VARIANCE-REDUCED SIMULATION OF LATTICE MARKOV CHAINS

BY

PETER A. MAGINNIS

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mechanical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2018

Urbana, Illinois

Doctoral Committee:

Professor Matthew West, Co-Chair
Professor Geir E. Dullerud, Co-Chair
Professor Rayadurgam Srikant
Professor David F. Anderson
Professor Narayana Aluru

Abstract

The focus of this dissertation is on reducing the cost of Monte Carlo estimation for lattice-valued Markov chains. We achieve this goal by manipulating the random inputs to stochastic processes (Poisson random variables in the discrete-time setting and Poisson processes in continuous-time) such that they become negatively correlated with some of their cohort while their individual marginal distributions are completely unaltered. In this way, we preserve the convergence properties of the Law of Large Numbers, but mean estimates, say, constructed from these sample paths exhibit dramatically reduced variance. The work is comprised of three main parts. First, we introduce algorithms to reduce the simulation costs for discrete-time Markov chains. We describe how to modify the simulation of sample trajectories that introduces negative correlation while introducing no additional computational cost and that are compatible with existing codes. We support this algorithm with theoretical results, including guarantee that such mean estimators will be unbiased and consistent with respect to the discrete-time distribution. Further, we prove a recursive relation that characterizes the evolution of mutual negative covariance over time in the general case as well as prove a sufficient condition in the case of linear rate functions. Lastly, we present several numerical experiments that demonstrate multiple orders-of-magnitude reduction in mean-square error (MSE) for both linear and nonlinear reaction rate systems.

In the next part, we show how insights gained from the discrete-time case can be used to inform a related approach in continuous-time. In these cases, we rely on a formulation of these lattice Markov chains called the random-time change representation. This allows us to translate the general problem of simulating anticorrelated trajectories of a given lattice Markov chain into the simpler problem of simulating anticorrelated pairs of unit-rate Poisson processes, which are the fundamental source of randomness that are input into random time-change representations. We systematically construct and analyze algorithms to produce negatively correlated, identically distributed Poisson processes. We prove closed form expressions for the MSE evolution of one of these systems, as well as present asymptotic performance lower bounds. We then show how to use these anticorrelated Poisson processes to simulate exact, identically distributed stochastic processes which are now significantly negatively correlated, and are thus suitable for variance-reduced Monte

Carlo. Numerical experiments on both linear and nonlinear systems demonstrate order-of-magnitude cost reduction. We also introduce error vs cost comparisons with existing standard methods.

Finally, we present extensions and refinements of the above algorithms. First is an approach to discrete-time simulation (specifically for tau-leaping systems) that leverages insights gained from the continuous-time approach in order to further strengthen the performance of the original algorithm in its weakest regime. This algorithm inherits several desirable properties from the antithetic discrete-time simulation case. In addition, we present numerical studies that show where this refinement outperforms the original algorithm. Finally, we present extensions of the anticorrelated simulation algorithms into both model predictive control and particle filtering.

To Jessica.

Table of Contents

List of Figures

List of Abbreviations

iid	independent and identically distributed
ODE	ordinary differential equation
CTMC	continuous-time Markov chain
DTMC	discrete-time Markov chain
RTC	random time-change
MSE	mean-squared error
CDF	cumulative distribution function
PRNG	pseudorandom number generator

Chapter 1

Introduction

This dissertation is concerned with reducing the cost of Monte Carlo simulation of a large class of stochastic systems with discrete state-spaces. Such systems, which we refer to as lattice Markov chains, include counting processes, jump systems, and reaction networks. They are broadly applied in numerous areas of research but are frequently intractable to analysis and exhibit inherent stochasticity that is difficult to approximate with deterministic models. Consequently, simulating these systems numerically to study their behavior and properties is frequently the best approach. Monte Carlo provides convergent estimates, but frequently requires many sample paths to achieve a desired degree of accuracy. When system trajectories become expensive to simulate, usually due to high-dimensionality or time-scale separation, the cost of Monte Carlo estimation can become prohibitive. This work provides algorithms, approaches and theory that reduce the variance (and thus the cost) of such estimators in order to dramatically reduce the number of sample trajectories needed to produce accurate estimates. While anticorrelating two sample trajectories is not inherently difficult (and is even trivial in the case of say Brownian motion), the key insight here is how to entirely preserve the marginal distributions of trajectories such that they remain identical, regardless of their particular system definitions or asymmetries. As we will show, these techniques can reduce estimator variance by one or more orders-of-magnitude with no or minimal computational overhead increase.

Lattice Markov chains systems are significant in the stochastic simulation literature, with applications including stochastic chemical systems [18], systems biology [40], aerosol modeling [33], gene expression systems [8], and HIV infection [6]. They are particularly useful when the number of particles of a population or one of its subspecies are small and are thus poorly approximated by large-concentration ODE limits. While a few ad hoc techniques were devised to simulate such systems historically, the first generalized algorithm was proposed by Gillespie [18] with the stochastic simulation algorithm (SSA). In the interim, both as computational resources have grown exponentially and the inherent stochasticity of many systems has become better understood [27], such methods have seen increased utility and development. As increasingly complex models are developed, the cost of Monte Carlo simulation for their study can become prohibitive. To address this issue, we seek algorithms to reduce the variance of unbiased Monte Carlo estimates, increasing their accuracy for a fixed or reduced number of sample trajectories.

We direct our attention to two primary classes of simulation: lattice continuous-time Markov chains (CTMCs) and lattice discrete-time Markov chains (DTMCs). Frequently, lattice CTMCs are of primary interest to researchers due to their strong connection to physical systems, and lattice DTMCs often arise as numerical approximations to CTMCs when their simulation costs become significant. For example, the tau-leaping algorithm [19] is a widely used time-stepping method to approximately simulate lattice CTMCs when one or more of their reactions occur very frequently. It yields a lattice DTMC. Our work presents algorithms that either reduce the cost of Monte Carlo simulation for lattice CTMCs or for lattice DTMCs, and the organization of the dissertation reflects this partition.

While the algorithms we present for continuous-time and discrete-time systems are distinct, they share a common macroscopic approach. We aim to reduce the cost of Monte Carlo simulation, which traditionally simulates independent, identically distributed (iid) sample paths to construct statistical estimates [34], by modifying it to slightly relax the independence assumption. Namely, we seek to modify the simulation algorithms used to draw trajectories such that small subsets of paths are negatively correlated with each other. The result is that estimators constructed using these sample paths will have reduced variance, sometimes by an order-of-magnitude or more. Importantly, none of our algorithms relax the identically distributed condition of Monte Carlo simulation, nor do any of them alter the original process distribution. The marginal distribution of any one of our anticorrelated paths must be *identical* to the iid version of the system we are simulating, whether in continuous- or discrete-time. Even though many of the lattice DTMCs we study will be biased approximations of lattice CTMCs, our algorithms will introduce no additional bias by negatively correlating sample paths. We achieve this in both the continuous- and discrete-time cases by manipulating the sources of random input to the models. In the continuous-time case, these can be expressed as unit-rate Poisson processes, and in the discrete-time case these are Poisson random variables. While DTMCs are frequently used to approximate CTMCs, we will present the discrete algorithms first as they are more straightforward and illustrative. With the intuition they provide, we then present the more involved CTMC algorithms.

In tau-leaping algorithms, the number of events within a time step are sampled from appropriate Poisson distributions; it is the exact analog of a deterministic Euler approximation in time of a lattice CTMC. The method was introduced by Gillespie [19], and is particularly desirable for the simulation of processes that have at least some reaction channels which experience many transitions on short time scales. The discrete-time approximation used in tau-leaping produces a biased lattice DTMC distribution with respect to the original lattice CTMC distribution, but stability and convergence to the stochastic simulation algorithm

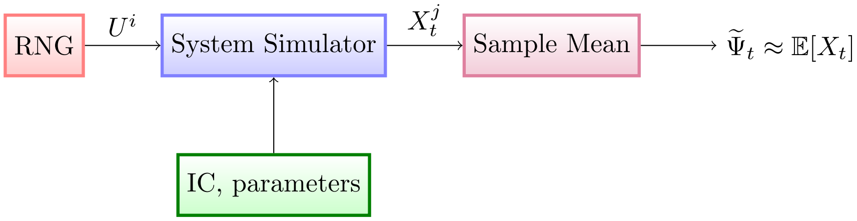

Figure 1.1: Traditional Monte Carlo mean estimation using iid stochastic simulation. Here, iid sequences $\{U^1, U^2, \ldots\}$ of standard uniform variates from a random number generator (RNG) drive a particular discrete-time process simulation. These sources of randomness, combined with initial conditions (IC) and system parameters, govern the system evolution, and are used to produce a sequence of iid sample paths $\{X_t^j\}_{j=1}^N$. This sequence is then used to compute the sample mean, $\widetilde{\Psi}_t = \frac{1}{N}\sum_{j=1}^N X_t^j$, in order to approximate $\mathbb{E}[X_t]$, the true mean of the process.

(SSA) [17, 18] (which simulates lattice CTMCs exactly) have been proven [31, 29] for sufficiently small step-size τ. Significant work has been done to produce enhanced tau-leaping algorithms, including the development of adaptive step size selection [11, 2] and implicit variants [30]. While we don't present any direct applications of the anticorrelated simulation techniques we present here to these more sophisticated tau-leaping algorithms, we do believe that adaptations of the work in this dissertation are straightforwardly applicable to several of these important refinements.

The methods we propose for discrete-time simulation require only the manipulation of uniform random inputs to the "black-box" system dynamics, and thus are easily implemented in any typical code. This feature of our approach is similar to that of other numerical algorithms that incorporate legacy simulation codes in a modular fashion, such as the recursive projection method [37] and equation-free methods [22]. The flow of traditional independent, identically distributed (iid) Monte Carlo simulation for mean estimation is shown in Figure 1.1, and is contrasted with Monte Carlo driven using our antithetically paired sample paths, shown in Figure 1.2.

Of particular interest are mean estimators that dominate traditional iid Monte Carlo estimates, meaning unbiased estimators whose MSE is lower than iid estimators over any operating parameters. Dominant mean estimators avoid the need to tune a given technique to a particular application and also provide performance guarantees. Over such a general class of models (with no apparent exploitable symmetries), a generalized variance reduction via anticorrelation is challenging; analytical guarantees are difficult to achieve for the same reason that Monte Carlo simulation is necessary, namely that such models are often analytically intractable.

In the continuous-time setting, the slow-scale SSA method [10] has been proposed to simulate fast and slow dynamics in a separate but coupled fashion. Other variance reduction techniques applied to the continuous simulation version of this class of processes include, for example the common random numbers and common reaction path methods proposed in Rathinam et al. [32] and an efficient finite-difference technique

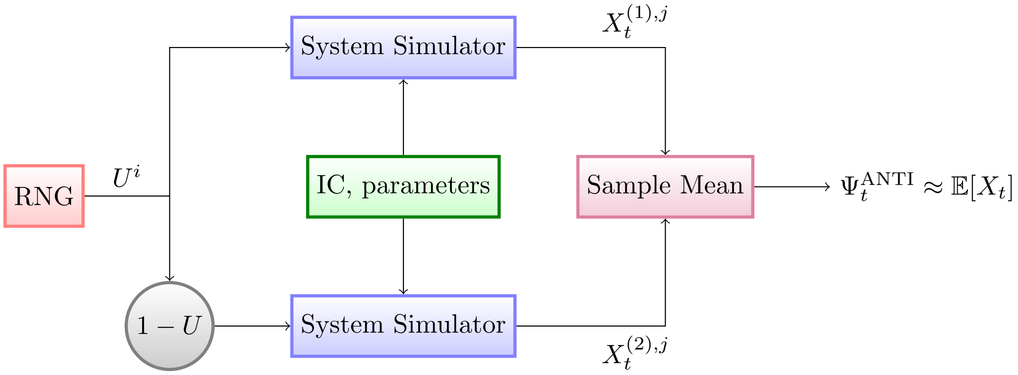

Figure 1.2: Variance-reduced Monte Carlo mean estimation using antithetic stochastic simulation as an alternative to the iid Monte Carlo estimation architecture shown in Figure 1.1. Again, a single stream of standard uniform variates $\{U^1, U^2, \ldots\}$ drive a particular discrete-time process simulation to produce a sequence of iid sample paths $\{X_t^{(1),j}\}_{j=1}^N$. Additionally, however, the corresponding antithetic standard uniform sequence $\{1-U^1, 1-U^2, \ldots\}$ (itself a sequence of uniform variables on $[0,1]$) is used as input for another realization of the same discrete-time process simulation with the same initial conditions and reaction rates to produce another iid sequence $\{X_t^{(2),j}\}_{j=1}^N$ of sample paths with identical marginal distribution. The most important feature of these sequences is that, for any j, $X_t^{(1),j}$ and $X_t^{(2),j}$ are correlated for each t. When this correlation is negative, their sample mean, $\Psi_t^{\text{ANTI}} = \frac{1}{2N}\sum_{j=1}^N \left(X_t^{(1),j} + X_t^{(2),j}\right)$, will be an unbiased estimator of the true mean $\mathbb{E}[X_t]$ and will have lower variance than $\widetilde{\Psi}_t$.

was proposed by Anderson [3], used to estimate parameter sensitivities. Anderson and Higham [4] have also proposed a multilevel technique that reduces variance using a type of iterated control variates. For multiscale systems in this class, exact, reduced cost sampling techniques using binning strategies [28] compatible with the techniques of this work have been shown to be effective.

The algorithms we present for the simulation of lattice CTMCs exploit the random time-change representation [14, 5] of a lattice CTMC in order to draw out their fundamental source of randomness: unit-rate Poisson processes. We propose a pair of related algorithms for simulating negatively correlated pairs of Poisson processes while preserving their marginal distributions. These anticorrelated Poisson process pairs can then be used as random input into random time-change representations of lattice CTMCs to produce negatively correlated sample paths of the lattice CTMC itself. These anticorrelated sample paths can then be used to construct unbiased, reduced-variance Monte Carlo mean estimates of the lattice CTMC distribution. As in the lattice DTMC case, by manipulating random inputs alone, we avoid any interactions with the particular system dynamics. Since the marginal distributions of our inputs are unchanged, and since we ensure that all random inputs used to construct a single path are mutually independent, we can simulate any such system without altering its dynamics or marginal distribution. Additionally, we will demonstrate with numerical experiments that the resulting lattice CTMC pair has significant negative correlation, even in nonlinear examples. While the algorithms we propose do result in some additional computational costs, the numerical cost-error studies we present show that the significant gains in performance still allow it to remain competitive with standard methods such as next-reaction methods [15, 1].

In both the discrete- and continuous-time cases, our focus is on typical system behavior and we invoke no measure changes as in well-known variance reduction techniques like importance sampling [20] or restarting [38]. In this work, we don't address systems that include time-delays [7, 9, 1]. Further, we do not directly address issues arising from time-scale separation between reaction channels [10]. Alternative variance reduction approaches for stochastic simulation include common reaction path methods [32] and finite-differencing [3]. Multi-level Monte Carlo methods [16] have been extended to and studied for this class of processes [4].

The outline of the dissertation is as follows. In Chapter 2, we remind the reader of several important mathematical background preliminaries that may be helpful in understanding the subsequent chapters. In Chapter 3, we introduce the antithetic simulation of lattice DTMCs. First, in Section 3.1 we introduce the algorithm to produce unbiased, variance-reduced mean estimators drawn for such systems. In this section, we also provide some insights to the implementation of such algorithms in practice, perhaps on parallel high performance computing architectures. We also define the error metrics we'll use in the discrete-time cases in this dissertation. Next, in Section 3.2, we present and prove several analytical results, including unbiasedness, covariance evolution over time, and a sufficient condition for variance-reduction in the affine rates case. Finally, in Section 3.3, we introduce three example systems and conduct numerical parameter studies to demonstrate significant variance-reduction in each case. The three systems are tau-leaping approximations of a gene expression system with affine rate functions, and two systems with nonlinear rate functions: a coagulation system in the presence of gravitational settling and an HIV infection system.

In Chapter 4, we develop algorithms for the anticorrelated simulation of lattice CTMCs. We begin in Section 4.1 by showing how to construct negatively correlated unit-rate Poisson processes using a pair of algorithms, which we refer to as the endpoint technique and its refinement, the binomial-midpoint technique. We also present analysis of the algorithms, including a closed form solution for the mean-squared error (MSE) of the endpoint technique, as well as numerical studies of each algorithm to demonstrate their dependence on operating parameters. Finally, in Section 4.2 we explain how these negatively correlated Poisson processes can be used as input to the random time-change representation to produce negatively correlated, exact realizations of lattice CTMC systems. We present as numerical examples the continuous-time versions of the gene expression and coagulation systems used in Section 3.3.

In Chapter 5, we present a new framework for variance-reduced lattice DTMC simulation that extends beyond antithetic simulation to any anticorrelated technique for generating Poisson random variables. In effect, this serves to de-couple the variance-reduction method from the dynamics of the system. In particular, we can use this new paradigm to further reduce variance in the simulation of tau-leaping systems that experience few reactions-per-time-step. We present one possible implementation of anticorrelated tau-leaping, based on the binomial midpoint technique in the continuous-time setting. In Section 5.2, we present the results of numerical studies of this initial implementation, again for the tau-leaping versions of the affine gene expression and nonlinear coagulation systems.

In Chapter 6, we present two other applications of variance-reduced simulation. In Section 6.1 we discuss how antithetic simulation techniques can be applied the stochastic model predictive control problem by reducing the cost of computing an estimated cost-to-go. This is particular useful for systems running stochastic controllers online that are subject to strict computational constraints. In Section 6.2, we propose a way to use antithetic discrete-time simulation to reduce the cost of the prediction step of bootstrap particle filtering algorithms.

Finally, in Chapter 7 we summarize the results of the dissertation and present some possible future directions for research.

Chapter 2

Preliminaries

The inspiration for several of the algorithms we will present is the classical technique to sample antithetic pairs of scalar random variables. For example, we may generate two Poisson-distributed random samples (X_1, X_2) by defining

$$\begin{aligned} X_1 &:= F_\tau^{-1}(U) \\ X_2 &:= F_\tau^{-1}(1-U), \end{aligned}$$

where $U \sim \text{Unif}[0,1]$ is a uniform random variable and F_τ^{-1} is the formal inverse of the Poisson cumulative distribution function (CDF) with parameter τ. It's easy to show that $\text{Cov}\,(X_1, X_2) \leq 0$ for all τ [25]. We will denote an antithetic draw from this distribution by $(X_1, X_2) \overset{\text{anti}}{\sim} \text{Pois}(\tau)$.

We will also use a related approach to simulate unit-rate Poisson process trajectories. Let $Y(t)$ denote a unit-rate Poisson process. For $s < t$, define $N(s,t) := Y(t) - Y(s)$ to be its increment. Recall that the unit-rate Poisson process is defined by three properties [35]:

- $Y(0) = 0$
- if $s_1 < t_1 \leq s_2 < t_2$, then $N(s_1, t_1)$ and $N(s_2, t_2)$ are independent
- the increment is Poisson distributed with parameter $t - s$, i.e., $N(s,t) \sim \text{Pois}(t-s)$

Further, recall that its arrival times conditioned on its value at the endpoints of any interval are uniformly distributed throughout that interval, and, its increment within the interval, conditioned on the same values, is binomially distributed. That is, for $s < u < t$,

$$N(s,u) | \{Y(s), Y(t)\} \sim \text{Binom}\left(N(s,t), \frac{u-s}{t-s}\right).$$

2.1 Lattice continuous-time Markov chains

The random time-change (RTC or Kurtz) representation [14] of a lattice CTMC expresses the process as a linear combination of unit-rate Poisson processes, each run at different time-rates determined by the rate functions and current state of the process. Consider a state vector $X(t) \in \mathbb{Z}^D$, $t \in [0, T]$. If the process has I event channels, each with propensity function $a^i(t, x)$, its RTC representation is given by

$$X(t) = x_0 + \sum_{i=1}^{I} Y^i \left(\int\limits_0^t a^i(s, X(s))\, \mathrm{d}s \right) \nu^i. \tag{2.1}$$

Here, $\{Y^i\}_{i=1}^I$ are independent, unit-rate Poisson processes and $\nu^i \in \mathbb{Z}^D$, $i = 1, \ldots, I$, are the state jump vectors. That is, $\nu^i = X(t^+) - X(t^-)$ if the ith event channel experiences a transition at time t.

2.2 Lattice discrete-time Markov chains

Consider the random time-change (or Kurtz) representation (2.1) above. The evolution of such a process can be studied alongside a corresponding a lattice discrete-time Markov chain. For fixed timestep increment τ, consider the discrete-time approximation $X_k \approx X(\tau k)$ for $k \in \{0, \ldots, K\}$, where $K := \max\{k : \tau k \leq T\}$. Then X_k evolves via

$$X_{k+1} = X_k + \sum_{i=1}^{I} S_k^i \left(a^i\big(\tau k, X_k\big)\tau \right) \nu^i, \tag{2.2}$$

where $S_k^i(\lambda) \sim \mathrm{Pois}(\lambda)$. For compactness, define $\lambda^i(k, X_k) = a^i\big(\tau k, X_k\big)\tau$ and denote $S_k^i \left(\lambda^i(k, X_k)\right)$ by S_k^i. Allowing for an abuse of notation, let t replace k for the discrete-time index (used as a subscript), and (6.52) becomes

$$X_{t+1} = X_t + \sum_{i=1}^{I} S_t^i \nu^i. \tag{2.3}$$

The technique presented in this dissertation demonstrates how to produce unbiased, anticorrelated ensembles of this discrete-time system (2.3) with respect to its own distribution. Variance-reduced simulation of the continuous-time system (2.1) is possible by using the tau-leaping method of [19] to produce a corresponding DTMC, but this introduces bias with respect to the CTMC distribution. Unbiased, variance-reduced simulation of the lattice CTMC system is the subject of future work. The discrete time system evolves at each timestep via a collection of independent, marginally-Poisson random variable draws with stochastic-valued parameters. This structure is crucial in the construction of anticorrelated path ensembles drawn from (2.3), as defined in Algorithm 1 below. Due to the discrete time approximation, there is a nonzero probability of transition to a state outside of the domain of the continuous time system (e.g. a negative number of particles). We handle this using the method of [31] in the tau-leaping context, namely by truncating the state to zero instead if it would transition to a negative state. For clarity, we restrict our attention to explicit tau-leaping with fixed step size, though extension to implicit [30] and/or adaptive [11, 2] variants are are of future interest.

2.3 Monte Carlo estimation of discrete-time systems

Our setting of interest is the Monte Carlo estimation of the mean behavior μ_t of a discrete-time stochastic process X_t. The classical Monte Carlo approach is to draw M iid samples of the process to produce an estimator $\widetilde{\Psi}_t^M \approx \mu_t = \mathbb{E}[X_t]$ (hereafter, the tilde will denote the use of iid simulation). Throughout the dissertation, we will compare this iid approach to our anticorrelated approach, which produces an alternative mean estimator Ψ_t^M, constructed from identically distributed but non-independent sample paths. We define the iid mean-estimator $\widetilde{\Psi}_t^M$ of a discrete-time stochastic process $X_t \in \mathbb{R}^D$ to be the sample-mean of an iid collection of M sample paths $\widetilde{X}_t^{(r)}$,

$$\widetilde{\Psi}_t^M := \frac{1}{M} \sum_{r=1}^{M} \widetilde{X}_t^{(r)}. \tag{2.4}$$

It is well established [34] that this is an unbiased mean-estimator that achieves the minimum variance possible using iid samples. To reduce the MSE of estimators constructed from system samples, we must reduce estimator variance. Indeed, these two quantities nearly coincide, as MSE can be expressed as a scaled trace of the variance matrix. Specifically,

$$\begin{aligned} \operatorname{MSE}(\widetilde{\Psi}_t^M) &= \mathbb{E}[\|\widetilde{\Psi}_t^M - \mu_t\|_2^2] \\ &= \operatorname{tr} \operatorname{Var}(\widetilde{\Psi}_t^M) = \frac{1}{M} \operatorname{tr} \operatorname{Var}(X_t). \end{aligned} \tag{2.5}$$

Of course, the MSE is reduced as the number of samples M increases, but more stochastic simulation can come at a significant computational cost. Consider, however, our alternative ensemble $X_t^{(r)} \sim X_t$ for $r \in \{1, \ldots, M\}$ such that the sample paths are mutually correlated while still identically distributed. In this case, we may reduce the variance of the estimator Ψ_t^M by ensuring that samples that compose it are negatively correlated. Analytically, the variance of our proposed mean estimator,

$$\Psi_t^M := \frac{1}{M} \sum_{r=1}^{M} X_t^{(r)}, \tag{2.6}$$

can be expressed as

$$\operatorname{MSE}(\Psi_t^M) = \frac{1}{M}\operatorname{tr}\operatorname{Var}(X_t) + \frac{1}{M^2}\sum_{r\neq p}\operatorname{tr}\operatorname{Cov}(X_t^{(r)}, X_t^{(p)}) \tag{2.7}$$

$$= \operatorname{MSE}(\widetilde{\Psi}_t^M) + \frac{1}{M^2}\sum_{r\neq p}\operatorname{tr}\operatorname{Cov}(X_t^{(r)}, X_t^{(p)}). \tag{2.8}$$

Thus if the paths $\{X_t^{(r)}\}_{r=1}^M$ are simulated in such a way to ensure mutual negative correlation between different samples, the MSE of the correlated ensemble estimator will be less than the MSE of the iid estimator. Note that, to generate any sufficiently accurate mean estimator, we may produce multiple iid realizations of anticorrelated collections of M paths $\{X^{(r)}\}_{r=1}^M$. However, we need only analyze estimators produced from a single collection of anticorrelated paths. In particular, for the purposes of antithetic mean estimation, the inclusion of additional antithetic pairs in an ensemble will reduce the error of mean estimates similarly to iid Monte Carlo, since pairs $(X_t^{(r)}, X_t^{(r+1)})$ are iid in $r \in \{1, 3, 5, \dots\}$. This fact is proven in Lemma 2. The challenge, of course, is how to produce anticorrelated paths with correct marginal distributions without any foreknowledge of the particular parameters of a stochastic system in a large class. The key motivation for our approach lies in the random time-change representation of a lattice CTMC and its corresponding lattice DTMC.

Chapter 3

Discrete-time simulation

3.1 Constructing antithetic sample paths for lattice DTMCs

We now define the Markov process samples used in the pathwise mean estimators. Define M iid Monte Carlo sample paths by

$$\widetilde{X}^{(r)} \overset{\text{iid}}{\sim} \widetilde{X}, \quad r \in \{1, \dots, M\} \tag{3.1}$$

so that the random value of the rth sample path at time t is denoted $\widetilde{X}_t^{(r)}$. Here, we explicitly construct antithetic pairs of stochastic paths $X^{(1)}, X^{(2)}$ as our anticorrelated method of choice; for the details in constructing stratified or hybrid ensembles of paths, see [24]. In each case, the analysis of variance reduction ultimately hinges on the value of $\mathrm{Cov}(X_t^{(r_1)}, X_t^{(r_2)})$ for $r_1 \neq r_2$ and for $t > 0$.

To generate anticorrelated sample paths of the Markov chain, several adaptations of classical random variable variance reduction techniques [34] are necessary. Since, in general, the parameter $\lambda^{i,(r)}$ of the Poisson random variable used to simulate the ith reaction channel of the Markov chain at time t depend on the current state $X_t^{(r)}$, the Poisson variables used by different sample paths may have different parameters. To produce unbiased sample paths for the Markov jump process, we produce antithetically paired random inputs (which are Poisson when conditioned on the random state value) as shown in Algorithm 1. Here F_λ is the Poisson CDF with parameter λ and $F_\lambda^{-1}(u) := \inf\{q : F_\lambda(q) \geq u\}$. Using a result shown in [39], this scheme is optimal for pairs of random variables with the same marginal distribution obtained via CDF inversion. While these marginal Poisson samples will not have the same parameters in general, this scheme is still a reasonable choice since the parameters are unknown *a priori*.

3.1.1 Implementation architecture

Implementing Monte Carlo simulations of large-scale stochastic systems often requires consideration of both memory usage and parallelizability. While direct implementation of Algorithm 1 is sufficient for many applications, in some situations it might not be desirable to store multiple trajectory instances simultaneously

Algorithm 1 Constructing antithetic paths for lattice DTMC systems (2.3)

```
**Intialize:** $X_0^{(j)} \leftarrow x_0$
**for** $t = 0$ to $T$ **do**
    **for** $i = 1$ to $I$ **do**
        sample iid $U_t^i \overset{\text{iid}}{\sim} \text{Unif}(0,1)$
        $S_t^{i,(1)} \leftarrow F^{-1}_{\lambda^i(t, X_t^{(1)})}(U_t^i) \sim \text{Pois}(\lambda^i(t, X_t^{(1)}))$
        $S_t^{i,(2)} \leftarrow F^{-1}_{\lambda^i(t, X_t^{(2)})}(1 - U_t^i) \sim \text{Pois}(\lambda^i(t, X_t^{(2)}))$
    **end for**
    **for** $r \in \{1, 2\}$ **do**
        $X_{t+1}^{(r)} \leftarrow X_t^{(r)} + \sum_{i=1}^I S_t^{i,(r)} \nu^i$
    **end for**
**end for**
```

in memory or to utilize only one processor. An important feature of lattice DTMC systems is that, for given system parameters and number of time steps, the number of uniform variates required to simulate a sample path is fixed. As a result, we may reproduce a complete sequence of uniform random numbers, and thus an entire trajectory, by storing or communicating the scalar state or seed of the pseudorandom number generator (PRNG) instead of an entire sequence of random numbers, as shown in Figure 1.2. Furthermore, an antithetic pair of sample trajectories needn't be computed simultaneously. A single processor may compute an antithetic pair of sample paths in series by storing the seed of the PRNG, avoiding the need to simultaneously store two trajectories. Furthermore, an antithetic pair of trajectories can easily be computed on separate processors with minimal communication overhead by simply passing each processor the PRNG seed corresponding to the pair.

3.1.2 Lattice DTMC error quantification

We define the Mean Square Error (MSE) of an estimation method at time t to be

$$\text{MSE}\left(\Psi_t^M\right) = \mathbb{E}\left[\left\|\Psi_t^M - \mathbb{E}[\Psi_t^M]\right\|_2^2\right], \tag{3.2}$$

where $\|\cdot\|_2$ denotes the Euclidean vector norm. As shown above, this estimator error is the trace of its variance,

$$\text{MSE}\left(\Psi_t^M\right) = \mathbb{E}\left[\left\|\Psi_t^M - \mathbb{E}[\Psi_t^M]\right\|_2^2\right] = \text{tr}\,\text{Var}\left(\Psi_t^M\right). \tag{3.3}$$

From this, we define our primary error measure, the pathwise MSE, given by

$$\mathrm{MSE}\left(\Psi^M\right) = \mathbb{E}\left[\left\|\Psi^M - \mathbb{E}[\Psi^M]\right\|^2\right], \tag{3.4}$$

where $\|\cdot\|$ denotes the Frobenius matrix norm. The stepwise MSE is related to the pathwise MSE by

$$\mathrm{MSE}\left(\Psi^M\right) = \sum_{t=0}^{T} \mathrm{MSE}\left(\Psi_t^M\right). \tag{3.5}$$

3.2 Analytical results for pathwise variance reduction algorithms

A simple yet important first result is that the local relaxation of the independence assumption in the law of large numbers does not sacrifice unbiasedness or consistency of a mean estimator. In the sequel, $X^{(r)}$ denotes the rth member of a possibly correlated ensemble and $\{X^{(r),j}\}_{r=1}^{M}$ is the jth iid realization of a complete M-element ensemble. In other words, the sequence of identically distributed sample paths $X^{(r),j} \sim X$, for $r \in \{1, \ldots, M\}$ and $j \in \{1, \ldots, N\}$ has the property that $X^{(r_1),j_1}$ and $X^{(r_2),j_2}$ are independent for $j_1 \neq j_2$ and are not necessarily independent for $j_1 = j_2$. Using such a sequence, we may construct a mean estimator

$$\Psi^{NM} = \frac{1}{NM}\sum_{j=1}^{N}\sum_{r=1}^{M} X^{(r),j} =: \frac{1}{N}\sum_{j=1}^{N} \Psi^{M,j}.$$

Lemma 1. *The mean estimator Ψ^{NM} is unbiased with respect to the lattice DTMC distribution* (2.3) *and, for fixed M, is consistent in N.*

Proof. Unbiasedness follows immediately by

$$\mathbb{E}[\Psi^{NM}] = \frac{1}{NM}\sum_{j=1}^{N}\sum_{r=1}^{M} \mathbb{E}[X^{(r),j}] = \mathbb{E}[X].$$

To prove consistency, note that while the path sequence $\{X^{(r),j},\ r = 1, \ldots, M,\ j = 1, \ldots, N\}$ is not iid, the sequence $\{\Psi^{M,j},\ j = 1, \ldots, N\}$ is iid and $\mathbb{E}[\Psi^{M,j}] = \mathbb{E}[X]$. So by the strong law of large numbers, for any fixed M, $\lim_{N\to\infty} \Psi^{NM} = \mathbb{E}[X]$ almost surely. □

Since such estimators are unbiased, reduction of their variance is tantamount to reduction of their MSE. For estimators formed from anticorrelated collections of samples as above, the number of such collections used to construct an estimator does not affect its performance relative to an iid estimator using the same number of samples, as shown in the following Lemma.

Lemma 2. *For Ψ^{NM} constructed as above,*

$$\frac{\mathrm{MSE}\left(\Psi^{NM}\right)}{\mathrm{MSE}\left(\widetilde{\Psi}^{NM}\right)} = \frac{\mathrm{MSE}\left(\Psi^{M}\right)}{\mathrm{MSE}\left(\widetilde{\Psi}^{M}\right)}.$$

Proof. Again, using the fact that $\{\Psi^{M,j},\, j = 1, \ldots, N\}$ is iid,

$$\frac{\mathrm{MSE}\left(\Psi^{NM}\right)}{\mathrm{MSE}\left(\widetilde{\Psi}^{NM}\right)} = \frac{\mathrm{MSE}\left(\frac{1}{N}\sum_{j=1}^{N}\Psi^{M,j}\right)}{\mathrm{MSE}\left(\frac{1}{N}\sum_{j=1}^{N}\widetilde{\Psi}^{M,j}\right)} = \frac{\frac{1}{N}\,\mathrm{MSE}\left(\Psi^{M,j}\right)}{\frac{1}{N}\,\mathrm{MSE}\left(\widetilde{\Psi}^{M,j}\right)} = \frac{\mathrm{MSE}\left(\Psi^{M}\right)}{\mathrm{MSE}\left(\widetilde{\Psi}^{M}\right)}.$$

□

In particular, we may analyze any antithetic mean estimator by examining only the estimator constructed from a single antithetic pair, $\{X^{(1)}, X^{(2)}\}$. To quantify variance reduction of such estimators, one must first characterize the evolution of the covariance of correlated paths, which can evolve in time in nonlinear, recursive fashion, as shown below.

Theorem 3. *If $X^{(1)}, X^{(2)} \in \mathbb{Z}^D \times \mathbb{N}$ are two realizations that satisfy* (2.3) *and are constructed using Algorithm 1, then their mutual covariance satisfies*

$$\begin{aligned}\mathrm{Cov}(X_{t+1}^{(1)}, X_{t+1}^{(2)}) = {} & \mathrm{Cov}(X_t^{(1)}, X_t^{(2)}) \\ & + \sum_{i=1}^{I} \nu^i \,\mathrm{Cov}(\lambda^i(t, X_t^{(1)}), X_t^{(2)}) + \sum_{i=1}^{I} \mathrm{Cov}(X_t^{(1)}, \lambda^i(t, X_t^{(2)})) {\nu^i}^\top \\ & + \sum_{i_1=1}^{I}\sum_{i_2=1}^{I} \nu^{i_1} {\nu^{i_2}}^\top \mathrm{Cov}(\lambda^{i_1}(t, X_t^{(1)}), \lambda^{i_2}(t, X_t^{(2)})) \\ & + \sum_{i=1}^{I} \nu^i {\nu^i}^\top \mathbb{E}\big[\big(S_t^{i,(1)} - \lambda^i(t, X_t^{(1)})\big) \cdot \big(S_t^{i,(2)} - \lambda^i(t, X_t^{(2)})\big)\big]. \end{aligned} \tag{3.6}$$

Proof. First, consider the simplified case of a single event channel. In this case, the system dynamics are given by

$$X_{t+1} = X_t + S_t \nu.$$

Draw any two anticorrelated sample paths $X^{(1)}$ and $X^{(2)}$, simulated as above, and consider their mutual covariance which can be expanded using the system definition and bilinearity as

$$\mathrm{Cov}(X_{t+1}^{(1)}, X_{t+1}^{(2)}) = \mathrm{Cov}(X_t^{(1)}, X_t^{(2)}) + \nu\,\mathrm{Cov}(S_t^{(1)}, X_t^{(2)}) + \mathrm{Cov}(X_t^{(1)}, S_t^{(2)})\nu^\top + \nu\nu^\top\,\mathrm{Cov}(S_t^{(1)}, S_t^{(2)}).$$

The treatment of the last term is informative for how to simplify the other terms.

$$\begin{aligned}
\mathrm{Cov}(S_t^{(1)}, S_t^{(2)}) &= \mathbb{E}\Big[(S_t^{(1)} - \mathbb{E}[S_t^{(1)}]) \cdot (S_t^{(2)} - \mathbb{E}[S_t^{(2)}])\Big] \\
&= \mathbb{E}\Big[\big(S_t^{(1)} - \lambda(t, X_t^{(1)})\big) \cdot \big(S_t^{(2)} - \lambda(t, X_t^{(2)})\big) \\
&\qquad + \big(S_t^{(1)} - \lambda(t, X_t^{(1)})\big) \cdot \big(\lambda(t, X_t^{(2)}) - \mathbb{E}[S_t^{(2)}]\big) \\
&\qquad + \big(S_t^{(2)} - \lambda(t, X_t^{(2)})\big) \cdot \big(\lambda(t, X_t^{(1)}) - \mathbb{E}[S_t^{(1)}]\big) \\
&\qquad + \big(\lambda(t, X_t^{(1)}) - \mathbb{E}[S_t^{(1)}]\big) \cdot \big(\lambda(t, X_t^{(2)}) - \mathbb{E}[S_t^{(2)}]\big)\Big] \\
&= \mathbb{E}\Big[\big(S_t^{(1)} - \lambda(t, X_t^{(1)})\big) \cdot \big(S_t^{(2)} - \lambda(t, X_t^{(2)})\big)\Big] \\
&\qquad + \mathbb{E}\Big[\mathbb{E}\Big[S_t^{(1)} - \lambda(t, X_t^{(1)}) | X_t^{(1)}, X_t^{(2)}\Big]\big(\lambda(t, X_t^{(2)}) - \mathbb{E}[S_t^{(2)}]\big)\Big] \\
&\qquad + \mathbb{E}\Big[\mathbb{E}\Big[S_t^{(2)} - \lambda(t, X_t^{(2)}) | X_t^{(1)}, X_t^{(2)}\Big]\big(\lambda(t, X_t^{(1)}) - \mathbb{E}[S_t^{(1)}]\big)\Big] \\
&\qquad + \mathrm{Cov}\left(\lambda(t, X_t^{(1)}), \lambda(t, X_t^{(2)})\right) \\
&= \mathbb{E}\Big[\big(S_t^{(1)} - \lambda(t, X_t^{(1)})\big) \cdot \big(S_t^{(2)} - \lambda(t, X_t^{(2)})\big)\Big] \\
&\qquad + \mathrm{Cov}\left(\lambda(t, X_t^{(1)}), \lambda(t, X_t^{(2)})\right),
\end{aligned}$$

where the last equality follows since the conditional expectations in the middle two terms are zero. It is easy to see, again using the law of total expectation, that the first term can only be non-zero when $S_t^{(1)}$ and $S_t^{(2)}$ are associated with the same event channel. Indeed, suppose $i_1 \neq i_2$. Then

$$\begin{aligned}
&\mathbb{E}\Big[\big(S_t^{i_1,(1)} - \lambda^{i_1}(t, X_t^{(1)})\big) \cdot \big(S_t^{i_2,(2)} - \lambda^{i_2}(t, X_t^{(2)})\big)\Big] \\
&\quad = \mathbb{E}\Big[\mathbb{E}\Big[\big(S_t^{i_1,(1)} - \lambda^{i_1}(t, X_t^{(1)})\big) \cdot \big(S_t^{i_2,(2)} - \lambda^{i_2}(t, X_t^{(2)})\big) | X_t^{(1)}, X_t^{(2)}\Big]\Big] \\
&\quad = \mathbb{E}\Big[\mathbb{E}\Big[S_t^{i_1,(1)} - \lambda^{i_1}(t, X_t^{(1)}) | X_t^{(1)}, X_t^{(2)}\Big] \cdot \mathbb{E}\Big[S_t^{i_2,(2)} - \lambda^{i_2}(t, X_t^{(2)}) | X_t^{(1)}, X_t^{(2)}\Big]\Big] \\
&\quad = 0.
\end{aligned}$$

Thus (5.1) follows immediately for the multi event channel case. □

The following result proves a useful connection between the properties of an anticorrelated Poisson variable and the analogous lattice DTMC constructed using Algorithm 1. First, however, we require a lemma regarding the antithetic Poisson sampling used in Algorithm 1.

Lemma 4. *For any $\lambda_1, \lambda_2 \in \mathbb{R}_+$, if $S^{(r)} \sim \mathrm{Pois}(\lambda_\mathrm{r})$, $r \in \{1, 2\}$ are simulated using the antithetic sampling technique used in steps 3 and 4 of Algorithm 1, then* $\mathrm{Cov}(S^{(1)}, S^{(2)}) \leq 0$.

Proof. By construction, for $U \sim \text{Unif}[0,1]$,

$$\text{Cov}(S^{(1)}, S^{(2)}) = \text{Cov}(F_{\lambda_1}^{-1}(U), F_{\lambda_2}^{-1}(1-U)) \tag{3.7}$$

$$= -\text{Cov}(F_{\lambda_1}^{-1}(U), -F_{\lambda_2}^{-1}(1-U)) \leq 0 \tag{3.8}$$

due to a result proven in [39]. There, the crucial fact is that $F_{\lambda_1}^{-1}(U)$ and $-F_{\lambda_2}^{-1}(1-U)$ are both non-decreasing functions of U for any $\lambda_1, \lambda_2 \geq 0$. □

Using Lemma 4, we prove a result that guarantees the non-positivity of the final term of (5.1).

Theorem 5. *Suppose that $X^{(1)}, X^{(2)}$ are two realizations simulated using Algorithm 1. Then, for any event channel i and for each time $t \geq 0$,*

$$\mathbb{E}\big[\big(S_t^{i,(1)} - \lambda^i(t, X_t^{(1)})\big) \cdot \big(S_t^{i,(2)} - \lambda^i(t, X_t^{(2)})\big)\big] \leq 0. \tag{3.9}$$

Proof. Suppose $S^{(1)}(t,x), S^{(2)}(t,x) \sim \text{Pois}(\lambda(t,x))$ are simulated using such an anticorrelated technique. Fix any $x_1, x_2 \in \mathbb{R}^D$, and, by Lemma 4,

$$\begin{aligned}0 &\geq \text{Cov}(S^{(1)}(t,x_1), S^{(2)}(t,x_2)) \\ &= \mathbb{E}\big[\big(S^{(1)}(t,x_1) - \lambda(t,x_1)\big) \cdot \big(S^{(2)}(t,x_2) - \lambda(t,x_2)\big)\big].\end{aligned}$$

Since this is true for any $x_i \in \mathbb{R}^D$, it is necessarily true that

$$0 \geq \mathbb{E}\big[\big(S_t^{i,(1)} - \lambda^i(t, X_t^{(1)})\big) \cdot \big(S_{r_2,t}^{i,(2)} - \lambda^i(t, X_t^{(2)})\big) | X_t^{(1)}, X_t^{(2)}\big], \tag{3.10}$$

almost surely, since this is exactly the same integral (in U) for given $X_t^{(1)}, X_t^{(2)}$ random. Taking expectation of both sides, we get (3.9). □

We remark here that, as noted above, anticorrelated sampling schemes other than antithetic sampling may be used in the framework of Algorithm 1, such as stratified or hybrid antithetic/stratified sampling [24]. To prove a result equivalent to Theorem 5 for these methods, we need only prove a corresponding version of Lemma 4 for any $X^{(r_1)}, X^{(r_2)}$ for $r_1, r_2 \in \{1, \ldots, M\}$.

The following corollary refines the previous results when more is required of the intensity functions $\lambda^i(t, X_t)$ beyond nonnegativity, namely that they be affine in the state X_t. The first condition can be used to greatly simplify (5.1).

Corollary 1. *Suppose that*

$$\lambda^i(t, X_t) = \left(a^i(t) + {\kappa^i}^\top X_t\right)\tau. \tag{3.11}$$

If the conditions of Theorem 3 are satisfied, then the following recursion is satisfied

$$\begin{aligned}\operatorname{Cov}(X_{t+1}^{(1)}, X_{t+1}^{(2)}) = \operatorname{Cov}(X_t^{(1)}, X_t^{(2)})& \\ &+ \sum_{i=1}^{I} \tau \nu^i {\kappa^i}^\top \operatorname{Cov}(X_t^{(1)}, X_t^{(2)}) + \sum_{i=1}^{I} \tau \operatorname{Cov}(X_t^{(1)}, X_t^{(2)}) \kappa^i {\nu^i}^\top \\ &+ \sum_{i_1=1}^{I} \sum_{i_2=1}^{I} \tau^2 \nu^{i_1} {\kappa^{i_1}}^\top \operatorname{Cov}(X_t^{(1)}, X_t^{(2)}) \kappa^{i_2} {\nu^{i_2}}^\top \\ &+ \sum_{i=1}^{I} \nu^i {\nu^i}^\top \mathbb{E}\big[\big(S_t^{i,(1)} - \lambda^i(t, X_t^{(1)})\big) \cdot \big(S_t^{i,(2)} - \lambda^i(t, X_t^{(2)})\big)\big].\end{aligned} \tag{3.12}$$

This expression can be more compactly written as

$$\operatorname{Cov}(X_{t+1}^{(1)}, X_{t+1}^{(2)}) = \mathcal{L}(\operatorname{Cov}(X_t^{(1)}, X_t^{(2)})) + \sum_{i=1}^{I} c_t^i \nu^i {\nu^i}^\top,$$

where $\mathcal{L}$ is a time invariant linear operator on the space of symmetric matrices and

$$c_t^i := \mathbb{E}\big[\big(S_t^{i,(1)} - \lambda^i(t, X_t^{(1)})\big) \cdot \big(S_t^{i,(2)} - \lambda^i(t, X_t^{(2)})\big)\big], \quad t \in \{1, \dots, T\}$$

are sequences of reals that depend on x_0 and $\{\lambda^i\}_{i=1}^I$. If, in addition, the conditions of Theorem 5 are satistfied, then $c_t^i \leq 0$ for every i and t.

In the affine rates case, we can derive a sufficient, testable condition for ensemble variance reduction of our algorithm. First, we require an algebraic proposition regarding matrix invariance in a half-space. Consider the set S_D of $D \times D$ symmetric matrices as a vector space together with the field $\mathbb{R}$ and the Frobenius inner product $\langle A, B\rangle := \operatorname{tr} AB^\top$. Define the half-space $\mathcal{H}_- := \{A \in S_D : \langle A, I_D\rangle \leq 0\}$. Define the cone $\mathcal{R} := \{\sum_{i=1}^I c^i \nu^i {\nu^i}^\top : c^i \leq 0 \text{ for each } i\} \subset \mathcal{H}_-$, and consider the following sufficient condition.

Proposition 1. *Suppose*

(i) the sequence $A_t \in S_D$ evolves according to

$$A_{t+1} = \mathcal{L}(A_t) + \sum_{i=1}^{I} c_t^i \nu^i {\nu^i}^\top, \tag{3.13}$$

where $\mathcal{L}$, c_t^i and ν^i are all defined as in Corollary 1, and $A_0 = 0$;

(ii) that for any $R \in \mathcal{R}$ and for every $t \geq 1$, $\mathcal{L}^t(R) \in \mathcal{H}_-$.

Then $A_t \in \mathcal{H}_-$ for every $t \geq 0$.

Proof. Define $R_t := \sum_{i=1}^{I} c_t^i \nu^i {\nu^i}^\top$. Then, for each $t \geq 0$, $R_t \in \mathcal{R}$. For each $t \geq 1$, a solution to (3.12) is given by

$$A_t := \sum_{\ell=0}^{t-1} \mathcal{L}^{t-\ell-1}(R_\ell). \tag{3.14}$$

Since $\mathcal{L}^{t-\ell-1}(R_\ell) \in \mathcal{H}_-$ and since the half-space $\mathcal{H}_-$ is closed under addition, $A_t \in \mathcal{H}_-$ for every $t \geq 0$. □

The final corollary shows that the above conditions are sufficient to prove the dominance of the antithetic estimator Ψ^M over the iid estimator $\widetilde{\Psi}^M$ in this affine rates setting.

Corollary 2. *Suppose that $X^{(r)}$ satsify the conditions of Corollary 1, that $\mathcal{L}$ satisfies the conditions of Proposition 1, and that* $\mathrm{Cov}(X_0^{(1)}, X_0^{(2)}) = 0$. *Then*

$$\mathrm{MSE}(\Psi^M) \leq \mathrm{MSE}(\widetilde{\Psi}^M). \tag{3.15}$$

Proof. It is easy to see that $\operatorname{tr}\mathrm{Cov}(\Psi_t^M) \leq \operatorname{tr}\mathrm{Cov}(\widetilde{\Psi}_t^M)$ for each $t \geq 0$ if $\operatorname{tr}\mathrm{Cov}(X_t^{(r_1)}, X_t^{(r_2)}) \leq 0$ for each $r_1 \neq r_2 \in \{1, \ldots, M\}$ and $t \geq 0$. Since the evolution equation (3.12) of $\mathrm{Cov}(X_t^{(1)}, X_t^{(2)})$ is identical to (3.13), we have by Proposition 1 that $\mathrm{Cov}(X_t^{(1)}, X_t^{(2)}) \in \mathcal{H}_-$ for every $t \geq 0$. That is, $\operatorname{tr}\mathrm{Cov}(X_t^{(r_1)}, X_t^{(r_2)}) \leq 0$ for every $t \geq 0$. Thus $\operatorname{tr}\mathrm{Cov}(\Psi_t^M) \leq \operatorname{tr}\mathrm{Cov}(\widetilde{\Psi}_t^M)$ for each $t \geq 0$, and the claim holds by (2.8). □

3.3 Numerical results for lattice DTMC samplers

We now introduce three example stochastic systems for numerical study of Algorithm 1, that illustrate the above analytical results and provide further intuition for its efficacy in more general cases. These systems are drawn from the literature and have been specifically chosen to exhibit increasingly complex rate functions. The first is a simple model of gene expression which appears in [8], modeling the production and decay of mRNA and protein molecules. This system has rate functions which are affine in the state variables, and thus corresponds to Corollary 1, and we prove that it satisfies the sufficient conditions of Corollary 2. The second system is a simplified model of coagulation of water molecules via gravitational settling. As presented in [36], the system is composed of two sizes of water molecules, large and small, where coagulations within sizes are rare but between sizes are frequent and are specified by a nonlinear propensity function. Finally, we present a seven-dimensional model of HIV infection with 19 reaction channels and nonlinear rate functions, as found in [6]. The latter two systems are chosen because their complexity extends beyond the scope of our analytical results. While we cannot yet provide analytical guarantees of improved performance using anticorrelated simulation for such models, the numerical results do demonstrate significant computational savings for each system. Thus, we provide promising evidence of the wider applicability of variance-reduced tau-leaping. First, we briefly introduce each of the three systems in Subsections 3.3.1, 3.3.2 and 3.3.3, then we study the performance of Algorithm 1 in all three settings via a parametric study of the scale of each system in Subsection 3.3.4.

3.3.1 Affine gene expression system

Consider a simple gene expression system, where mRNA is produced and decays, and it produces a protein which also decays. This simple model is quite commonly studied, the specific formulation and parameter values appear here as in [8], and are taken to be unitless. That is,

$$\begin{aligned}
\emptyset &\xrightarrow{k_r} \text{mRNA}\\
\text{mRNA} &\xrightarrow{\gamma_r} \emptyset\\
\text{mRNA} &\xrightarrow{k_p} \text{protein} + \text{mRNA}\\
\text{protein} &\xrightarrow{\gamma_p} \emptyset.
\end{aligned}$$

Define the state of the system to be number of mRNA and protein particles, respectively, as a vector $X \in \mathbb{Z}^2$, with initial condition $X_0 = V \cdot [1.0\ 0.5]^\top$ (where V is a system volume scaling parameter) with $I = 4$ reaction channels, given by:

$$\begin{aligned}
\nu^1 &= [1\ 0]^\top & a^1(X_t) &= k_r V\\
\nu^2 &= [-1\ 0]^\top & a^2(X_t) &= \gamma_r X_{t,1}\\
\nu^3 &= [0\ 1]^\top & a^3(X_t) &= k_p X_{t,1}\\
\nu^4 &= [0\ \ -1]^\top & a^4(X_t) &= \gamma_p X_{t,2},
\end{aligned}$$

for $k_r, \gamma_r, k_p, \gamma_p > 0$, and where $X_{t,d}$ denotes the dth component of the state vector at time t. The corresponding discrete time approximation is simulated using the tau-leaping approximation (6.52) with $\tau = 1$ (i.e. $\lambda^i = a^i$) and run from time $t = 0$ to time $t = T = 100$. A pair of antithetic sample trajectories are shown in Figure 3.1. Our primary interest is the normalized estimator MSE for the gene expression estimator, shown in Figure 3.3; this shows the degree of MSE reduction of antithetic simulation compared to iid simulation over a large range of system scales. This behavior will be discussed in detail in 3.3.4. As mentioned above, the number of antithetic pairs used in the variance reduced mean estimator is irrelevant for comparison to iid estimators using an equal number of sample paths, since both decay at the same rate as proven in Lemma 2, so we consider mean estimators composed of a pair of antithetic paths.

Note that this system satisfies the conditions of Corollary 1 and Proposition 1. Indeed, the rate functions are affine in the state variables, and, with respect to (3.11),

$$\begin{aligned}\kappa^1 &= [0\ 0]^\top\\ \kappa^2 &= [\gamma_r\ 0]^\top\\ \kappa^3 &= [k_p\ 0]^\top\\ \kappa^4 &= [0\ \gamma_p]^\top.\end{aligned}$$

It is easy to verify then that, if we vectorize the 2×2 covariance matrix objects as vectors in $\mathbb{R}^4$, then we may identify the linear operator $\mathcal{L}$ with left multiplication by a matrix $L \in \mathbb{R}^{4\times 4}$ given by

$$L = \begin{pmatrix} (1-\gamma_r\tau)^2 & 0 & 0 & 0 \\ k_p\tau(1-\gamma_r\tau) & (1-\gamma_r\tau)(1-\gamma_p\tau) & 0 & 0 \\ k_p\tau(1-\gamma_r\tau) & 0 & (1-\gamma_r\tau)(1-\gamma_p\tau) & 0 \\ k_p^2\tau^2 & k_p\tau(1-\gamma_p\tau) & k_p\tau(1-\gamma_p\tau) & (1-\gamma_p\tau)^2 \end{pmatrix},$$

a lower triangular matrix. Furthermore, the negative cone $\mathcal{R}$ can be identified with the set

$$\{(c^1,\ 0,\ 0,\ c^2)^\top : c^1, c^2 \leq 0\} \subset \mathbb{R}^4, \tag{3.16}$$

so for any $R \in \mathcal{R}$, and for any $t \geq 1$, $\mathcal{L}^t(R)$ can be identified with $L^t \operatorname{vec}(R)$, which, if $\gamma_r\tau \leq 1$ and $\gamma_p\tau \leq 1$, is the product of a lower triangular matrix with non-negative entries and the standard vectorization of an element of $\mathcal{R}$. Therefore, the first and fourth components of the product will have the form

$$\begin{aligned}\left(L^t \operatorname{vec}(R)\right)_1 &= \left(L^t\right)_{1,1} c^1 && \leq 0\\ \left(L^t \operatorname{vec}(R)\right)_4 &= \left(L^t\right)_{4,1} c^1 + \left(L^t\right)_{4,4} c^2 && \leq 0\end{aligned}$$

for every $t \geq 1$, where subscripts are used here to denote vector and matrix components. So $\mathcal{L}^t(R) \in \mathcal{H}_-$ for every $t \geq 1$, and thus $\operatorname{tr}\operatorname{Cov}(X_t^{(1)}, X_t^{(2)}) \leq 0$ for every $t \geq 0$, as long as $\gamma_r\tau, \gamma_p\tau \leq 1$. Thus the antithetic

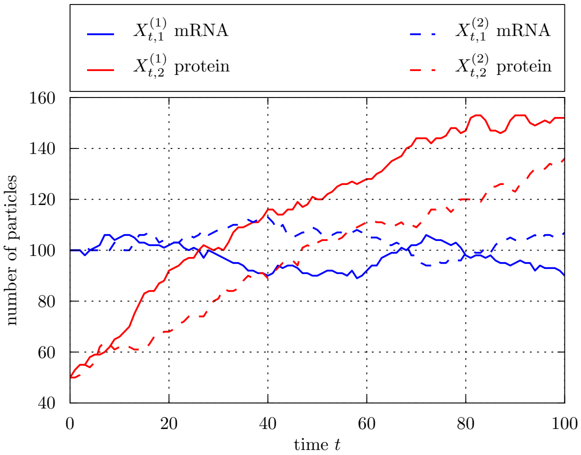

Figure 3.1: Antithetic pair of sample path trajectories of the gene expression system using timestep $\tau = 1$ and rate parameters $(k_r, \gamma_r, k_p, \gamma_p) = (0.01, 0.03, 0.06, 0.0066)$ and initial condition $X_0 = [100\ 50]^\top$ (i.e. volume parameter $V = 100$) plotted versus time.

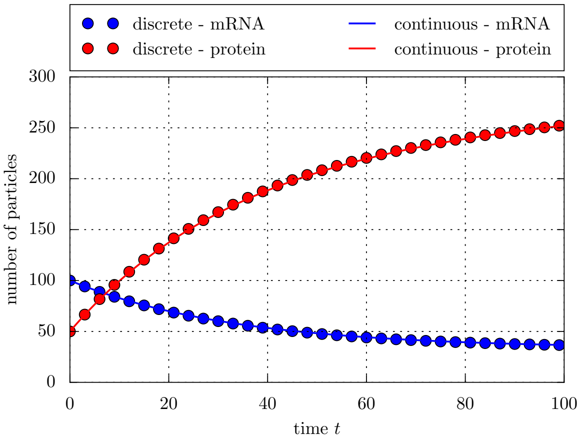

Figure 3.2: Mean trajectory of the lattice CTMC gene expression system obtained by directly solving the master equation and the mean trajectory of the lattice DTMC system obtained using Monte Carlo simulation plotted versus time. Both systems used parameters timestep $\tau = 1$, rate parameters $(k_r, \gamma_r, k_p, \gamma_p) = (0.01, 0.03, 0.06, 0.0066)$ and initial condition $X_0 = [100\ 50]^\top$ (i.e. volume parameter $V = 100$).

mean estimator is a dominant mean estimator for the lattice DTMC distribution.

Because the propensity functions in this example are affine with respect to the state, the exact mean evolution of the corresponding continuous-time system is obtainable. Following the approach used in [13], the mean evolution of the lattice CTMC corresponding to the affine gene expression system is given by the solution to:

$$\dot{m}(t) = \nu A m(t) + \nu B, \qquad m(0) = X_0 \tag{3.17}$$

where $m(t) := \mathbb{E}[X(t)]$, $\nu = [\nu^1\ \nu^2\ \nu^3\ \nu^4] \in \mathbb{R}^{2\times 4}$, and, using the propensities defined above, $a(X_t) := [a^i(X_t)]_{i=1}^4 = AX_t + B$ for $A \in \mathbb{R}^{4\times 2}$ and $A \in \mathbb{R}^{4\times 1}$. The solution of the above ordinary differential equation is

$$m(t) = e^{\nu A t} m(0) + \int_0^t e^{\nu A(t-\tau)} \nu B \, \mathrm{d}\tau,$$

and it is plotted together with the mean of the discrete-time version obtained via Monte Carlo simulation in Fig. 3.2. Our antithetic mean estimates are unbiased with respect to the discrete-time distribution, which in turn is biased with respect to the continuous-time distribution due to the tau-leaping approximation, though this bias is small for the values of τ and V shown in the plot.

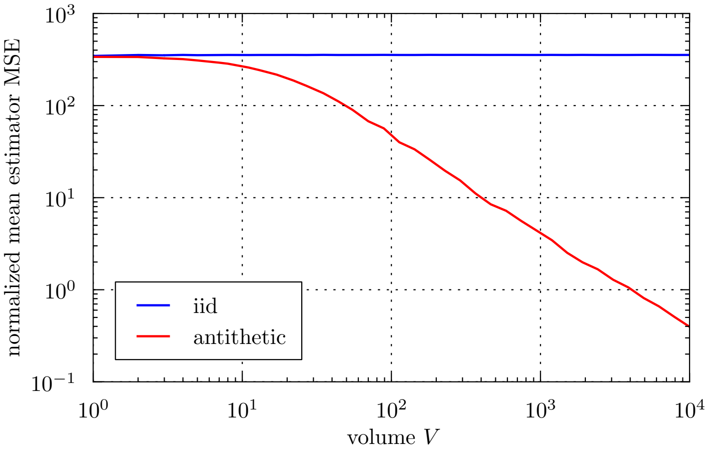

Figure 3.3: Normalized pathwise $\mathrm{MSE}(\Psi^M)M/V$ of an $M = 4$ sample estimator of the mean $\mathbb{E}[X_t]$ of the gene expression system using the iid and antithetic sampling techniques plotted versus volume scale V, for timestep $\tau = 1$s, rate parameters $(k_r, \gamma_r, k_p, \gamma_p) = (0.01, 0.03, 0.06, 0.0066)$, initial condition $X_0 = [V\ V/2]^\top$ and timesteps from $t = 0$ to $t = 100$. Pathwise MSE calculated from 10^6 sample paths, error bars omitted.

3.3.2 Nonlinear coagulation via gravitational settling

Following the treatment in [36], Chapter 13, consider a system of water particles falling in the atmosphere under the influence of gravity. The system falls in a control volume and is made up of two classes of particles, large and small. The system evolves via the coagulation of a large particle and a small particle (the coagulation rate is driven by differences in terminal settling velocity, so particles of similar size are unlikely to coagulate) or by the emission of new small particles into the volume V. We may specify the state of the system as (N_s, M_s, N_l, M_l), or the number of small particles, total mass of small particles, the number of large particles, and the total mass of the large particles, respectively. The probability rate at which a single small particle coagulates with a single large particle in volume V is given by (13.A.4) in [36]:

$$K_{sl}^{\mathrm{GS}} = \frac{\pi}{4}\frac{1}{V}(D_l + D_s)^2|v_l - v_s|$$

where D_s and D_l are the diameters of the small and large particles, respectively, and v_s and v_l are the terminal settling velocities of the small and large particles, respectively. For simplicity, we take the collision efficiency to be 1. By Stokes' Law,

$$v_s = \frac{1}{18}\frac{(\rho_p - \rho_f)}{\mu} g {D_s}^2,$$

where ρ_p is the density of the particle, ρ_f is the density of the fluid, μ is the viscosity of the fluid, and g is the acceleration due to gravity. A similar equation holds for the terminal settling velocity v_l of the large particles. For simplicity, we consider the case where the number of large particles, N_l scales directly with the

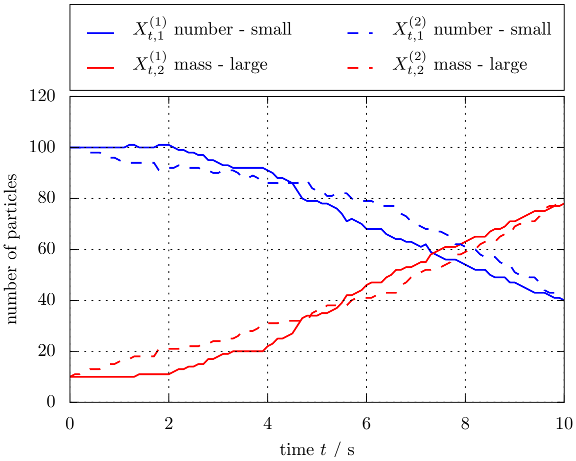

Figure 3.4: Antithetically paired sample trajectories for the nonlinear coagulation system with small particle mass $m = 1$, proportionality constant $\alpha = 5 \cdot 10^{-4}$ into a control volume with $V = 1$, using timestep $\tau = 0.1$ s, and timesteps from $t = 0$ to $T = 100$.

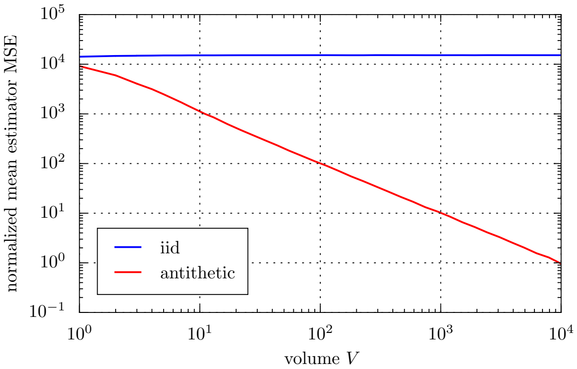

Figure 3.5: Normalized pathwise $\mathrm{MSE}(\Psi^M)M/V$ of an $M = 4$ sample mean estimator for the nonlinear coagulation system versus source area from $V = 10^0$ to $V = 10^4$. Computation of pathwise MSE from 10^6 simulations, where each simulation uses timestep $\tau = 0.1$ s, and timesteps from $t = 0$ to $t = 100$. Error bars are small and are thus omitted.

volume V and where the mass of each small particle is uniformly fixed as m. The state of the system then becomes (N_s, mN_s, V, M_l), and we need only track the smaller state $X = (N_s, M_l)$. Simplifying the master equation (13.81) in [36], and using the fact that $D_s \propto \sqrt[3]{m}$ and $D_l \propto \sqrt[3]{\frac{M_l}{N_l}}$, we can specify the reaction channels of this system by:

$$\nu^1 = [m\ 0]^\top \qquad a^1(X_t) = V$$

$$\nu^2 = [-m\ m]^\top \qquad a^2(X_t) = \alpha K_{sl}^{\mathrm{GS}} V X_{t,1}$$

where α is a proportionality constant and, for simplicity

$$K_{sl}^{\mathrm{GS}} = \frac{1}{V}\left(\sqrt[3]{X_{t,2}/V} + \sqrt[3]{m}\right)^3 \left(\sqrt[3]{X_{t,2}/V} - \sqrt[3]{m}\right).$$

The state is initialized from $X_0 = V \cdot [100, 10]$ and a corresponding lattice DTMC system is obtained via the tau-leaping approximation (6.52) for $\tau = 0.1$s, and is simulated for 10s from timestep $t = 0$ to timestep $T = 100$. We take $\alpha = 5 \cdot 10^{-4}$ and for simplicity we take $m = 1$ so that the state $X_t \in \mathbb{Z}^2$. Concentration sample trajectories are shown in Figure 3.4. The estimator MSE of the coagulation system plotted versus the system scale parameter V shown in Figure 3.5. This behavior will be discussed in in 3.3.4.

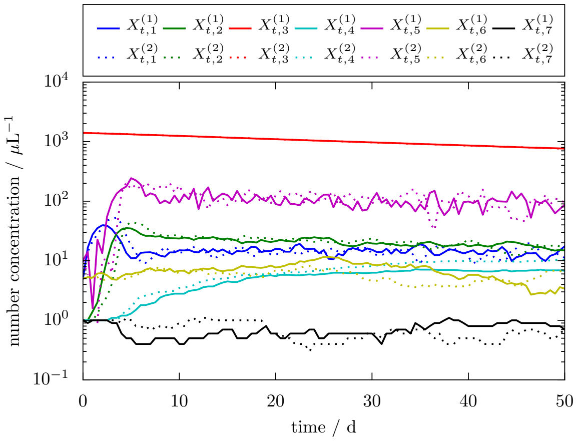

Figure 3.6: Antithetic pair of sample trajectories of cell concentration for the HIV infection system for volume $V = 10$ μL, timestep $\tau = 0.005$ days, and timesteps from $t = 0$ to $t = 10\,000$ plotted versus time. All other parameter values taken from [6].

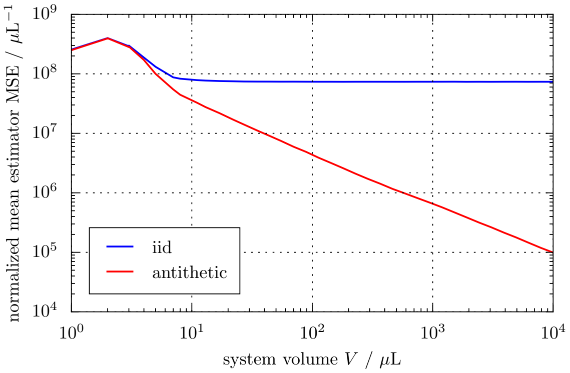

Figure 3.7: Normalized pathwise $\mathrm{MSE}(\Psi^M)M/V$ of an $M = 4$ sample mean estimator for the expected path $\mathbb{E}[X_t]$ of the HIV infection system plotted versus system volume V from 1 μL to $10\,000$ μL. Simulations use a timestep $\tau = 0.005$ days, and timestep from $t = 0$ to $t = 10\,000$. All other parameter values taken from [6]. Pathwise MSE computed using 10^4 estimator samples, error bars omitted.

3.3.3 Nonlinear HIV infection system

Following [6], consider a model for HIV infection with state $X \in \mathbb{R}^7$ representing concentrations of uninfected and infected activated CD4+ T-cells ($X_{t,1}$ and $X_{t,2}$, respectively), uninfected and infected resting CD4+ T-cells ($X_{t,3}$ and $X_{t,4}$, respectively), infectious free virus ($X_{t,5}$), and HIV-specific effector and memory CD8+ T-cells ($X_{t,6}$ and $X_{t,7}$, respectively). The state is initialized from $X_0 = V \cdot [5\ 1\ 1400\ 1\ 10\ 5\ 1]^\top$ with $I = 19$ reaction channels with nonlinear rates [6], save that we maintain the scaling parameter V, the volume of the system. This system is discretized with the tau-leaping approximation (6.52) with $\tau = 0.005$ days and simulated from timestep $t = 0$ to time $t = T = 10\,000$. Sample trajectories are shown in Figure 3.6 and normalized estimator MSE is shown in Figure 3.7.

3.3.4 Parameter variations

Figures 3.3, 3.5 and 3.7 illustrate the dependence of estimator error on a parameter that governs the number of particles of each system. Each figure plots a pathwise estimator's MSE divided by a scaling parameter V that governs the "volume" (number of computational "particles" in a typical simulation), versus the same scaling parameter. This parameter controls the "speed" of the system, i.e., the number of reactions that occur at each timestep; equivalently it governs the typical mean Poisson parameter sampled at each timestep. Each mean estimator is constructed using $M = 4$ samples, either 4 iid samples or 2 antithetic pairs to ensure a fair comparison. It is easy to see that using this normalization removes from consideration any reduction of MSE gained by generating more iid samples or pairs of samples for sufficiently large-scale systems, and thus there is no need to draw any more samples than a small M for comparison.

The volume scale V is roughly proportional to the typical Poisson parameter used to generate random samples, though in nonlinear cases such as the HIV system, this relationship will also tend to be nonlinear, as we observe quite clearly for the small volume region of Figure 3.7. Since the variance of a mean estimator scales with the variance of an individual sample path, the cost of iid Monte Carlo increases roughly linearly for large V without bound (or remains constant if simulations are normalized as concentrations). So increasing V corresponds to both increased resolution of simulations as well as increased cost. However, as V becomes large and reactions occur more frequently, we expect greater relative gains in MSE for antithetic sampling by analogy to the Poisson variable case [23]. Indeed, we observe antithetic MSE that is nearly constant even for large V (or, in the particle concentration (normalized) setting, linearly inversely proportional to V), as opposed to linear growth for the iid estimator (or constant behavior when normalized). Thus when many jumps are typically observed in each timestep, the gains produced via antithetic simulation are most dramatic. It is worth noting that this is precisely the same operating regime that maximizes the desirability of tau-leaping over exact continuous-time simulation, via say SSA [19].

Figures 3.3, 3.5 and 3.7, each show that the antithetic mean estimator has equal or lower MSE than the iid mean estimator for all observed values of V. Note that the affine gene expression estimator satisfies the sufficient conditions of Corollary 2 as shown above, and thus provably dominates the iid Monte Carlo mean estimator. In this case, we also know that a study of any other parameter and range will show that MSE is lower for the antithetic estimator. However, the nonlinear coagulation and HIV infection estimators also produce greatly reduced MSE over observed values of V, despite having nonlinear state-dependent propensity functions which make potential proof of dominance more difficult. These numerical results show that the affine conditions, while sufficient, are certainly not necessary to produce non-increased pathwise MSE over at least this parameter range. Thus we posit that these variance reduction techniques can be effective in a much larger class of models, even if analytical proof is not yet available.

Chapter 4

Continuous-time simulation

4.1 Anticorrelated unit-rate Poisson processes

We begin with the problem of simulating antithetic pairs of unit-rate Poisson processes. We will proceed by introducing two algorithms to achieve this goal, and provide some theoretical and numerical analysis of their behavior and performance. In Section 4.2, we will apply these algorithms to the problem of simulating lattice CTMCs.

4.1.1 Endpoint method for simulating antithetic Poisson processes

The inspiration for the algorithms we have created to simulate antithetic Poisson process paths is the classical technique to sample antithetic pairs of scalar random variables. This generates two Poisson-distributed random samples (X_1, X_2) with

$$\begin{aligned} X_1 &:= F_\tau^{-1}(U) \\ X_2 &:= F_\tau^{-1}(1-U), \end{aligned}$$

where $U \sim \mathrm{Unif}[0,1]$ is a uniform random variable and F_τ^{-1} is the formal inverse of the Poisson CDF with parameter τ. It's easy to show that $\mathrm{Cov}\,(X_1, X_2) \leq 0$ for all τ [25]. We will denote an antithetic draw from this distribution by $(X_1, X_2) \overset{\text{anti}}{\sim} \mathrm{Pois}(\tau)$. Recall that the distribution of the increment $N(s,t)$ of a unit-rate Poisson process Y over an interval $[s,t]$ is given by $N(s,t) := Y(t) - Y(s) \sim \mathrm{Pois}(t-s)$, and is independent of increments of the same Poisson process over other, disjoint intervals. Further, recall that its arrival times conditioned on its value at the endpoints of any interval are uniformly distributed throughout that interval.

The first algorithm, the endpoint technique, produces an antithetic pair of Poisson process paths (Y^1, Y^2) over an interval $[0, \tau_\mathrm{f}]$. One way to specify a pair of continuous-time Poisson process trajectories is to simulate the sequences $\left(\mathcal{A}^1_{\tau_\mathrm{f}}, \mathcal{A}^2_{\tau_\mathrm{f}}\right)$ of their arrival times. We achieve this as follows. For some step-increment $\tau_\mathrm{s} > 0$, sample $\left(Y^1(\tau_\mathrm{s}), Y^2(\tau_\mathrm{s})\right) = \left(N^1_\mathrm{E}(0,\tau_\mathrm{s}), N^2_\mathrm{E}(0,\tau_\mathrm{s})\right) \overset{\text{anti}}{\sim} \mathrm{Pois}(\tau_\mathrm{s})$, the state of the pair of Poisson processes at time τ_s as an antithetic pair of Poisson random variables. Here we denote the increments of process Y^j constructed using the endpoint technique from time s to time t with $N_E^j(s,t)$. Then, we can sample the state of the processes at time $2\tau_s$ by sampling the next process increment as an independent antithetic pair, $\left(N_E^1(\tau_s, 2\tau_s), N_E^2(\tau_s, 2\tau_s)\right) \overset{\text{anti}}{\sim} \text{Pois}(\tau_s)$, and setting $Y^i(2\tau_s) = N_E^i(0,\tau_s) + N_E^i(\tau_s, 2\tau_s)$. We may proceed until we have the sampled values $\{Y^i(\tau_s), Y^i(2\tau_s), \dots, Y^i((N+1)\tau_s)\}_{i=1}^2$ of the state of the antithetic pair of paths evaluated at multiples of τ_s, where $N := \lfloor \tau_f/\tau_s \rfloor$ is the total number of full steps. We can then finish simulating the paths by sampling the jump times, which are iid uniform random variables. That is, within each interval $(n\tau_s, (n+1)\tau_s]$, we know that $N_E^i(n\tau_s, (n+1)\tau_s)$ jumps occurred, and that each jump is uniformly distributed over the interval. More formally, $t_{j,n}^i \overset{\text{i.i.d.}}{\sim} \text{Unif}(n\tau_s, (n+1)\tau_s)$ for $j = 1, \dots, N_E^i(n\tau_s, (n+1)\tau_s)$ and for $n = 0, \dots, N$. Finally, we re-index the jump times in j such that they are sorted in increasing order and we discard any arrival times that lie outside $[0, \tau_f]$. The Endpoint technique is summarized in Algorithm 2.

Algorithm 2 Endpoint Method: Antithetic Poisson process paths via concatenation of step size τ_s

```
Intialize:  $n \leftarrow 0$, $\left(\mathcal{A}^1_{\tau_f}, \mathcal{A}^2_{\tau_f}\right) \leftarrow (\emptyset, \emptyset)$
while $n\tau_s < \tau_f$ do
  Sample antithetic Poisson random variables: $\left(N_E^1(0,\tau_s), N_E^2(0,\tau_s)\right) \overset{\text{anti}}{\sim} \text{Pois}(\tau_s)$
  for $j = 1, \dots, N_E^i(0,\tau_s)$ do
    Simulate iid jump times in the interval: $t_{j,n}^i \overset{\text{i.i.d.}}{\sim} \text{Unif}(0,\tau_s)$, $i \in \{1,2\}$
  end for
  Sort and append arrival times: $\mathcal{A}^i_{\tau_f} \leftarrow \mathcal{A}^i_{\tau_f} \cup \text{sort}\left(\left\{n\tau_s + t_{j,n}^i\right\}_{j=1}^{N_E^i}\right)$, $i \in \{1,2\}$
  $n \leftarrow n+1$
end while
$\mathcal{A}^i_{\tau_f} \leftarrow \mathcal{A}^i_{\tau_f} \cap [0,\tau_f]$, $i \in \{1,2\}$
return  $\left(\mathcal{A}^1_{\tau_f}, \mathcal{A}^2_{\tau_f}\right)$
```

The reason we apply antithetic sampling to the state values of $Y(t)$ instead of the jump times themselves, is that we are ultimately concerned with quantities like Cov $\left(Y^1(t), Y^2(t)\right)$, and producing anticorrelation in the state space is significantly more effective than indirectly anticorrelating via the time dimension. While the anticorrelation is only injected at sample points $\{\tau_s, 2\tau_s, \dots\}$ and jump times are simulated iid, we will show in 4.1.3 that negative correlation is felt throughout the time domain of the process, not just at the antithetic sample points.

While a more complete and precise analysis is provided in Section 4.1.3, we attempt here to provide some intuition regarding the performance and limitations of the endpoint method, in order to motivate the development of the other algorithm we will present in this work, the binomial midpoint method.

First, we define a useful performance metric. Since all of the mean estimators we construct in this work are unbiased, we define the scaled mean square error (MSE) of a pathwise mean estimator δ at time t to be

$$\mathrm{MSE}(t) := N\,\mathrm{Var}\left(\delta(t)\right) = N\,\mathrm{Var}\left(\frac{1}{N}\sum_{i=1}^{N} Y^i(t)\right), \tag{4.1}$$

where N is the number of sample paths used to produce the mean estimate. This quantity is of interest both because it is invariant to the inclusion of additional iid paths (or pairs of antithetic paths, as the case may be) and because it cleanly relates to other quantities of interest. For example, for any mean estimator

$$\delta_{2M}(t) = \frac{1}{2M}\sum_{i=1}^{M}\left[Y^{1,i}(t) + Y^{2,i}(t)\right], \tag{4.2}$$

where the pairs $(Y^{1,i}, Y^{2,i})$ are iid in i but their elements could be correlated. Then

$$\begin{aligned}
\mathrm{MSE}_{\delta_{2M}}(t) &= 2M\,\mathrm{Var}\left(\frac{1}{2M}\sum_{i=1}^{M}\left[Y^{1,i}(t) + Y^{2,i}(t)\right]\right)\\
&= \frac{2}{M}\sum_{i=1}^{M}\mathrm{Var}\left(\frac{Y^{1,i}(t) + Y^{2,i}(t)}{2}\right)\\
&= 2\,\mathrm{Var}\left(\delta_2(t)\right)\\
&= \mathrm{MSE}_{\delta_2}(t)\\
&= \mathrm{Var}\left(Y^1(t)\right) + \mathrm{Cov}\left(Y^1(t), Y^2(t)\right)\\
&= t + \mathrm{Cov}\left(Y^1(t), Y^2(t)\right).
\end{aligned}$$

In particular, the scaled MSE at time t of a mean estimator constructed from any number of iid sample paths is simply $\mathrm{MSE}_{\widetilde{\delta}_M}(t) = t$. For convenience, we will frequently denote $\mathrm{MSE}_{\delta}(t)$ as simply $\mathrm{MSE}(t)$.

As we will prove in the sequel, a mean estimator $\delta(t)$ constructed using the endpoint method has $\mathrm{MSE}(\tau_{\mathrm{s}}) = \tau_{\mathrm{s}} + \mathrm{Cov}\left(N^1_{\mathrm{E}}(0,\tau_{\mathrm{s}}), N^2_{\mathrm{E}}(0,\tau_{\mathrm{s}})\right) \leq \tau_{\mathrm{s}}$. That is, its MSE is that of the iid estimator plus the negative covariance between the antithetically sampled Poisson random variables with parameter τ_{s}. Note that this MSE is also strictly greater than 0, since the Poisson distribution is not symmetric. Due to iid jump times, $\mathrm{MSE}(t)$ for this mean estimate is a piecewise concave quadratic function for $t \in (0, \tau_{\mathrm{s}})$. It is determined by $\mathrm{MSE}(0) = 0$, the value of $\mathrm{MSE}(\tau_{\mathrm{s}})$ (which is fixed by the covariance between two antithetic Poisson random variables), and the fact that $\frac{\mathrm{d}}{\mathrm{dt}}\,\mathrm{MSE}(t)|_{t=0^+} = 1$ (see Lemma 6). This is illustrated by the red curve in Figure 4.1, for $\tau_{\mathrm{s}} = 10$ over an interval $[0, \tau_{\mathrm{f}} = 10]$.

We may further reduce the MSE over most of this interval by reducing τ_{s}, and thus injecting negative correlation more frequently in the interval. Compare the endpoint MSE (red) curve in Figure 4.1 ($\tau_{\mathrm{s}} = 10$)

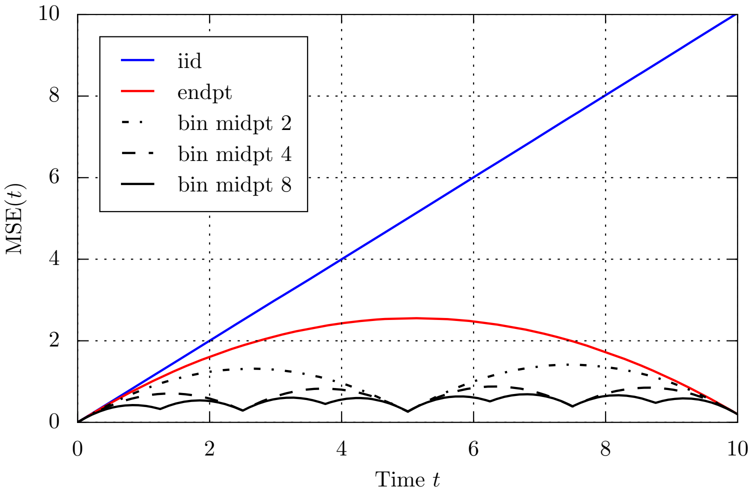

Figure 4.1: The scaled MSE of 2-sample mean estimators, each produced from a pair of iid, endpoint, or binomial midpoint antithetic Poisson process paths, all simulated for a single step of length $\tau_\mathrm{s} = 10$. For comparison, we show binomial midpoint estimators constructed using 2, 4, and 8 substeps. Note that each subsequent mean estimator dominates the previous one, meaning it has lower MSE(t) for all t.

to the same curve in Fig. 4.2 ($\tau_\mathrm{s} \approx 2.5$) to observe this process. However, note that the gains here present a tradeoff. Indeed, we can see that MSE(10) is increased by taking four steps instead of one. Using independent increments of the endpoint method at τ_s step intervals:

$$\begin{aligned}\mathrm{MSE}(4\tau_\mathrm{s}) &= 4\tau_\mathrm{s} + \sum_{n=0}^{3} \mathrm{Cov}\left(N_\mathrm{E}^1(n\tau_\mathrm{s}, (n+1)\tau_\mathrm{s}), N_\mathrm{E}^2(n\tau_\mathrm{s}, (n+1)\tau_\mathrm{s})\right) \\ &= 4\tau_\mathrm{s} + 4\,\mathrm{Cov}\left(N_\mathrm{E}^1(0, \tau_\mathrm{s}), N_\mathrm{E}^2(0, \tau_\mathrm{s})\right) \\ &\geq 4\tau_\mathrm{s} + \mathrm{Cov}\left(N_\mathrm{E}^1(0, 4\tau_\mathrm{s}), N_\mathrm{E}^2(0, 4\tau_\mathrm{s})\right), \qquad (4.3)\end{aligned}$$

since the covariance between antithetic Poisson variables is sub-linear in their parameter [23]. So each time we step forward by τ_s, we accumulate MSE from the previous endpoint and this accumulation exceeds the MSE at the endpoint of a single, larger step. This difference is small at first, and still reduces the overall MSE in the interval, but as we let τ_s get even smaller, eventually the MSE not only becomes significantly larger at the endpoint of the interval than before, it is also larger over the majority of the interval, as shown in Fig. 4.3 ($\tau_\mathrm{s} \approx 0.0625$). In fact, the expression we will prove in Theorem 7 shows that as $\tau_\mathrm{s} \to 0$, $\mathrm{MSE}_\delta(t) \to \mathrm{MSE}_{\widetilde{\delta}}(t)$, the MSE of the iid estimator. So then, is there a way to reduce MSE more evenly for $t \in (0, \tau_\mathrm{s})$? The next algorithm we present does exactly that.

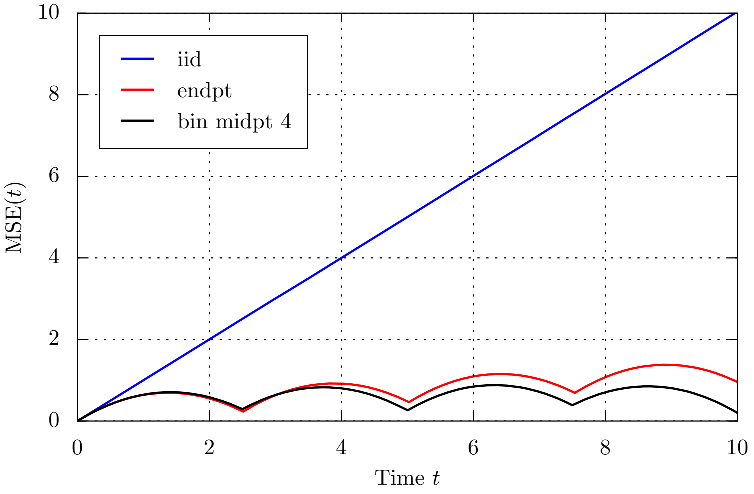

Figure 4.2: The scaled MSE of iid, endpoint and binomial midpoint 2-sample mean estimators. Endpoint technique uses $\tau_s \approx 2.5$. Binomial midpoint technique uses $\tau_s = 10.0$ with 4 sub-steps, or sub-step size 2.5. Note that the endpoint estimator achieves similar performance to the binomial midpoint estimator, but accumulates slightly more MSE with each step, as shown in (4.3). These two antithetic estimators require an almost identical number of random variable draws to simulate on average.

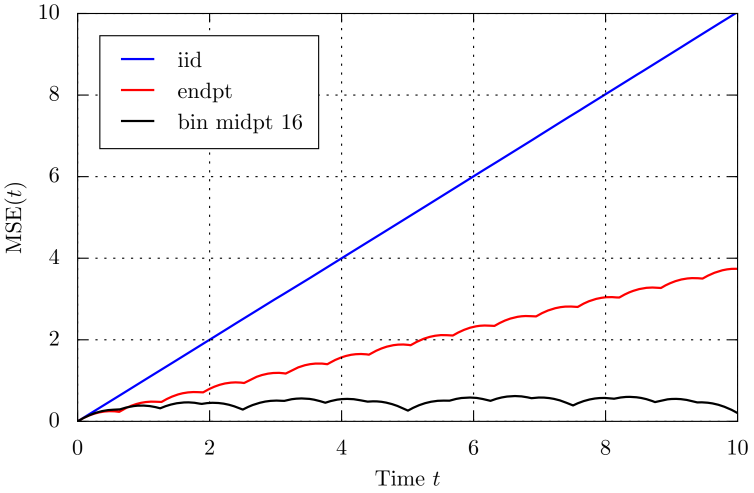

Figure 4.3: The scaled MSE of iid, endpoint and binomial midpoint 2-sample mean estimators. Endpoint technique uses $\tau_s \approx 0.63$. Binomial midpoint technique uses $\tau_s = 10.0$ with 16 sub-steps, for sub-step size of $10/16 = 0.625$. When the step size of the endpoint technique becomes sufficiently small, its MSE accumulates rapidly. For the binomial midpoint technique, MSE(10) is not affected by the number of sub-steps it takes in $[0, 10.0]$. These two antithetic estimators require an almost identical number of random variable draws to simulate on average.

4.1.2 Binomial midpoint method for increased variance reduction

As motivated above, τ_s is the primary parameter that governs the reduction in MSE for antithetic simulation of unit-rate Poisson processes using these techniques. There are limitations to what modifying τ_s alone can do, however. Indeed, as we illustrated, with changing τ_s there is a tradeoff where near-time performance (within a fixed time-window, say) competes with long-term performance (after compounding many small steps, say). Instead of reducing τ_s to improve near-time performance, we may instead antithetically sub-sample previous times (using the conditional binomial distribution) so that we improve local performance in much the same way that, say, halving τ_s does, but without sacrificing endpoint performance. We refer to this approach as the binomial midpoint method.

The binomial midpoint method injects more negative correlation into the Poisson process pair by antithetically sampling values in the interior of a step after sampling its endpoint. Here we exploit the fact that, conditioned on past and future values, the Poisson process has binomial distribution. First, we may simulate $\left(Y^1(\tau_\mathrm{s}), Y^2(\tau_\mathrm{s})\right)$, exactly as in the endpoint method using antithetic Poisson sampling. But then, instead of merely sampling the iid jump times over $[0, \tau_\mathrm{s}]$ as in the endpoint method, we first conditionally sample additional antithetic values of the process at interior time points. For example, $\left(Y^1(\tau_\mathrm{s}/2), Y^2(\tau_\mathrm{s}/2)\right)$, which, conditioned on $\left(Y^1(\tau_\mathrm{s}), Y^2(\tau_\mathrm{s})\right)$ are binomially distributed, i.e., $Y^i(\tau_\mathrm{s}/2)|Y^i(\tau_\mathrm{s}) \sim \mathrm{Bin}\left(Y^i(\tau_\mathrm{s}), 1/2\right)$ for $i = 1, 2$, may be sampled antithetically by inverting their respective binomial CDFs. Note that this antithetic pair will no longer have identical conditional CDFs, since, for a particular pair of trajectories $Y^1(\tau_\mathrm{s}) \neq Y^2(\tau_\mathrm{s})$. But we can still introduce additional negative correlation at this point since both distribution functions are still non-decreasing. And, importantly, we haven't increased $\mathrm{MSE}(\tau_\mathrm{s})$ since it has already been sampled. We find it best in practice to sub-sample at time points that bisect the interval formed by times where the Poisson process has already been sampled, since the binomial parameter $p = 1/2$ symmetrizes the binomial CDF, enhancing the efficacy of antithetic sampling. Thus, additional negative correlation can be introduced at subsequent dyadic intervals by conditioning on the nearest previous and future values that have already been sampled. For instance $N^i(3\tau_\mathrm{s}/4) := \left(Y^i(3\tau_\mathrm{s}/4) - Y^i(\tau_\mathrm{s}/2)\right)|Y^i(\tau_\mathrm{s}/2), Y^i(\tau_\mathrm{s}) \sim \mathrm{Bin}\left(Y^i(\tau_\mathrm{s}) - Y^i(\tau_\mathrm{s}/2), 1/2\right)$ for $i = 1, 2$. For the sake of compactness, we denote the conditional sub-increment of process Y^i from time s to time $(s+t)/2$ (i.e., from the beginning of the interval $[s, t]$ to its midpoint) as

$$Q^i(s,t) := \left(Y^i((s+t)/2) - Y^i(s)\right)|Y^i(s), Y^i(t), \tag{4.4}$$

where $Q^i(s,t) \sim \mathrm{Bin}\left(Y^i(t) - Y^i(s), 1/2\right)$. The binomial midpoint technique is summarized in Algorithm 3.

Note that the binomial midpoint method with no partitions between steps (i.e., when the order $L = 0$,

Algorithm 3 Binomial Midpoint Method: Conditional sub-sampling in τ_{s}-increment on 2^L dyadic points

```
Intialize:  $n \leftarrow 0, \left(\mathcal{A}^1_{\tau_{\mathrm{f}}}, \mathcal{A}^2_{\tau_{\mathrm{f}}}\right) \leftarrow (\emptyset, \emptyset)$
while $n\tau_{\mathrm{s}} < \tau_{\mathrm{f}}$ do
  $Y^i_{\mathrm{B}}(0) \leftarrow 0, \mathcal{A}^i_{\mathrm{B}} \leftarrow \emptyset,\ i \in \{1,2\}$,
  Sample antithetic Poisson random variables: $\left(N^1_{\mathrm{B}}(0,\tau_{\mathrm{s}}), N^2_{\mathrm{B}}(0,\tau_{\mathrm{s}})\right) \overset{\mathrm{anti}}{\sim} \mathrm{Pois}(\tau_{\mathrm{s}})$
  Set state value at endpoint $\tau_{\mathrm{s}}$: $Y^i_{\mathrm{B}}(\tau_{\mathrm{s}}) \leftarrow N^i_{\mathrm{B}}(0,\tau_{\mathrm{s}}),\ i \in \{1,2\}$
  for $\ell = 1$ to $L$ do
    for $k = 1, 3, 5, \dots, 2^\ell - 1$ do
      Conditionally sample increments at midpoint as antithetic binomial variables (see Eq. (4.4)):
      $Q^i\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}, \frac{k+1}{2^\ell}\tau_{\mathrm{s}}\right) \overset{\mathrm{anti}}{\sim} \mathrm{Bin}\left(Y^i_{\mathrm{B}}\left(\frac{k+1}{2^\ell}\tau_{\mathrm{s}}\right) - Y^i_{\mathrm{B}}\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}\right), \frac{1}{2}\right),\ i \in \{1,2\}$
      Set state value at midpoint: $Y^i_{\mathrm{B}}\left(\frac{k}{2^\ell}\tau_{\mathrm{s}}\right) \leftarrow Y^i_{\mathrm{B}}\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}\right) + Q^i\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}, \frac{k+1}{2^\ell}\tau_{\mathrm{s}}\right),\ i \in \{1,2\}$
    end for
  end for
  for $k = 1$ to $2^L - 1$ do
    for $j = 1$ to $Y^i_{\mathrm{B}}\left(\frac{k+1}{2^L}\tau_{\mathrm{s}}\right) - Y^i_{\mathrm{B}}\left(\frac{k}{2^L}\tau_{\mathrm{s}}\right)$ do
      Simulate iid jump times: $t^i_{j,k} \overset{\mathrm{i.i.d.}}{\sim} \mathrm{Unif}\left(\frac{k}{2^L}\tau_{\mathrm{s}}, \frac{k+1}{2^L}\tau_{\mathrm{s}}\right),\ i \in \{1,2\}$
    end for
    Sort and append jump times: $\mathcal{A}^i_{\mathrm{B}} \leftarrow \mathrm{sort}\left(\{n\tau_{\mathrm{s}} + t^i_{j,k}\}_j\right),\ i \in \{1,2\}$
  end for
  $\mathcal{A}^i_{\tau_{\mathrm{f}}} \leftarrow \mathcal{A}^i_{\tau_{\mathrm{f}}} \cup \mathcal{A}^i_{\mathrm{B}},\ i \in \{1,2\}$
  $n \leftarrow n + 1$
end while
$\mathcal{A}^i_{\tau_{\mathrm{f}}} \leftarrow \mathcal{A}^i_{\tau_{\mathrm{f}}} \cap [0, \tau_{\mathrm{f}}],\ i \in \{1,2\}$
return  $\left(\mathcal{A}^1_{\tau_{\mathrm{f}}}, \mathcal{A}^2_{\tau_{\mathrm{f}}}\right)$
```

which can be interpreted as the number of times we halve the sub-interval length) is precisely the endpoint method. Note also that we introduce additional cost (in the form of additional random variable samples) to achieve this variance reduction. If we divide a particular step of length τ_{s} into 2^L equal-length sub-intervals, we require $2^L - 1$ additional CDF inversions. As we will show in the sequel, this cost-error tradeoff is profitable for a finite order L that depends on the operating parameters of the system and particular simulation. Additionally, while MSE will always be reduced by increasing L, note that this reduction decreases sharply for large L. As a result, it should not be thought of as an asymptotic parameter that drives MSE toward zero for large values and fixed τ_{s}. Indeed, note that as $\tau_{\mathrm{s}}/2^L$ becomes small, the first parameter of the corresponding binomial distributions will also be small, since the expected change in the process will be small over this sub-interval. This limits the impact of antithetic sampling. In this case, the corresponding reduction in MSE is small, and it would have been more efficient to partition into 2^{L-1} sub-intervals instead, incurring roughly half the computational cost. So L is best thought of as a finite parameter that has significant benefit for small values and saturates quickly for large values. This saturation point will largely be determined by the quantity $\tau_{\mathrm{s}}/2^L$.

4.1.3 Analysis of antithetic endpoint Poisson processes

We now provide some analysis of the antithetic algorithm above. In this section, we present two useful metrics to quantify the expected error from mean-estimators constructed using the algorithms: the scaled mean-square error and the integrated scaled mean-square error. Next, we define a special function related to the antithetic simulation of Poisson random *variables* that will help us analyze the behavior of antithetically simulated Poisson random *processes*. Finally, we present several results that explicitly and exactly quantify the scaled MSE and integrated scaled MSE behavior of antithetic endpoint Poisson process simulation, which in particular we then use to obtain asymptotic performance bounds.

Let $\left(\widetilde{Y}^1, \widetilde{Y}^1\right)$ denote a pair of iid, unit-rate Poisson process, so that $\operatorname{Cov}\left(\widetilde{Y}^1(t), \widetilde{Y}^2(t)\right) = 0$ for all $t \geq 0$. Let $\left(Y^1, Y^2\right)$ denote the antithetic, unit-rate Poisson processes constructed using Algorithm 2 above, so that Y^1 and Y^2 are correlated (indeed, we will show that $\operatorname{Cov}\left(Y^1(t), Y^2(t)\right) \leq 0$ for all $t \geq 0$). Let $\widetilde{\delta}(t)$ and $\delta(t)$ denote the 2-sample mean estimators obtained by averaging the iid and endpoint Poisson process pairs, respectively. For brevity, we will refer to mean estimators by the method used to simulate their constituent sample paths (e.g. iid estimator, endpoint estimator, binomial midpoint estimator). Recall the scaled MSE defined in Eq. (4.1) given by

$$\mathrm{MSE}(t) := N \operatorname{Var}\left(\delta(t)\right) = N \operatorname{Var}\left(\frac{1}{N}\sum_{i=1}^{N} Y^i(t)\right),$$

where N is the number of sample paths used to construct the mean estimate.

Note that, as with any Monte Carlo scheme, we may produce more accurate mean estimates by increasing the number of samples used to construct the estimator. Practitioners can simulate a sequence of many antithetic pairs which are iid with respect to each other (each with 2 correlated components, of course) to create mean estimates of sufficient accuracy for their particular application. This decrease in variance will scale in the usual way ($1/N$, or $1/\sqrt{N}$ w.r.t. the standard deviation), so we restrict our analyses to mean estimates constructed from a single antithetic pair of random paths. Further, all comparisons are made to iid mean estimates constructed using two independent sample paths.

Recall that $(X_1, X_2) \overset{\text{anti}}{\sim} \operatorname{Pois}(\tau)$ denotes the anticorrelated scalar Poisson variable pair, i.e.,

$$X_1 := F_\tau^{-1}(U)$$
$$X_2 := F_\tau^{-1}(1-U),$$

where $U \sim \operatorname{Unif}[0,1]$ is a uniform random variable and F_τ^{-1} is the formal inverse of the Poisson CDF with

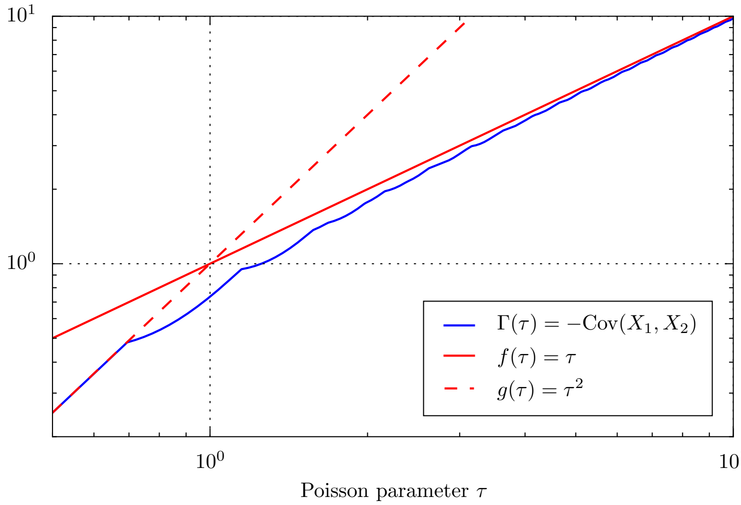

Figure 4.4: The special function $\Gamma(\tau) = -\operatorname{Cov}(X_1, X_2)$ (where $(X_1, X_2) \overset{\text{anti}}{\sim} \text{Pois}(\tau)$) plotted versus Poisson parameter τ. For reference, we also show the simple functions $f(\tau) = \tau$ and $g(\tau) = \tau^2$. Note that $0 \leq \Gamma(\tau) < \tau$ for all τ and, for $\tau < \ln 2$, $\Gamma(\tau) = \tau^2$.

parameter τ. For such an antithetic pair, define the special function

$$\Gamma(\tau) := -\operatorname{Cov}(X_1, X_2) = \tau^2 - \int_{u=0}^{1} F_\tau^{-1}(u) F_\tau^{-1}(1-u)\, \mathrm{d}u \geq 0, \tag{4.5}$$

the negative covariance of a pair of antithetically sampled Poisson scalar random variables. This function will appear frequently in the analysis of the variance properties of antithetically simulated Poisson process paths. It has several useful properties. In particular, note that $\Gamma(\tau) \leq \tau = \operatorname{Var}(X_1)$ by definition and $\Gamma(\tau) \geq 0$ for all τ [25]. Note that $\Gamma(\tau) = \tau^2$ for all $\tau < \ln 2$ [23]. These relationships, as well as Γ itself, are shown in Fig. 4.4. The scaled MSE of a mean estimator constructed from (X_1, X_2) is related to Γ by $\text{MSE} = 2 \operatorname{Var}\left(\frac{X_1+X_2}{2}\right) = \operatorname{Var}(X_1) - \Gamma(\tau) = \tau - \Gamma(\tau)$. This quantity is plotted in Fig. 4.5.

We proceed by exactly characterizing the variance properties of a mean estimator constructed from two antithetic sample paths (Y^1, Y^2) of the unit-rate, Poisson process simulated using the endpoint technique defined above. First, we provide an expression for the variance of the estimator at every time, then we motivate and provide an expression for a more useful quantity, the integral of the estimator variance over a fixed time window $[0, \tau_\mathrm{f}]$. We begin with a lemma that characterizes how the covariance between two correlated Poisson processes propagates from points of direct anticorrelation to times where direct anticorrelation is *not* applied.

Lemma 6. *For antithetic unit-rate Poisson processes Y^1, Y^2, and for $0 \leq T_1 < T_2$, denote by $\mathcal{G}(T_1, T_2) := \sigma\left\{Y^1(T_1), Y^1(T_2), Y^2(T_1), Y^2(T_2)\right\}$, the sigma algebra generated by the 4 random variables obtained by eval-*

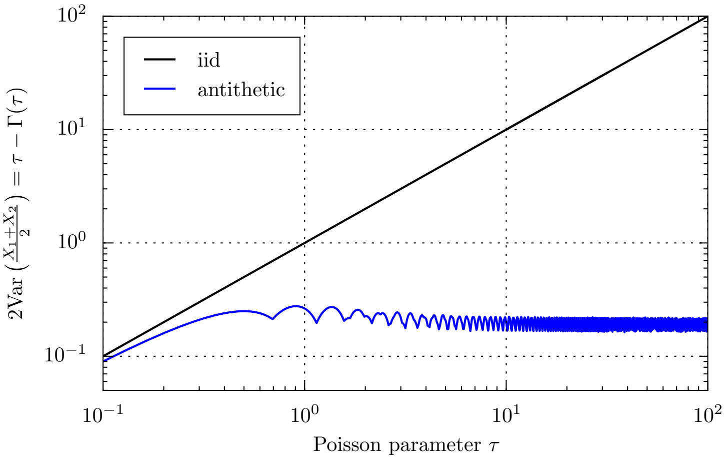

Figure 4.5: Scaled MSE of a mean estimate constructed using two Poisson random variables, sampled either using iid or antithetic sampling, plotted versus Poisson parameter τ. Note that this variance remains bounded below by a small positive constant, even for large τ. This suggests that $\Gamma(\tau)$ does not converge to τ as $\tau \to \infty$.

uating each process at each endpoint. Then, if $Y^1(t)$ and $Y^2(t)$ are conditionally independent given $\mathcal{G}(T_1, T_2)$ for every $t \in (T_1, T_2)$, then:

$$\text{Cov}\left(Y^1(t), Y^2(t)\right) = \text{Cov}\left(Y^1(T_1), Y^2(T_1)\right) + \frac{(t-T_1)^2}{(T_2-T_1)^2}\,\text{Cov}\left(N^1(T_1,T_2), N^2(T_1,T_2)\right), \tag{4.6}$$

for every $t \in [T_1, T_2]$, where $N^i(T_1, T_2) := Y^i(T_2) - Y^i(T_1)$ is the increment of the process. .

Proof. We proceed using the Law of Total Expectation, the conditional independence hypothesis, and the independent increments property as follows:

$$\begin{aligned}
\mathbb{E}\left[Y^1(t)Y^2(t)\right] &= \mathbb{E}\left[\mathbb{E}\left[Y^1(t)Y^2(t)\middle|\mathcal{G}\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[Y^1(t)\middle|\mathcal{G}\right]\mathbb{E}\left[Y^2(t)\middle|\mathcal{G}\right]\right] \\
&= \mathbb{E}\left[\left(Y^1(T_1) + \frac{t-T_1}{T_2-T_1}N^1(T_1,T_2)\right)\cdot\left(Y^2(T_1) + \frac{t-T_1}{T_2-T_1}N^2(T_1,T_2)\right)\right] \\
&= \mathbb{E}\left[Y^1(T_1)Y^2(T_1)\right] + 2T_1(t-T_1) + \frac{(t-T_1)^2}{(T_2-T_1)^2}\mathbb{E}\left[N^1(T_1,T_2)N^2(T_1,T_2)\right].
\end{aligned}$$

Note also that

$$t^2 = T_1^2 + 2T_1(t-T_1) + \frac{(t-T_1)^2}{(T_2-T_1)^2}(T_2-T_1)^2, \tag{4.7}$$

so that

$$\begin{aligned}\mathrm{Cov}(Y^1(t), Y^2(t)) &= \mathbb{E}\left[Y^1(t)Y^2(t)\right] - t^2 \\ &= \mathbb{E}\left[Y^1(T_1)Y^2(T_1)\right] - T_1^2 + \frac{(t-T_1)^2}{(T_2-T_1)^2}\left(\mathbb{E}\left[N^1(T_1,T_2)N^2(T_1,T_2)\right] - (T_2-T_1)^2\right) \\ &= \mathrm{Cov}(Y^1(T_1), Y^2(T_1)) + \frac{(t-T_1)^2}{(T_2-T_1)^2}\mathrm{Cov}(N^1(T_1,T_2), N^2(T_1,T_2)),\end{aligned}$$

and the claim holds. □

Lemma 6 allows us to express the scaled MSE of an endpoint estimator with step-size τ_s at any time t in terms of the special function Γ evaluated at τ_s. This explicit expression is derived in Theorem 7.

Theorem 7. *For anticorrelated unit-rate Poisson processes (Y^1, Y^2), sampled using the antithetic endpoint technique with step size τ_s, the scaled MSE of the corresponding mean estimator is piecewise quadratic, and is given exactly by:*

$$\mathrm{MSE}(t) = 2\,\mathrm{Var}\left(\frac{Y^1(t)+Y^2(t)}{2}\right) = t - n\Gamma(\tau_\mathrm{s}) - \frac{(t-n\tau_\mathrm{s})^2}{{\tau_\mathrm{s}}^2}\Gamma(\tau_\mathrm{s}), \tag{4.8}$$

for every $t \in [n\tau_\mathrm{s}, (n+1)\tau_\mathrm{s}]$.

Proof. We proceed by first noting that the covariance between the Poisson increments $\mathrm{Cov}\left(N^1(T_1,T_2), N^2(T_1,T_2)\right)$ present in the last term of (4.6) is exactly equal to $-\Gamma(T_2-T_1)$, since $\left(N^1(T_1,T_2), N^2(T_1,T_2)\right)$ are just antithetically sampled Poisson random variables with parameter $T_2 - T_1$. We proceed by induction on n. First, note that the conditions of Lemma 6 are satisfied for $T_1 = 0 < T_2 = \tau_\mathrm{s}$ since, conditioned on the σ-algebra $\mathcal{G}(0,\tau_\mathrm{s}) = \sigma\left\{Y^1(0)=0, Y^2(0)=0, Y^1(\tau_\mathrm{s}), Y^2(\tau_\mathrm{s})\right\}$, the random variables $Y^1(t), Y^2(t)$ are independent for all $t \in (0,\tau_\mathrm{s})$. Thus

$$\begin{aligned}\mathrm{Cov}\left(Y^1(t), Y^2(t)\right) &= \mathrm{Cov}\left(Y^1(0), Y^2(0)\right) + \frac{t^2}{{\tau_\mathrm{s}}^2}\mathrm{Cov}\left(N^1(0,\tau_\mathrm{s}), N^2(0,\tau_\mathrm{s})\right) \\ &= -\frac{t^2}{{\tau_\mathrm{s}}^2}\Gamma(\tau_\mathrm{s}),\end{aligned}$$

and

$$\begin{aligned}\mathrm{MSE}(t) &= \mathrm{Var}\left(Y^1(t)\right) + \mathrm{Cov}\left(Y^1(t), Y^2(t)\right) \\ &= t - \frac{t^2}{{\tau_\mathrm{s}}^2}\Gamma(\tau_\mathrm{s})\end{aligned}$$

for all $t \in [0,\tau_\mathrm{s}]$ (i.e., $n = 0$).

Now, suppose that the claim holds for $n-1$, namely that, for $t \in [(n-1)\tau_{\mathrm{s}}, n\tau_{\mathrm{s}}]$,

$$\mathrm{MSE}(t) = 2\,\mathrm{Var}\left(\frac{Y^1(t)+Y^2(t)}{2}\right) = t - (n-1)\Gamma(\tau_{\mathrm{s}}) - \frac{(t-(n-1)\tau_{\mathrm{s}})^2}{{\tau_{\mathrm{s}}}^2}\Gamma(\tau_{\mathrm{s}}),$$

and in particular that

$$\begin{aligned} \mathrm{Var}\left(Y^1(n\tau_{\mathrm{s}})\right) + \mathrm{Cov}\left(Y^1(n\tau_{\mathrm{s}}), Y^2(n\tau_{\mathrm{s}})\right) &= n\tau_{\mathrm{s}} - (n-1)\Gamma(\tau_{\mathrm{s}}) - \frac{(n\tau_{\mathrm{s}}-(n-1)\tau_{\mathrm{s}})^2}{{\tau_{\mathrm{s}}}^2}\Gamma(\tau_{\mathrm{s}}) \\ \implies \mathrm{Cov}\left(Y^1(n\tau_{\mathrm{s}}), Y^2(n\tau_{\mathrm{s}})\right) &= -(n-1)\Gamma(\tau_{\mathrm{s}}) - \frac{(\tau_{\mathrm{s}})^2}{{\tau_{\mathrm{s}}}^2}\Gamma(\tau_{\mathrm{s}}) \\ &= -n\Gamma(\tau_{\mathrm{s}}). \end{aligned}$$

By construction, for $t \in (n\tau_{\mathrm{s}}, (n+1)\tau_{\mathrm{s}})$, $Y^1(t)$ and $Y^2(t)$ are independent conditioned on $\mathcal{G}(n\tau_{\mathrm{s}}, (n+1)\tau_{\mathrm{s}})$, since all random sampling inside the interval is iid uniform, given the endpoints. So the conditions of Lemma 6 again hold, and

$$\begin{aligned} \mathrm{Cov}\left(Y^1(t), Y^2(t)\right) &= \mathrm{Cov}\left(Y^1(n\tau_{\mathrm{s}}), Y^2(n\tau_{\mathrm{s}})\right) + \frac{(t-n\tau_{\mathrm{s}})^2}{((n+1)\tau_{\mathrm{s}} - n\tau_{\mathrm{s}})^2}\,\mathrm{Cov}\left(N^1(n\tau_{\mathrm{s}}, (n+1)\tau_{\mathrm{s}}), N^2(n\tau_{\mathrm{s}}, (n+1)\tau_{\mathrm{s}})\right) \\ &= -n\Gamma(\tau_{\mathrm{s}}) - \frac{(t-n\tau_{\mathrm{s}})^2}{{\tau_{\mathrm{s}}}^2}\Gamma(\tau_{\mathrm{s}}), \end{aligned}$$

for all $t \in [n\tau_{\mathrm{s}}, (n+1)\tau_{\mathrm{s}}]$. So then, for all $t \in [n\tau_{\mathrm{s}}, (n+1)\tau_{\mathrm{s}}]$,

$$\begin{aligned} \mathrm{MSE}(t) &= \mathrm{Var}\left(Y^1(t)\right) + \mathrm{Cov}\left(Y^1(t), Y^2(t)\right) \\ &= t - n\Gamma(\tau_{\mathrm{s}}) - \frac{(t-n\tau_{\mathrm{s}})^2}{{\tau_{\mathrm{s}}}^2}\Gamma(\tau_{\mathrm{s}}) \end{aligned}$$

and the claim holds for n. □

The expression (4.8) proven in Theorem 7 combined with our intuition about the function Γ suggests that the larger step size τ_{s} we take, the greater variance reduction we will observe over a long period of time. In practice, however, stochastic simulation will often be performed over a relatively fixed finite time window, dictated by the system parameters or problem of interest. Reductions in estimator variance beyond that window of interest, which we will denote by $[0, \tau_{\mathrm{f}}]$, are of little benefit since they will never be observed, and in particular they may adversely affect the performance of mean estimates in the window of simulation. Thus, a better metric for comparison between techniques is the total MSE over a fixed finite time interval $[0, \tau_{\mathrm{f}}]$. For simplicity, we proceed using the $\mathcal{L}_1$-integral of MSE as our metric of choice.

Lemma 8. *For the endpoint technique implemented with step-size τ_{s} over time interval $[0, \tau_{\mathrm{f}}]$, let $N :=$*

$\lfloor \tau_{\rm f}/\tau_{\rm s} \rfloor$ *be the number of incremental Poisson samples (i.e., full steps) taken in the interval. Then*

$$\begin{aligned}\int_0^{\tau_{\rm f}} \mathrm{MSE}(t)\,\mathrm{dt} &= \int_0^{\tau_{\rm f}} 2\,\mathrm{Var}\left(\frac{Y^1(t)+Y^2(t)}{2}\right)\,\mathrm{d}t \\ &= \frac{N(N-1)}{2}\tau_{\rm s}\left(\tau_{\rm s}-\Gamma(\tau_{\rm s})\right)+\frac{N}{6}\tau_{\rm s}\left(3\tau_{\rm s}-2\Gamma(\tau_{\rm s})\right) \\ &+ N\left(\tau_{\rm s}-\Gamma(\tau_{\rm s})\right)\left(\tau_{\rm f}-N\tau_{\rm s}\right)+\frac{\left(\tau_{\rm f}-N\tau_{\rm s}\right)^2}{2}-\frac{\left(\tau_{\rm f}-N\tau_{\rm s}\right)^3}{3\tau_{\rm s}^2}\Gamma(\tau_{\rm s}). \qquad (4.9)\end{aligned}$$

The proof is lengthy but straightforward, and is omitted here for brevity. While this expression is exact, it can be difficult to parse in the general case. For comparison, note that the integrated variance of the iid mean estimator is given by

$$\int_0^{\tau_{\rm f}} \mathrm{MSE}_{\widetilde{\delta}}(t)\,\mathrm{d}t = \int_0^{\tau_{\rm f}} 2\,\mathrm{Var}\left(\frac{\widetilde{Y}^1(t)+\widetilde{Y}^2(t)}{2}\right)\,\mathrm{d}t = \int_0^{\tau_{\rm f}} t\,\mathrm{d}t = \frac{1}{2}\tau_{\rm f}^2. \qquad (4.10)$$

As will be shown in Fig. 4.7, we can see that the integrated MSE of the endpoint estimator is always less than the same quantity for the iid mean estimator. Two simple extremal cases are also illustrative. Consider the case when $\tau_{\rm s} > \tau_{\rm f}$, i.e., when less than one step is used for simulation. In this case, $N = 0$ and (4.9) reduces to

$$\int_0^{\tau_{\rm f}} \mathrm{MSE}(t)\,\mathrm{d}t = \frac{1}{2}\tau_{\rm f}^2 - \frac{\Gamma(\tau_{\rm s})}{3\tau_{\rm s}^2}\tau_{\rm f}^3. \qquad (4.11)$$

For fixed $\tau_{\rm f}$, as $\tau_{\rm s} \to \infty$, the performance of the antithetic estimator will degrade back to the iid estimator, since $\Gamma(\tau_{\rm s}) \le \tau_{\rm s}$. Alternatively, for fixed $\tau_{\rm f}$, when $\tau_{\rm s} \to 0$, i.e., many Poisson steps are being taken during the course of a simulation, $0 \le \tau_{\rm f} - N\tau_{\rm s} \le \tau_{\rm s} \to 0$ and $\Gamma(\tau_{\rm s}) = \tau_{\rm s}^2$. Thus (4.9) becomes

$$\int_0^{\tau_{\rm f}} \mathrm{MSE}(t)\,\mathrm{d}t \approx \frac{1}{2}\tau_{\rm f}^2 - \left(\tau_{\rm f}^2/2+\tau_{\rm f}\right)\tau_{\rm s} + \mathcal{O}(\tau_{\rm s}^2), \qquad (4.12)$$

and performance again degrades to the iid case. This suggests that the best performing $\tau_{\rm s}$ is one that is neither too large nor too small relative to the window of interest, a claim that is further supported by the numerical results in Section 4.1.4.

4.1.4 Numerical results for antithetic Poisson processes

We now support the analytical results of the previous section with numerical experiments. In particular, we are interested in examining the relationship between the choice of Poisson simulation step time $\tau_{\rm s}$ and the integrated scaled MSE over the time window $[0, \tau_{\rm f}]$. For each of the Poisson process simulation algorithms, iid, endpoint, and binomial midpoint, a pair of unit-rate Poisson processes are simulated from $t = 0$ to $\tau_{\rm f}$

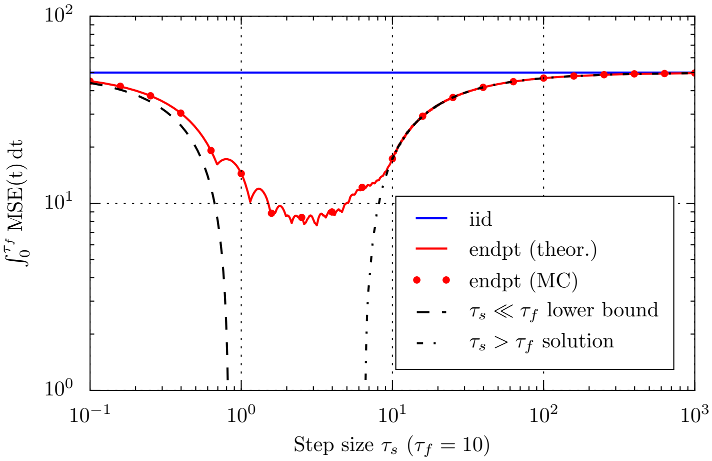

Figure 4.6: Exact analytical solution for the integrated scaled MSE of the endpoint estimator (4.9) versus step size τ_s, compared to empirical observation of the same. The right lower bound is defined in (4.11) and the left lower bound is given by (4.12). Empirical results obtained using Monte Carlo simulation using ensemble sizes of 1 440 000 samples or more. Error bars are very small and are thus omitted.

using a step size of τ_s, and averaged to produce a single, 2-sample mean estimate of the Poisson process. This sampling is repeated to form an ensemble of such mean estimators, and the ensemble is then used to construct an estimate of the integrated MSE for each algorithm. This process is repeated for a wide range of τ_s and the results are plotted as follows. First, we examine the endpoint mean estimator in order to verify both the exact analytical expression proven in (4.9) and the asymptotic bounds given by (4.12) and (4.11). The results are collected in Fig. 4.6. Next, we compare the performance of each of the proposed algorithms with each other, and examine how they vary with τ_s for fixed τ_f. These results are collected in Fig. 4.7.

It is important to note here that the operating points of each of these methods and values of τ_s correspond to different computational costs, which we will define in this work as the expected number of random variable draws required to simulate a path. For example, suppose we are simulating unit-rate Poisson processes over the interval $[0, \tau_\mathrm{f}]$ using the endpoint method ($L = 0$) with step size τ_s. On average, we will draw roughly $\tau_\mathrm{f}/\tau_\mathrm{s}$ antithetic pairs of Poisson random variables for each step we take in the interval. Then, we will sample approximately τ_f random uniform jump times to simulate a path. The details of these costs are sensitive to the many optimizations that are possible for a particular implementation of continuous-time antithetic stochastic simulation. For the purpose of this work we focus on a crude implementation: steps of length τ_s are taken until the final time τ_f is strictly exceeded. For each step taken, every corresponding uniform jump time is simulated, including those lying outside $[0, \tau_\mathrm{f}]$. Thus we will tend to incur significant overhead relative to MSE reduction when $\tau_\mathrm{s} \gg \tau_\mathrm{f}$ or when $\tau_\mathrm{s}] \ll \tau_\mathrm{f}$ compared to existing methods such as SSA or Next-Reaction.

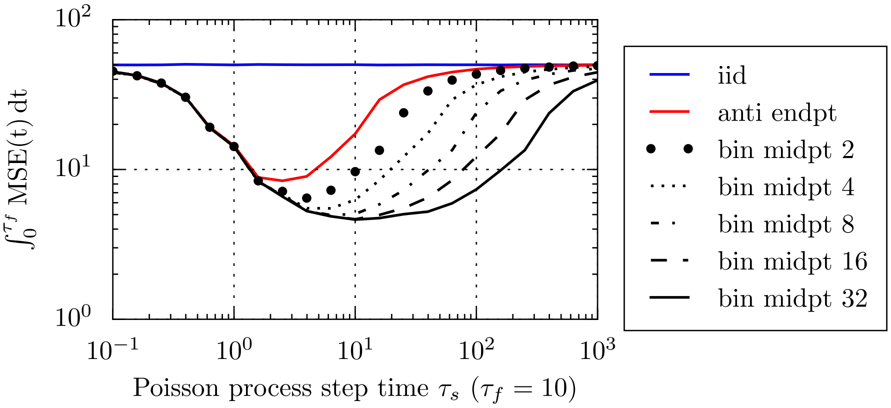

Figure 4.7: Integrated scaled variance versus step size τ_s for mean estimators produced using iid, endpoint and binomial midpoint simulation. Results obtained using Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are very small and are thus omitted.

For this particular implementation, we thus estimate the number of random draws necessary for an antithetic method with order L using the expression $2^L\left(\lceil \tau_\mathrm{f}/\tau_\mathrm{s}\rceil \vee 1\right) + \lceil \tau_\mathrm{f}/\tau_\mathrm{s}\rceil \tau_\mathrm{s} \vee \tau_\mathrm{s}$, where $\vee$ denotes the maximum operator. The lower limit for this cost is τ_f, achieved for example using various next-reaction methods [15], so we will use this as our baseline for comparison. We will restrict our attention to τ_s values that lie in the Pareto front, the region of values for which error and cost cannot be simultaneously improved. To illustrate this region, consider Fig. 4.8, where we plot the MSE results for the antithetic endpoint method that appear in Fig. 4.7 along with the estimated cost of simulation versus τ_s.

Restricting our attention only to Pareto values of τ_s, we may plot the error vs cost as shown in Fig. 4.9. As discussed above, the antithetic endpoint ($L = 0$) or binomial midpoint ($L = 1, 2, 3$) methods can offer significant performance improvement at relatively modest cost increases. These gains saturate relative to cost for larger L values (e.g. $L = 5$). Thus we may conclude that, in practice, τ_s should be tuned to the native speed of the process (relative to the time window τ_f that we are studying) and that significant performance gains can be achieved using binomial midpoint sample with moderate order L.

4.2 Antithetic simulation of lattice CTMCs

We can employ negatively correlated pairs of unit-rate Poisson processes (as simulated using Algorithms 2 and 3 shown in Section 4.1) to simulate negatively correlated pairs of lattice continuous-time Markov chains (CTMCs). We define this anticorrelated pair of stochastic processes $(X^{(1)}, X^{(2)})$ as follows:

$$X^{(j)}(t) = x_0 + \sum_{i=1}^{I} Y^{i,j}\left(\int_0^t a^i(s, X^{(j)}(s))\,\mathrm{d}s\right)\nu^i, \tag{4.13}$$

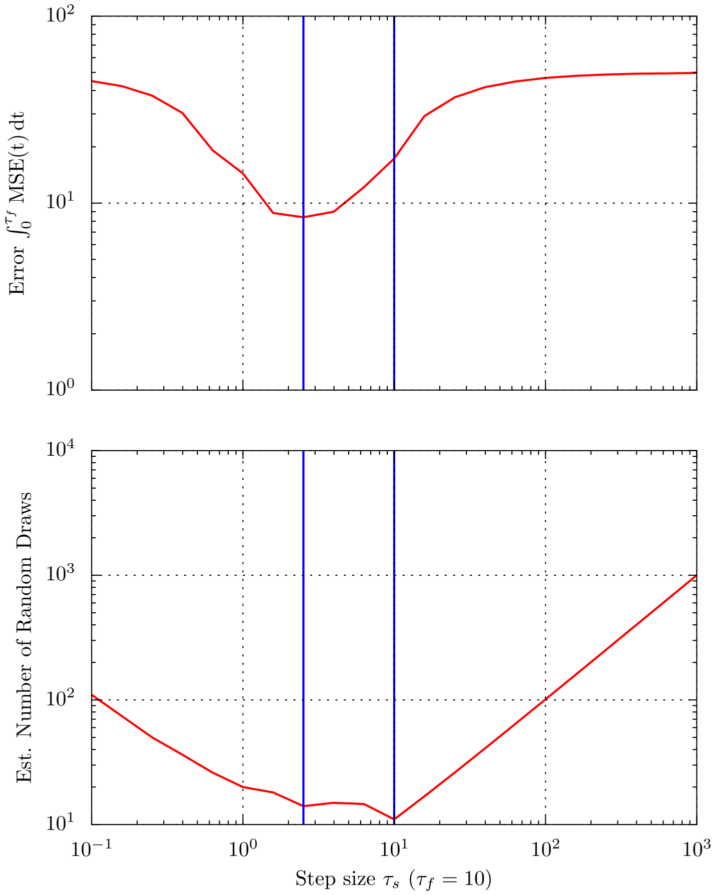

Figure 4.8: Integrated scaled variance (error) and estimated number of random draws (cost), each plotted versus step size τ_s for a mean estimator produced using endpoint simulation. The τ_s values shown in blue are the Pareto front: the set of values for which error (integrated MSE) and cost (expected number of random draws) cannot be simultaneously improved. In this case, the Pareto front is composed of two points. Results obtained using Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are very small and are thus omitted.

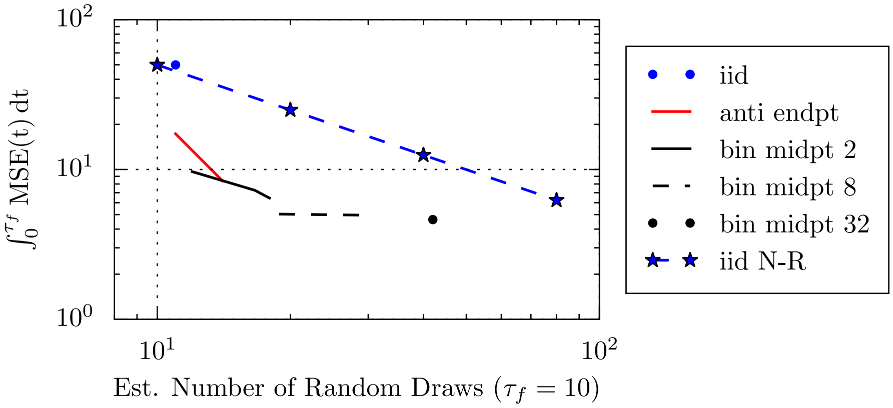

Figure 4.9: Integrated scaled variance (error) versus estimated number of random draws (cost), for various estimators constructed using iid, endpoint and binomial midpoint Poisson process simulation. Only points corresponding to τ_s values in the Pareto region are shown. The cost baseline for iid simulation is a next-reaction algorithm that simulates trajectories using no excess random variable draws. For cost comparison, we plot the unscaled, integrated MSE for the next-reaction estimator as the number of iid sample paths used in the N-R estimate is repeatedly doubled. Results obtained using Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are very small and are thus omitted.

for $j \in \{1, 2\}$. In other words, to simulate a pair of trajectories of a lattice CTMC system with I reaction channels, we simulate I antithetic pairs of unit-rate Poisson processes and assign one element of each pair to a reaction channel in each path $X^{(j)}$.

Note that, to simulate these CTMCs, we use the fact that each trajectory is piecewise constant while waiting for the next jump to occur. So for each reaction channel, we can use the value of the reaction rate to compute the time until the next transition occurs for that Poisson process. The smallest of these times is the one that will occur first, so we may move each process forward until this event occurs, update the state of the system and repeat. Thus we can simulate a process trajectory using only the ordered jump-times $\mathcal{A}$ of I unit-rate Poisson processes.

By construction, each stochastic process path is simulated using I iid, unit-rate Poisson processes, and the exact marginal distribution of the system is preserved. The only difference is that the pair of lattice CTMC paths are now negatively correlated, and will produce variance-reduced mean estimates $\delta = \frac{X^{(1)}+X^{(2)}}{2}$. To quantify this reduction in variance, we define the scaled stochastic process MSE to be

$$\mathrm{MSE}(t) = 2 \,\mathrm{tr}\, \mathrm{Var}\left(\delta(t)\right). \tag{4.14}$$

As discussed above, the scaled MSE is insensitive to the inclusion of additional anticorrelated pairs of paths, so we need only consider estimators constructed from a single pair of stochastic processes for the subsequent studies. As above, we use the integrated scaled MSE to quantify estimator performance. We use the iid mean estimator as a baseline, as its MSE does not depend on the choice of τ_{s}. In both numerical studies, we construct an ensemble of 2-sample mean estimators each built from a single pair of system trajectories simulated using either iid, endpoint or binomial midpoint simulation for a given value of τ_{s}. We then use this ensemble to estimate the integrated MSE of each of these estimators, and let τ_{s} vary over a large range of values to examine the dependence of estimator MSE. In both systems, we will see that a similar relationship between MSE and τ_{s} holds as in the Poisson process case, save that the artifact τ_{f} determining the time-window of interest is now replaced by the interaction between T, the final time of simulation, and the reaction rates and particular trajectory of the system.

We now introduce two example systems to illustrate the performance of the anticorrelated Monte Carlo for stochastic process paths using the RTC (2.1), driven by the above algorithms for generating antithetic Poisson process pairs. The first is a gene expression system with rates that are an affine function of the system state, and the second is an aerosol coagulation system driven by rates that are a nonlinear function of the system state. In both cases, the endpoint and binomial midpoint algorithms are used to generate the unit-rate Poisson processes $\{(Y^{i,1}, Y^{i,2})\}_{i=1}^{I}$ that are the sole source of random input to the models.

4.2.1 Gene-expression

First, we examine a linear gene expression system. The system has two components: mRNA that is produced and decays, and a protein it produces which also decays. This particular model appears in [8], and its reactions are given by

$$\begin{aligned} \emptyset &\xrightarrow{k_r} \text{mRNA} \\ \text{mRNA} &\xrightarrow{\gamma_r} \emptyset \\ \text{mRNA} &\xrightarrow{k_p} \text{protein} + \text{mRNA} \\ \text{protein} &\xrightarrow{\gamma_p} \emptyset. \end{aligned}$$

The system state is a vector $X \in \mathbb{Z}^2$ whose components represent the number of mRNA and protein particles. We set the initial condition $x_0 = V \cdot [1.0\ 0.5]^\top$ (where V is a system volume scaling parameter, fixed here at $V = 100$) with $I = 4$ reaction channels, given by:

$$\begin{aligned} \nu^1 &= [1\ 0]^\top & a^1(X(t)) &= k_r V \\ \nu^2 &= [-1\ 0]^\top & a^2(X(t)) &= \gamma_r X_1(t) \end{aligned}$$

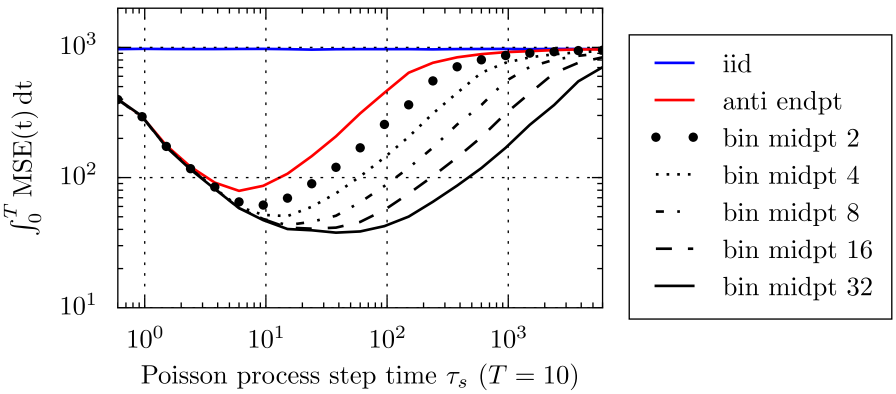

Figure 4.10: Integrated scaled MSE versus step size τ_s for mean estimators of the gene expression system produced using iid, endpoint and binomial midpoint simulation of unit-rate Poisson processes. The system is simulated with volume parameter $V = 100$, rate parameters $(k_r, \gamma_r, k_p, \gamma_p) = (0.01, 0.03, 0.06, 0.0066)$, and initial condition $X_0 = [V\ V/2]^\top$. MSE estimates are obtained using Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are very small and are thus omitted.

$$\nu^3 = [0\ 1]^\top \qquad\qquad a^3(X(t)) = k_p X_1(t)$$

$$\nu^4 = [0\ -1]^\top \qquad\qquad a^4(X(t)) = \gamma_p X_2(t),$$

for $k_r, \gamma_r, k_p, \gamma_p > 0$, and where $X_d(t)$ denotes the dth component of the state vector at time t. The system is simulated using the random time-change representation (2.1) run from time $t = 0$ to time $t = T = 10$. Our primary interest is the dependence of the integrated scaled MSE for the gene expression estimator on the choice of step size τ_s for the Poisson process trajectories. The results of this study are shown in Figure 4.10. Again, it's instructive to compare the different implementations on the basis of cost (as measured by estimated number of random variable simulations). Restricting our attention to only Pareto-optimal points, we obtain the error vs cost relationship shown in Fig. 4.11.

4.2.2 Nonlinear aerosol coagulation due to gravitational settling

Finally, we examine the MSE of a nonlinear lattice CTMC when we apply antithetic simulation to its driving Poisson processes. We consider a water aerosol system subject to gravitational settling that undergoes coagulation events as it falls. This system can be found in [36], Chapter 13, and the underlying assumptions and construction of the model used here is discussed in some detail in [25]. For the sake of brevity, we omit those details here. The system is composed of large and small water particles falling in a control volume. These classes of particles have different terminal velocities and thus may experience collisions as they fall leading to coagulation events. We fix the velocity of the control volume to be the same velocity as the large particles, so that small particles may enter the system and also coagulate with large particles. The state of

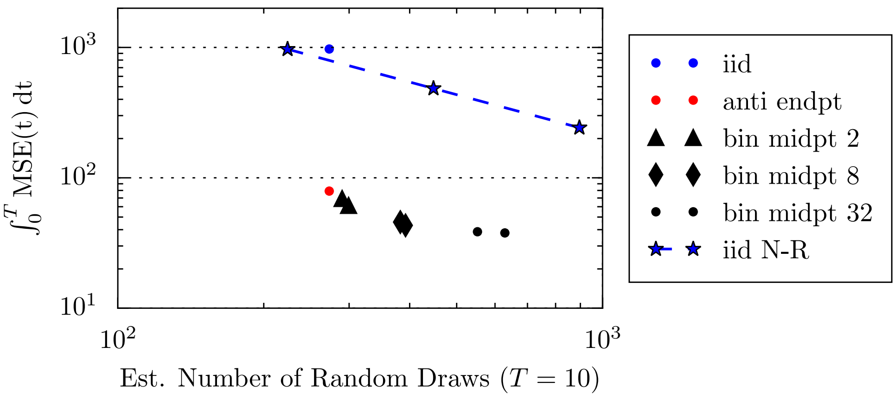

Figure 4.11: Integrated scaled variance (error) versus estimated number of random draws (cost), for various estimators of the gene expression system constructed using iid, endpoint and binomial midpoint Poisson process simulation. The cost baseline for iid simulation is a next-reaction algorithm that simulates trajectories using no excess random variable draws. For cost comparison, we plot the unscaled, integrated MSE for the next-reaction estimator as the number of iid sample paths used in the N-R estimate is repeatedly doubled. The system is simulated with volume parameter $V = 100$, rate parameters $(k_r, \gamma_r, k_p, \gamma_p) = (0.01, 0.03, 0.06, 0.0066)$, and initial condition $X_0 = [V\ V/2]^\top$. Only points corresponding to τ_{s} values in the Pareto region are shown. Results obtained using Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are very small and are thus omitted.

the system can be expressed as $X = (N_s, M_l) \in \mathbb{R}^2$, where N_s denotes the number of small particles and M_l is the total mass of the large particles. For convenience, we set the mass of the small particles to be $m = 1$, and the reaction channels and rates of the system are given by:

$$\begin{aligned} \nu^1 &= [1\ 0]^\top & a^1(X(t)) &= V \\ \nu^2 &= [-1\ 1]^\top & a^2(X(t)) &= \alpha K_{sl}^{\mathrm{GS}} V X_1(t) \end{aligned}$$

where $\alpha = 5 \cdot 10^{-4}$ is a proportionality constant and

$$K_{sl}^{\mathrm{GS}} = \frac{1}{V} \left(\sqrt[3]{X_2(t)/V} + \sqrt[3]{m} \right)^3 \left(\sqrt[3]{X_2(t)/V} - \sqrt[3]{m} \right).$$

The state is initialized from $X_0 = V \cdot [100, 10]$ and is simulated for 10s from $t = 0$ to $T = 10$. The integrated scaled MSE of the coagulation system plotted versus the Poisson process step-size τ_{s} is shown in Figure 4.12. As above, the cost-error tradeoff is visualized by restricting attention to Pareto-optimal points, and is shown in Fig. 4.13.

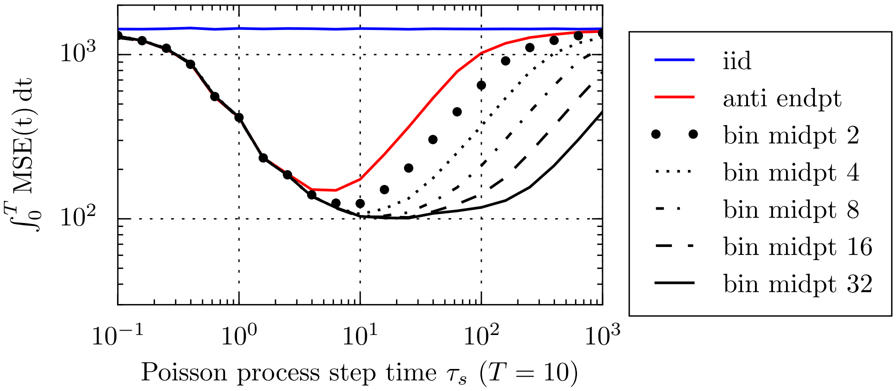

Figure 4.12: Integrated scaled MSE versus step size τ_s for mean estimators of the nonlinear aerosol coagulation system produced using iid, endpoint and binomial midpoint simulation of unit-rate Poisson processes. We take volume parameter $V = 100$, proportionality constant $\alpha = 5 \cdot 10^{-4}$, and small particle mass $m = 1$. Results obtained via Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are small and are thus omitted.

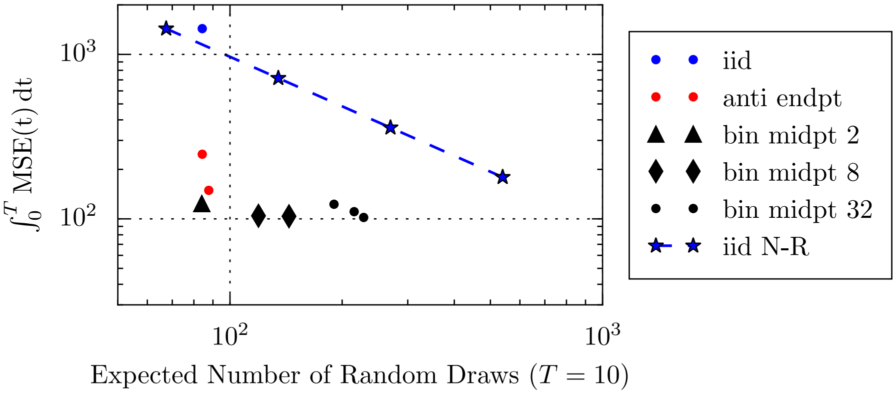

Figure 4.13: Integrated scaled variance (error) versus estimated number of random draws (cost), for various estimators of the nonlinear coagulation system constructed using iid, endpoint and binomial midpoint Poisson process simulation. The cost baseline for iid simulation is a next-reaction algorithm that simulates trajectories using no excess random variable draws. For cost comparison, we plot the unscaled, integrated MSE for the next-reaction estimator as the number of iid sample paths used in the N-R estimate is repeatedly doubled. We take volume parameter $V = 100$, proportionality constant $\alpha = 5 \cdot 10^{-4}$, and small particle mass $m = 1$. Only points corresponding to τ_s values in the Pareto region are shown. Results obtained using Monte Carlo simulation using ensemble sizes of 360 000 samples or more. Error bars are small and are thus omitted.

Chapter 5

Conditional tau-leaping

We return now to the context of lattice DTMCs, specifically simulation using the tau-leaping technique. In the first part of this dissertation, we show how, when a tau-leaping algorithm requests a Poisson random variable to simulate the evolution of a tau-leaping trajectory, we may construct two paths simultaneously by returning an antithetic pair of Poisson random variables and assigning one to each path. Recall that the parameters of these Poisson random variables represent the expected number of system jumps to occur during the discrete time-step, and they may vary at each time as the state and thus the reaction rates of the path evolve. As shown in Section 3.3, when these parameters were typically large (say when the volume or speed of the system were particularly high), the negative correlation between these sample paths, and thus the corresponding reduction in estimator variance was also correspondingly large. Intuitively, this comes from the enhanced symmetry of Poisson CDFs for large parameter values, as is shown in Figures 4.4 and 4.5 in Sec. 4.1.3.

The simplicity of this algorithm comes at a cost, however. By always returning a Poisson random variable pair by inverting a Poisson CDF with the same parameter, we effectively couple the performance of the algorithm to the natural scales of the system. When a system or a particular reaction channel is not expected to experience many transitions, the Poisson parameters will be small and the negative correlation between paths will be correspondingly weakened. This is particularly salient, for example, when consistency and stability concerns [31] may restrict the maximum time-step size used in the tau-leaping algorithm, restricting the number of jumps that are expected to occur in each step.

Another way to illustrate why this coupling might be undesirable is to consider the tau-leaping simulation of a system with a single reaction channel. At each time step, a pair of antithetic Poisson random variables is simulated, and each one is added to the state of their respective paths after being multiplied by the jump vector. The pathwise MSE of an estimator constructed using these paths will depend on the sum of the covariances between each of these pairs. This is analogous to the Poisson sampling performed in the endpoint method for continuous-time simulation. The key distinction is that, instead of a fixed τ_s used as a step-size (and thus Poisson parameter), here the Poisson parameter can vary with each step and for each path. For a process with a constant reaction rate, for example, the analogy becomes an equivalence, modulo some scaling due to the jump vector. Consider then, the red curves in Figs. 4.2 and 4.3. Ignoring the absence of uniform jump-time sampling, the performance of the above tau-leaping algorithm will be similar to the local minima of the red curves of these plots. When τ-step sizes are large, as in Fig. 4.2, the sum of these quantities is relatively small. As typical jump sizes get smaller as in Fig. 4.3, however, the MSE accumulates much more quickly as the negative covariance term becomes small relative to the native variance of a Poisson random variable. This is also clearly shown in Fig. 4.4, where for small parameter values, the negative covariance $\Gamma(\tau)$ due to antithetic sampling becomes small relative to τ. In effect, this illustrates the mechanism by which antithetic tau-leaping's performance degrades as the system slows.

As we will show, this coupling can be broken. The key requirement of the tau-leaping algorithm from a simulation perspective is that each trajectory receives a collection of (conditionally) Poisson-distributed random variables at each time-step, one variable for each reaction channel in order to update its state. Recall by construction in Sec. 2.2, for a given path, each of these Poisson variables were mutually independent (their anticorrelated pair was given to another sample path for use in the corresponding reaction channel) to ensure that the marginal distribution of the trajectory remain unchanged from the iid case. Thus we may generalize Algorithm 1 which simulates lattice DTMC paths using antithetic Poisson variables to describe a new Algorithm 4 that produces lattice DTMC paths using *anticorrelated* Poisson variables.

Algorithm 4 Constructing generalized anticorrelated paths for lattice DTMC systems (2.3)

```
Intialize: $X_0^{(j)} \leftarrow x_0$
for $t = 0$ to $T$ do
    for $i = 1$ to $I$ do
        sample pair $\left(S_t^{i,(1)}, S_t^{i,(2)}\right)$ such that:
        $\mathrm{Cov}\left(S_t^{i,(1)}, S_t^{i,(2)}\right) \leq 0$,
        $S_t^{i,(1)} \sim \mathrm{Pois}(\lambda^i(t, X_t^{(1)}))$,
        $S_t^{i,(2)} \sim \mathrm{Pois}(\lambda^i(t, X_t^{(2)}))$,
        and $\left\{S_t^{i,(r)}\right\}_{i=1}^{I}$ are independent, conditioned on $X_t^{(r)}$ for each $r \in \{1, 2\}$.
    end for
    for $r \in \{1, 2\}$ do
        $X_{t+1}^{(r)} \leftarrow X_t^{(r)} + \sum_{i=1}^{I} S_t^{i,(r)} \nu^i$
    end for
end for
```

Estimators constructed using Alg. 4 will satisfy all of the requirements of Lemma 1 and Theorem 3. That is, they will be unbiased, consistent, and their covariance will evolve in time in a fashion similar to antithetic tau-leaping. For convenience:

Lemma 9. *The mean estimator* Ψ^{NM} *constructed using Alg. 4 is unbiased with respect to the lattice DTMC distribution* (2.3) *and, for fixed* M*, is consistent in* N*.*

Theorem 10. *If* $X^{(1)}, X^{(2)} \in \mathbb{Z}^D \times \mathbb{N}$ *are two realizations that satisfy* (2.3) *and are constructed using Algorithm 4, then their mutual covariance satisfies*

$$\begin{aligned}\operatorname{Cov}(X^{(1)}_{t+1}, X^{(2)}_{t+1}) = &\operatorname{Cov}(X^{(1)}_t, X^{(2)}_t) \\ &+ \sum_{i=1}^{I} \nu^i \operatorname{Cov}(\lambda^i(t, X^{(1)}_t), X^{(2)}_t) + \sum_{i=1}^{I} \operatorname{Cov}(X^{(1)}_t, \lambda^i(t, X^{(2)}_t)) {\nu^i}^\top \\ &+ \sum_{i_1=1}^{I} \sum_{i_2=1}^{I} \nu^{i_1} {\nu^{i_2}}^\top \operatorname{Cov}(\lambda^{i_1}(t, X^{(1)}_t), \lambda^{i_2}(t, X^{(2)}_t)) \\ &+ \sum_{i=1}^{I} \nu^i {\nu^i}^\top \mathbb{E}\big[\big(S^{i,(1)}_t - \lambda^i(t, X^{(1)}_t)\big) \cdot \big(S^{i,(2)}_t - \lambda^i(t, X^{(2)}_t)\big)\big]. \end{aligned} \tag{5.1}$$

The negative covariance between Poisson samples is thus embedded in whatever sampling technique you use to generate negatively correlated Poisson random variables, as opposed to being derived as in say Lemma 4 for the antithetic algorithm. The question then becomes, what alternative to CDF inversion should we consider for anticorrelated Poisson random variable simulation?

5.1 Conditional Poisson sampling

Algorithm 4 allows us to generate Poisson variables using any method we desire that produces negative correlation. The space of such methods is enormous, so to proceed, we return to the analogy we used above, namely that of a single-channel process. We argued that antithetic tau-leaping is analogous to the endpoint method, where the step sizes are variable and correspond to the Poisson parameters being requested by the tau-leaping algorithm. Effectively, this analogy shows that we can use increments of a unit-rate Poisson process to provide inputs to a tau-leaping algorithm. In particular, we will use increments from a pair of negatively correlated unit-rate Poisson processes to produce an anticorrelated pair of Poisson random variables. These Poisson random variables can be returned to a tau-leaping algorithm just as the antithetic pairs obtained by CDF inversion in Alg. 1. In this way, we obtain a source of anticorrelated Poisson random variables whose negative correlation properties are de-coupled from the rates of the system.

Again, there are many potential choices for how exactly to generate antithetic Poisson processes and thus increments that will affect the degree to which samples are negatively correlated. Here, we present one possibility as a jumping off point. Returning to our analogy, one way to improve upon endpoint sampling with small tau-leaps is to take a large Poisson leap into the future, so that the negative covariance between samples is high. Then, as needed we may backfill in required missing process values using antithetic binomial midpoint sampling. As long as we are careful to condition on the future Poisson process values that we've sampled, we can draw from prior points in the process without biasing their distributions. In effect, we consider an algorithm that performs antithetic binomial midpoint sampling (without simulating uniform jump times) using, say, step size τ_{s} and exponential order L to pre-sample a few points on a pair of negatively correlated Poisson processes. Then, when a tau-leaping algorithm request a pair of Poisson samples of parameter $\left(\lambda^1, \lambda^2\right)$, we simply extract a Poisson process increment of length λ^j from process j by antithetically sampling its value once again from a conditional binomial distribution. This algorithm to sample lattice points of the Poisson process in order to sample anticorrelated Poisson process increments can be interpreted as a discrete sub-sampling of Alg. 3 (since no continuous-time iid jump times are required), is summarized in Alg. 5.

Algorithm 5 Binomial Midpoint Method: Conditional Poisson lattice sub-sampling in τ_{s}-increment on 2^L dyadic points

```
$Y_{\mathrm{B}}^i(0) \leftarrow 0,\ i \in \{1,2\}$
Sample antithetic Poisson random variables: $\left(N_{\mathrm{B}}^1(0,\tau_{\mathrm{s}}), N_{\mathrm{B}}^2(0,\tau_{\mathrm{s}})\right) \overset{\text{anti}}{\sim} \mathrm{Pois}(\tau_{\mathrm{s}})$
Set state value at endpoint $\tau_{\mathrm{s}}$: $Y_{\mathrm{B}}^i(\tau_{\mathrm{s}}) \leftarrow N_{\mathrm{B}}^i(0,\tau_{\mathrm{s}}),\ i \in \{1,2\}$
for $\ell = 1$ to $L$ do
  for $k = 1,3,5,\ldots,2^\ell - 1$ do
    Conditionally sample increments at midpoint as antithetic binomial variables (see Eq. (4.4)):
    $Q^i\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}, \frac{k+1}{2^\ell}\tau_{\mathrm{s}}\right) \overset{\text{anti}}{\sim} \mathrm{Bin}\left(Y_{\mathrm{B}}^i\left(\frac{k+1}{2^\ell}\tau_{\mathrm{s}}\right) - Y_{\mathrm{B}}^i\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}\right), \frac{1}{2}\right),\ i \in \{1,2\}$
    Set state value at midpoint: $Y_{\mathrm{B}}^i\left(\frac{k}{2^\ell}\tau_{\mathrm{s}}\right) \leftarrow Y_{\mathrm{B}}^i\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}\right) + Q^i\left(\frac{k-1}{2^\ell}\tau_{\mathrm{s}}, \frac{k+1}{2^\ell}\tau_{\mathrm{s}}\right),\ i \in \{1,2\}$
  end for
end for
Return two $(2^L+1) \times 2$-arrays $A_1$ and $A_2$, where the elements of the first column in each are the sample
times $A_i[k,0] \leftarrow \tau_{\mathrm{s}} k/2^L,\ k \in \{0,\ldots,2^L\}$ and the second columns are the corresponding sample values
$A_i[k,1] \leftarrow Y_{\mathrm{B}}^i\left(\tau_{\mathrm{s}}\frac{k}{2^L}\right)$.
```

The final component of this discrete-time simulation algorithm is mapping sub-sampled Poisson process increments to Poisson random variable samples. Algorithm 6 describes one possible implementation of this scheme. Two negatively correlated arrays of future Poisson process sample values are maintained by appending repeated samples of Alg. 5. When a Poisson random pair is requested with parameters (τ_1, τ_2), each array is searched to find the nearest neighboring points in time that have already been sampled. These nearest neighbors are then used as input into a binomial distribution in order to sample the exact value of the Poisson process τ_i from the previous state. The difference between these values and the iniitial state can be returned as the Poisson random variable $S^{(i)}$. Once both arrays have moved forward in time, values that lie in the intersection of their past values can be discarded.

Algorithm 6 Anticorrelated Poisson variable sampling for parameters (τ_1, τ_2) using pre-sampled Poisson process increments from Alg. 5 with τ_{s}-increment on 2^L dyadic points

```
if $(H_1, H_2) == \emptyset$ then
    Initialize state:
    $(H_1, H_2) \leftarrow (A_1, A_2)$ where $(A_1, A_2)$ is the output of Alg. 5 for $\tau_{\text{s}}, L$
    $\tau_i^{\text{curr}} \leftarrow 0, Y^i(\tau_i^{\text{curr}}) \leftarrow 0$, $i \in \{1, 2\}$
    $\tau^{\text{last samp}} \leftarrow H[-1, 0]$ (where index 0 denotes the first array element and index $-1$ denotes the last element)
    $Y^i(\tau_i^{\text{last samp}}) \leftarrow H_i[-1, 1]$
    $\tau^{\text{final}} \leftarrow \tau_{\text{s}}$
end if
Given input $(\tau_1, \tau_2)$:
$\tau_i^{\text{next}} \leftarrow \tau_i^{\text{curr}} + \tau_i$, $i \in \{1, 2\}$
while $\tau_i^{\text{next}} \geq \tau^{\text{last samp}}$ for some $i$ do
    Sample $(A_1, A_2)$ using Alg. 5 with parameters $\tau_{\text{s}}, L$
    $A_i[k, 0] \leftarrow A_i[k, 0] + \tau^{\text{final}}, \tau^{\text{final}} \leftarrow \tau^{\text{final}} + \tau_{\text{s}}$
    $A_i[k, 1] \leftarrow A_i[k, 1] + Y^i(\tau_i^{\text{last samp}})$, $i \in \{1, 2\}$
    Append $(A_1, A_2)$ to $(H_1, H_2)$
    $Y^i(\tau_i^{\text{last samp}}) \leftarrow H_i[-1, 1] \, i \in \{1, 2\}$
end while
for  $i \in \{1, 2\}$ do
    Find $k_i$ such that $H_i[k, 0] < \tau_i^{\text{next}} < H_i[k + 1, 0]$
    Sample $B_i \sim \text{Binom}\left(H_i[k_i + 1, 1] - H_i[k_i, 1], \frac{\tau_i^{\text{next}} - H_i[k_i, 0]}{H_i[k_i + 1, 0] - H_i[k_i, 0]}\right)$
    $Y^i(\tau_i^{\text{next}}) \leftarrow H_i[k_i, 1] + B_i$
    Create vector $V_i \in \mathrm{R}^2$ such that $V[0] \leftarrow \tau_i^{\text{next}}$, $V_i[1] \leftarrow Y^i(\tau_i^{\text{next}})$.
    Insert row $V_i$ into $H_i$ after row $k_i$
    $S^{(i)} \leftarrow Y^i(\tau_i^{\text{next}}) - Y^i(\tau_i^{\text{curr}})$
    $\tau_i^{\text{curr}} \leftarrow \tau_i^{\text{next}}$, $Y^i(\tau_i^{\text{curr}}) \leftarrow Y^i(\tau_i^{\text{next}})$, $i \in \{1, 2\}$
end for
Remove all rows of $H_i$ before index $\ell$, where $H_i[\ell, 0] = \min\{\tau_1^{\text{curr}}, \tau_2^{\text{curr}}\}$
Return $\left(S^{(1)}, S^{(2)}\right)$
```

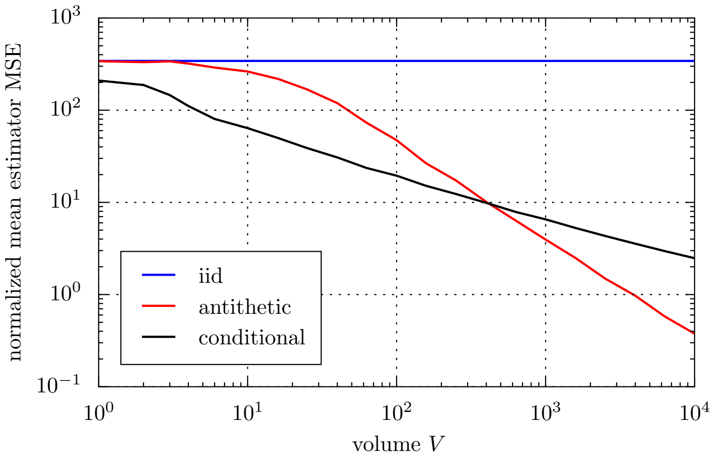

Figure 5.1: Normalized pathwise $\mathrm{MSE}(\Psi^M)M/V$ of an $M = 2$ sample estimator of the mean $\mathbb{E}[X_t]$ of the gene expression system using the iid, antithetic, and conditional tau-leaping sampling techniques plotted versus volume scale V, for timestep $\tau = 1$s, rate parameters $(k_r, \gamma_r, k_p, \gamma_p) = (0.01, 0.03, 0.06, 0.0066)$, initial condition $X_0 = [V\ V/2]^\top$ and timesteps from $t = 0$ to $t = 100$. The conditional Poisson sampling algorithm is run with parameters $\tau_\mathrm{s} = 100$ and $L = 6$. Pathwise MSE calculated from 2.56×10^5 sample paths, error bars are small and are thus omitted.

5.2 Numerical results

First, we consider the same tau-leaping simulation of the gene expression system found in Section 3.3.1. In this parameter sweep, we fix the conditional tau-leaping parameters as step-size $\tau_\mathrm{s} = 100$ and exponential order $L = 6$. We then perform the same parameter sweep as appears in Fig. 3.3 for comparison to the iid and antithetic tau leaping algorithms. The results are shown in Fig. 5.1. Note that, as predicted, for small volume values, i.e., when relatively few transitions occur during each time step, the conditional algorithm that takes large Poisson leaps then back-samples increments to produce Poisson sample values performs significantly better. Note also that, as V becomes large, the typical Poisson parameter sampled by the antithetic algorithm is larger on average than $\tau_s = 100$. This suggests that, in practice, either the value of τ_s should be selected adaptively based on the system parameters. Alternatively, for fixed τ_s, leaps that exceed τ_s could be performed with a simple endpoint leap of length requested, i.e., antithetic tau-leaping as in Chapter 3.

In Fig. 5.2, we present the same parameter study for the coagulation system of Sec. 3.3.2. In this case, the system experiences more rapid transitions, so we increase the conditional tau-leaping parameter to $\tau_\mathrm{s} = 1000$ and keep $L = 6$. Again, the results demonstrate superior performance for small V.

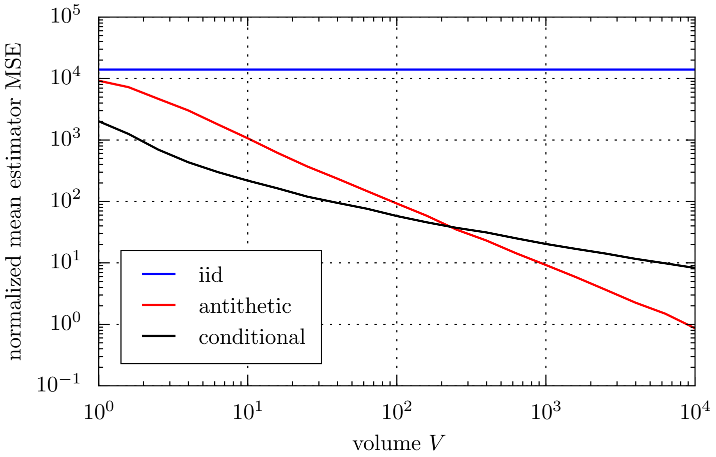

Figure 5.2: Normalized pathwise $\mathrm{MSE}(\Psi^{M})M/V$ of an $M = 2$ sample mean estimator for the nonlinear coagulation system using the iid, antithetic, and conditional tau-leaping sampling techniques plotted versus source area from $V = 10^{0}$ to $V = 10^{4}$. Computation of pathwise MSE from 10^{6} simulations, where each simulation uses timestep $\tau = 0.1$ s, and timesteps from $t = 0$ to $t = 100$. The conditional Poisson sampling algorithm is run with parameters $\tau_{\mathrm{s}} = 1000$ and $L = 6$.Error bars are small and are thus omitted.

Chapter 6

Extensions and Applications

6.1 Variance Reduced Stochastic MPC

Model predictive control (MPC) is a control strategy which seeks to approximate optimal, infinite time horizon feedback control via optimal solution of open loop, finite time horizon problems [26]. The control at time t takes in information about the current state and past control actions to simulate the cost of taking a given set of control actions over a finite time window $[t, t+H-1]$. From these simulations, an optimal control action over this time window can be found, and the *first* of these actions is implemented as the current control action. The state information is then updated, and a new optimal open loop solution is found for the next window $[t+1, t+H]$ and so forth.

In this context, we will focus on control of a perfectly observed Markov process on a countable state space where we attempt to minimize the cumulative sum of a cost function $g(x, u)$. Suppose we have a Markov decision process X described by

$$X_{t+1} = f(X_t, u_t) \tag{6.1}$$

where u_t is a particular control action at time t. Suppose we want to find an optimal policy $u_t = \mu(x_t) \in \mathcal{U}$ such that

$$\mu \in \underset{m \in \mathcal{F}(\mathbb{R}^n, \mathcal{U})}{\operatorname{argmin}} \ \mathbb{E}\left[\sum_{t=0}^{\infty} \beta^t g(X_t, m(X_t))\right], \tag{6.2}$$

where $\mathcal{U}$ is some admissible set of control actions, $\mathcal{F}(\mathbb{R}^n, \mathcal{U})$ is the set of measurable functions from $\mathbb{R}^n$ to $\mathcal{U}$, and $\beta \in (0, 1)$ is a discount factor to ensure boundedness of the sum. This problem is of course challenging for most Markov processes X, and often impossible to solve in closed form. We attempt, however to find an approximate realization of this policy along a particular trajectory by implementing MPC. Specifically, at time t, suppose that our controlled process $X_t = x_t$. We will obtain

$$u^{t,H} \in \underset{\tilde{u}\in\mathcal{U}_t^H}{\operatorname{argmin}} \mathbb{E}\left[\sum_{s=t}^{t+H-1} g(X_s, \tilde{u}_s)|X_t = x_t\right], \tag{6.3}$$

where $u^{t,H}$ is an H-vector of control actions over the finite horizon, and $\mathcal{U}_t^H$ is the admissible set of such sequences at time t, and we consider β close to 1. This optimization problem is over a much smaller space; even naive optimization strategies will suffice for small problems. We then set our current control action to be the first element of $u^{t,H} = (u_t^{t,H}, \dots, u_{t+H-1}^{t,H})$:

$$u_t = \mu^{\mathrm{MPC}}(x_t) := u_t^{t,H}, \tag{6.4}$$

ignoring the rest of the finite horizon optimizer. Time can then be updated to $t+1$, and the control window shifted to $[t+1, t+H]$ to solve for $\mu^{\mathrm{MPC}}(x_{t+1})$. Note here that we never solve for an approximation of the actual optimal policy μ for every state in our countable state space. Instead we solve for an approximation $\mu^{\mathrm{MPC}}(x_t)$ of $\mu(x_t)$, i.e. the evaluation of μ at a particular point on our controlled trajectory. In other words, the algorithm approximately *implements* the optimal policy rather than solving for it in a closed form.

Regardless of the optimization routine used, some approximation of the expectation in (6.3) will be required in order to find a minimizing control action over the finite horizon. Given that our selected control action will depend on minimizing this expectation, errors in approximating it can result in selecting a less optimal policy, producing worse performance in the model predictive controller. Typically this is done via a Monte Carlo ensemble of a large number sample paths initialized at x_t where we sum the cost for each trajectory, and average these costs to accurately approximate the expectation. For complex, noisy or large systems, this repeated simulation can become very costly for accurate estimates, and often actual run-time requirements will impose strict constraints on the available number of Monte Carlo sample paths.

To mitigate this problem, we propose implementing anticorrelated stochastic simulation of the finite horizon window to produce accurate estimates of the expected cost of a control sequence while using fewer Monte Carlo sample paths than traditional iid Monte Carlo simulation. By simulating process paths using Algorithm 1, we may immediately improve estimates of the desired expectation, and as we will show in the next subsection, this results in improved expected cost incurred by the resulting MPC policy. Algorithm 7 summarizes this approach for available Monte Carlo resources of N sample paths.

Algorithm 7 Variance Reduced MPC at time t

```
input: $x_t$
for $\tilde{u} \in \mathcal{U}_t^H$ do
    for $k = 1$ to $N/2$ do
        simulate $\{X_{1,s}^k, X_{2,s}^k\}$ for $s \in [t, t+H-1]$, $X_{j,t}^k = x_t$, and $u = \tilde{u}$ using Algorithm 1
    end for
    compute sample mean: $C(\tilde{u}) \leftarrow \frac{1}{N} \sum_{k=1}^{N/2} \sum_{s=t}^{t+H-1} [g(X_{1,s}^k, \tilde{u}_s) + g(X_{2,s}^k, \tilde{u}_s)]$
end for
select $u^{t,H}$ that minimizes $C(\tilde{u})$
$\mu_t^{\mathrm{MPC}}(x_t) \leftarrow u_t^{t,H}$
```

6.1.1 Numerical Results

Consider the following simple, nonlinear chemical reaction system:

$$\emptyset \overset{\rho_{\mathrm{A}}}{\to} \mathrm{A}$$

$$\mathrm{A} + \mathrm{A} \overset{\rho_{\mathrm{R}}}{\to} \mathrm{B}$$

$$\mathrm{B} \overset{\rho_{\mathrm{B}}}{\to} \emptyset$$

where the reaction rates ρ_{R} and ρ_{B} are given by mass action kinetics

$$\rho_{\mathrm{R}}(x) = \frac{1}{2} \kappa_{\mathrm{R}} x^{\mathrm{A}} (x^{\mathrm{A}} - 1)$$

$$\rho_{\mathrm{B}}(x) = \kappa_{\mathrm{B}} x^{\mathrm{B}},$$

and $\rho_{\mathrm{A}}(u) = \kappa_{\mathrm{A}} u$ is the control input. For simplicity, take $\mathcal{U} = \{u_{\mathrm{LO}} = 10 \,\text{molecules/s}, u_{\mathrm{HI}} = 100 \,\text{molecules/s}\}$ to be binary. Let the state $X_t = (X_t^{\mathrm{A}}, X_t^{\mathrm{B}})^\top$ denote the number of particles of each species at time t. Consider the τ-leaping simulation of this system

$$X_{t+1} = X_t + \sum_{i=1}^{I} S_t^i \zeta^i, \tag{6.5}$$

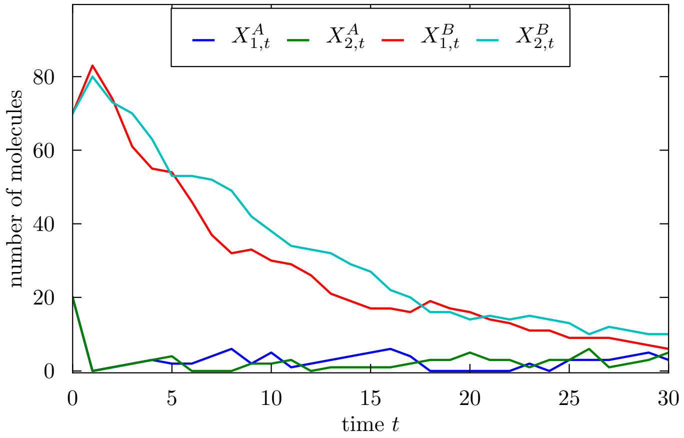

Figure 6.1: Two anticorrelated sample paths of the chemical reaction system with a constant input of $u_{\text{LO}} = 10$ molecules/s.

where $S^i_t \sim \text{Pois}(\lambda^i(X_t, u_t))$ and

$$\begin{aligned}
\zeta^1 &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} & \lambda^1(X_t, u_t) & &= \rho_{\text{A}}(u_t)\tau \\
\zeta^2 &= \begin{pmatrix} -2 \\ 1 \end{pmatrix} & \lambda^2(X_t, u_t) & &= \rho_{\text{R}}(X_t)\tau \\
\zeta^3 &= \begin{pmatrix} 0 \\ -1 \end{pmatrix} & \lambda^3(X_t, u_t) & &= \rho_{\text{B}}(X_t)\tau. \qquad (6.6)
\end{aligned}$$

An antithetic pair of sample open loop trajectories are shown in Figure 6.1 for $u \equiv u_{\text{LO}}$, $\kappa_{\text{A}} = \kappa_{\text{R}} = \kappa_{\text{B}} = 0.1$ and $\tau = 1.0\,\text{s}$.

We define the cost function so that closed-loop trajectories try to stabilize the number of molecules of species B:

$$g(x, u) = |x^{\text{B}} - x_{\text{ref}}| \qquad (6.7)$$

where $x_{\text{ref}} = 30$ molecules. Further, we take actions to be 5 second step functions, so that a decision is made every 5 steps of simulation time. We take the length of the finite time horizon $H = 15$ seconds, so that the optimization problem is over 3 actions and thus brute force search over the action space requires only $|\mathcal{U}^3| = 8$ checks. The exhaustive search clearly scales poorly as the size of the admissible control set or window length grow, but is used here for simplicity. Future work would include a more sophisticated optimization technique. An example closed-loop trajectory computed using either 2 iid sample paths or one antithetic pair of sample paths (i.e. N=2) is shown in Figure 6.2, and its corresponding action sequence is

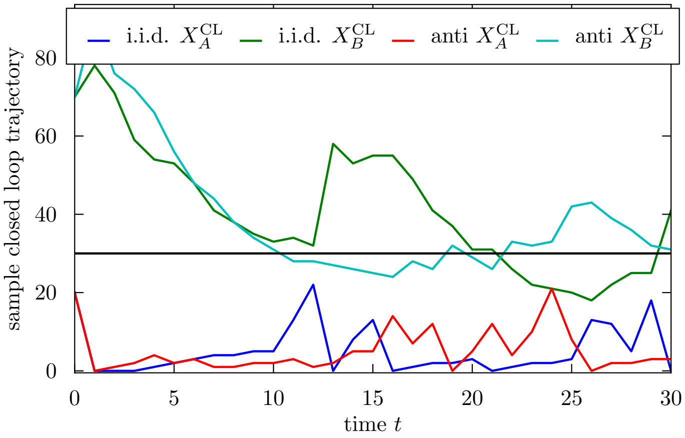

Figure 6.2: Two closed loop sample paths of the chemical reaction system with access to only 2 sample paths to estimate the expected value in (6.3). To estimate the expected cost of a candidate control sequence while running MPC, iid MPC uses two iid sample paths and the antithetic MPC uses two antithetically paired sample paths.

shown in Figure 6.3.

Because the closed loop trajectories, policies and costs are all stochastic, to compare the performance of iid and antithetic MPC we must take a large ensemble of closed loop realizations for each fixed value of N to compute the expected cost of each algorithm. Figure 6.4 plots a Monte Carlo estimate of this cost (along with error bars corresponding to the standard error of the mean, an approximation of a single standard deviation of the average cost) using 3.84e3 samples, versus the number of Monte Carlo sample paths to which the model predictive controller has access. While these average cost estimates are somewhat noisy due to high variance in cost incurred by a closed loop trajectory, we can see marked improvement in the antithetic MPC, achieving roughly the same cost using only 2 Monte Carlo samples as the iid MPC achieves using 4 samples. Note that since both the iid and antithetic estimates of the expectation in (6.3) are consistent [23], the expected cost incurred by each should converge as the ensemble resources N become large.

6.2 Particle filtering with anticorrelated predictions

We proceed by introducing notation used in Crisan and Doucet [12]. Suppose $X = \{X_t, t \in \mathbb{N}\}$ is a stochastic signal process in $\mathbb{R}^{n_x}$ and $Y = \{Y_t, t \in \mathbb{N} \setminus \{0\}\}$ is a corresponding observation process in $\mathbb{R}^{n_y}$. Let the signal process X be Markov with initial distribution $X_0 \sim \mu(\mathrm{d}x_0)$ and probability transition kernel $K(\mathrm{d}x_t|x_{t-1})$

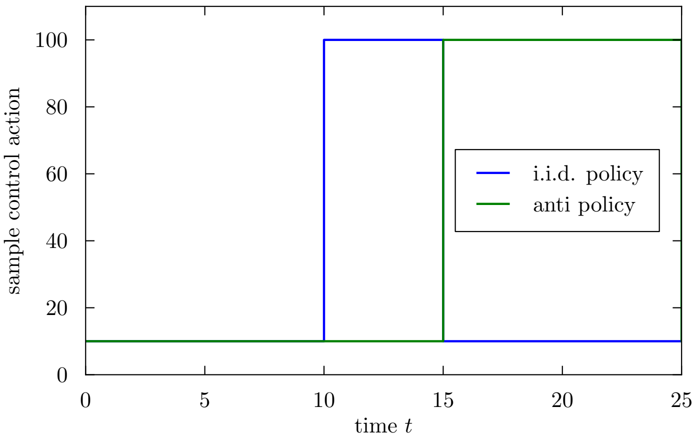

Figure 6.3: The implemented policies used by the closed loop paths in Fig. 6.2. To estimate the expected cost of a candidate control sequence while running MPC, iid MPC uses two iid sample paths and the antithetic MPC uses two antithetically paired sample paths.

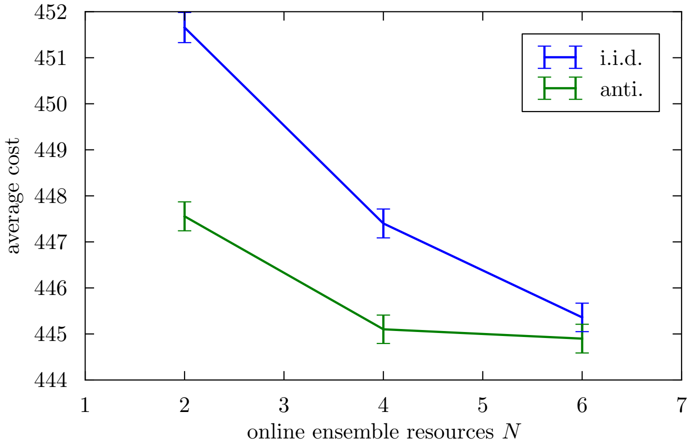

Figure 6.4: The estimated expected closed loop cost incurred by iid MPC and antithetic MPC versus the number of Monte Carlo samples to which they have access for online estimation of expected cost. Average costs are computed using 38,400 sample closed loop paths. Note the antithetic technique requires approximately half the ensemble resources to achieve the same average cost. The error bars show +/- standard error of the mean, which is approximately one standard deviation of the sample average cost.

so that

$$\begin{aligned}\mathbb{P}(X_t \in A | X_{t-1} = x_{t-1}) &= \int_A K(\mathrm{d}x_t | x_{t-1}),\ A \in \mathcal{B}(\mathbb{R}^{n_x}) \\ \mathbb{P}(Y_t \in B | X_t = x_t) &= \int_B g(\mathrm{d}y_t | x_t), \quad B \in \mathcal{B}(\mathbb{R}^{n_y}),\end{aligned}$$

where $\mathcal{B}(\mathbb{R}^n)$ denotes the Borel σ-algebra on $\mathbb{R}^n$. Consider a probability measure $\pi_{s|t}$ the solution of the optimal filtering problem that denotes the law of X_s conditioned on $Y_1, \ldots, Y_t$. We may obtain $\pi_{t|t}$ via a standard, two-step recursive version of Bayes' Theorem, given by:

$$\begin{aligned}&\text{Prediction:} \\ &\pi_{t|t-1}(\mathrm{d}x_t) \\ &\qquad = \int_{\mathbb{R}^{n_x}} \pi_{t-1|t-1}(\mathrm{d}x_{t-1}) K(\mathrm{d}x_t | x_{t-1}) \\ &\text{Update:} \\ &\pi_{t|t}(\mathrm{d}x_t) \\ &\qquad = g(y_t | x_t) \pi_{t|t-1}(\mathrm{d}x_t) \left[\int_{\mathbb{R}^{n_x}} g(y_t | x_t) \pi_{t|t-1}(\mathrm{d}x_t) \right]^{-1}.\end{aligned}$$

For φ a function, ν a measure, and Ξ a Markov transition kernel, define the standard notation:

$$\begin{aligned}(\nu, \varphi) &:= \int \varphi(x) \nu(\mathrm{d}x) \\ \nu\Xi(A) &:= \int \nu(\mathrm{d}x) \Xi(A | x) \\ \Xi\varphi(x) &:= \int \Xi(\mathrm{d}z | x) \varphi(z).\end{aligned}$$

Thus we may more compactly characterize the optimal filtering recursion by:

$$\begin{aligned}(\pi_{t|t-1}, \varphi) &= (\pi_{t-1|t-1}, K\varphi) && \text{Prediction} \\ (\pi_{t|t}, \varphi) &= (\pi_{t|t-1}, \varphi g)(\pi_{t|t-1}, g)^{-1} && \text{Update}\end{aligned}$$

where φ is any continuous, bounded, real-valued function on $\mathbb{R}^{n_x}$.

6.2.1 Particle Filtering

A particle filtering method approximates the optimal filter by maintaining a collection $\{x_t^{(i)}\}_{i=1}^N$ of N samples of the state, known as particles, indexed by time. The empirical measure $\pi_{t|t}^N$ of this collection, given by:

$$\pi_{t|t}^N(\mathrm{d}x_t) := \frac{1}{N}\sum_{i=1}^N \delta_{x_t^{(i)}}(\mathrm{d}x_t), \tag{6.8}$$

is meant to approximate $\pi_{t|t}$, the optimal filtering measure. Note here that δ_x denotes the Dirac delta measure at x. Given this object at time $t-1$, we may draw samples that are approximately distributed according to $\pi_{t|t-1}$ by drawing

$$\tilde{x}^{(i)} \overset{\text{i.i.d.}}{\sim} \pi_{t-1|t-1}^N K(\mathrm{d}x_t) = \frac{1}{N}\sum_{k=1}^N K(\mathrm{d}x_t|x_{t-1}^{(k)}).$$

The update step is performed by computing normalized weights for each particles given the information $Y_t = y_t$, and a resampling is performed to close the loop. One particle filtering algorithm is given by Crisan and Doucet [12]:

```
At time $t = 0$.
Step 0: Initialization
  For $i = 1, \dots, N$, sample $x_0^{(i)} \overset{\text{i.i.d.}}{\sim} \pi_{0|0}$ and set $t = 1$.
At time $t \geq 1$.
Step 1: Importance Sampling
  For $i = 1, \dots, N$, sample $\tilde{x}_t^{(i)} \overset{\text{i.i.d.}}{\sim} \pi_{t-1|t-1}^N K$.
  For $i = 1, \dots, N$, calculate the normalized importance weights $w_t^{(i)}$ :
    $w_t^{(i)} \propto g(y_t|\tilde{x}_t^{(i)})$ such that $\sum_{i=1}^N w_t^{(i)} = 1$.
Step 2: Resampling
  For $i = 1, \dots, N$, sample $x_t^{(i)} \overset{\text{i.i.d.}}{\sim} \tilde{\pi}_{t|t}^N := (1/N)\sum_{i=1}^N w_t^{(i)} \delta_{\tilde{x}_t^{(i)}}$
```

6.2.2 Anticorrelated Sampling

Beyond the antithetic random variable generation discussed above, several other examples of the larger class of anticorrelated sampling techniques are introduced in [23]. In addition to antithetic sampling, stratified sampling as well as a hybridization of stratified and antithetic sampling are examined. They are also defined in terms of the simulation of random variables via inversion of their CDFs. In particular, one may simulate a sequence of i.i.d. random variables X_k with distribution function F and law ν by sampling:

$$U_k \overset{\text{i.i.d.}}{\sim} \text{Unif}(0,1) \tag{6.9}$$

$$X_k := F^{-1}(U_k), \tag{6.10}$$

where we formally invert the CDF by defining:

$$F^{-1}(u) := \inf\{x : F(x) \geq u\}. \tag{6.11}$$

An empirical estimate of ν is a random measure that approximates ν. It can be computed via:

$$\nu^{N,\omega} = \frac{1}{N}\sum_{k=1}^{N} \delta_{X_k(\omega)} \approx \nu, \tag{6.12}$$

for N sufficiently large. For convenience we may suppress the explicit dependence on the outcome ω, but all empirical measures constructed in this way are random. We may attempt to accelerate this convergence by introducing localized correlation into the samples, say antithetically, by:

$$U_k^{\text{A}} \overset{\text{i.i.d.}}{\sim} \text{Unif}(0,1) \tag{6.13}$$

$$X_{k,1}^{\text{A}} := F^{-1}(U_k^{\text{A}}) \tag{6.14}$$

$$X_{k,2}^{\text{A}} := F^{-1}(1 - U_k^{\text{A}}) \tag{6.15}$$

$$\nu^{\text{A},N} = \frac{1}{N}\sum_{k=1}^{N/2} \Big(\delta_{X_{k,1}^{\text{A}}} + \delta_{X_{k,2}^{\text{A}}}\Big), \tag{6.16}$$

or via uniform stratification of the random variate sampled in $[0,1)$:

$$A_j := \left[\tfrac{j-1}{M}, \tfrac{j}{M}\right) \text{ for } j = 1, \ldots, M \tag{6.17}$$

$$U_{k,j}^{\text{S}} \overset{\text{i.i.d.}}{\sim} \text{Unif}(A_j) \text{ for } j = 1, \ldots, M \tag{6.18}$$

$$X_{k,j}^{\text{S}} := F^{-1}(\tilde{U}_{k,j}^{\text{S}}) \text{ for } j = 1, \ldots, M \tag{6.19}$$

$$\tilde{U}_{k,j}^{\text{S}} := (\Pi^M \vec{U}_k^{\text{S}})_j, \tag{6.20}$$

where Π^M is a random $M \times M$ permutation matrix. The empirical measure constructed from these samples is given by:

$$\nu_M^{\mathrm{S},Mn} := \frac{1}{Mn}\sum_{k=1}^{n}\sum_{j=1}^{M}\delta_{X_{k,j}^{\mathrm{S}}}. \tag{6.21}$$

A hybridization of these techniques is easily constructed by

$$U_{k,j}^{\mathrm{H}} \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(A_j) \text{ for } j = 1, \dots, \tfrac{M}{2} \tag{6.22}$$

$$U_{k,j}^{\mathrm{H}} := 1 - U_{k,M-j+1}^{\mathrm{H}} \text{ for } j = (\tfrac{M}{2}+1), \dots, M \tag{6.23}$$

$$X_{k,j}^{\mathrm{H}} := F^{-1}(\tilde{U}_{k,j}^{\mathrm{H}}) \text{ for } j = 1, \dots, M \tag{6.24}$$

$$\text{where } \tilde{U}_{k,j}^{\mathrm{H}} := (\Pi^M \vec{U}_k^{\mathrm{H}})_j \tag{6.25}$$

$$\nu_M^{\mathrm{H},Mn} := \frac{1}{Mn}\sum_{k=1}^{n}\sum_{j=1}^{M}\delta_{X_{k,j}^{\mathrm{H}}}, \tag{6.26}$$

for Π^M a random permutation matrix. In order to make precise the concept of convergence of measures, we introduce (weak) convergence of a sequence of measures $\{\mu_N\}_{N=1}^{\infty}$ to another measure μ if, for any $\varphi \in C_b(\mathbb{R}^{n_x})$ the space of continuous, bounded functions:

$$\lim_{N\to\infty}(\mu_N, \varphi) = (\mu, \varphi). \tag{6.27}$$

Further, one can choose a countable set $\mathcal{A} = \{\varphi_i : i \in \mathbb{N}\}$ such that the above condition holds for every $\varphi \in \mathcal{A}$ if and only if μ_N converges weakly to μ as $N \to \infty$. We may also define a metric d on the set $\mathcal{P}(\mathbb{R}^{n_x})$ of probability measures on $\mathbb{R}^{n_x}$ which induces this weak topology, given by:

$$d(\mu, \nu) = \sum_{i=1}^{\infty}\frac{|(\mu, \varphi_i) - (\nu, \varphi_i)|}{2^i\|\varphi_i\|}, \tag{6.28}$$

where $\|\varphi\| := \sup_{x\in\mathbb{R}^{n_x}}|\varphi(x)|$, the supremum norm on $C_b(\mathbb{R}^{n_x})$. That is, $\mu_N \to \mu$ weakly as $N \to \infty$ if and only if $\lim_{N\to\infty} d(\mu_N, \mu) = 0$ (see Crisan and Doucet [12]).

6.2.3 Anticorrelated particle filter in one dimension

We may express a version of the particle filter in terms of a composition of mappings from $\mathcal{P}(\mathbb{R})$ to itself. First, we construct the two continuous maps used in the optimal filter. Define $b_t : \mathcal{P}(\mathbb{R}) \to \mathcal{P}(\mathbb{R})$ to be

$$b_t(\nu)(\mathrm{d}x_t) := \nu K(\mathrm{d}x_t) = \int_{\mathbb{R}} K(\mathrm{d}x_t|x_{t-1})\nu(\mathrm{d}x_{t-1}), \tag{6.29}$$

for any $\nu \in \mathcal{P}(\mathbb{R})$. It is shown in [12] that this map is continuous in the sense of weak topology if the Markov transition kernel K is Feller. Observe that for any $\varphi \in \mathcal{C}_b(\mathbb{R})$,

$$(b_t(\nu), \varphi) = (\nu, K\varphi), \tag{6.30}$$

and also that

$$\pi_{t|t-1} = b_t(\pi_{t-1|t-1}). \tag{6.31}$$

Similarly, we may define another map on measures, a_t, by its pairing with an arbitrary continuous, bounded function φ as:

$$(a_t(\nu), \varphi) = (\nu, g)^{-1}(\nu, \varphi g), \tag{6.32}$$

where a sufficient condition for the continuity of a_t is that $g(y_t|\cdot)$ be bounded, continuous and strictly positive. Here, it is also clear that

$$\pi_{t|t} = a_t(\pi_{t|t-1}) = a_t(b_t(\pi_{t-1|t-1})). \tag{6.33}$$

We may also further define the maps k_t and $k_{1:t}$ to be

$$k_t := a_t \circ b_t \tag{6.34}$$

$$k_{1:t} := k_t \circ k_{t-1} \circ \cdots \circ k_1, \tag{6.35}$$

which are continuous if a_t and b_t are for every t. Note then that for initial distribution μ, we may express the optimal filter as

$$\pi_{t|t} = k_t(\pi_{t-1|t-1}) = k_{1:t}(\mu). \tag{6.36}$$

Now, in order to express a class of particle filters in these terms, define the random mapping $c^{N,\omega}$ from a measure ν to its empirical measure by:

$$c^{N,\omega}(\nu)(\mathrm{d}x_t) = \frac{1}{N}\sum_{j=1}^{N} \delta_{V_j(\omega)}(\mathrm{d}x_t), \tag{6.37}$$

where $V_j \sim \nu$ are i.i.d.. It is clear from the strong law of large numbers that $c^{N,\omega}(\nu) \to \nu$ weakly as $N \to \infty$ for almost every ω by pairing with an arbitrary $\varphi \in \mathcal{C}_b(\mathbb{R})$. In terms of the above notation, $\nu^{N,\omega} = c^{N,\omega}(\nu)$. The above particle filtering algorithm can thus be expressed as:

$$\begin{aligned} &\text{Prediction:} && \pi^N_{t|t-1} = c^N \circ b_t(\pi^N_{t-1|t-1}) \\ &\text{Update and Resampling:} && \pi^N_{t|t} = c^N \circ a_t(\pi^N_{t|t-1}), \end{aligned}$$

initialized at $\mu^N := c^N(\mu)$. Similarly, define maps k^N_t and $k^N_{1:t}$ to be

$$k^N_t := c^N \circ a_t \circ c^N \circ b_t \tag{6.38}$$

$$k^N_{1:t} := k^N_t \circ k^N_{t-1} \circ \cdots \circ k^N_1, \tag{6.39}$$

so that

$$\pi^N_{t|t} = k^N_t(\pi^N_{t-1|t-1}) = k^N_{1:t}(\mu^N). \tag{6.40}$$

It was shown in [12] that this version of the particle filter converges in the weak sense to the optimal filter (almost surely in the random map $c^{N,\omega}$). We now propose an extension of this particle filter to methods which use negative correlation in the prediction step. Instead of sampling the distributions in an i.i.d. way, sample them according to the above algorithms. For example, construct the antithetic empirical map

$$c^{\mathrm{A},N}(\nu)(\mathrm{d}x_t) := \nu^{\mathrm{A,N}} = \frac{1}{N}\sum_{k=1}^{N/2}\Big(\delta_{V^{\mathrm{A}}_{k,1}} + \delta_{V^{\mathrm{A}}_{k,2}}\Big), \tag{6.41}$$

where the samples $(V^{\mathrm{A}}_{k,1}, V^{\mathrm{A}}_{k,2}),\ k = 1, \dots, N/2$ used are i.i.d. in k and $V^{\mathrm{A}}_{k,1}$ and $V^{\mathrm{A}}_{k,2}$ are pairwise antithetically sampled as above. We may then define the antithetic particle filter $\pi^{\mathrm{A},N}_{t|t}$

$$\begin{aligned}
&\text{Initialization:} && \pi^{\mathrm{A},N}_{0|0} = \mu^{\mathrm{A},N} := c^{\mathrm{A},N}(\mu)\\
&\text{Prediction:} && \pi^{\mathrm{A},N}_{t|t-1} = c^{\mathrm{A},N} \circ b_t(\pi^{\mathrm{A},N}_{t-1|t-1})\\
&\text{Update and Resampling:} && \pi^{\mathrm{A},N}_{t|t} = c^{N} \circ a_t(\pi^{\mathrm{A},N}_{t|t-1}).
\end{aligned}$$

Similarly, we may define stratified and hybrid particle filters using M uniform strata of $[0,1)$ in terms of the empirical map

$$c^{\alpha,N}(\nu)(\mathrm{d}x_t) := \nu^{\alpha,N}_{M} = \frac{1}{N}\sum_{k=1}^{N/M}\sum_{j=1}^{M}\delta_{V^{\alpha}_{k,j}}, \tag{6.42}$$

for $\alpha \in \{\mathrm{S}, \mathrm{H}\}$, that is, where $V^{\mathrm{S}}_{k,j} \sim \nu$ and $V^{\mathrm{H}}_{k,j} \sim \nu$ are sampled via stratification and the hybrid technique, respectively. We construct the stratified and hybrid particle filters exactly as above with A replaced with S and H, respectively. We now show convergence of these techniques in the weak sense almost surely, and we expect these techniques to be MSE non-increasing by results proven in [25].

6.2.4 Almost Sure Convergence

We proceed by proving a strong property of the convergence of $c^{\alpha,N}$ to identity, and then the desired result follows immediately via a Lemma in [12].

Lemma 11. *For $c^{\alpha,N,\cdot}$ as above and any sequence of measures $\{\nu_N\}_{N=1}^{\infty}$ such that $\nu_N \to \nu$ as $N \to \infty$, then $c^{\alpha,N,\cdot}(\nu_N) \to \nu$ almost surely for $\alpha \in \{\mathrm{A}, \mathrm{S}, \mathrm{H}\}$*

Proof. Suppose $\{\nu_N\}_{N=1}^{\infty}, \nu \in \mathcal{P}(\mathbb{R})$ are any such measures. For any $\varphi_i \in \mathcal{A}$, any empirical map $c^{\alpha,N}, \alpha \in \{\mathrm{A}, \mathrm{S}, \mathrm{H}\}$, and any number M of correlated variables per i.i.d. sample (i.e. $M = 2$ for antithetic sampling, and M is the number of strata for stratified or hybrid),

$$\begin{aligned}
&\mathbb{E}\left[\left((c^{\alpha,N}(\nu_N), \varphi_i) - (\nu_N, \varphi_i)\right)^4\right]\\
&= \quad \mathbb{E}\left[\left(\frac{1}{N}\sum_{k=1}^{N/M}\sum_{j=1}^{M}[\varphi_i(V^{\alpha}_{k,j}) - (\nu_N, \varphi_i)]\right)^4\right].
\end{aligned} \tag{6.43}$$

For compactness, define $a^i_{k,j} := \varphi_i(V^{\alpha}_{k,j}) - (\nu_N, \varphi_i)$ and observe that $a^i_{k,j},\ k \in \{1, \dots, n/M\}, j \in \{1, \dots, M\}$ are random, independent in k, correlated in j and $\mathbb{E}[a^i_{k,j}] = 0$ since $V^{\alpha}_{k,j} \sim \nu_N$. Then we have, using these facts

$$\begin{aligned}
\mathbb{E}\left[\left((c^{\alpha,N}(\nu_N),\varphi_i)-(\nu_N,\varphi_i)\right)^4\right] = \quad & \mathbb{E}\left[\left(\frac{1}{N}\sum_{k=1}^{N/M}\sum_{j=1}^{M} a^i_{k,j}\right)^4\right] \\
= \quad & \frac{1}{N^4}\mathbb{E}\left[\sum_{k=1}^{N/M}\left(\sum_{j=1}^{M} a^i_{k,j}\right)^4\right] \\
& + \frac{6}{N^4}\mathbb{E}\left[\sum_{\substack{k_1=1\\k_2=k_1+1}}^{N/M}\left(\sum_{j_1=1}^{M} a^i_{k,j}\right)^2\left(\sum_{j_2=1}^{M} a^i_{k,j}\right)^2\right] \\
& = 2^4\|\varphi_i\|^4\frac{M^3+3NM^2-3M^3}{N^3} \\
& \le \frac{48M^2\|\varphi_i\|^4}{N^2},
\end{aligned}$$

for sufficiently large N. Thus

$$\mathbb{E}\left[\sum_{N=1}^{\infty}\left((c^{\alpha,N}(\nu_N),\varphi_i)-(\nu_N,\varphi_i)\right)^4\right] \quad \le 48M^2\|\varphi_i\|^4\sum_{N=1}^{\infty}\frac{1}{N^2}<\infty.$$

Thus, with probability 1

$$\sum_{N=1}^{\infty}\left((c^{\alpha,N}(\nu_N),\varphi_i)-(\nu_N,\varphi_i)\right)^4<\infty \tag{6.44}$$

$$\implies \lim_{N\to\infty}|(c^{\alpha,N}(\nu_N),\varphi_i)-(\nu_N,\varphi_i)|=0 \tag{6.45}$$

for any $i\in\mathbb{N}$. Thus $\lim_{N\to\infty} d(c^{\alpha,N}(\nu_N),\nu_n)=0$, almost surely, and by the triangle inequality

$$\begin{aligned}
&\lim_{N\to\infty} d(c^{N,\omega}(\nu_N),\nu) \\
&\quad \le \lim_{N\to\infty} d(c^{N,\omega}(\nu_N),\nu_N)+\lim_{N\to\infty} d(\nu_N,\nu)=0,
\end{aligned}$$

and almost sure convergence follows. □

Theorem 12. *For transition kernel K Feller and likelihood function g bounded, continuous and strictly positive, then $\lim_{N\to\infty}\pi^{\alpha,N}_{t|t}=\pi_{t|t}$ almost surely for $\alpha\in\{\mathrm{A},\mathrm{S},\mathrm{H}\}$*

Proof. As was proven in [12], for a_t, b_t continuous, $c^N, c^{\alpha,N}$ endowed with the property proven in Lemma 11, for $\lim_{N\to\infty} \mu^{\alpha,N} = \lim_{N\to\infty} c^{\alpha,N}(\mu) = \mu$,

$$\lim_{N\to\infty} \pi_{t|t}^{\alpha,N} = \lim_{N\to\infty} k_{1:t}^{\alpha,N}(\mu^{\alpha,N}) = k_{1:t}(\mu) = \pi_{t|t}. \tag{6.46}$$

□

6.2.5 Room Population Dynamics

As a concrete illustration of the proof above, consider X_t to be the population of a single, initially empty room into which people enter at exponentially spaced times, and their arrivals are independent of each other. This can be modeled as a Poisson process with, say, unit rate. Suppose that, at every time t, an observer attempts to count the number of occupants but has some probability of over or undercounting. Let this count be denoted Y_t and suppose that we can model the accuracy of the observer by

$$Y_t = X_t + N_t \tag{6.47}$$

where N_t is an i.i.d. noise process with distribution with pmf

$$\mathbb{P}(N_t = z) = \begin{cases} \frac{\kappa}{z^4} & \text{if } z \in \mathbb{Z} \setminus \{0\} \\ \kappa & \text{if } z = 0, \end{cases} \tag{6.48}$$

where $\kappa = (\frac{\pi^4}{45} + 1)^{-1}$. We will henceforward refer to this distribution as a quartic power law.

To accurately guess the state X_t, given all of the past measurements and knowledge of the system, we must solve the optimal Bayesian filtering problem for the measure $\pi_{t|t}$. However, this is in general a difficult quantity to calculate, so we seek some computationally efficient method of approximation, the particle filtering distribution $\pi^N t|t$.

We initialize a population of particles $\{X_t^{(i)}\}_{i=1}^N$ at the point $X_0^{(i)} = 0$. Now, given a particle population at time $t-1$, we use our knowledge of the process X_t to predict the next transition. This problem is simple enough that we can efficiently determine the distribution function of the measure $\pi_{t|t}^N K \in \mathcal{P}(\mathbb{R})$, which is just a set of points which advance with independent Poisson distributions. In this case, it is given by

$$F(z) = \mathbb{P}(\tilde{X}_t^{(i)} \leq z) = \frac{1}{N} \sum_{k=1}^{N} F_\lambda(\max\{z - X_{t-1}^{(k)}, 0\}), \tag{6.49}$$

where $F_\lambda(z)$ is the CDF of a Poisson distribution with rate λ. Given access to a CDF, implementation of the

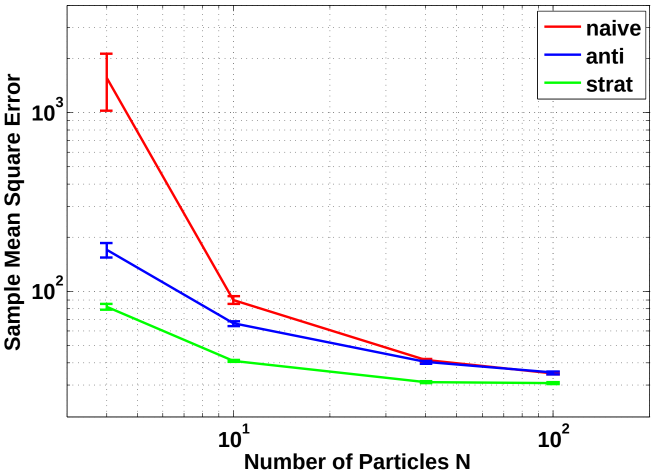

Figure 6.5: Plot of sample mean square error of the mean of the empirical particle distribution from an ensemble of random sample paths of the Poisson process. Error bars shown are standard error of the mean. Here, ensemble size is 100. Note the apparent convergence to the Bayesian limit.

antithetic or stratified filters is trivial, so we may correlate our predictions as above. Finally, we compute likelihood weights for each of our samples, and resample to get a new population.

6.2.6 Numerical Results

Consider a single sample of this process, comprised of a sample trajectory X_t^e, a sample measurement process Y_t^e, and a particle filter of N particles, with corresponding mean estimator

$$\hat{X}_t^{e,N} = \frac{1}{N}\sum_{j=1}^{N} X_t^{e,(j)}. \tag{6.50}$$

If, for every e we compute the mean square error (MSE) $|X^e - \hat{X}^e|^2$ and average over an ensemble $\{e\}_{e=1}^{100}$, we will approximate the expected MSE of the particle filter estimator, $\mathbb{E}[|X - (\pi^N, x)|^2]$. Further, as the number of particles N becomes sufficiently large, this quantity should approach the optimal Bayesian limit $\mathbb{E}[|X - (\pi, x)|^2] = \mathbb{E}[|X - \mathbb{E}[X|Y]|^2] > 0$ for a non-fully observed system. Fig. 6.5 collects the results of these numerical experiments. Note that, relative to the Bayesian limit, the anticorrelated samplers can produce more than an order of magnitude reduction MSE of the naive sampler.

6.2.7 Multidimensional anticorrelated particle filtering

One way to extend the variance reduced algorithms presented above to a multi-dimensional setting is via the tau-leaping method of Gillespie [19] for simulating Markov processes with a finite number of event channels.

Consider the random time-change representation of a Markov process $X(t) \in \mathbb{R}^D$, $t \in [0, T]$, with I event channels, each with propensity function $\rho^i(t, X(t))$, defined by

$$X(t) = X(0) + \sum_{i=1}^{I} \Upsilon^i \left(\int_0^t \rho^i(s, X(s))\, \mathrm{d}s \right) \zeta^i, \tag{6.51}$$

where Υ^i is a unit-rate Poisson process and $\zeta^i \in \mathbb{R}^D$ are the state jump vectors, so that $\zeta^i = X(t^+) - X(t^-)$, if the ith event channel experiences a transition at time t. The evolution of such a process can be approximated in discrete time using the tau-leaping method. For time-step increment τ, let $t_\ell = \ell\tau$ and $\widetilde{X}_\ell \approx X(t_\ell)$ for $\ell \in \{0, \ldots, L\}$, where $L := \max\{\ell : t_\ell \leq T\}$. Then $\widetilde{X}_\ell$ evolves via

$$\widetilde{X}_{\ell+1} = \widetilde{X}_\ell + \sum_{i=1}^{I} S_\ell^i \left(\rho^i(t_\ell, \widetilde{X}_\ell)\tau \right) \zeta^i, \tag{6.52}$$

where $S_\ell^i(\lambda) \sim \text{Pois}(\lambda)$. For compactness, define $\lambda_\ell^i = \rho^i\big(t_\ell, \widetilde{X}_\ell\big)\tau$ and denote $S_\ell^i\left(\lambda_\ell^i\right)$ by S_ℓ^i. Thus (6.52) becomes

$$\widetilde{X}_{\ell+1} = \widetilde{X}_\ell + \sum_{i=1}^{I} S_\ell^i \zeta^i. \tag{6.53}$$

If we couple this discrete time stochastic process with an observation process Y_t, say

$$\widetilde{X}_{t+1} = \widetilde{X}_t + \sum_{i=1}^{I} S_t^i \zeta^i \tag{6.54}$$

$$Y_t = f(X_t, N_t), \tag{6.55}$$

where N_t is some independent noise process, and $X_0 \sim \mu$, then we have produced a whole class of mulitdimensional filtering problems for finding $\pi_{t|t}$, the law of X_t given the observations $Y_1, Y_2, \ldots, Y_t$. The particle filtering approximation to the solution of this problem can be obtained exactly as above, though often practitioners will implement a slightly modified algorithm known as the bootstrap particle filter [21]. The primary difference between a bootstrap filter and the particle filter presented above is that, in the prediction step, instead of drawing

$$\tilde{x}_t^{(j)} \overset{\text{i.i.d.}}{\sim} \pi_{t-1|t-1}^N K(\mathrm{d}x_t) = \frac{1}{N} \sum_{k=1}^{N} K(\mathrm{d}x_t | x_{t-1}^{(k)}).$$

we merely simulate the particle $x_t^{(j)}$ moving forward in time in isolation, that is

$$\tilde{x}_t^{(j)} \sim K(\mathrm{d}x_{t-1} | x_{t-1}^{(j)}).$$

So in this case, given a particle population $\{x_t^{(j)}\}_{j=1}^N$ at time t, to perform the prediction step, for each particle $x_t^{(j)}$ we sample S_t^i for each $i \in \{1, \dots, I\}$ and compute

$$\tilde{x}_{t+1}^{(j)} = x_t^{(j)} + \sum_{i=1}^{I} S_t^i \zeta^i, \tag{6.56}$$

and the rest of the algorithm follows as before. We can see now that the special structure of this model alows us to reduce the stochastic simulation of a multidimensional process to the simulation of I random, real-valued variables, corresponding with each reaction process. Thus the technique we propose for implementing anticorrelated particle filtering is the application of the one dimensional techniques to each of these random variables. Since the techniques as constructed produced fully realized (yet correlated) samples from their respective distributions, transitions simulated this way are fair sample transitions of the process. To proceed we first discuss the application of the stratified technique for M strata A_j constructed as before. For each $j \in \{1, \dots, M\}$ and for any $i \in \{1, \dots, I\}$ and any time step t, take $U_{j,t}^{\mathrm{S},i} \overset{\text{ind.}}{\sim} \mathrm{Unif}(A_j)$. For a uniformly distributed random permutation $\Pi_t^{\mathrm{S},M}$, set $\vec{V}_t^{\mathrm{S},i} = \Pi_t^{\mathrm{S},M} \vec{U}_t^{\mathrm{S},i}$. Now, for each $r \in \{1, \dots, M\}$ define

$$\widetilde{X}_{r,t+1}^{\mathrm{S}} = \widetilde{X}_{r,t}^{\mathrm{S}} + \sum_{i=1}^{I} S_{r,t}^{\mathrm{S},i} \zeta^i, \tag{6.57}$$

for given deterministic $\widetilde{X}_{r,0}^{\mathrm{S}} = X_r^{\mathrm{S}}(0)$, where we sample $S_{r,t}^{\mathrm{S},i}$ via

$$S_{r,t}^{\mathrm{S},i} := F_{\lambda_{r,t}^{\alpha,i}}^{-1} \left(V_{r,t}^{\mathrm{S},i} \right), \tag{6.58}$$

and where $\lambda_{r,t}^{\alpha,i} := \rho^i\big(t_t, \widetilde{X}_{r,t}^{\mathrm{S}}\big)\tau$ and F_λ is the Poisson CDF with parameter λ.

The construction of the samples used for the other variance-reduced pathwise mean estimators follow a similar development. Hybrid paths $\widetilde{X}_{r,t}^{\mathrm{H}}$ are simulated almost identically to the stratified case, save that

$$U_{j,t}^{\mathrm{H},i} \sim \mathrm{Unif}(A_j) \qquad \text{if } 1 \le j \le \frac{M}{2} \tag{6.59}$$

$$U_{j,t}^{\mathrm{H},i} = 1 - U_{M+1-j,t}^{\mathrm{H},i} \qquad \text{otherwise.} \tag{6.60}$$

Then, for a uniformly distributed random permutation $\Pi_t^{\mathrm{H},M}$, $\vec{V}_t^{\mathrm{H},i} := \Pi_t^{\mathrm{H},M} \vec{U}_t^{\mathrm{H},i}$, and $\widetilde{X}_{r,t}^{\mathrm{H}}$ is constructed as above.

Finally, in the case of the antithetic estimator, to generate an even number M of paths $\widetilde{X}_r^{\mathrm{A}}$, simulate $V_{r,t}^{\mathrm{A},i} \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(0,1)$ for each $r \in \{1, \dots, \frac{M}{2}\}$, and for each event channel i and timestep t. We then simulate

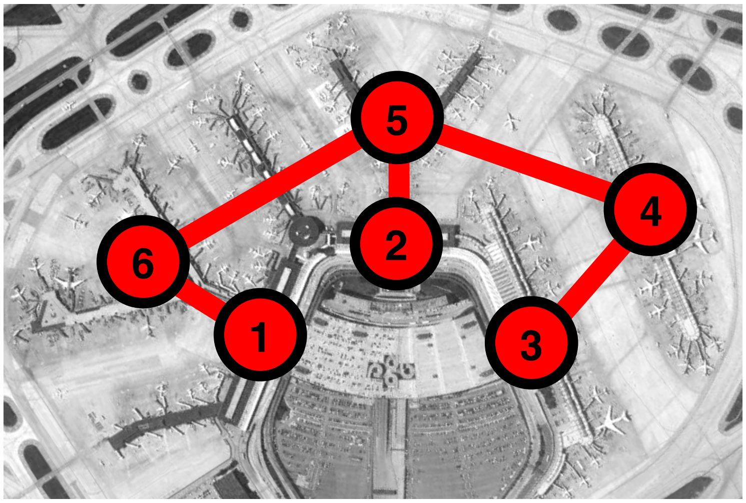

Figure 6.6: Six node graph of O'Hare International Airport's domestic terminals (Source: www.allairports.net/ chicago/chicago-airport-terminal-map.htm). State $X_t \in \mathbb{R}^6$ is the population of each node. Measurements Y_t are taken at nodes 1, 2, 3, and 5.

$\widetilde{X}^{\mathrm{A}}_{r,t}$ as above, save that we define

$$S^{\mathrm{A},i}_{r,t} := \begin{cases} F^{-1}_{\lambda^{\alpha,i}_{r,t}}\left(V^{\mathrm{A},i}_{r,t}\right) & \text{if } 1 \leq r \leq \frac{M}{2} \\ F^{-1}_{\lambda^{\alpha,i}_{r,t}}\left(1 - V^{\mathrm{A},i}_{M+1-r,t}\right) & \text{if } \frac{M}{2}+1 \leq r \leq M. \end{cases} \tag{6.61}$$

While the multidimensional extension of the technique presented here appears natural and well motivated, extension of the proof of Theorem 12 to the multidimensional case may require a different mathematical approach, and proof of such a theorem is reserved for future work.

6.2.8 Building Population Dynamics

Take as an example a simple model of the movement of people through departures side of the three domestic terminals of O'Hare International airport. The airport is represented by a six node graph, with each terminal comprised of a pre- and post-security node, connected as shown in Figure 6.6. The state $X_t \in \mathbb{R}^6$ represents the population at each node at time t. There are source type event channels at nodes 1, 2 and 3 which correspond to jump vectors $\zeta^i = u_i$ the ith standard unit vector in $\mathbb{R}^6$ for $i \in \{1, 2, 3\}$. The corresponding rates at which these transitions occur are simple time dependent, piecewise linear functions (though they needn't be, in general), $\rho_i(t, X_t) = \rho_i(t)\eta$, where $\eta = 100$ is a model scaling factor. There are sinks at nodes 4, 5 and 6, with corresponding jump vectors $\zeta^i = -u_i$ for $i \in \{4, 5, 6\}$, and time dependent, linear rate functions $\rho_i(t, X_t) = \rho_i(t)(X_t)_i$, where $(x)_i$ denotes the ith component of $x \in \mathbb{R}^6$, and $\rho_i(t)$ is also piecewise linear. A jump transition i where a person moves from node ℓ to node m along an edge of the graph has corresponding jump vector $\zeta^i = u_m - u_\ell$. Such transitions from node $\ell \in \{1,2,3\}$ to node $m \in \{4,5,6\}$ along an edge of the graph are taken to be linear in the state, with no explicit time dependence, of the form $\rho_i(t, X_t) = \frac{3\gamma}{2}(X_t)_\ell$, where $\gamma = 0.12$ is another scaling factor. Transitions in the opposite direction, that is from $m \in \{4,5,6\}$ to $\ell \in \{1,2,3\}$ are taken to have rate 0. Finally people are free to move in any direction along a single edge from node ℓ to node m, where $\ell, m \in \{4,5,6\}$. Such a transition i is taken to have nonlinear rate function

$$\rho_i(t, X_t) = \frac{\gamma}{2}\left((X_t)_5 + \frac{(X_t)_5^2}{2\eta}\right) \tag{6.62}$$

if the transition is leaving node 5 and

$$\rho_i(t, X_t) = \frac{\gamma}{3}\left((X_t)_\ell + \frac{(X_t)_\ell^2}{2\eta}\right),\ \ell \in \{4,6\} \tag{6.63}$$

if the transition is entering node 5. Noisy, low resolution measurements of part of the state are available at nodes 1, 2, 3, and 5, and are given by

$$(Y_t)_\ell = \chi\left(\left\lceil \frac{(X_t)_\ell}{\chi} \right\rceil + N_t\right),\ \ell \in \{1,2,3,5\} \tag{6.64}$$

where χ is a scaling factor determining resolution as well as variance of the noise, and N_t is an i.i.d. noise sequence drawn from a quartic power law.

6.2.9 Numerical Results

We see in Fig. 6.7 a single sample path of this process, along with the mean estimator produced by a 100 particle filter. We can repeat the experiment in the scalar case, save that we now compute MSE in terms of norms squared $\|X^e - \hat{X}^e\|^2$ and approximate the expected MSE of the multidimensional particle filter estimator, $\mathbb{E}[\|X - (\pi^N, x)\|^2]$. Again, as the number of particles N becomes sufficiently large, this quantity should converge to the optimal Bayesian limit $\mathbb{E}[\|X - (\pi, x)\|^2] = \mathbb{E}[\|X - \mathbb{E}[X|Y]\|^2] > 0$. Fig. 6.8 plots these estimated MSEs versus number of particles for each of the techniques presented here. Observe that the estimators appear to closely approximate the limit in a relatively small number of particles, and that the MSEs of the anticorrelated particle filter mean estimators appear to be upper bounded by the naive particle filter. Also note that relative MSE reductions do not appear to be as drastic as in the one dimensional case.

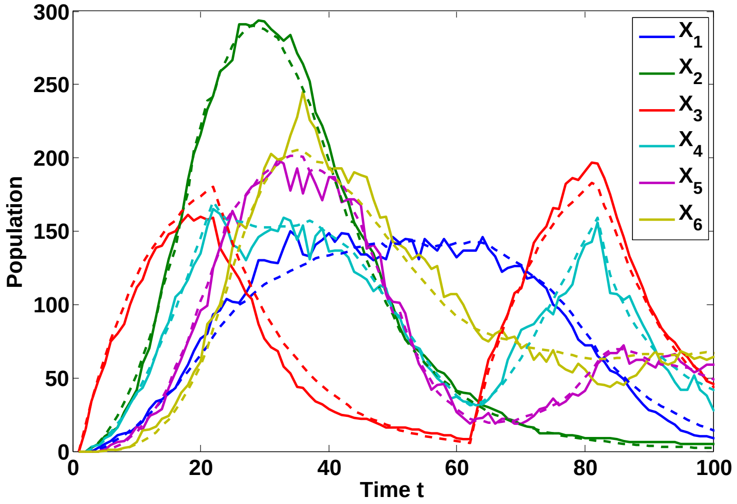

Figure 6.7: Illustrative sample path X_t, shown with mean path estimator (dashed lines) of a corresponding particle filter with 100 particles. Note that here the measurements have a resolution of only $\chi = 50$ and nodes 4 and 6 are not even observed directly.

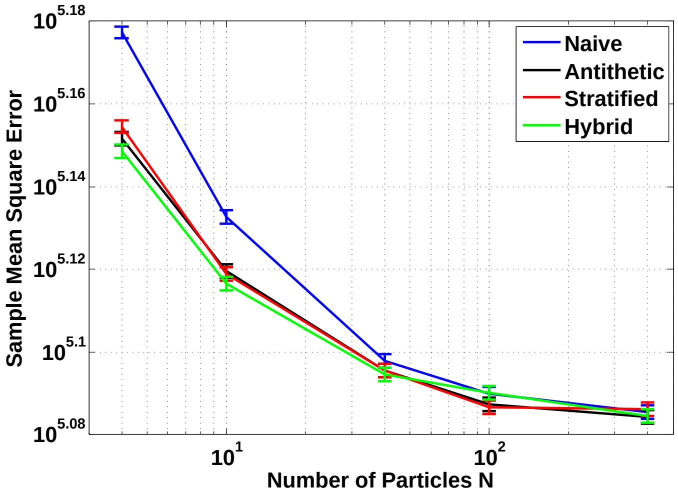

Figure 6.8: Plot of sample mean square error of the mean of the empirical particle distribution from an ensemble of random sample paths of the airport model. Error bars shown are standard error of the mean. Here, ensemble size is 16000. Note the apparent convergence to the Bayesian limit.

Chapter 7

Conclusions

In this dissertation, we presented a number of algorithms, approaches and ideas to reduce the cost of Monte Carlo simulation of a large collection of lattice Markov chains. While we considered distinct algorithms in both discrete- and continuous-time, the overall approach was to modify the stochastic simulation algorithms to relax the independence assumption of traditional Monte Carlo. By introducing strong, local (with respect to ensemble) negative correlation between sample trajectories, we were able to demonstrate significant reductions in estimator variance for both linear and nonlinear systems. Crucially, all algorithms we present are both agnostic to the particular parameters of the lattice Markov chains of interest and none of them alter the marginal distributions of sample trajectories from their iid counterparts. Thus the variance reduction our algorithms produce comes with no additional sources of bias or approximation.

In the discrete-time case, we introduced an algorithm that requires only slight modification of the uniform random inputs to the system in order to produce strongly negatively correlated trajectories, effectively extending the classical technique of antithetic variates to the pathwise stochastic simulation domain. Requiring no additional computational overhead and remaining fully embarrassingly parallelizable, our antithetic simulation algorithms produced multiple orders-of-magnitude variance reduction for Monte Carlo mean estimators for affine (gene expression) and nonlinear (coagulation and HIV infection) simulations. Additionally, we proved that these estimators are unbiased and consistent with respect to the discrete-time distribution of interest such as tau-leaping, for example. We also derived a recurrence relation that governs the time evolution of the negative covariance our algorithms introduce, and use this relation to prove a sufficient condition for variance reduction in the affine rates case. We discuss possible further refinements of this algorithm below.

In the continuous-time setting, we introduced a pair of algorithms, the endpoint method and the binomial midpoint method, to simulate exact, negatively correlated sample trajectories of the unit-rate Poisson process. For one of the algorithms, we derived closed form expressions for the evolution of the scaled MSE of mean estimators constructed using these anticorrelated processes in terms of a special function related to antithetic scalar Poisson variable generation. Further, we provided simple, asymptotic bounds for performance of this algorithm for both large and small parameter regimes. Using the random time-change representation of a lattice Markov chain, we showed how these anticorrelated Poisson processes could be used as the random inputs to simulate any lattice Markov chain. Again using both affine and nonlinear systems, we numerically demonstrated performance for a large parameter range, demonstrating order-of-magnitude MSE reduction. Further, while these algorithms introduce some computational overhead relative to the most efficient continuous-time simulation techniques, we provided error vs cost analysis that showed that the gains in MSE were significant enough to make them cost competitive when operating in Pareto regimes when compared with iid next reaction methods. Future work in this algorithm might include analysis to determine the relationship between stochastic system parameters like rate functions and the Pareto optimal parameters for the variance-reduction algorithm. Our experiments suggest that an adaptive anticorrelated algorithm may be attainable.

Finally, we presented several extensions and applications of these anticorrelated simulation algorithms. We extended antithetic tau-leaping to allow for any anticorrelated set of Poisson random variables to be used as inputs in a modular fashion. To illustrate this new paradigm, we presented a conditional tau-leaping algorithm that antithetically pre-sampled large sections of Poisson process trajectory in order to return variance-reduced, conditionally Poisson random variables to a tau-leaping algorithm. The resulting approach significantly improved the MSE performance of anticorrelated tau-leaping estimators in the regime where their performance was weakest. While this particular implementation of anticorrelated stochastic simulation introduced non-trivial computational overhead, future work could greatly improve the efficiency of this technique. For example, since the variance-reduction algorithm parameters are now uncoupled from the system dynamics, the parameters could be adaptively modified during simulation depending on the system rates. We also showed how these anticorrelated techniques might extend beyond a pure Monte Carlo simulation context into control and filtering of stochastic processes. We presented an algorithm to reduce the cost of estimating the expected cost-to-go in a version of the stochastic model predictive control problem. This approach could be particularly useful when online computational resources are constrained. Lastly, we presented a way to simulate the prediction step in a particle filtering algorithm using anticorrelated methods. In particular, we showed that the re-sampling methods typically used to implement the update step of the particle filter are particularly inhospitable to maintaining persistent negative correlation between trajectories over time. Consequently, we suggest that any future work in applying anticorrelated sampling to particle filters focus on pairing it with an alternative update step.

References

[1] D. F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. Journal of Chemical Physics, 127(21), 2007.

[2] D. F. Anderson. Incorporating postleap checks in tau-leaping. The Journal of Chemical Physics, 128(5):054103, 2008.

[3] D. F. Anderson. An efficient finite difference method for parameter sensitivities of continuous time Markov chains. SIAM J. Numer. Anal., 50(5):2237–2258, 2012.

[4] D. F. Anderson and D. H. Higham. Multilevel Monte Carlo for continuous time Markov chains, with applications in biochemical kinetics. Multiscale Model. Simul., 10(1):146–179, 2012.

[5] D. F. Anderson and T. G. Kurtz. Stochastic Analysis of Biochemical Systems. Springer, 2015.

[6] H. T. Banks, S. Hu, M. Joyner, A. Broido, B. Canter, K. Gayvert, and K. Link. A comparison of computational efficiencies of stochastic algorithms in terms of two infection models. Mathematical Biosciences and Engineering, 9(3):487–526, 2012.

[7] D. Bratsun, D. Volfson, L. S. Tsimring, and J. Hasty. Delay-induced stochastic oscillations in gene regulation. PNAS, 102(41):1459314598, 2005.

[8] C. Briat and M. Khammash. Computer control of gene expression: robust setpoint tracking of protein mean and variance using integral feedback. In Proceedings of the IEEE Conference on Decision and Control, December 2012.

[9] X. Cai. Exact stochastic simulation of coupled chemical reactions with delays. Journal of Chemical Physics, 126(12):124108, 2007.

[10] Y. Cao, D. T. Gillespie, and L. R. Petzold. The slow-scale stochastic simulation algorithm. J. Chem. Phys., 122(1):014116, 2005.

[11] Y. Cao, D. T. Gillespie, and L. R. Petzold. Efficient stepsize selection for the tau-leaping simulation method. Journal of Chemical Physics, 124:044109, 2006.

[12] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. IEEE Transactions on Signal Processing, 50:736–746, 2002.

[13] S. Engblom. Computing the moments of high dimensional solutions of the master equation. Applied Mathematics and Computation, 180(2):498–515, 2006.

[14] S. N. Ethier and T. G. Kurtz. Markov Processes: Characterization and Convergence. John Wiley and Sons, 1986.

[15] M. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. Journal of Physical Chemistry A, 105:1876, 2000.

[16] M. B. Giles. Multilevel monte carlo path simulation. Operations Research, 56:607–617, 2008.

[17] D. T. Gillespie. An exact method for numerically simulating the stochastic coalescence process in a cloud. Journal of the Atmospheric Sciences, 32:1977–1989, 1975.

[18] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J. Comput. Phys., 22:403–434, 1976.

[19] D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. Journal of Chemical Physics, 115(4):1716–1733, 2001.

[20] P. W. Glynn and D. L. Iglehart. Importance sampling for stochastic simulation. Management Science, 35:1367 – 1392, 1989.

[21] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. Proceedings of the IEEE. Institute of Electrical and Electronics Engineers, 140:107–113, 1993.

[22] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runberg, and C. Theodoropoulos. Equation-free, coarse-grained multiscale computation: enabling microscopic simulators to perform system-level tasks. Communications in Mathematical Sciences, 1(4):715–762, 2003.

[23] P. A. Maginnis. Variance reduction for Poisson and Markov jump processes. Master's thesis, University of Illinois at Urbana-Champaign, 2011.

[24] P. A. Maginnis, M. West, and G. E. Dullerud. Anticorrelated discrete-time stochastic simulation. In Proceedings of the IEEE Conference on Decision and Control, pages 618–623, December 2013.

[25] P. A. Maginnis, M. West, and G. E. Dullerud. Variance-reduced simulation of lattice discrete-time markov chains with applications in reaction networks. Journal of Computational Physics, 322:400–414, 2016.

[26] D. Q. Mayne. Model predictive control: recent developments and future promise. Automatica, 50:2967–2986, 2014.

[27] Harley H. McAdams and Adam Arkin. Stochastic mechanisms in gene expression. Proceedings of the National Academy of Sciences, 94(3):814–819, 1997.

[28] M. D. Michelotti, M. T. Heath, and M. West. Binning for efficient stochastic multiscale particle simulations. Multiscale Model. Simul., 11(4):1071–1096, 2013.

[29] M. Rathinam. Convergence of moments of tau leaping schemes for unbounded Markov processes on integer lattices. SIAM Journal on Numerical Analysis, 54(1):415–439, 2016.

[30] M. Rathinam, L. R. Petzold, Y. Cao, and D. T. Gillespie. Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method. Journal of Chemical Physics, 119:12784, 2003.

[31] M. Rathinam, L. R. Petzold, Y. Cao, and D. T. Gillespie. Consistency and stability of tau-leaping schemes for chemical reaction systems. Multiscale Model. Simul., 4(3):867–895, 2005.

[32] M. Rathinam, P. W. Sheppard, and M. Khammash. Efficient computation of parameter sensitivities of discrete stochastic chemical reaction networks. Journal of Chemical Physics, 132(3):034103, 2010.

[33] N. Riemer, M. West, R. A. Zaveri, and R. C. Easter. Simulating the evolution of soot mixing state with a particle-resolved aerosol model. Journal of Geophysical Research, 114:D09202, 2009.

[34] C. P. Robert and G. Casella. Monte Carlo statistical methods. Springer, 2nd edition, 2004.

[35] S. M. Ross. Stochastic Processes. Wiley, 2nd edition, 1996.

[36] J. H. Seinfeld and S. N. Pandis. Atmospheric Chemistry and Physics. John Wiley and Sons, Hoboken, NJ, 2006.

[37] G. M. Shroff and H. B. Keller. Stabilization of unstable procedures: the recursive projection method. SIAM Journal on Numerical Analysis, 30(4):1099–1120, 1993.

[38] M. Villen-Altamirano. Rare event simulation: The RESTART methods. In Proceedings of the International Conference on High Performance Computing and Simulation, pages 32–41, July 2012.

[39] W. Whitt. Bivariate distributions with given marginals. Annals of Statistics, 4(6):1280–1289, 1976.

[40] D. J. Wilkinson. Stochastic Modelling for Systems Biology. CRC Press, 2011.