*. Springer, pp. 123–138.

Bortolussi, Luca and Roberta Lanciani (2014). "Stochastic Approximation of Global Reachability Probabilities of Markov Population Models." In: *Computer Performance Engineering -*

11th European Workshop, EPEW 2014, Florence, Italy, September 11-12, 2014. Proceedings, pp. 224–239.

Bortolussi, Luca, Roberta Lanciani, and Laura Nenzi (2018). "Model checking Markov population models by stochastic approximations." In: *Inf. Comput.* 262, pp. 189–220.

Bortolussi, Luca, Dimitrios Milios, and Guido Sanguinetti (2015). "Efficient stochastic simulation of systems with multiple time scales via statistical abstraction." In: *International Conference on Computational Methods in Systems Biology*. Springer, pp. 40–51.

Breuer, Lothar (2003). *From Markov jump processes to spatial queues*. Springer.

Broemeling, Lyle D (2017). *Bayesian Inference for Stochastic Processes*. CRC Press.

Buchholz, Peter (1994). "Exact and ordinary lumpability in finite Markov chains." In: *Journal of applied probability*, pp. 59–75.

Byrne, George D. and Alan C. Hindmarsh (1975). "A polyalgorithm for the numerical solution of ordinary differential equations." In: *ACM Transactions on Mathematical Software (TOMS)* 1.1, pp. 71–96.

Cao, Wei-Lu and William J Stewart (1985). "Iterative aggregation/disaggregation techniques for nearly uncoupled Markov chains." In: *Journal of the ACM (JACM)* 32.3, pp. 702–719.

Cao, Yang, Daniel T Gillespie, and Linda R Petzold (2005). "The slow-scale stochastic simulation algorithm." In: *The Journal of chemical physics* 122.1, p. 014116.

Cardelli, Luca and Attila Csikász-Nagy (2012). "The cell cycle switch computes approximate majority." In: *Scientific reports* 2, p. 656.

Ceska, Milan and Jan Kretínský (2019). "Semi-quantitative Abstraction and Analysis of Chemical Reaction Networks." In: *Computer Aided Verification - 31st International Conference, CAV 2019, New York City, NY, USA, July 15-18, 2019, Proceedings, Part I*, pp. 475–496.

Chen, Taolue, Marco Diciolla, Marta Kwiatkowska, and Alexandru Mereacre (2011). "Time-bounded verification of CTMCs against real-time specifications." In: *International Conference on*

Formal Modeling and Analysis of Timed Systems. Springer, pp. 26–42.

Chen, Taolue, Tingting Han, Joost-Pieter Katoen, and Alexandru Mereacre (2009). "Quantitative model checking of continuous-time Markov chains against timed automata specifications." In: *2009 24th Annual IEEE Symposium on Logic In Computer Science*. IEEE, pp. 309–318.

Cheng, Russell CH (1978). "Analysis of simulation experiments under normality assumptions." In: *Journal of the Operational Research Society* 29.5, pp. 493–497.

Daigle Jr, Bernie J, Min K Roh, Dan T Gillespie, and Linda R Petzold (2011). "Automated estimation of rare event probabilities in biochemical systems." In: *The Journal of Chemical Physics* 134.4, 01B628.

David, Alexandre, Kim G Larsen, Axel Legay, Marius Mikučionis, Danny Bøgsted Poulsen, and Sean Sedwards (2015). "Statistical model checking for biological systems." In: *International Journal on Software Tools for Technology Transfer* 17.3, pp. 351–367.

Dayar, Tuğrul, Holger Hermanns, David Spieler, and Verena Wolf (2011). "Bounding the equilibrium distribution of Markov population models." In: *Numerical linear algebra with applications* 18.6, pp. 931–946.

Dayar, Tuğrul and William J Stewart (1997). "Quasi lumpability, lower-bounding coupling matrices, and nearly completely decomposable Markov chains." In: *SIAM Journal on Matrix Analysis and Applications* 18.2, pp. 482–498.

Dehnert, Christian, Sebastian Junges, Joost-Pieter Katoen, and Matthias Volk (2017). "A storm is coming: A modern probabilistic model checker." In: *International Conference on Computer Aided Verification*. Springer, pp. 592–600.

Diamond, Steven and Stephen Boyd (2016). "CVXPY: A Python-Embedded Modeling Language for Convex Optimization." In: *Journal of Machine Learning Research* 17.83, pp. 1–5.

Dowdy, Garrett R and Paul I Barton (2018a). "Bounds on stochastic chemical kinetic systems at steady state." In: *The Journal of chemical physics* 148.8, p. 084106.

Dowdy, Garrett R and Paul I Barton (2018b). "Dynamic bounds on stochastic chemical kinetic systems using semidefinite programming." In: *The Journal of chemical physics* 149.7, p. 074103.

Engblom, Stefan (2006). "Computing the moments of high dimensional solutions of the master equation." In: *Applied Mathematics and Computation* 180.2, pp. 498–515.

Ethier, Stewart N and Thomas G Kurtz (2009). *Markov processes: characterization and convergence*. Vol. 282. John Wiley & Sons.

Feller, William (1971). *An Introduction to Probability Theory and Its Applications*. Wiley & Sons.

Gast, Nicolas, Luca Bortolussi, and Mirco Tribastone (2019). "Size expansions of mean field approximation: Transient and steady-state analysis." In: *Performance Evaluation* 129, pp. 60 –80.

Geva-Zatorsky, Naama, Nitzan Rosenfeld, Shalev Itzkovitz, Ron Milo, Alex Sigal, Erez Dekel, Talia Yarnitzky, Yuvalal Liron, Paz Polak, Galit Lahav, et al. (2006). "Oscillations and variability in the p53 system." In: *Molecular systems biology* 2.1, pp. 2006–0033.

Ghusinga, Khem Raj, Andrew Lamperski, and Abhyudai Singh (2018). "Estimating stationary characteristic functions of stochastic systems via semidefinite programming." In: *2018 European Control Conference (ECC)*. IEEE, pp. 2720–2725.

Ghusinga, Khem Raj, Cesar A Vargas-Garcia, Andrew Lamperski, and Abhyudai Singh (2017). "Exact lower and upper bounds on stationary moments in stochastic biochemical systems." In: *Physical biology* 14.4, 04LT01.

Gihman, Iosif I and Anatoli V Skorohod (1975). *The Theory of Stochastic Processes II*. Springer.

Gillespie, Daniel T (1977). "Exact stochastic simulation of coupled chemical reactions." In: *The journal of physical chemistry* 81.25, pp. 2340–2361.

Gillespie, Daniel T (2000). "The chemical Langevin equation." In: *The Journal of Chemical Physics* 113.1, pp. 297–306.

Gillespie, Daniel T (2001). "Approximate accelerated stochastic simulation of chemically reacting systems." In: *The Journal of Chemical Physics* 115.4, pp. 1716–1733.

Glasserman, Paul and Bin Yu (2005). "Large sample properties of weighted monte carlo estimators." In: *Operations Research* 53.2, pp. 298–312.

Golightly, Andrew and Chris Sherlock (2019). "Efficient sampling of conditioned Markov jump processes." In: *Statistics and Computing* 29.5, pp. 1149–1163.

Golightly, Andrew and Darren J Wilkinson (2005). "Bayesian inference for stochastic kinetic models using a diffusion approximation." In: *Biometrics* 61.3, pp. 781–788.

Golightly, Andrew and Darren J Wilkinson (2011). "Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte Carlo." In: *Interface focus* 1.6, pp. 807–820.

Großmann, Gerrit, Michael Backenköhler, and Verena Wolf (2020). "Importance of Interaction Structure and Stochasticity for Epidemic Spreading: A COVID-19 Case Study." In: *17th International Conference on Quantitative Evaluation of SysTems (QEST)*. Vol. 12289. Lecture Notes in Computer Science. Springer, pp. 211–229.

Gupta, Ankit, Corentin Briat, and Mustafa Khammash (2014). "A scalable computational framework for establishing long-term behavior of stochastic reaction networks." In: *PLoS Comput Biol* 10.6, e1003669.

Gupta, Ankit, Jan Mikelson, and Mustafa Khammash (2017). "A finite state projection algorithm for the stationary solution of the chemical master equation." In: *The Journal of chemical physics* 147.15, p. 154101.

Hajnal, Matej, Morgane Nouvian, David Šafránek, and Tatjana Petrov (2019). "Data-Informed Parameter Synthesis for Population Markov Chains." In: *International Workshop on Hybrid Systems Biology*. Springer, pp. 147–164.

Harris, Charles R. et al. (2020). "Array programming with NumPy." In: *Nature* 585, pp. 357–362.

Hasenauer, Jan, Verena Wolf, Atefeh Kazeroonian, and Fabian J Theis (2014). "Method of conditional moments (MCM) for the chemical master equation." In: *Journal of mathematical biology* 69.3, pp. 687–735.

Hausdorff, Felix (1921). "Summationsmethoden und Momentfolgen." In: *Mathematische Zeitschrift* 9, pp. 74–109.

Hayden, Richard A, Anton Stefanek, and Jeremy T Bradley (2012). "Fluid computation of passage-time distributions in large Markov models." In: *Theoretical Computer Science* 413.1, pp. 106–141.

He, Shaobo, Yuexi Peng, and Kehui Sun (2020). "SEIR modeling of the COVID-19 and its dynamics." In: *Nonlinear Dynamics*, pp. 1–14.

Helmes, Kurt, Stefan Röhl, and Richard H Stockbridge (2001). "Computing moments of the exit time distribution for Markov processes by linear programming." In: *Operations Research* 49.4, pp. 516–530.

Henzinger, Thomas A, Maria Mateescu, and Verena Wolf (2009). "Sliding window abstraction for infinite Markov chains." In: *International Conference on Computer Aided Verification*. Springer, pp. 337–352.

Hespanha, Joao (2008). "Moment closure for biochemical networks." In: *2008 3rd International Symposium on Communications, Control and Signal Processing*. IEEE, pp. 142–147.

Hinton, Andrew, Marta Kwiatkowska, Gethin Norman, and David Parker (2006). "PRISM: A tool for automatic verification of probabilistic systems." In: *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, pp. 441–444.

Huang, Lirong, Loic Pauleve, Christoph Zechner, Michael Unger, Anders Hansen, and Heinz Koeppl (2016). "Reconstructing dynamic molecular states from single-cell time series." In: *Journal of The Royal Society Interface* 13.122, p. 20160533.

Iyer-Biswas, Srividya and Anton Zilman (2016). "First-Passage Processes in Cellular Biology." In: *Advances in Chemical Physics* 160, pp. 261–306.

Jahnke, Tobias and Wilhelm Huisinga (2007). "Solving the chemical master equation for monomolecular reaction systems analytically." In: *Journal of mathematical biology* 54.1, pp. 1–26.

Kashima, Kenji and Reiichiro Kawai (2009). "Polynomial programming approach to weak approximation of Lévy-driven

stochastic differential equations with application to option pricing." In: *2009 ICCAS-SICE*. IEEE, pp. 3902–3907.

Kazeroonian, Atefeh, Fabian J Theis, and Jan Hasenauer (2014). "Modeling of stochastic biological processes with non-polynomial propensities using non-central conditional moment equation." In: *IFAC Proceedings Volumes* 47.3, pp. 1729–1735.

Knuth, Donald E (1993). "Johann Faulhaber and sums of powers." In: *Mathematics of Computation* 61.203, pp. 277–294.

Kuntz, Juan, Philipp Thomas, Guy-Bart Stan, and Mauricio Barahona (2019a). "Bounding the stationary distributions of the chemical master equation via mathematical programming." In: *The Journal of chemical physics* 151.3, p. 034109.

Kuntz, Juan, Philipp Thomas, Guy-Bart Stan, and Mauricio Barahona (2019b). "The exit time finite state projection scheme: bounding exit distributions and occupation measures of continuous-time Markov chains." In: *SIAM Journal on Scientific Computing* 41.2, A748–A769.

Kuntz, Juan, Philipp Thomas, Guy-Bart Stan, and Mauricio Barahona (2021a). "Approximations of Countably Infinite Linear Programs over Bounded Measure Spaces." In: *SIAM Journal on Optimization* 31.1, pp. 604–625.

Kuntz, Juan, Philipp Thomas, Guy-Bart Stan, and Mauricio Barahona (2021b). "Stationary distributions of continuous-time Markov chains: a review of theory and truncation-based approximations." In: *SIAM Review* 63.1, pp. 3–64.

Kurasov, Pavel, Alexander Lück, Delio Mugnolo, and Verena Wolf (2018). "Stochastic hybrid models of gene regulatory networks–a PDE approach." In: *Mathematical biosciences* 305, pp. 170–177.

Kurtz, Thomas G (1981). *Approximation of population processes*. Vol. 36. Regional conference series in applied mathematics. SIAM.

Kuwahara, Hiroyuki and Ivan Mura (2008). "An efficient and exact stochastic simulation method to analyze rare events in biochemical systems." In: *The Journal of chemical physics* 129.16, 10B619.

Kwiatkowska, Marta, Gethin Norman, and David Parker (2011). "PRISM 4.0: Verification of probabilistic real-time systems." In:

International conference on computer aided verification. Springer, pp. 585–591.

L'Ecuyer, Pierre (1994). "Efficiency improvement and variance reduction." In: *Proceedings of the 26th conference on Winter simulation*. Society for Computer Simulation International, pp. 122–132.

Lapin, Maksim, Linar Mikeev, and Verena Wolf (2011). "SHAVE: stochastic hybrid analysis of Markov population models." In: *Proceedings of the 14th international conference on Hybrid systems: computation and control*, pp. 311–312.

Lasserre, Jean-Bernard (2010). *Moments, positive polynomials and their applications*. Vol. 1. World Scientific.

Lasserre, Jean-Bernard, Tomas Prieto-Rumeau, and Mihail Zervos (2006). "Pricing a class of exotic options via moments and SDP relaxations." In: *Mathematical Finance* 16.3, pp. 469–494.

Lavenberg, Stephen S, Thomas L Moeller, and Peter D Welch (1982). "Statistical results on control variables with application to queueing network simulation." In: *Operations Research* 30.1, pp. 182–202.

Lipshtat, Azi, Adiel Loinger, Nathalie Q Balaban, and Ofer Biham (2006). "Genetic toggle switch without cooperative binding." In: *Physical review letters* 96.18, p. 188101.

Loinger, Adiel, Azi Lipshtat, Nathalie Q Balaban, and Ofer Biham (2007). "Stochastic simulations of genetic switch systems." In: *Physical Review E* 75.2, p. 021904.

MOSEK ApS (2018). *MOSEK Optimizer API for C 8.1.0.67*. URL: https://docs.mosek.com/8.1/capi/index.html.

Mateescu, M, V Wolf, F Didier, and TA Henzinger (2010). "Fast adaptive uniformisation of the chemical master equation." In: *IET systems biology* 4.6, pp. 441–452.

Mélykúti, Bence, Joao P Hespanha, and Mustafa Khammash (2014). "Equilibrium distributions of simple biochemical reaction systems for time-scale separation in stochastic reaction networks." In: *Journal of The Royal Society Interface* 11.97, p. 20140054.

Meyn, Sean P and Richard L Tweedie (1993). "Stability of Markovian processes III: Foster-Lyapunov criteria for continuous-time processes." In: *Advances in Applied Probability*, pp. 518–548.

Meyn, Sean P and Richard L Tweedie (2012). *Markov chains and stochastic stability*. Springer.

Meyn, Sean P, Richard L Tweedie, et al. (1994). "Computable bounds for geometric convergence rates of Markov chains." In: *The Annals of Applied Probability* 4.4, pp. 981–1011.

Mikeev, Linar, Martin R Neuhäußer, David Spieler, and Verena Wolf (2013). "On-the-fly verification and optimization of DTA-properties for large Markov chains." In: *Formal Methods in System Design* 43.2, pp. 313–337.

Mikeev, Linar and Werner Sandmann (2019). "Approximate Numerical Integration of the Chemical Master Equation for Stochastic Reaction Networks." In: *arXiv preprint arXiv:1907.10245*.

Mikeev, Linar, Werner Sandmann, and Verena Wolf (2011). "Efficient calculation of rare event probabilities in Markovian queueing networks." In: *Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools*, pp. 186–196.

Mikeev, Linar, Werner Sandmann, and Verena Wolf (2013). "Numerical approximation of rare event probabilities in biochemically reacting systems." In: *International Conference on Computational Methods in Systems Biology*. Springer, pp. 5–18.

Milias-Argeitis, Andreas and Mustafa Khammash (2014). "Optimization-based Lyapunov function construction for continuous-time Markov chains with affine transition rates." In: *53rd IEEE Conference on Decision and Control*. IEEE, pp. 4617–4622.

Milner, Peter, Colin S Gillespie, and Darren J Wilkinson (2013). "Moment closure based parameter inference of stochastic kinetic models." In: *Statistics and Computing* 23.2, pp. 287–295.

Mode, Charles J and Candace K Sleeman (2000). *Stochastic processes in epidemiology: HIV/AIDS, other infectious diseases, and computers*. World Scientific.

Munsky, Brian and Mustafa Khammash (2006). "The finite state projection algorithm for the solution of the chemical master equation." In: *The Journal of chemical physics* 124.4, p. 044104.

Munsky, Brian, Ilya Nemenman, and Golan Bel (2009). "Specificity and completion time distributions of biochemical processes." In: *The Journal of chemical physics* 131.23, 12B616.

Nelson, Barry L (1990). "Control variate remedies." In: *Operations Research* 38.6, pp. 974–992.

Neupane, Thakur, Chris J Myers, Curtis Madsen, Hao Zheng, and Zhen Zhang (2019). "STAMINA: STochastic approximate model-checker for INfinite-state analysis." In: *International Conference on Computer Aided Verification*. Springer, pp. 540–549.

O'Donoghue, B., E. Chu, N. Parikh, and S. Boyd (Nov. 2017). *SCS: Splitting Conic Solver, version 2.1.0*. URL: https://github.com/cvxgrp/scs.

Pardoux, Etienne (2008). *Markov processes and applications: algorithms, networks, genome and finance*. Vol. 796. John Wiley & Sons.

Parrilo, Pablo A (2003). "Semidefinite programming relaxations for semialgebraic problems." In: *Mathematical programming* 96.2, pp. 293–320.

Porter, Mason A and James P Gleeson (2016). "Dynamical systems on networks." In: *Frontiers in Applied Dynamical Systems: Reviews and Tutorials* 4.

Rabiner, Lawrence R and Biing-Hwang Juang (1986). "An introduction to hidden Markov models." In: *IEEE ASSP Magazine* 3.1, pp. 4–16.

Sakurai, Yuta and Yutaka Hori (2017). "A convex approach to steady state moment analysis for stochastic chemical reactions." In: *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*. IEEE, pp. 1206–1211.

Sakurai, Yuta and Yutaka Hori (2019). "Bounding Transient Moments of Stochastic Chemical Reactions." In: *IEEE Control Systems Letters* 3.2, pp. 290–295.

Särkkä, Simo (2013). *Bayesian filtering and smoothing*. Vol. 3. Cambridge University Press.

Schnoerr, David, Botond Cseke, Ramon Grima, and Guido Sanguinetti (2017). "Efficient Low-Order Approximation of First-Passage Time Distributions." In: *Phys. Rev. Lett.* 119 (21), p. 210601.

Schnoerr, David, Guido Sanguinetti, and Ramon Grima (2014). "Validity conditions for moment closure approximations in stochastic chemical kinetics." In: *The Journal of chemical physics* 141.8, 08B616_1.

Schnoerr, David, Guido Sanguinetti, and Ramon Grima (Nov. 2015). "Comparison of different moment-closure approximations for stochastic chemical kinetics." In: *The Journal of Chemical Physics* 143.18, p. 185101.

Schnoerr, David, Guido Sanguinetti, and Ramon Grima (Mar. 2017). "Approximation and inference methods for stochastic biochemical kinetics—a tutorial review." In: *Journal of Physics A: Mathematical and Theoretical* 50.9, p. 093001.

Schweitzer, Paul J (1991). "A survey of aggregation-disaggregation in large Markov chains." In: *Numerical solution of Markov chains* 8, pp. 63–88.

Siegal-Gaskins, Dan, Maria Katherine Mejia-Guerra, Gregory D Smith, and Erich Grotewold (2011). "Emergence of switch-like behavior in a large family of simple biochemical networks." In: *PLoS Comput Biol* 7.5, e1002039.

Singh, Abhyudai and Joao Pedro Hespanha (2006). "Lognormal moment closures for biochemical reactions." In: *Proceedings of the 45th IEEE Conference on Decision and Control*. IEEE, pp. 2063–2068.

Spieler, David (2010). *Geobound*. URL: https://mosi.uni-saarland.de/tools/geobound.

Spieler, David (2014). "Numerical analysis of long-run properties for Markov population models." PhD thesis. Saarland University.

Spieler, David, Ernst Moritz Hahn, and Lijun Zhang (2011). "Model checking CSL for Markov population models." In: *arXiv preprint arXiv:1111.4385*.

Stekel, Dov J and Dafyd J Jenkins (2008). "Strong negative self regulation of prokaryotic transcription factors increases the

intrinsic noise of protein expression." In: *BMC systems biology* 2.1, p. 6.

Stewart, William J (1994). *Introduction to the numerical solution of Markov chains*. Princeton University Press.

Stewart, William J (2009). *Probability, Markov chains, queues, and simulation: the mathematical basis of performance modeling*. Princeton university press.

Strasser, Michael, Fabian J Theis, and Carsten Marr (2012). "Stability and multiattractor dynamics of a toggle switch based on a two-stage model of stochastic gene expression." In: *Biophysical journal* 102.1, pp. 19–29.

Szechtman, Roberto (2003). "Control variate techniques for monte carlo simulation: control variates techniques for monte carlo simulation." In: *Proceedings of the 35th conference on Winter simulation: driving innovation*. Winter Simulation Conference, pp. 144–149.

Tweedie, Richard L (1975). "Sufficient conditions for regularity, recurrence and ergodicity of Markov processes." In: *Mathematical Proceedings of the Cambridge Philosophical Society* 78.1, pp. 125–136.

Ullah, Mukhtar and Olaf Wolkenhauer (July 2009). "Stochastic Approaches for Systems Biology." In: *Wiley interdisciplinary reviews. Systems biology and medicine* 2, pp. 385–97.

Ullah, Mukhtar and Olaf Wolkenhauer (2011). *Stochastic approaches for systems biology*. Springer.

Van Kampen, Nicolaas Godfried (1992). *Stochastic processes in physics and chemistry*. Vol. 1. Elsevier.

Vandenberghe, Lieven (2010). *The CVXOPT linear and quadratic cone program solvers*. URL: http://cvxopt.org/documentation/coneprog.pdf.

Virtanen, Pauli et al. (2020). "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." In: *Nature Methods* 17, pp. 261–272.

Vorst, Henk A Van der (1992). "Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems." In: *SIAM Journal on scientific and Statistical Computing* 13.2, pp. 631–644.

Wildner, Christian and Heinz Koeppl (2019). "Moment-based variational inference for Markov jump processes." In: *International Conference on Machine Learning*. PMLR, pp. 6766–6775.

Wilkinson, Darren J (2018). *Stochastic modelling for systems biology*. CRC press.

Wilson, James R (1984). "Variance reduction techniques for digital simulation." In: *American Journal of Mathematical and Management Sciences* 4.3-4, pp. 277–312.

Zapreev, Ivan and Joost-Pieter Katoen (2006). "Safe on-the-fly steady-state detection for time-bounded reachability." In: *Third International Conference on the Quantitative Evaluation of Systems-(QEST'06)*. IEEE, pp. 301–310.

LIST OF FIGURES

LIST OF TABLES

LIST OF MODELS

ACRONYMS

CDF	cumulative density function
CME	chemical master equation
CRN	chemical reaction network
CSL	concurrent separation logic
CV	control variate
CTMC	continuous-time Markov chain
DNA	deoxyribonucleic acid
DTMC	discrete-time Markov chain
FPT	first-passage time
FSP	finite state projection
GMP	generalized moment problem
HMM	hidden Markov model
IVP	initial value problem
MFPT	mean first-passage time
MPM	Markovian population model
MTL	metric temporal logic
ODE	ordinary differential equation
pCTMC	population CTMC
SEIR	susceptible-exposed-infected-removed
SDP	semi-definite program
SSA	stochastic simulation algorithm
SRN	stochastic reaction network

COLOPHON

This document was typeset in LaTeX using a design based upon `classicthesis v4.6` developed by André Miede and Ivo Pletikosić.

Hermann Zapf's *Palatino* and *Euler* type faces (Type 1 PostScript fonts *URW Palladio L* and *FPL*) are used. The "typewriter" text is typeset in *Bera Mono*, originally developed by Bitstream, Inc. as "Bitstream Vera". (Type 1 PostScript fonts were made available by Malte Rosenau and Ulrich Dirr.)

Final Version as of July 20, 2021.

NOTE: The custom size of the textblock was calculated using the directions given by Mr. Bringhurst (pages 26–29 and 175/176). 10 pt Palatino needs 133.21 pt for the string "abcdefghijklmnopqrstuvwxyz". This yields a good line length between 24–26 pc (288–312 pt). Using a "*double square textblock*" with a 1:2 ratio this results in a textblock of 312:624 pt (which includes the headline in this design). A good alternative would be the "*golden section textblock*" with a ratio of 1:1.62, here 312:505.44 pt. For comparison, `DIV9` of the `typearea` package results in a line length of 389 pt (32.4 pc), which is by far too long. However, this information will only be of interest for hardcore pseudo-typographers like me.

To make your own calculations, use the following commands and look up the corresponding lengths in the book:

```
\settowidth{\abcd}{abcdefghijklmnopqrstuvwxyz}
\the\abcd\ % prints the value of the length
```

Please see the file `classicthesis.sty` for some precalculated values for Palatino and Minion.

abcdefghijklmnopqrstuvwxyzabcdefghijklmnopqrstuvwxyz
145.86469pt