

AUTOMATED STATE-DEPENDENT IMPORTANCE SAMPLING FOR MARKOV JUMP PROCESSES VIA SAMPLING FROM THE ZERO-VARIANCE DISTRIBUTION

ADAM W. GRACE * ** AND
DIRK P. KROESE,* *The University of Queensland*
WERNER SANDMANN,*** *University of Derby*

Abstract

Many complex systems can be modeled via Markov jump processes. Applications include chemical reactions, population dynamics, and telecommunication networks. Rare-event estimation for such models can be difficult and is often computationally expensive, because typically many (or very long) paths of the Markov jump process need to be simulated in order to observe the rare event. We present a state-dependent importance sampling approach to this problem that is adaptive and uses Markov chain Monte Carlo to sample from the zero-variance importance sampling distribution. The method is applicable to a wide range of Markov jump processes and achieves high accuracy, while requiring only a small sample to obtain the importance parameters. We demonstrate its efficiency through benchmark examples in queueing theory and stochastic chemical kinetics.

Keywords: Importance sampling; adaptive; automated; improved cross entropy; state dependent; zero-variance distribution; Markov jump process; continuous-time Markov chain; stochastic chemical kinetics; queueing system

2010 Mathematics Subject Classification: Primary 60J28
Secondary 62M05

1. Introduction

Many real-world random systems can be described mathematically via *Markov jump processes*. Typical examples can be found in stochastic chemical kinetics [10], [17], [20], [25], population dynamics [5], [19], and queueing networks [1], [13], [26].

Rare-event probability estimation by direct simulation of the model (crude Monte Carlo) is computationally difficult because typically a huge simulation effort is required to observe only a few rare events. A more sophisticated approach is to use *importance sampling*, where the system is simulated under a change of measure; see, e.g. [2], [11], and [14]. However, the efficiency of this technique depends crucially on (a) the appropriate choice of a parameterized family of importance sampling distributions, and (b) the judicious selection of an appropriate member within that family; that is, the correct choice of an importance sampling parameter. Adaptive techniques such as the *cross-entropy method* [22] are useful for estimating good importance sampling parameters, but in high-dimensional settings it may take a large simulation effort [4].

Received 29 October 2012; revision received 30 August 2013.

* Postal address: School of Mathematics and Physics, The University of Queensland, Brisbane 4072, QLD, Australia.

** Email address: adam.grace@uqconnect.edu.au

*** Postal address: School of Computing and Mathematics, University of Derby, Derby DE22 1GB, UK.

The matter is further complicated by the fact that good importance sampling distributions often need to be *state dependent*, where the importance sampling parameter depends on the state of the process.

In this paper we propose a new method for obtaining state-dependent importance sampling parameters that exploits the dynamic nature of Markov jump processes. We demonstrate how to obtain, for a broad class of Markov jump processes, a state-dependent change of measure that gives improved accuracy compared to standard importance sampling approaches and recent state-dependent methods [3], [7], [8], [9], [21]. The method requires a considerably smaller sample run than current techniques to obtain suitable importance sampling parameters, due to the fact that the pilot samples are generated directly from the zero-variance distribution, using the *hit-and-run* Markov chain Monte Carlo sampler [23], [24]. The effectiveness of the approach is demonstrated via a range of examples in stochastic chemical kinetics and queueing theory, showing a marked improvement over existing benchmark models.

The rest of the paper is organized as follows. In Section 2 we present the underlying framework for Markov jump processes used in this paper. In Section 3 we outline the background for rare-event estimation and explain current approaches to estimating rare-event probabilities. We then present various improvements involving sampling from the zero-variance distribution and state-dependence importance sampling. In Section 4 we describe how to sample from the zero-variance distribution using hit-and-run Markov chain Monte Carlo sampling. In Section 5 we demonstrate the applicability and improved accuracy of the new methods via comparisons with benchmark cases in stochastic chemical kinetics and queueing theory.

2. Markov jump processes

The class of *Markov jump processes* (henceforth abbreviated to MJPs) that we wish to consider has some particular structure. Firstly, the transition rates are assumed to be of the form

$$q(\mathbf{x}, \mathbf{x} + \mathbf{y}_i), \quad i = 1, \dots, p, \quad (1)$$

where $\mathbf{y}_1, \dots, \mathbf{y}_p$ are fixed vectors. Thus, from each state \mathbf{x} , there are (at most) p different *transition types* that can be made by the MJP. Secondly, the transition rates depend on parameter vector $\mathbf{u} = (u_1, \dots, u_p)$ (with $u_i > 0$, $i = 1, \dots, p$), in such a way that

$$q_i(\mathbf{x}; u_i) := q(\mathbf{x}, \mathbf{x} + \mathbf{y}_i; u_i) = u_i h_i(\mathbf{x}), \quad i = 1, \dots, p, \quad (2)$$

where $h_i(\mathbf{x})$, $i = 1, \dots, p$, are nonnegative functions of \mathbf{x} . Finally, we only study the MJP in some fixed interval $[0, T]$. In practice, many MJPs are of this form with small or moderate p .

Any *path* $\{\mathbf{X}(t), t \in [0, T]\}$ corresponds uniquely to a sequence $(t_1, r_1), \dots, (t_K, r_K)$, where t_k is the time at which the k th transition occurs and r_k is the k th transition type. The random variable K is the number of transitions before, and including, time T . We call $\mathbf{J} = (t_1, r_1, \dots, t_K, r_K)$ the *trajectory vector*, or simply the *trajectory*.

Since the sojourn times $\{s_k\}$, where $s_k = t_k - t_{k-1}$, are exponentially distributed, and the jump probabilities are proportional to the transition rates, for a fixed K , the probability density function (PDF) of a trajectory \mathbf{J} is given by

$$f(\mathbf{j}; \mathbf{u}) = \prod_{k=1}^K q_{r_k}(\mathbf{x}(t_{k-1}); u_{r_k}) \exp\left(-\left(\sum_{i=1}^p q_i(\mathbf{x}(t_{k-1}); u_i)\right)(t_k - t_{k-1})\right). \quad (3)$$

Simulation of the sojourn times and the transition types is straightforward via the inverse-transform method [14].

3. Rare-event probability estimation for MJPs

We wish to accurately estimate small probabilities of the form

$$\ell = \mathbb{P}(\mathbf{J} \in \mathcal{J}),$$

where \mathcal{J} is the set of all possible trajectories under which the rare event occurs. The crude Monte Carlo (CMC) estimator for ℓ is

$$\hat{\ell}_{\text{CMC}} = \frac{1}{N} \sum_{n=1}^N \mathbf{1}_{\{\mathbf{J}^{(n)} \in \mathcal{J}\}},$$

where $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} f(\mathbf{j}; \mathbf{u})$. A measure for the accuracy of any unbiased estimator $\hat{\ell}$ of ℓ is the *relative error* of the estimator, defined as $\text{RE} = \sqrt{\text{var}[\hat{\ell}]/\ell}$.

3.1. Importance sampling

A well-known technique to estimate rare-event probabilities more accurately than CMC is importance sampling [2], [11], [14]. The idea is to simulate the trajectories $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)}$ from a different PDF g rather than f in (3), and to estimate ℓ via the *importance sampling estimator*

$$\hat{\ell} = \frac{1}{N} \sum_{n=1}^N \mathbf{1}_{\{\mathbf{J}^{(n)} \in \mathcal{J}\}} W(\mathbf{J}^{(n)}), \tag{4}$$

where $\{\mathbf{J}^{(n)}\} \stackrel{\text{i.i.d.}}{\sim} g$ and

$$W(\mathbf{J}^{(n)}) = \frac{f(\mathbf{J}^{(n)}; \mathbf{u})}{g(\mathbf{J}^{(n)})}$$

is the *likelihood ratio*.

Ideally, we would like an importance sampling PDF g that results in the importance sampling estimator having the smallest possible variance. The best possible PDF is the *zero-variance* importance sampling PDF g^* defined as the conditional PDF of $\mathbf{J} \sim f(\cdot; \mathbf{u})$ given $\mathbf{J} \in \mathcal{J}$, that is,

$$g^*(\mathbf{j}) = \frac{f(\mathbf{j}; \mathbf{u}) \mathbf{1}_{\{\mathbf{j} \in \mathcal{J}\}}}{\ell}. \tag{5}$$

An obvious difficulty is that g^* cannot be evaluated, since ℓ is typically not known. A useful strategy is to restrict g to the same parametric family as f and then choose from this family a member that is close, in some sense, to g^* . Thus, we consider $g(\mathbf{j})$ of the form $f(\mathbf{j}; \mathbf{v})$, where $\mathbf{v} = (v_1, \dots, v_p)$ is the changed parameter (instead of $\mathbf{u} = (u_1, \dots, u_p)$), and aim to find a suitable \mathbf{v} under which to carry out the importance sampling. A possible method to use is the *cross-entropy* (CE) method [22], which selects a \mathbf{v}^* that minimizes the Kullback-Leibler (or cross-entropy) distance between g^* and $f(\cdot; \mathbf{v})$. The standard CE method uses a multistage approach, where the final importance sampling parameter $\hat{\mathbf{v}}^*$ (an estimate of \mathbf{v}^*) is adaptively obtained through a sequence of parameters $\mathbf{v}^{[1]} = \mathbf{u}, \mathbf{v}^{[2]}, \dots, \hat{\mathbf{v}}^*$ corresponding to a sequence of increasingly smaller sets $\mathcal{J}^{[1]} \supset \mathcal{J}^{[2]} \supset \dots \supset \mathcal{J}$. The optimal CE parameter \mathbf{v}^* is the solution to the cross-entropy optimization program [22, Section 3.3]

$$\max_{\mathbf{v}} \mathbb{E}_{\mathbf{u}}[W(\mathbf{J}; \mathbf{u}, \mathbf{v}) \mathbf{1}_{\{\mathbf{J} \in \mathcal{J}\}} \log f(\mathbf{J}; \mathbf{v})], \tag{6}$$

where f is given in (3) and $W(\mathbf{J}; \mathbf{u}, \mathbf{v}) = f(\mathbf{J}; \mathbf{u})/f(\mathbf{J}; \mathbf{v})$. Moreover, \mathbf{v}^* can be estimated from simulation by solving the stochastic counterpart of (6), replacing the expectation with a sample average. For the type of MJPs considered in this paper, this leads to the following *updating formula* for the importance sampling parameters at the b th stage of the CE procedure:

$$v_i^{[b]} = \frac{\sum_{n=1}^N W(\mathbf{J}^{(n)}; \mathbf{u}, \mathbf{v}^{[b-1]}) m_i^{(n)} \mathbf{1}_{\{\mathbf{J}^{(n)} \in \mathcal{J}^{[b]}\}}}{\sum_{n=1}^N W(\mathbf{J}^{(n)}; \mathbf{u}, \mathbf{v}^{[b-1]}) \sum_{k=1}^{K^{(n)}} h_i(\mathbf{x}^{(n)}(t_{k-1}^{(n)}))(t_k^{(n)} - t_{k-1}^{(n)}) \mathbf{1}_{\{\mathbf{J}^{(n)} \in \mathcal{J}^{[b]}\}}}. \tag{7}$$

Here $v_i^{[b]}$ is the i th component of $\mathbf{v}^{[b]}$, with $b = 1, 2, 3, \dots$, $m_i^{(n)}$ is the number of type- i transitions in $\mathbf{J}^{(n)}$, and $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} f(\mathbf{j}; \mathbf{v}^{[b-1]})$ are the trajectories generated at stage $b - 1$. At the b th stage of the CE procedure, $\mathcal{J}^{[b]}$ is defined as the set containing all possible trajectories (not only those that are generated) that are ‘at least as rare’ (according to some problem-dependent criterion) as the $N\rho$ th rarest from the sample $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} f(\mathbf{j}; \mathbf{v}^{[b-1]})$. The parameter $\rho \in (0, 1)$ is fixed in advance; we typically choose $\rho = 0.1$. We can simply define $\mathcal{J}^{[1]}$ as the set of all possible trajectories \mathbf{J} .

3.2. State-dependent importance sampling

The family of importance sampling PDFs obtained by changing the parameters $\{u_i\}$ to $\{v_i\}$ in (2) may not be close enough to the ideal importance sampling PDF g^* to yield accurate importance sampling estimates. It is natural to seek to enlarge the class of importance sampling distributions to include *state-dependent* changes of measure, in which the parameters $\{v_i\}$ depend on the state of the process. We propose to replace the transition rate of type- i transitions, $q_i(\mathbf{x}; u_i)$ in (2), with the transition rate $q_i(\mathbf{x}; v_i(\mathbf{x})) = h_i(\mathbf{x})v_i(\mathbf{x})$, where

$$v_i(\mathbf{x}) = \sum_{j=1}^{\beta_i} v_{ij} \mathbf{1}_{\{\mathbf{x} \in B_{ij}\}}, \quad i = 1, \dots, p, \tag{8}$$

where the bins B_{ij} , $j = 1, \dots, \beta_i$, $i = 1, \dots, p$, form p separate *partitions* of the state space E of the MJP, that is, for each i , the sets $B_{i1}, \dots, B_{i\beta_i}$ are disjoint and their union is E . Thus, if $\mathbf{x} \in B_{ij}$, the transition rate from \mathbf{x} to $\mathbf{x} + \mathbf{y}$ (see (1)) is $v_{ij}h_i(\mathbf{x})$. This family includes the original PDF—set $\beta_i = 1$, and take $B_{i1} = E$ and $v_{i1} = u_i$ for $i = 1, \dots, p$.

Because the corresponding family of importance sampling distributions includes the state-independent importance sampling discussed in the previous section, importance sampling with the extended family can offer higher accuracy in estimation, provided the parameters $\{v_{ij}\}$ are chosen appropriately. Again, this can be achieved adaptively via the CE method. Similarly to (7) and (12), the parameters in the multilevel CE procedure are updated as

$$v_{ij}^{[b]} = \frac{\sum_{n=1}^N \mathbf{1}_{\{\mathbf{J}^{(n)} \in \mathcal{J}^{[b]}\}} W(\mathbf{J}^{(n)}; \mathbf{u}, \mathbf{v}^{[b-1]}) \sum_{k=1}^{K^{(n)}} (\mathbf{1}_{\{r_k=i\}} \mathbf{1}_{\{\mathbf{x}(t_{k-1}) \in B_{ij}\}})}{\sum_{n=1}^N \mathbf{1}_{\{\mathbf{J}^{(n)} \in \mathcal{J}^{[b]}\}} W(\mathbf{J}^{(n)}; \mathbf{u}, \mathbf{v}^{[b-1]}) \sum_{k=1}^{K^{(n)}} (h_i(\mathbf{x}(t_{k-1})))(t_k - t_{k-1}) \mathbf{1}_{\{\mathbf{x}(t_{k-1}) \in B_{ij}\}}}, \tag{9}$$

where $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} f(\mathbf{j}; \mathbf{v}^{[b-1]})$. Note that we have suppressed the superscript ‘ (n) ’ on r_k, \mathbf{x}, t_k , and t_{k-1} for ease of notation. Also, the set of importance sampling parameters $\{v_{ij}^{[b]}\}$ is summarized as $\mathbf{v}^{[b]}$.

How to determine the partition $\{B_{ij}\}$ for each i is problem dependent. A natural approach is to partition the state space into regions within which the transition rates are roughly constant. Of course, there is a tradeoff between a higher number of smaller regions with near-constant

transition rate (thus leading to a more accurate change of measure) and fewer regions that possibly differ significantly but offer much less computational burden. A convenient way to partition the state space is to assign, for each type of transition i , a numerical value to the states, say $\phi_i(\mathbf{x}) \in [0, 1]$, and to define

$$B_{ij} = \{\mathbf{x} : \phi_i(\mathbf{x}) \in (c_i^{(j-1)}, c_i^{(j)}]\}, \quad j = 1, \dots, \beta_i, i = 1, \dots, p,$$

where, for each i , the values $0 = c_i^{(0)} < c_i^{(1)} < \dots < c_i^{(\beta_i)} = 1$ partition the unit interval $(0, 1]$. Such an approach was taken in [21], where the function ϕ_i is defined as the probability that the MJP leaves state \mathbf{x} via the i th type of transition *under the original measure*, that is,

$$\phi_i(\mathbf{x}) = q_i(\mathbf{x}; u_i) / \sum_{k=1}^p q_k(\mathbf{x}; u_k), \quad i = 1, \dots, p. \quad (10)$$

We use such a partition for all the chemistry-related examples in this paper. In general, the choice of the partition function is problem dependent and might require some experimentation to identify the regions $\{B_{ij}\}$ of interest. In, e.g. the queueing example, Example 4 below, we use instead $\phi_i(\mathbf{x}) = \min\{x_j/C_i, 1\}$ for some constant C_i and some index $j \in \{1, \dots, d\}$.

We next describe a simple automatic procedure for choosing the $\{c_i^{(j)}\}$ at the b th stage of the CE method, based on an independent and identically distributed (i.i.d.) sample of trajectories $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)}$ from $f(\mathbf{j}; \mathbf{v}^{[b-1]})$. Let $\mathbf{x}_k^{(n)}$, $k = 1, \dots, K^{(n)}$, $n = 1, \dots, N$, be the subsequent states visited by the trajectories/paths, and denote by ϕ_i^{\min} the smallest of the $\phi_i(\mathbf{x}_k^{(n)})$ for which the rare event at the b th stage is reached. Thus, we take into account only those $\mathbf{x}_k^{(n)}$ for which $\mathbf{J}^{(n)} \in \mathcal{J}^{[b]}$. Similarly, for such $\mathbf{x}_k^{(n)}$, let ϕ_i^{\max} be the maximum of the ϕ_i values. For a given β_i (say $\beta_i = 10$), choose

$$c_i^{(j)} = \phi_i^{\min} + \frac{\phi_i^{\max} - \phi_i^{\min}}{\beta_i} j, \quad j = 1, \dots, \beta_i - 1. \quad (11)$$

Recall that $c_i^{(0)} = 0$ and $c_i^{(\beta_i)} = 1$ by definition.

Although (11) provides a convenient initial construction of the partitions, the $\{c_i^{(j)}\}$ (or, equivalently, the $\{B_{ij}\}$) should be chosen such that the CE optimal parameters v_{ij}^* can be accurately estimated via (9). This requires that enough transitions of type i occur when the MJP is in each set $\{B_{ij}, j = 1, \dots, \beta_i\}$. In particular, we require that there are at least ω type- i transitions occurring in each set, where ω is specified in advance; in this paper we take $\omega = 20$ for all examples.

For partitions of the form (10), this requirement can be met (and automated) by *merging* adjacent intervals $(c_i^{(j)}, c_i^{(j+1)}]$ and $(c_i^{(j+1)}, c_i^{(j+2)}]$. The merging can be implemented in various ways.

As a numerical example, let $\omega = 5$ and suppose (for a certain i) that the interval $(0, 1]$ is initially partitioned as $(0, 0.3]$, $(0.3, 0.4]$, $(0.4, 0.5]$, $(0.5, 0.6]$, $(0.6, 0.7]$, $(0.7, 1]$. Suppose that 20 type- i transitions have taken place that have reached the (intermediate) rare event and that the counts of the numbers of times that $\phi_i(\mathbf{x})$ falls in one of the above six sets are 2, 3, 4, 2, 8, 1. Merging intervals from left to right until the combined count is more than ω , the modified partition is $(0, 0.4]$, $(0.4, 0.6]$, $(0.6, 1]$, with counts 5, 6, 9.

Algorithms 1 and 2 below explain how the corresponding state-dependent cross-entropy method is implemented, using the method described above and as in [21]. Below, N_v is the size of the sample used to calculate the importance parameter vector \mathbf{v} . Typically, N_v is smaller than N which is used in the final simulation (4) to estimate ℓ .

Algorithm 1. (*Adaptive state-dependent cross-entropy method.*)

1. Choose parameters ω , N , N_v , ρ , β_1, \dots, β_p , and functions ϕ_i , $i = 1, \dots, p$. Set $b = 1$, and $B_{i1} = E$ and $v_{i1} = u_i$ for all $i = 1, \dots, p$.
2. Generate via Algorithm 2 below an i.i.d. sample $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N_v)}$ from $f(\mathbf{j}; \mathbf{v}^{[b]})$.
3. Let $b = b + 1$, and identify the *elite sample* as the $N_v \rho$ rarest of the $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N_v)}$ (according to some problem-dependent criterion). If more than $N_v \rho$ of the generated samples fall in \mathcal{J} , set $\mathcal{J}^{[b]} = \mathcal{J}$; otherwise, define $\mathcal{J}^{[b]}$ as the set containing all possible trajectories that are ‘at least as rare’ as the worst of the elite sample.
4. Using the elite sample, determine the $c_i^{(j)}$ according to (11) and further modify these to obtain at least ω counts in each set B_{ij} (this may change the β_i).
5. Use the *same* elite sample to find the importance sampling parameters $v_{ij}^{[b]}$ via (9).
6. If $\mathcal{J} \neq \mathcal{J}^{[b]}$, repeat steps 2 to 5; otherwise, proceed to step 7.
7. Calculate $\hat{\ell}$ via (4) using $\mathbf{J}^{[1]}, \dots, \mathbf{J}^{[N]}$ i.i.d. $\sim f(\mathbf{j}; \hat{\mathbf{v}}^*)$, where $\hat{\mathbf{v}}^* = \mathbf{v}^{[b]}$.

In general, the trajectory $\mathbf{J} = (t_1, r_1, \dots, t_K, r_K)$ should be generated long enough to produce the entire path $\{\mathbf{X}(t), t \in [0, T]\}$. However, often the rare event is of the form $(t_1, r_1, \dots, t_M, r_M) \in \mathcal{J}$ for some stopping time $M < K$ (with respect to the sequence of $(t_1, r_1), (t_2, r_2), \dots$). In that case the generation of \mathbf{J} can be stopped immediately after the rare event occurs, say at some random stopping time $\tau < T$. All numerical examples in this paper are of this form and the following generation algorithm is formulated accordingly.

Algorithm 2. (*Trajectory/path generation with importance sampling.*)

1. Identify $\mathcal{J}^{[b]}$ and the importance sampling parameters $v_{ij}^{[b]}$. Let N_v be the sample size, and let T be the maximum time length. Set $n = 1$.
2. Set $t_0 = 0$, $W^{(n)} = 1$, and $k = 1$; initialize $\mathbf{X}(0) = \mathbf{x}(0)$.
3. Generate the transition type r_k and sojourn time s_k using $q_i(\mathbf{x}; v_i(\mathbf{x})) = h_i(\mathbf{x})v_i(\mathbf{x})$ instead of $q_i(\mathbf{x}; u_i)$, where $v_i(\mathbf{x})$ is given in (8). Set $k = k + 1$.
4. If $t_{k-1} + s_k > T$, stop the trajectory generation and continue to step 5. Otherwise, set $\mathbf{X}(t_k) = \mathbf{X}(t_{k-1}) + \mathbf{y}_{r_k}$, $t_k = t_{k-1} + s_k$, and update the likelihood ratio of the n th trajectory using

$$W^{(n)} = W^{(n)} \frac{u_{r_k} \exp\left(-\left(\sum_{i=1}^p u_i h_i(\mathbf{x}(t_{k-1}))\right)s_k\right)}{v_{r_k}(\mathbf{x}(t_{k-1})) \exp\left(-\left(\sum_{i=1}^p v_i(\mathbf{x}(t_{k-1}))h_i(\mathbf{x}(t_{k-1}))\right)s_k\right)}.$$

If the current $\mathbf{J} \in \mathcal{J}$, stop the trajectory generation and continue to step 5. Otherwise, repeat from step 3.

5. If $n = N_v$, stop and output $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N_v)}$ and the corresponding likelihood ratios $W^{(1)}, \dots, W^{(N_v)}$; otherwise, set $n = n + 1$ and repeat from step 2.

3.3. Improved CE

The standard multilevel CE approach requires a large enough sample in each level to estimate the final importance sampling parameter vector \mathbf{v}^* accurately enough. Difficulties may arise when the number of parameters to be estimated is large. An alternative approach, discussed in [4] is to estimate \mathbf{v}^* via a *single* step, by (approximately) generating directly from the zero-variance PDF g^* using, for example, Markov chain Monte Carlo methods. Specifically, the estimate $\hat{\mathbf{v}}^*$ of the optimal importance sampling parameter \mathbf{v}^* is in this case the solution to the optimization problem

$$\max_{\mathbf{v}} \sum_{n=1}^N \log f(\mathbf{J}^{(n)}; \mathbf{v}),$$

where $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} g^*$. Note that this solution is simply the maximum likelihood estimate of \mathbf{v} based on the data from g^* . Here every simulated trajectory $\mathbf{J}^{(n)}$ is in \mathcal{J} and, therefore, the optimal importance parameter for the MJP is estimated in a single step with a simpler version of (7):

$$\hat{v}_i^* = \frac{\sum_{n=1}^N m_i^{(n)}}{\sum_{n=1}^N \sum_{k=1}^{K^{(n)}} h_i(\mathbf{x}^{(n)}(t_{k-1}^{(n)}))(t_k^{(n)} - t_{k-1}^{(n)})} \tag{12}$$

for a sample of N trajectories $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} g^*$. This parameter updating formula is more accurate than (7). Therefore, only a very small sample from g^* is required. In Section 4 we discuss how the *hit-and-run* sampler can be used efficiently to sample (approximately) from g^* .

We extend the idea of state-dependent importance sampling outlined in Section 3.2 by capitalizing on the improved CE formula (12). As a direct result of improved statistical inference due to sampling from g^* , the number of bins β described in Section 3.2 can be increased. An increase in the number of bins, all with a reduced size, results in a more ‘fine-tuned’ change of measure that offers more accuracy in estimation. The importance sampling parameters are calculated, similarly to (9), via

$$\hat{v}_{ij}^* = \frac{\sum_{n=1}^N \sum_{k=1}^{K^{(n)}} (\mathbf{1}_{\{r_k=i\}} \mathbf{1}_{\{\mathbf{x}(t_{k-1}) \in B_{ij}\}})}{\sum_{n=1}^N \sum_{k=1}^{K^{(n)}} (h_i(\mathbf{x}(t_{k-1}))(t_k - t_{k-1}) \mathbf{1}_{\{\mathbf{x}(t_{k-1}) \in B_{ij}\}})} \tag{13}$$

where $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N)} \stackrel{\text{i.i.d.}}{\sim} g^*$. Note that here there are no likelihood ratio terms and the parameters are updated in a single step.

4. Sampling from the zero-variance distribution

In this section we explain how Markov chain Monte Carlo, in particular hit-and-run sampling, can be used to sample trajectories that are approximately from the zero-variance distribution. The hit-and-run sampler has been shown to be very useful in rare-event simulation (and estimation) in [12]. We present a new and powerful adaptation of hit-and-run sampling. Specifically, we apply it to static uniform vectors representing transformed dynamic MJPs.

Via the inverse transform method a vector $\boldsymbol{\xi} = (\xi_1, \dots, \xi_V)$ of i.i.d. $\mathbf{U}(0, 1)$ random variables can be transformed into a sequence of exponential and categorical random variables to simulate random trajectories as in (3). Here V is a fixed integer large enough to account for the times and types of any trajectory. Any given trajectory \mathbf{J} can therefore be represented as $\mathbf{J} = \boldsymbol{\psi}(\boldsymbol{\xi})$ for some function $\boldsymbol{\psi}$.

Our aim is to generate a Markov chain of trajectories $\xi^{(0)}, \dots, \xi^{(N_v)}$ that, when transformed, approximately give a random sample $\mathbf{J}^{(0)}, \dots, \mathbf{J}^{(N_v)}$ from g^* given in (5). In Section 4.2 we explain how to obtain the initial trajectory $\mathbf{J}^{(0)}$.

4.1. Zero-variance sampling via hit-and-run sampler

In this section we explain how the hit-and-run sampler [23], [24] is used to construct a Markov process that converges in distribution to the uniform distribution on the set

$$\{\xi \in [0, 1]^V : \psi(\xi) = \mathbf{j} \in \mathcal{J}\}. \tag{14}$$

The hit-and-run method was chosen, instead of, for example, the Metropolis–Hastings or Gibbs samplers, because of its fast mixing properties [16] and ease of implementation. In this paper we use a modification of the hit-and-run sampler in [14, Example 6.5, p. 242]. The idea is that sampling uniformly on a region R in the hypercube is equivalent to first sampling from an n -dimensional standard normal distribution in a region $\Phi(R) \subset \mathbb{R}^n$, where Φ is the n -dimensional normal cumulative distribution function (CDF), and then applying the inverse map Φ^{-1} . The sampling in \mathbb{R}^n can be done via ordinary hit-and-run. We found that this approach leads to much faster convergence to the target distribution than direct hit-and-run sampling on the hypercube. The intuitive reason is that the ordinary hit-and-run sampler will spend a lot of time in the corners of the hypercube, whereas the transformed sampler will search the hypercube more evenly. Algorithm 3 outlines how, starting from an initial random vector $\xi^{(0)} \in [0, 1]^V$ such that $\psi(\xi^{(0)}) = \mathbf{J}^{(0)} \in \mathcal{J}$, we can generate a Markov chain whose limiting distribution is the uniform distribution on (14).

Algorithm 3. (*Hit-and-run sampler.*)

1. Obtain a $\xi^{(0)}$ such that $\psi(\xi^{(0)}) = \mathbf{J}^{(0)} \in \mathcal{J}$, and set $n = 0$.
2. Generate a random direction column vector \mathbf{d} uniformly on the surface of the n -dimensional hypersphere. This can be done via $\mathbf{d} = \mathbf{d}_1 / \sqrt{\mathbf{d}_1^\top \mathbf{d}_1}$, with $\mathbf{d}_1 \sim \mathbf{N}(\mathbf{0}, I_V)$.
3. Generate a point $\eta^{(n+1)} = \Phi(\Phi^{-1}(\xi^{(n)}) + \lambda \mathbf{d})$ on the line segment \mathcal{L} , via

$$\lambda = -\mathbf{d}^\top \Phi^{-1}(\xi^{(n)}) + Z,$$

where $Z \sim \mathbf{N}(0, 1)$, and Φ and Φ^{-1} are the n -dimensional normal CDF and n -dimensional inverse normal CDF, respectively.

4. If $\psi(\eta^{(n+1)}) = \mathbf{J}^{(n+1)} \in \mathcal{J}$, set $\xi^{(n+1)} = \eta^{(n+1)}$. Otherwise, repeat step 3.
5. Let $n = n + 1$, and repeat from step 2.

Since the vectors $\xi^{(0)}, \xi^{(1)}, \xi^{(2)}, \dots$ in the Markov chain are dependent, it is necessary to apply *thinning* in order to obtain an approximately i.i.d. sample (of size N_v) from the zero-variance distribution. In particular, we can keep the k th, $2k$ th, $3k$ th, etc. generated vectors and discard the rest. There is obviously a tradeoff between computational time to generate a sample and how closely the sample resembles a true random sample that is i.i.d. from g^* . For the types of MJPs investigated in this paper, we have selected k to be rather high, because in general Markov samplers in high-dimensional spaces tend to mix poorly. Therefore, a total of kN_v vectors corresponding to rare events should be generated to obtain a sample of N_v i.i.d. vectors from g^* . After transforming, the resulting $\lfloor kN_v/k \rfloor$ vectors represent a random sample

from the zero-variance distribution: $\mathbf{J}^{(k)}, \mathbf{J}^{(2k)}, \mathbf{J}^{(3k)}, \dots \stackrel{\text{i.i.d.}}{\sim} g^*$. Note that $\mathbf{J}^{(0)}$ should be discarded as it was not generated from g^* .

4.2. Markov sampling to obtain the initial rare-event trajectory

Hit-and-run sampling is used to obtain the initial rare-event trajectory $\mathbf{J}^{(0)} = \psi(\boldsymbol{\xi}^{(0)})$ which is then used to generate the Markov chain described in Algorithm 3. Beginning with a random uniform vector $\boldsymbol{\xi} \in \mathbf{U}(0, 1)^V$, hit-and-run Markov sampling is applied such that every subsequent vector in the chain is rejected unless it is ‘as rare’ or ‘more rare’ than the current step. Algorithm 4 below describes this process.

Algorithm 4. (*Generate rare-event trajectory.*)

1. Generate $\boldsymbol{\xi}^{(0)} \sim \mathbf{U}(0, 1)^V$, and set $n = 0$.
2. Same as step 2 in Algorithm 3.
3. Same as step 3 in Algorithm 3.
4. Set

$$\boldsymbol{\xi}_{n+1} = \begin{cases} \boldsymbol{\eta}^{(n+1)} & \text{if } \psi(\boldsymbol{\eta}^{(n+1)}) = \mathbf{J}^{(n+1)} \text{ is ‘at least as rare’ as } \psi(\boldsymbol{\xi}^{(n)}) = \mathbf{J}^{(n)}, \\ \boldsymbol{\xi}^{(n)} & \text{otherwise.} \end{cases}$$

5. Let $n = n + 1$, and repeat steps 2 to 4 until $\psi(\boldsymbol{\xi}^{(n+1)}) = \mathbf{J}^{(n+1)} \in \mathcal{J}$.

Remark 1. Steps 3 and 4 in Algorithm 3 involve standard acceptance–rejection sampling on a line. We denote the average ‘acceptance rate’ by κ , that is, the expected number of samples required to generate a new point in the hit-and-run algorithm. Therefore, the total number of trajectories required to be simulated in order to obtain N_v i.i.d. trajectories from g^* is approximately $k\kappa N_v$. For all the problems in this paper, we have used $\omega = 20$, $k = 100$, and $V = 1000$. All of these parameters could have been lower without significantly compromising accuracy.

We have found that the computational effort required to (i) generate $\mathbf{J}^{(0)}$, (ii) merge the B_{ij} , (iii) calculate the importance parameters (via CE formulae), and (iv) select the appropriate importance parameter at each step in the trajectory, is mostly negligible compared to the subsequent effort required to generate $\mathbf{J}^{(1)}, \dots, \mathbf{J}^{(N_v)}$. Therefore, it is reasonable to compare the effort required for rare-event estimation in this paper by simply comparing $N + N_v$ (number of iterations necessary to obtain \mathbf{v}^*), for the iterative importance sampling approaches, to $N + k\kappa N_v$ for the zero-variance approach. Note that often the final simulation is more computationally intensive than the techniques of obtaining the importance parameters. Thus, this investigation focuses primarily on the accuracy of rare-event estimation techniques.

Empirical results have shown that our proposed approach of combining Algorithms 3 and 4 is a practical method for generating rare events from the types of MJPs investigated in this paper. The computational effort required by our method is discussed and compared with the outlined importance sampling approaches where relevant in this paper.

5. Numerical results

In this section we demonstrate, via a range of examples, that the proposed method provides an effective approach to rare-event estimation for MJPs. The improvement outlined in Section 3.3 is completely adaptive and automatic. In particular, no prior knowledge of any optimal

importance distribution is required. The advantages of the approach are greater accuracy in estimating rare-event probabilities, general applicability, and only a small size is required to estimate the importance sampling parameters via standard maximum likelihood.

We also illustrate that Markov chain Monte Carlo can be effective, even for multidimensional MJPs, in sampling from the zero-variance distribution. In particular, the inverse transform method combined with hit-and-run sampling is a practical way to sample vectors that are from, or extremely close to, g^* . This has another advantage in that even specific structures in trajectories from g^* are preserved when using this method, as will be illustrated in Example 5.

5.1. Stochastic chemical kinetics

In this section we consider rare-event estimation for MJPs that represent the numbers of species in a chemical reaction.

Example 1. (*Chemical reaction with two species.*) In [21] a chemical reaction with two species of molecules is considered which can be modeled via an MJP $\{(X_1(t), X_2(t)), t \geq 0\}$ on the positive integers, with reaction rates

$$\begin{aligned} q((x_1, x_2), (x_1 - 1, x_2 + 1)) &= u_1 x_1, \\ q((x_1, x_2), (x_1 + 1, x_2 - 1)) &= u_2 x_2. \end{aligned}$$

In Table 1 we present the results of three importance sampling procedures for the estimation of the probability $\mathbb{P}(\max_{0 \leq t \leq 10} X_2(t) \geq 30)$. The first method (standard CE) uses the standard state-independent CE method, where the change of measure involves the change of parameters from (u_1, u_2) to (v_1, v_2) . The second method (state-dependent CE) implements the state-dependent multilevel Algorithm 1. In the third method (zero-variance (ZV) state-dependent CE), the single-level, state-dependent CE procedure is used, where the importance sampling parameters are found via (13) by sampling from the zero-variance distribution. Recall that N_v is the sample size required to estimate v and β is the number of bins for each transition (reaction) type.

As can be seen in Table 1 the relative error decreases when β is increased, even when N_v , the sample size used to obtain the importance sampling parameters, is decreased. Two parameter updates were required for the first two methods but only a single update was necessary for the ZV state-dependent CE method. The acceptance rate value was $\kappa \approx 1.2$. These estimates are consistent with the true value $\ell = 1.191 \times 10^{-5}$ in [21].

The numerical results below show that the proposed method(s) can deal effectively with much more elaborate systems.

Example 2. (*Protein cycle process.*) As a benchmark model for stochastic chemical kinetics, we consider a pheromone-induced G-protein cycle in *Saccharomyces cerevisiae* (see [6] and [21]), which can be modeled via a seven-dimensional MJP with system state at time t

TABLE 1: Numerical comparisons for $\mathbb{P}(\max_{0 \leq t \leq 10} X_2(t) \geq 30)$ starting from $X(0) = (100, 0)$.

Method	Relative error	$\hat{\ell}$	N	N_v	β
Standard CE	0.093	1.42×10^{-5}	10^6	10^5	—
State-dependent CE	0.0016	1.192×10^{-5}	10^6	10^5	10
ZV state-dependent CE	0.00079	1.191×10^{-5}	10^6	10^3	50

TABLE 2: Estimation of $\mathbb{P}(\max_{0 \leq t \leq 5} X_6(t) \geq 40)$ starting from $X(0) = (50, 2, 0, 50, 0, 0, 0)$.

Method	Relative error	$\hat{\ell}$	N	N_v	β
Standard CE	0.14	2.48×10^{-11}	10^6	10^5	—
State-dependent CE	0.031	2.35×10^{-11}	10^6	10^5	10
ZV state-dependent CE	0.017	2.35×10^{-11}	10^6	10^3	50

represented by the vector

$$X(t) = (X_1(t), X_2(t), X_3(t), X_4(t), X_5(t), X_6(t), X_7(t))^\top$$

and with transition rates

$$\begin{aligned} q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1 + 1, x_2, x_3, x_4, x_5, x_6, x_7)) &= u_1, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1 - 1, x_2, x_3, x_4, x_5, x_6, x_7)) &= u_2 x_1, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1 - 1, x_2, x_3 + 1, x_4, x_5, x_6, x_7)) &= u_3 x_1 x_2, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1 + 1, x_2, x_3 - 1, x_4, x_5, x_6, x_7)) &= u_4 x_3, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1, x_2, x_3 - 1, x_4 - 1, x_5 + 1, x_6 + 1, x_7)) &= u_5 x_3 x_4, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1, x_2, x_3, x_4, x_5 - 1, x_6, x_7 + 1)) &= u_6 x_5, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1, x_2, x_3, x_4 + 1, x_5, x_6 - 1, x_7 - 1)) &= u_7 x_6 x_7, \\ q((x_1, x_2, x_3, x_4, x_5, x_6, x_7), (x_1, x_2, x_3 + 1, x_4, x_5, x_6, x_7)) &= u_8. \end{aligned}$$

In this example there are $p = 8$ transition types corresponding to eight different types of chemical reactions, where

$$\mathbf{u} = (0.0038, 0.0004, 0.042, 0.010, 0.011, 0.100, 1050, 3.21).$$

Suppose that the rare event of interest here is $\{\max_{0 \leq t \leq 5} X_6(t) \geq 40\}$, starting from state $X(0) = (50, 2, 0, 50, 0, 0, 0)$. We implemented the standard CE, the state-dependent multilevel CE, and the state-dependent improved CE method to calculate \mathbf{v} , and, for each method, used importance sampling with the same sample size $N = 10^6$. The results are shown in Table 2.

The results improve those in [21] by providing a significantly lower relative error for the last method, while requiring a similar sized pilot run. For this problem, the acceptance rate was $\kappa \approx 2.58$, giving an initial sample size of approximately $\kappa k N_v = 258\,000$. Note that the discrepancy in reported values of $\hat{\ell}$ in Table 2 and [21] is due to rounding errors in the reported transition vector \mathbf{u} in [21].

We see that ZV state-dependent CE achieved higher accuracy than state-dependent CE (and standard CE). Increasing the number of bins β increases the accuracy of the estimate, as long as the importance sampling parameters for each bin can be calculated with sufficient accuracy. The minimum number of samples per bin ω was set at 20.

The standard CE and state-dependent CE methods required four parameter updates, each using $N_v = 10^5$ for each update to obtain the importance sampling parameters. Sampling from the zero-variance distribution needed a single update with only $N_v = 10^3$. As the rare event ℓ to be estimated becomes smaller, sampling from g^* becomes increasingly more efficient compared with standard updating CE methods.

5.2. Tandem Jackson queue

Rare-event probability estimation problems have been studied extensively in the context of queueing systems. An important benchmark model is the tandem Jackson network. In this system customers arrive to the first queue according to a Poisson process with rate λ and are served by a single server in the order in which they arrive. If a customer arrives when the server is busy serving another customer, the arriving customer joins the end of the waiting line. Once the service is completed, the customer moves on to the second queue, where the customer waits until served by the second server. The first and second servers have exponential service times with rates μ_1 and μ_2 , respectively.

Let $X_1(t)$ and $X_2(t)$ be the number of customers in the first and second queues, respectively (including the customers who are being served), at time t . Then $\{(X_1(t), X_2(t)), t \geq 0\}$ is an MJP with $p = 3$ transition types, with transition rates

$$\begin{aligned} q((x_1, x_2), (x_1 + 1, x_2)) &= \lambda, \\ q((x_1, x_2), (x_1 - 1, x_2 + 2)) &= \mu_1 \mathbf{1}_{\{x_1 \geq 1\}}, \\ q((x_1, x_2), (x_1, x_2 - 1)) &= \mu_2 \mathbf{1}_{\{x_2 \geq 1\}}, \end{aligned}$$

where $\mathbf{1}_{\{x_i \geq 1\}}$ is the indicator that there is at least one customer in the i th queue.

This model is known to exhibit interesting rare-event behavior depending on the choice of the parameters λ , μ_1 , and μ_2 . A typical rare-event quantity of interest is the probability that the second queue reaches some high level y before it becomes empty. When $\mu_1 > \mu_2$ (the second queue is the bottleneck), a simple and effective way to estimate the rare event is to use state-independent importance sampling with the change of parameters $\tilde{\lambda} = \mu_2$, $\tilde{\mu}_1 = \mu_1$, and $\tilde{\mu}_2 = \lambda$. However, for the case $\mu_1 < \mu_2$ (the first queue is the bottleneck), a state-independent change of measure does not work. In [15] a state-dependent change of measure is derived using the theory of Markov-additive processes. Similar changes of measure are outlined in [18]. The effort required to obtain \hat{v}^* in [15] and [18] is considered negligible compared to the total simulation effort.

Example 3. (*Tandem queue (simple).*) Consider a tandem queue with transition rates $\lambda = 1$, $\mu_1 = 4$, and $\mu_2 = 2$. Thus, the second queue is the bottleneck. We wish to estimate the probability ℓ that the second queue reaches 25 before it becomes empty, starting from state $(1, 1)$. In Table 3 we compare the estimates and relative error from [15] (KN) with the zero-variance state-dependent CE method. For the latter, we used a function ϕ_i of the form (10) to partition the state space. Note that in this case there are only two possible values for each ϕ_i . This is because the MJP stops as soon as $X_2 = 0$, thus making

$$\phi_i(\mathbf{x}) = q_i / (\lambda + \mu_1 + \mu_2) \quad \text{or} \quad \phi_i(\mathbf{x}) = q_i / (\lambda + \mu_2),$$

the latter occurring when $x_1 = 0$, where $q_i = \lambda$, $\mu_1 \mathbf{1}_{\{x_1 \geq 1\}}$, and $\mu_2 \mathbf{1}_{\{x_2 \geq 1\}}$ for $i = 1, 2$, and 3 , respectively.

TABLE 3: Estimation of the probability ℓ that the second queue reaches level 25 before becoming empty, starting from $X(0) = (1, 1)$.

Method	Relative error	$\hat{\ell}$	N	N_v	β
KN state dependent	0.0011	4.47×10^{-8}	10^6	—	—
ZV state-dependent CE	0.0011	4.48×10^{-8}	10^6	10^4	2

We see that both methods had the same level of accuracy, showing that the automated method successfully finds the state-independent importance sampling parameters for this case.

Example 4. (*Tandem queue (difficult 1).*) We consider the same problem as in the previous example, but now with parameter values $\lambda = 1$, $\mu_1 = 2$, and $\mu_2 = 3$, which makes queue 1 the bottleneck. As mentioned before, in this case state-independent importance sampling does not work. For this problem, the bins were calculated in relation to the population sizes at each queue, rather than the relative probabilities as in (10). In particular, we took $\phi_1(\mathbf{x}) = \min(x_1/C_1, 1)$, $\phi_2(\mathbf{x}) = \min(x_1/C_2, 1)$, and $\phi_3(\mathbf{x}) = x_2/C_3$, where C_1 , C_2 , and C_3 are all 20. This is somewhat similar to the type of dependence along the state space boundaries used in general in [18]. The nature of ϕ_1 , ϕ_2 , and ϕ_3 used here is more suitable for this problem, and queueing problems in general. This is because the transition functions have indicator variables in relation to the distribution (population in each queue). As such, ϕ_i functions similar to (10), which was implemented in Example 3, do not utilize all of the possible statistical information to break up the state space and assign importance sampling parameters—Example 3 only had two bins for each transition! The method applied to this particular example takes into account how many customers are in each queue at every step of the Markov process. We applied zero-variance state-dependent CE and compared it with the state-dependent method in [15] (KN). The results are shown in Table 4.

This particular choice for the ϕ functions (which are similar in nature to [18]) results here in a better change of measure than in [15], as is evident from the reduction in relative error for the ZV state-dependent CE method compared to the KN state-dependent method used in [15]. The state-dependent approach in [15] requires a numerical calculation of the importance parameters at each predefined bin in the state space, which is done prior to the main Monte Carlo simulation. The two other methods are adaptive and automatic in their approach to calculate the state-dependent importance parameters via deterministic equations.

Example 5. (*Tandem queue (difficult 2).*) As a further illustration of the methodology, consider the same model as in Example 4 where ℓ is now the probability that the second queue reaches level 20 before *both* queues become empty, starting from (1, 1). In Table 5 we present the results for the combination-specific importance sampling, using the same ϕ_i functions as Example 4.

A typical rare-event trajectory for this example is shown in Figure 1. Here X_1 builds up first, and while it empties into the second buffer, X_2 increases ‘linearly’ towards level 20. We have included this plot to show how sampling from the zero-variance distribution via the hit-and-run sampler can yield important information on the manner in which a rare event takes place.

TABLE 4: Estimation of the probability ℓ that the second queue reaches level 20 before becoming empty, starting from $\mathbf{X}(0) = (1, 1)$.

Method	Relative error	$\hat{\ell}$	N	N_v	β
KN state dependent	0.0049	2.05×10^{-11}	10^6	—	—
ZV state-dependent CE	0.0024	2.04×10^{-11}	10^6	10^4	10

TABLE 5: Estimation of the probability that the second queue reaches level 20 before both queues become empty, starting from $\mathbf{X}(0) = (1, 1)$.

Method	Relative error	$\hat{\ell}$	N	N_v	β
ZV state-dependent CE	0.0021	1.263×10^{-9}	10^6	10^4	10

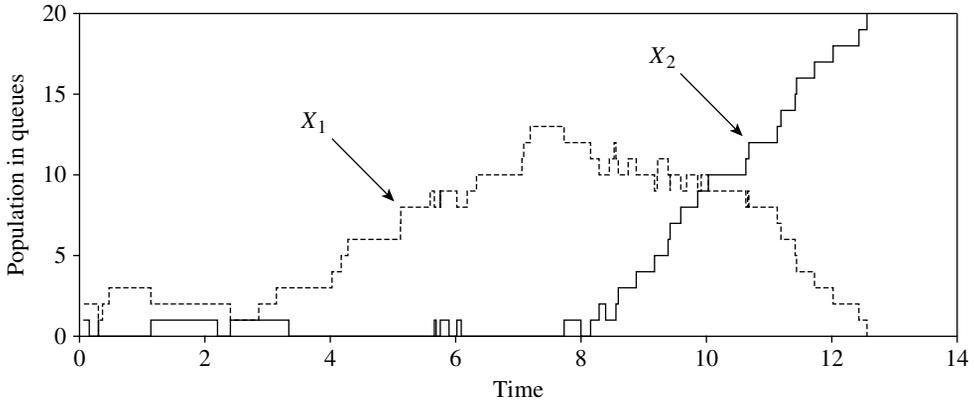


FIGURE 1: The first queue X_1 fills up, at which point the second queue X_2 proceeds to overflow while X_1 proceeds to empty.

Acknowledgement

This work was supported by the Australian Research Council under grant number DP0985-177.

References

- [1] ASMUSSEN, S. (1987). *Applied Probability and Queues*. John Wiley, Chichester.
- [2] ASMUSSEN, S. AND GLYNN, P. W. (2007). *Stochastic Simulation: Algorithms and Analysis*. Springer, New York.
- [3] BLANCHET, J. AND LAM, H. (2012). State-dependent importance sampling for rare-event simulation: an overview and recent advances. *Surveys Operat. Res. Manag. Sci.* **17**, 38–59.
- [4] CHAN, J. C. C. AND KROESE, D. P. (2012). Improved cross-entropy method for estimation. *Statist. Comput.* **22**, 1031–1040.
- [5] COHEN, J. E. (1972). Markov population processes as models of primate social and population dynamics. *Theoret. Pop. Biol.* **3**, 119–134.
- [6] DAIGLE, B. J., JR., ROH, M. K., GILLESPIE, D. T. AND PETZOLD, L. R. (2011). Automated estimation of rare event probabilities in biochemical systems. *J. Chem. Phys.* **134**, 044110.
- [7] DE BOER, P. T. AND NICOLA, V. F. (2002). Adaptive state-dependent importance sampling simulation of Markovian queueing networks. *Europ. Trans. Telecommun.* **13**, 303–315.
- [8] DUPUIS, P. AND WANG, H. (2009). Importance sampling for Jackson networks. *Queueing Systems* **62**, 113–157.
- [9] DUPUIS, P., SEZER, A. D. AND WANG, H. (2007). Dynamic importance sampling for queueing networks. *Ann. Appl. Prob.* **17**, 1306–1346.
- [10] GILLESPIE, D. T. (1977). Exact stochastic simulation of coupled chemical reactions. *J. Physical Chem.* **81**, 2340–2361.
- [11] GLYNN, P. W. AND IGLEHART, D. L. (1989). Importance sampling for stochastic simulations. *Manag. Sci.* **35**, 1367–1392.
- [12] GRACE, A. W. AND WOOD, I. A. (2012). Approximating the tail of the Anderson–Darling distribution. *Comput. Statist. Data Anal.* **56**, 4301–4311.
- [13] HEIDELBERGER, P. (1995). Fast simulation of rare events in queueing and reliability models. *ACM Trans. Modeling Comput. Simul.* **5**, 43–85.
- [14] KROESE, D. P., TAIMRE, T. AND BOTEV, Z. I. (2011). *Handbook of Monte Carlo Methods*. John Wiley, Hoboken, NJ.
- [15] KROESE, D. P. AND NICOLA, V. F. (2002). Efficient simulation of a tandem Jackson network. *ACM Trans. Modeling Comput. Simul.* **12**, 119–141.
- [16] LOVÁSZ, L. (1999). Hit-and-run mixes fast. *Math. Programming* **86**, 443–461.
- [17] MCQUARRIE, D. A. (1967). Stochastic approach to chemical kinetics. *J. Appl. Prob.* **4**, 413–478. (Correction: **5** (1968), 484–485.)
- [18] NICOLA, V. F. AND ZABURNENKO, T. S. (2007). Efficient importance sampling heuristics for the simulation of population overflow in Jackson networks. *ACM Trans. Modeling Comput. Simul.* **17**, 2, article 10.

- [19] PEARL, R. (1925). *The Biology of Population Growth*. Knopf, New York.
- [20] POLLETT, P. K. AND VASSALLO, A. (1992). Diffusion approximations for some simple chemical reaction schemes. *Adv. Appl. Prob.* **24**, 875–893.
- [21] ROH, M. K., DAIGLE, B. J., JR., GILLESPIE, D. T. AND PETZOLD, L. R. (2011). State-dependent doubly weighted stochastic simulation algorithm for automatic characterization of stochastic biochemical rare events. *J. Chem. Phys.* **135**, 234108.
- [22] RUBINSTEIN, R. Y. AND KROESE, D. P. (2004). *The Cross-Entropy Method. A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*. Springer, New York.
- [23] SMITH, R. L. (1984). Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operat. Res.* **32**, 1296–1308.
- [24] SMITH, R. L. (1996). The hit-and-run sampler: a globally reaching Markov chain sampler for generating arbitrary multivariate distributions. In *Proc. 28th Conf. Winter Simulation*, IEEE, Washington, DC, pp. 260–264.
- [25] VAN KAMPEN, N. G. (1992). *Stochastic Processes in Physics and Chemistry*. North Holland, Amsterdam.
- [26] WOLF, R. W. (1989). *Stochastic Modeling and the Theory of Queues*. Prentice Hall, Englewood Cliffs, NJ.