Proceedings of the 2011 Winter Simulation Conference
S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, eds.

AN IMPORTANCE SAMPLING METHOD BASED ON A ONE-STEP LOOK-AHEAD DENSITY FROM A MARKOV CHAIN

Zdravko I. Botev
Pierre L'Ecuyer

DIRO, Université de Montreal
C.P. 6128, Succ. Centre-Ville
Montréal (Québec), H3C 3J7, CANADA

Bruno Tuffin

INRIA Rennes Bretagne Atlantique
Campus Universitaire de Beaulieu
35042 Rennes Cedex, FRANCE

ABSTRACT

We propose a new importance sampling method that constructs an importance sampling density which approximates the zero-variance sampling density nonparametrically as follows. In a first stage, it generates a sample (possibly approximately) from the zero-variance density using, for example, Markov chain Monte Carlo methodology. In a second stage, the method constructs a kernel density estimator of the zero-variance density based on the sample in the first stage. The most important aspect of the method is that, unlike other kernel estimation methods, the kernel of the estimator is defined as the one-step transition density of a Markov chain whose stationary distribution is the zero-variance one. We give examples where this one-step transition density is available analytically and provide numerical illustrations in which the method performs very well.

1 INTRODUCTION

Importance sampling (IS) is an extremely powerful variance reduction method for stochastic simulation. With an optimal sampling density, it can actually produce a zero-variance estimator. However, finding an admissible sampling density that is a good approximation of the zero-variance one and from which it is easy to sample, can be difficult. It is also well-known that a poor choice of a sampling density can be disastrous; the variance can increase by an arbitrary large factor, and can even become infinite (Asmussen and Glynn 2007, L'Ecuyer, Mandjes, and Tuffin 2009).

In this paper, we propose a new two-stage approach that constructs an IS sampling density and uses it to compute a low-variance unbiased estimator. This is done in a setting where we want to estimate a finite-dimensional integral that represents the mathematical expectation of a function of some random vector with given density. In the first stage (a learning phase), we obtain a sample of m observations (not necessarily independent) approximately distributed according to the zero-variance distribution. In our examples, we use Markov chain Monte Carlo (MCMC) methodology to generate this sample. This MCMC approach constructs an artificial Markov chain whose stationary distribution is the zero-variance distribution, runs this chain, and picks the set of visited states, or a subset of those, as the m observations. Based on this sample, the method constructs a kernel density to be used in the second stage as an IS sampling density to sample n new states independently. This produces an unbiased IS estimator, together with an unbiased variance estimator, from which a confidence interval can be computed. The kernel density is constructed by selecting a transition density of a Markov chain whose corresponding stationary distribution is the zero-variance one (it could be the same as for the Markov chain in the first stage, or a different one). Our method applies under the assumption that we can generate a sample approximately from the zero-variance density. We argue that this is often the case and give examples where this is achieved by constructing a Markov chain whose stationary distribution equal to the zero-variance one. We call the method *Markov* *chain importance sampling* (MCIS), to reflect the fact that it combines Markov chain methodology with IS.

Other adaptive IS methods where the sampling density is learned in a first stage and used in the second stage have been proposed in the past (Juneja and Shahabuddin 2006, Asmussen and Glynn 2007, Rubinstein and Kroese 2007, L'Ecuyer, Mandjes, and Tuffin 2009). Some of these methods select a *parametric class* of IS densities and try to estimate in a first stage a value of the parameter that minimizes the variance. In the context of rare-event simulation, this is often done in an incremental fashion, step by step, where the first step uses a much easier problem formulation in which the "rare event" is not so rare (say, its probability is 5% or more). At any given step, the original problem is modified so that the rare event is less likely to occur (more rare) than for the previous step, and the IS parameter estimated at the previous step is used in an IS scheme to estimate the optimal parameter for the current step, until we reach the original problem formulation with the smallest rare-event probability. The estimated optimal parameter at each step can be taken directly as the one that minimizes the empirical variance of the IS estimator as a function of the sampling parameter, as discussed in Rubinstein and Shapiro (1993), for example. This is the *variance minimization* (VM) method.

A more general problem formulation replaces the variance minimization by an optimization problem where the objective is to minimize a general measure of distance or divergence between the zero-variance density and the density that corresponds to the selected parameter (Rubinstein and Kroese 2007). Rubinstein (1997) proposed a special case of this approach with the Kullback-Leibler (or cross-entropy) divergence, motivated by the fact that it yields an expression that is sometimes more stable and is much easier to minimize when the parametric class of sampling densities is from an exponential family. He called it the *cross-entropy* (CE) method. This method has been experimented in a large variety of settings, and also generalized to other measures of divergence (Rubinstein and Kroese 2007).

Restricting ourselves to a parametric class of IS densities has the disadvantage that even the best parameter choice may yield a poor sampling density if the zero-variance density is far from the parameterized family. That is, a good choice of a parametric class depends very much on the application and can be difficult to find in general. One way to address this problem might be to make the parametric class more general and flexible, but then the parameters are more numerous and more difficult to optimize. Going further in that direction leads to estimating IS sampling densities nonparametrically, as in Zhang (1996). However, the performance of these methods is usually disappointing in practice, especially when the dimension increases, primarily because of the difficulty of accurate density estimation in high dimensions. Another approach, popular in statistics, is the method of Chib (1995), which exploits the fact that the zero-variance density is often known explicitly up to a multiplicative constant (it is proportional to the integrand in the expectation), and that this constant is equal to the value of the integral. Chib (1995) estimates this constant (i.e., the integral) by estimating the zero-variance density at a single point (where this density is large enough), using a Gibbs sampling scheme. However, this estimator is biased, can be very noisy, and its computation is time-consuming. Moreover, the method does not deliver independent samples, so it becomes difficult to compute a confidence interval.

Another class of methods approximate the zero-variance density by using asymptotic approximations of the expectation conditional on the realizations of any given subset of coordinates of the random vector, and plugging these approximations in recurrence formulas (Asmussen and Glynn 2007, Juneja and Shahabuddin 2006, L'Ecuyer and Tuffin 2008, L'Ecuyer, Mandjes, and Tuffin 2009). This frequently leads to densities that are exponentially twisted versions of the original ones. These approximations must be available a priori and must be easily computable. However, good approximations of that type are not always available, and when they are, they are usually accurate only in situations where the rare event is very rare.

The MCIS method proposed here constructs a nonparametric density estimate that is naturally adapted to the problem and uses this density to compute independent replicates of an unbiased estimator, from which a confidence interval of the unknown expectation is easy to obtain. Our empirical experiments show that this estimator can outperform other available IS schemes on certain types of applications. A key requirement for the effectiveness of the method is the ability to construct a Markov chain whose stationary distribution is the zero-variance one, and whose one-step transition density is known explicitly. In principle, this can always be achieved via a Metropolis-Hastings type of construction, but the Markov chain must also be easy to sample from and should be rapidly mixing for good performance. Our algorithm is appropriate when this is possible.

In the next section, we state our MCIS algorithm, after defining the setting in which it applies and recalling some basic facts on IS and MCMC. In Section 3, we give numerical results for two examples. Section 4 provides a conclusion and points out topics that deserve further attention.

2 MARKOV CHAIN IMPORTANCE SAMPLING

2.1 Problem Setting and Importance Sampling

We consider the problem of estimating a mathematical expectation written as a d-dimensional integral of the form

$$\ell = \mathbb{E}_f[h(\mathbf{Y})] = \int_{\mathbb{R}^d} f(\mathbf{y})h(\mathbf{y})\,\mathrm{d}\mathbf{y}, \tag{1}$$

where $h : \mathbb{R}^d \to \mathbb{R}$ can be seen as some cost or reward function and f is the probability density function (pdf) of the random variable $\mathbf{Y}$ over $\mathbb{R}^d$. Note that this density can be nonzero over only a strict subset of $\mathbb{R}^d$. Ordinary Monte Carlo estimates ℓ by sampling n independent realizations of $\mathbf{Y}$ from density f, and averaging the n corresponding realizations of $h(\mathbf{Y})$.

With *importance sampling*, the n independent realizations of $\mathbf{Y}$, say $\mathbf{Y}_1 \ldots, \mathbf{Y}_n$, are sampled from another density g for which $g(\mathbf{y}) > 0$ whenever $h(\mathbf{y})f(\mathbf{y}) \neq 0$. The (unbiased) IS estimator of ℓ is (Hammersley and Handscomb 1964, Asmussen and Glynn 2007):

$$\hat{\ell} = \frac{1}{n}\sum_{i=1}^{n} \frac{h(\mathbf{Y}_i)f(\mathbf{Y}_i)}{g(\mathbf{Y}_i)}. \tag{2}$$

An unbiased estimator of $\mathrm{Var}[\hat{\ell}]$ is S_n^2/n where S_n^2 is the sample variance of the $h(\mathbf{Y}_i)f(\mathbf{Y}_i)/g(\mathbf{Y}_i)$'s, and this variance estimate can be used to compute a confidence interval for ℓ in a standard way, using a normal approximation. The main difficulty in applying this method is to find a good IS sampling density g.

For simplicity, we assume in this paper that h is never negative (otherwise, we can write it as $h = h^+ - h^-$ and apply the method to $h^+ \geq 0$ and $h^- \geq 0$ separately). Then, the optimal g (minimizing the variance of $\hat{\ell}$) is $g = \pi$ where

$$\pi(\mathbf{x}) = \frac{h(\mathbf{x})f(\mathbf{x})}{\ell}. \tag{3}$$

This choice of density π reduces the variance to zero. But unfortunately, this π depends on the unknown normalizing constant ℓ, which is what we want to estimate in the first place. The practical goal is to find a g that approximates π as well as possible. The quality of this approximation can be measured in a variety of ways, for example by a measure of divergence between the two densities (Rubinstein and Kroese 2007), and this could be useful for convergence analysis, but we will not pursue this in the present paper.

One class of approaches to find a good g is to select a parameterized family of densities $\{g(\cdot;\theta),\, \theta \in \Theta\}$, and try to optimize the parameter θ. In the VM method, θ is selected as the parameter value that minimizes the empirical variance of the IS estimator obtained in a preliminary stage. For the CE method, θ is taken as the value that minimizes the (empirical) cross-entropy between $g(\cdot;\theta)$ and the density π based on the sample from the preliminary stage.

2.2 Constructing Markov Chains Whose Stationary Density is the Zero-Variance One

MCMC consists in constructing an artificial Markov chain $\{\mathbf{X}_i,\, i \geq 0\}$ having a given (desired) stationary distribution. In our context, this desired stationary distribution has density π. A general way of constructing such a Markov chain is the Metropolis-Hastings (MH) algorithm, defined as follows (Robert and Casella 2004, Asmussen and Glynn 2007, Kroese, Taimre, and Botev 2011). We first select $\{q(\cdot \mid \mathbf{x}), \mathbf{x} \in \mathbb{R}^d\}$, which represents the transition density of a Markov chain and is taken as the proposal density at each step. We also define

$$\alpha(\mathbf{x},\mathbf{y}) \stackrel{\text{def}}{=} \min\left\{\frac{\pi(\mathbf{y})\, q(\mathbf{x} \mid \mathbf{y})}{\pi(\mathbf{x})\, q(\mathbf{y} \mid \mathbf{x})}, 1\right\} \tag{4}$$

for all $\mathbf{x},\mathbf{y} \in \mathbb{R}^d$.

Algorithm 1 : Metropolis-Hastings for MCMC

```
Initialize $\mathbf{X}_0$ to some state for which $\pi(\mathbf{X}_0) > 0$
for $i = 1$ to $m$ do
    given $\mathbf{X}_{i-1}$, generate $\mathbf{Y}$ from density $q(\cdot \mid \mathbf{X}_{i-1})$
    with probability $\alpha(\mathbf{X}_{i-1},\mathbf{Y})$, put $\mathbf{X}_i \leftarrow \mathbf{Y}$, else $\mathbf{X}_i \leftarrow \mathbf{X}_{i-1}$
```

The artificial Markov chain $\{\mathbf{X}_i, i \geq 0\}$ generated by this algorithm has stationary density π. Moreover, if this chain is π-irreducible, which means that every measurable set $B \subset \mathbb{R}^d$ such that $\int_B \pi(\mathbf{x})\mathrm{d}\mathbf{x} > 0$ is visited infinitely often with probability one, and which we shall assume in this paper, then the empirical distribution of $\mathbf{X}_1,\ldots,\mathbf{X}_m$ also converges in total variation to the zero-variance distribution (Asmussen and Glynn 2007, Theorem 3.5).

In general, we may not want to directly use the sequence $\mathbf{X}_1,\ldots,\mathbf{X}_m$ produced by this algorithm, but rather a decimated subsequence, such as $\mathbf{X}_s,\mathbf{X}_{2s},\ldots,\mathbf{X}_{ms}$ for some integer s, to reduce the dependence between successive retained states.

Two important special cases of Algorithm 1 are the hit-and-run sampler (Chen and Schmeiser 1993, Chen, Shao, and Ibrahim 2000) and Gibbs sampling (Asmussen and Glynn 2007, Kroese, Taimre, and Botev 2011). In our examples, we will use Gibbs sampling, defined as follows, where $\mathbf{X}_i = (X_{i,1},\ldots,X_{i,d})$ and $\mathbf{Y} = (Y_1,\ldots,Y_d)$.

Algorithm 2 : Gibbs sampling

```
Initialize $\mathbf{X}_0$ to a state for which $\pi(\mathbf{X}_0) > 0$
for $i = 1$ to $m$ do
    for $j = 1$ to $d$ do
        generate $Y_j$ from density $\pi(\cdot \mid Y_1,\ldots,Y_{j-1},X_{i-1,j+1},\ldots,X_{i-1,d})$
    $\mathbf{X}_i \leftarrow \mathbf{Y}$
```

In this algorithm, $\alpha(\mathbf{x},\mathbf{y}) \equiv 1$ and the transition density is the product

$$\kappa(\mathbf{y} \mid \mathbf{x}) = \prod_{j=1}^{d} \pi(y_j \mid y_1,\ldots,y_{j-1},x_{j+1},\ldots,x_d). \tag{5}$$

Other techniques can be used as well to obtain the sample $\mathbf{X}_1,\ldots,\mathbf{X}_m$. The only condition is that it must be generated approximately from π. In one of our examples, we will in fact obtain this sample via the generalized splitting methodology of Botev and Kroese (2010).

MCMC technology permits us to sample states approximately from the zero-variance density, but this sample cannot be used directly to define an unbiased IS estimator, because the corresponding sampling density g is not known explicitly. Moreover, the sampled states $\mathbf{X}_i$ are not independent. A first (naive) idea might be to just use the empirical distribution of $\mathbf{X}_1,\ldots,\mathbf{X}_m$ for IS sampling. But this distribution does not satisfy the requirement of having a nonzero density wherever $h(\mathbf{x})f(\mathbf{x}) > 0$. To satisfy this requirement, a second idea is to fit a kernel density estimate to this sample, based on a kernel whose density is nonzero wherever $h(\mathbf{x})f(\mathbf{x}) > 0$. However, kernel density estimation by standard methods is difficult in high dimensions. What we do instead is to use the density one step ahead in the Markov chain with transition density κ, when the chain is in state $\mathbf{X}_i$, and average this over the m states $\mathbf{X}_1,\dots,\mathbf{X}_m$. In other words, regardless of how the transition density κ is constructed, we approximate the density π by $\hat{\pi}$, defined as

$$\hat{\pi}(\mathbf{y}) = \frac{1}{m}\sum_{i=1}^{m} \kappa(\mathbf{y} \mid \mathbf{X}_i) \tag{6}$$

for $\mathbf{y} \in \mathbb{R}^d$. Sampling from $\hat{\pi}$ is easy: It suffices to generate an index J uniformly over $\{1,\dots,m\}$ and then generate $\mathbf{Y}$ from density $\kappa(\cdot \mid \mathbf{X}_J)$.

If we know that the support of $\kappa(\cdot \mid \mathbf{x})$ covers the support of π for each $\mathbf{x}$, then $\hat{\pi}$ is guaranteed to be a valid IS sampling density (that is, it is nonzero whenever $h(\mathbf{x})f(\mathbf{x}) > 0$). Otherwise, we may have to modify $\hat{\pi}$ to make sure that it is nonzero over the entire support of π. One very simple way of doing this is to take a mixture of $\hat{\pi}$ with the original density f. That is, we can use the IS sampling density

$$\hat{\pi}_w(\mathbf{y}) = w\, f(\mathbf{x}) + (1-w)\,\hat{\pi}(\mathbf{x})\,, \tag{7}$$

where $0 \le w < 1$.

2.3 The MCIS Algorithm

Putting together the ingredients described in the previous subsection, we obtain the following MCIS algorithm, where we assume that the mixture parameter w is specified in advance and is possibly zero.

Algorithm 3 : Markov Chain Importance Sampling

```
First stage, MC:
Generate a sample $\mathbf{X}_1,\dots,\mathbf{X}_m$ (approximately) iid with density $\pi$
Second stage, IS:
for $i = 1$ to $n$ do
    generate $U$ uniformly over $(0,1)$
    if $U < w$ then
        generate $\mathbf{Y}_i \sim f(\cdot)$
    else
        generate $J$ uniformly over the set $\{1,\dots,m\}$
        generate $\mathbf{Y}_i$ from density $\kappa(\cdot \mid \mathbf{X}_J)$
return the MCIS estimator
```

$$\hat{\ell} = \frac{1}{n}\sum_{i=1}^{n} \frac{h(\mathbf{Y}_i)f(\mathbf{Y}_i)}{\hat{\pi}_w(\mathbf{Y}_i)}. \tag{8}$$

One can also compute the sample variance S_n^2 of the n realizations $\{h(\mathbf{Y}_i)f(\mathbf{Y}_i)/\hat{\pi}_w(\mathbf{Y}_i), i = 1,\dots,n\}$, which are independent conditional on $\hat{\pi}$, and estimate the relative error of $\hat{\ell}$ by $n^{-1/2}S_n/\hat{\ell}$. This estimate can eventually be used to compute a confidence interval for ℓ.

In this algorithm, the computation of $\hat{\pi}(\mathbf{Y}_i)$ in the denominator can be time-consuming when m is large. We emphasize that the returned estimators $\hat{\ell}$ and S_n^2 are unbiased for the mean and the variance even if $\mathbf{X}_1,\dots,\mathbf{X}_m$ are not independent and do not have density π exactly, and regardless of the quality of the approximation in the first step of the algorithm. However, the performance of the MCIS algorithm certainly depends on the quality of this approximation.

3 NUMERICAL ILLUSTRATIONS

3.1 Example: A Shortest Path Problem

For our first (small) example, we consider the bridge network in Figure 1, in which we want to estimate the expected length ℓ of the shortest path between nodes A and B. The lengths of the five links are independent

Table 1: Performance comparison of some IS methods on the bridge network.

method	estimate	relative error	VRF
Monte Carlo	0.93	0.43%	1.00
CE	0.9289	0.25%	2.95
VM	0.9295	0.24%	3.21
Chib	0.9296	0.09%	22.8
MCIS	0.92978	0.015%	810.9

random variables $X_1,\ldots,X_5$, where X_j is uniform over $(0,a_j)$ and $(a_1,\ldots,a_5)=(1,2,3,2,1)$. The density $f(\mathbf{x})$ is uniform over the rectangle $\prod_{j=1}^{5}(0,a_j)$. Given $\mathbf{X}=(X_1,\ldots,X_5)$, $h(\mathbf{X})$ is the length of the shortest path.

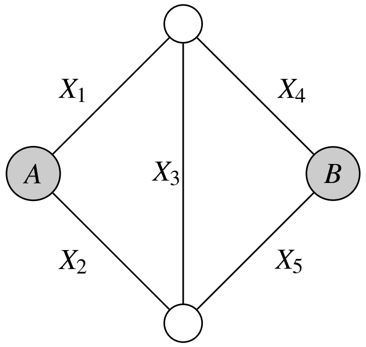

Figure 1: A bridge network with four nodes and five links.

In the MCIS algorithm, we use a Markov chain whose transition density is defined by systematic Gibbs sampling (Algorithm 2). With $\mathbf{x}_{-j}$ denoting $\mathbf{x}$ without its coordinate j, the conditional density $\pi(y_j \mid \mathbf{x}_{-j})$ for coordinate j turns out to be linear in y_j in the interval $(0,\eta_j]$ and constant in the interval (η_j,a_j), where η_j is the threshold above which link j does not belong to the shortest path, given $\mathbf{x}_{-j}$. The actual expressions are a bit lengthy but not difficult to obtain;

For the first stage of the MCIS algorithm, we apply $m=100$ steps of Gibbs sampling, starting at $\mathbf{X}_0=(a_1/2,\ldots,a_5/2)$, to obtain a population of 100 states. In the second stage, we apply IS with $n=10^4$, using the IS sampling density $\hat{\pi}$ in (6) with κ also defined by the Gibbs sampling scheme described above. We also implemented and ran the VM, CE, and Chib's method, all with sample size $n=10^4$. For the CE and VM methods we used the importance sampling density

$$g(\mathbf{y};\theta)=\prod_{i=1}^{5}\frac{\theta_i}{a_i}\left(\frac{y_i}{a_i}\right)^{\theta_i-1}, \quad y_i\in(0,a_i),$$

where $\theta=(\theta_1,\ldots,\theta_5)\in\mathbb{R}^+$ is a parameter vector. Note that $g(\mathbf{y};\mathbf{1})$ yields the original uniform distribution. The estimated optimal parameter for the VM method was $\hat{\theta}_{\text{VM}}=(1.26,1.08,1.01,1.23,1.06)$ and for the CE method $\hat{\theta}_{\text{CE}}=(1.27,1.12,1.00,1.32,1.07)$; see Kroese, Taimre, and Botev (2011), Chapter 9) for more details. For Chib's method, we took $\mathbf{x}^*=(a_1/2,\ldots,a_5/2)$ as the point of density estimation and replicated the method 10 times to estimate the relative error. Other values of $\mathbf{x}^*$ did not perform as well. Table 3.1 reports the estimate of ℓ, the estimated relative error, and the estimated variance reduction factor (VRF) with respect to crude Monte Carlo for all these methods.

We find that for this particular example, the MCIS method gives the smallest relative error and the largest variance reduction factor, by a large margin. These comparisons do not account for the CPU times, but these computational times were about the same for the MCIS method and Chib's estimator in our implementation. The comparisons also do not account for the preliminary computations required to estimate θ in the VM and CE algorithms, and which make them even less competitive.

3.2 Example: A Static Network Reliability Problem

Our second example is a static network reliability problem, already considered by many authors; see Cancela et al. (2009), Cancela et al. (2009), Gertsbakh and Shpungin (2010) and the references therein. We want to estimate the probability ℓ that nodes A and B are *not* connected in the network of Figure 2, if each of the 30 edges (or links) fails with probability ε, and those failures are independent.

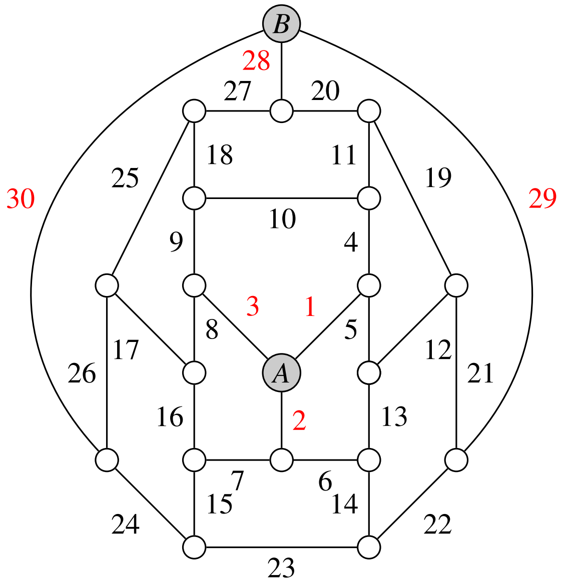

Figure 2: A dodecahedron graph with 20 nodes and 30 links.

More specifically, suppose each edge $e \in \{1,\dots,d\}$ is assigned a random weight X_e and $X_1,\dots,X_d$ are independent exponential random variables with rate $\lambda = -\log(\varepsilon)$. Then, $\varepsilon = \mathbb{P}(X_e > 1)$ for all edges and the event $\{X_e > 1\}$ is equivalent to the event that edge e has failed. Let $\mathscr{P} = \{P_j\}$ denote the set of all paths connecting nodes A and B (P_j represents a sequence of edges connecting nodes A and B). The failure probability ℓ can be expressed as

$$\ell = \mathbb{P}(S(\mathbf{X}) > 1),$$

where $\mathbf{X} = (X_1,\dots,X_d)$ has density f and S is defined via

$$S(\mathbf{X}) = \min_{P_j \in \mathscr{P}} \max_{e \in P_j} X_e\ . \tag{9}$$

When ε is small, say, $\varepsilon = 10^{-3}$, the probability ℓ is typically a rare-event probability. We will apply the MCIS Algorithm 3 to estimate ℓ in this situation.

In the first (MC) stage, we use the adaptive splitting algorithm from Botev and Kroese (2010) to generate a Markov chain sample $\mathbf{X}_1,\dots,\mathbf{X}_m$ with approximate density π. This is only one of many ways in which to generate the sample $\mathbf{X}_1,\dots,\mathbf{X}_m$. We summarize this splitting algorithm in the appendix. For the current example we use a splitting factor of $s = 2$ in that algorithm.

Table 2: Reliability of dodecahedron network for various values of ε.

link reliability ε	MCIS estimate $\hat{\ell}$	MCIS relative error	CE relative error
10^{-3}	1.98×10^{-9}	1.0%	63%
10^{-4}	2.02×10^{-12}	1.3%	68%
10^{-5}	2.01×10^{-15}	1.1%	98%
10^{-6}	1.97×10^{-18}	1.2%	87%
10^{-7}	1.98×10^{-21}	1.2%	63%

In the second (IS) stage, we use the mixture (7) with $w = 0.5$ as the importance sampling density and (5) as the Markov transition density. Here, the conditional pdfs of $\pi(\mathbf{x})$ are:

$$\pi(x_e \,|\, \mathbf{X}_{-e}) = \lambda\, \mathrm{e}^{-\lambda x_e} \left(\mathbb{I}\{S_e > 1\} + \frac{1 - \mathbb{I}\{S_e > 1\}}{\varepsilon} \mathbb{I}\{x_e > 1\} \right) = \begin{cases} \lambda\, \mathrm{e}^{-\lambda x_e} & \text{if } S_e > 1 \\ \lambda\, \mathrm{e}^{-\lambda (x_e - 1)} \mathbb{I}\{x_e > 1\} & \text{otherwise} \end{cases},$$

for $e = 1, \ldots, d$, where $\{S_e > 1\} \equiv \{S(X_1, \ldots, X_{e-1}, 0, X_{e+1}, \ldots, X_d) > 1\}$ is the event that the nodes are not connected given that edge e is forced to work (which is the same as $X_e < 1$). That is, if adding link e does not make the network operational, no IS is applied when sampling this link, otherwise a distribution truncated to $[1, \infty)$ is used for IS.

Thus, if $\mathbf{Y} \sim \hat{\pi}_w(\mathbf{y})$, the likelihood ratio for the importance sampling estimator (8) simplifies to

$$\frac{f(\mathbf{Y})\mathbb{I}\{S(\mathbf{Y}) \geq 1\}}{\hat{\pi}_w(\mathbf{Y})} = \left(w + \frac{1-w}{n} \sum_{i=1}^{n} \prod_{e=1}^{d} \left(\mathbb{I}\{S_e^{(i)} > 1\} + \frac{1 - \mathbb{I}\{S_e^{(i)} > 1\}}{\varepsilon} \mathbb{I}\{Y_e > 1\} \right) \right)^{-1},$$

where $S_e^{(i)}$ corresponds to $\mathbf{X}_i$. Table 3.2 gives numerical results for various values of ε with $m = 20$ and $n = 10^4$. The second and third columns give the MCIS estimate (8) and the corresponding estimated relative error (in percentage). The fourth column gives the estimated relative error for the cross entropy method using the importance sampling density

$$g(\mathbf{y}; \theta) = \prod_{e=1}^{d} \theta_e\, \mathrm{e}^{-\theta_e y_e}, \quad \mathbf{y} \in \mathbb{R}^+$$

parameterized by the vector $\theta = (\theta_1, \ldots, \theta_d) \in \mathbb{R}^+$, and with the same simulation effort of 10^4 and θ estimated in the preliminary stage. From the table we can see that in this case the MCIS method estimates the zero-variance density better than the CE method and as a result the resulting importance sampling scheme is more efficient.

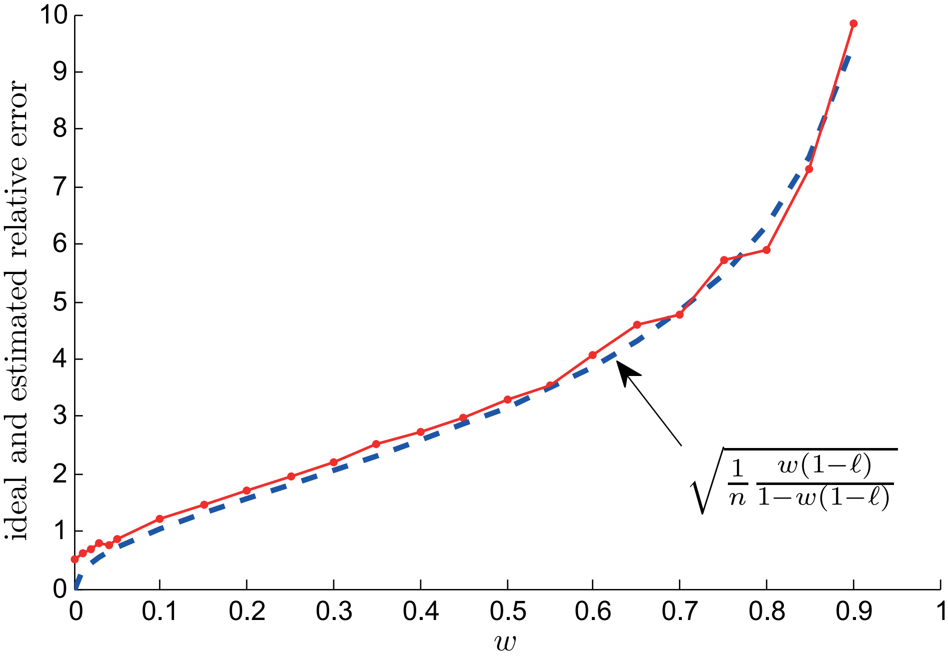

Figure 3: The estimated (solid red curve) versus the ideal (dashed curve) relative error (in percentage) as a function of w.

Suppose that within the mixture (7) we have estimated π perfectly, that is, $\hat{\pi} \equiv \pi$. Then, the importance sampling density is $\hat{\pi}_w \equiv \pi_w$, where

$$\pi_w(\mathbf{y}) = w\, f(\mathbf{x}) + (1-w)\,\pi(\mathbf{x})\,,$$

and we have the following lower bound for the squared relative error of (8):

$$\frac{\mathrm{Var}(\hat{\ell})}{\ell^2} = \frac{1}{n}\mathrm{Var}\left(\frac{\pi(\mathbf{Y})}{\hat{\pi}_w(\mathbf{Y})}\right) \geq \frac{1}{n}\int \frac{(\pi(\mathbf{y}) - \pi_w(\mathbf{y}))^2}{\pi_w(\mathbf{y})}\,\mathrm{d}\mathbf{y} = \frac{w(1-\ell)}{n(1-w(1-\ell))}.$$

This lower bound is achieved in the ideal case when $\hat{\pi} \equiv \pi$; that is, when we have eliminated the error arising from the estimation of π. It is thus interesting to compare the estimated relative error of (8) with the relative error when $\hat{\pi} \equiv \pi$. Figure 3 shows the behavior of the estimated and ideal relative errors for various values of w with $\varepsilon = 10^{-3}$, $m = 100$, $n = 10^3$. The ideal relative error (in percentage),

$$\sqrt{w(1-\ell)/(n - n w\,(1-\ell))}\,,$$

is plotted as a dashed curve, and the estimated relative error is plotted as a red curve. From the graph we can see that the estimated relative error is slightly larger than the ideal one (it gets slightly smaller at some points due to simulation noise). In addition, while the ideal relative error goes to zero when $w \to 0$, the estimated relative error remains above 0.5% for all values of w. The reason for this is that $\hat{\pi} \not\equiv \pi$ and hence the ideal lower bound is not reached.

4 CONCLUSION

We have presented a novel method of approximating the optimal importance sampling density, which combines importance sampling with Markov chain sampling. The method requires that we generate a population of random variables (approximately) distributed according to the zero-variance importance sampling density, and that we have an analytical expression for a Markov transition density with stationary distribution equal to the zero-variance importance sampling density. This Markov transition density is then used as the kernel of a nonparametric kernel density estimator of the zero-variance importance sampling density. Numerical results indicate that this approach is viable and that it can be more effective than alternative methods for estimating the zero-variance density such as, for example, the cross entropy and variance minimization methods.

An issue that calls for further study is the dependence of the MCIS estimator on the quality of the random sample $\mathbf{X}_1,\ldots,\mathbf{X}_m$. Ideally, we would like to have an iid sample from the optimal importance sampling density, but this is rarely possible and we have to resort to MCMC sampling. Thus, the performance of the MCIS sampler depends on the rate of convergence of the MCMC sampler. Such an analysis will give insight into the optimal allocation of simulation effort between the MC step (learning phase) and the IS step (estimation phase).

ACKNOWLEDGMENTS

This work has been supported by INRIA's associated team MOCQUASIN to all authors, and NSERC-Canada and a Canada Research Chair to the second author.

APPENDIX

Here we briefly present the MCMC sampling algorithm that is used to generate an approximate sample from the zero-variance density π in Example 2. The main idea is to use the splitting method (Botev and Kroese 2010). Suppose we are given an integer $s \geq 2$, called the splitting factor. Initially, we generate $m \times s$ independent states $\mathbf{X}$ from density f, and determine a threshold parameter γ_1 so that exactly m of them have $S(\mathbf{X}) \geq \gamma_1$. Then at each step t, for $t = 2,3,\ldots,$ we run for s steps a Markov chain with stationary density $f(\mathbf{x})\mathbb{I}\{S(\mathbf{x}) \geq \gamma_{t-1}\}/\mathbb{P}(S(\mathbf{x}) \geq \gamma_{t-1})$, from each of those m states $\mathbf{X}$ for which $S(\mathbf{X}) \geq \gamma_{t-1}$. We denote the transition kernel density of this Markov chain by $\tilde{\kappa}_{t-1}$. This gives another $m \times s$ states and we select a parameter γ_t so that exactly m of them have $S(\mathbf{X}) \geq \gamma_t$. This is done until $\gamma_t \geq 1$ for some t. This iterative procedure is summarized in the following algorithm.

Algorithm 4 Adaptive splitting sampler.

```
Require: an integer $s \geq 2$
  $q \leftarrow m \times s - m$
  $\mathscr{X}_1 \leftarrow \emptyset$
  for $i = 1$ to $m \times s$ do
      generate a vector $\mathbf{Y}$ from density $f$ and add it to $\mathscr{X}_1$
  sort the elements of $\mathscr{X}_1$ by increasing order of $S(\mathbf{X})$, say $\mathbf{X}_{(1)},\ldots,\mathbf{X}_{(m\times s)}$
  $\gamma_1 \leftarrow [S(\mathbf{X}_{(q)}) + S(\mathbf{X}_{(q+1)})]/2$
  $t \leftarrow 1$
  while $\gamma_t \leq 1$ do
      $t \leftarrow t+1$
      $\mathscr{X}_{t-1} \leftarrow \{\mathbf{X}_{(q)},\ldots,\mathbf{X}_{(m\times s)}\}$      // retain only the best $m$ elements from $\mathscr{X}_{t-1}$
      $\mathscr{X}_t \leftarrow \emptyset$
      for all $\mathbf{X}_0 \in \mathscr{X}_{t-1}$ do
          for $j = 1$ to $s$ do
              sample $\mathbf{X}_j$ from the density $\tilde{\kappa}_{t-1}(\cdot \mid \mathbf{X}_{j-1})$ and add it to $\mathscr{X}_t$
      sort the elements of $\mathscr{X}_t$ by increasing order of $S(\mathbf{X})$, say $\mathbf{X}_{(1)},\ldots,\mathbf{X}_{(m\times s)}$
      $\gamma_t \leftarrow \min\{[S(\mathbf{X}_{(q)}) + S(\mathbf{X}_{(q+1)})]/2, 1\}$
  return $\mathbf{X}_1 \ldots,\mathbf{X}_m$, for which $S(\mathbf{X}) \geq 1$, as a sample with approximate density $\pi(\mathbf{x}) = f(\mathbf{x})\mathbb{I}\{S(\mathbf{x}) \geq 1\}/\ell$
```

In this algorithm, $\mathscr{X}_t$ denotes a set of vectors $\mathbf{X}$ for which $S(\mathbf{X}) > \gamma_{t-1}$. When this set contains $m \times s$ elements, we sort it to retain the m vectors having the largest value of $S(\mathbf{X})$, and we remove the other vectors from this set. The threshold parameter γ_t is placed midway between the m-th and the $(m+1)$-th largest values of $S(\mathbf{X})$. In our example we select the transition density $\tilde{\kappa}_{t-1}$ to be the transition density of the systematic Gibbs sampler with stationary density $f(\mathbf{x})\mathbb{I}\{S(\mathbf{x}) \geq \gamma_{t-1}\}/\mathbb{P}(S(\mathbf{x}) \geq \gamma_{t-1})$.

REFERENCES

Asmussen, S., and P. W. Glynn. 2007. *Stochastic Simulation*. New York: Springer-Verlag.

Botev, Z. I., and D. P. Kroese. 2010. "Efficient Monte Carlo Simulation via the Generalized Splitting Method". *Statistics and Computing*. to appear.

Cancela, H., M. El Khadiri, and G. Rubino. 2009. "Rare Event Analysis by Monte Carlo Techniques in Static Models". In *Rare Event Simulation Using Monte Carlo Methods*, edited by G. Rubino and B. Tuffin, 145–170. Wiley. Chapter 7.

Cancela, H., P. L'Ecuyer, M. Lee, G. Rubino, and B. Tuffin. 2009. "Analysis and Improvements of Path-Based Methods for Monte Carlo Reliability Evaluation of Static Models". In *Simulation Methods for Reliability and Availability of Complex Systems*, edited by J. Faulin, A. A. Juan, S. Martorell, and E. Ramirez-Marquez, 65–84. Springer Verlag.

Chen, M.-H., and B. W. Schmeiser. 1993. "Performance of the Gibbs, Hit-and-Run, and Metropolis Samplers". *Journal of Computational and Graphical Statistics* 2:251–272.

Chen, M.-H., Q. M. Shao, and J. G. Ibrahim. 2000. *Monte Carlo Methods in Bayesian Computations*. New York: Springer-Verlag.

Chib, S. 1995. "Marginal Likelihood from Gibbs Output". *Journal of the American Statistical Association* 90 (432): 1313–1321.

Gertsbakh, I. B., and Y. Shpungin. 2010. *Models of Network Reliability*. Boca Raton, FL: CRC Press.

Hammersley, J. M., and D. C. Handscomb. 1964. *Monte Carlo Methods*. London: Methuen.

Juneja, S., and P. Shahabuddin. 2006. "Rare Event Simulation Techniques: An Introduction and Recent Advances". In *Simulation*, edited by S. G. Henderson and B. L. Nelson, Handbooks in Operations Research and Management Science, 291–350. Amsterdam, The Netherlands: Elsevier. Chapter 11.

Kroese, D. P., T. Taimre, and Z. I. Botev. 2011. *Handbook of Monte Carlo Methods*. New York: John Wiley and Sons.

L'Ecuyer, P., M. Mandjes, and B. Tuffin. 2009. "Importance Sampling and Rare Event Simulation". In *Rare Event Simulation Using Monte Carlo Methods*, edited by G. Rubino and B. Tuffin, 17–38. Wiley. Chapter 2.

L'Ecuyer, P., and B. Tuffin. 2008, December. "Approximate Zero-Variance Simulation". In *Proceedings of the 2008 Winter Simulation Conference*, edited by S. J. Mason, R. R. Hill, L. Moench, O. Rose, T. Jefferson, and J. W. Fowler, 170–181. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Robert, C. P., and G. Casella. 2004. *Monte Carlo Statistical Methods*. second ed. New York, NY: Springer-Verlag.

Rubinstein, R., and D. P. Kroese. 2007. *Simulation and the Monte Carlo Method*. second ed. New York: John Wiley & Sons.

Rubinstein, R. Y. 1997. "Optimization of Computer Simulation Models with Rare Events". *European Journal of Operations Research* 99:89–112.

Rubinstein, R. Y., and A. Shapiro. 1993. *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*. New York: Wiley.

Zhang, P. 1996. "Nonparametric Importance Sampling". *Journal of the American Statistical Association* 91 (435): 1245–1253.

AUTHOR BIOGRAPHIES

ZDRAVKO I. BOTEV is a postdoctoral fellow at Université de Montréal, Canada. He obtained his Ph.D. in Mathematics from The University of Queensland, Australia, in 2010. His main research interest are in Monte Carlo methods and computational statistics. He has written jointly with D. P. Kroese and T. Taimre a *Handbook of Monte Carlo Methods* published by John Wiley & Sons in 2011. He will join the Statistics Department of the University of New South Wales, in Sydney, Australia, in 2012. His email address is: botev@iro.umontreal.ca

PIERRE L'ECUYER is Professor in the Département d'Informatique et de Recherche Opérationnelle, at the Université de Montréal, Canada. He holds the Canada Research Chair in Stochastic Simulation and Optimization. He is a member of the CIRRELT and GERAD research centers. His main research interests are random number generation, quasi-Monte Carlo methods, efficiency improvement via variance reduction, sensitivity analysis and optimization of discrete-event stochastic systems, and discrete-event simulation in general. He is currently Editor-in-Chief for *ACM Transactions on Modeling and Computer Simulation*, and Associate/Area Editor for *ACM Transactions on Mathematical Software*, *Statistics and Computing*, *International Transactions in Operational Research*, and *Cryptography and Communications*. More information and his recent research articles are available on-line from his web page: http://www.iro.umontreal.ca/~lecuyer

BRUNO TUFFIN received his PhD degree in applied mathematics from the University of Rennes 1 (France) in 1997. Since then, he has been with INRIA in Rennes. He spent eight months as a postdoc at Duke University in 1999. His research interests include developing Monte Carlo and quasi-Monte Carlo simulation techniques for the performance evaluation of telecommunication systems, and developing new Internet-pricing schemes and telecommunication-related economical models. He is currently Associate Editor for *INFORMS Journal on Computing*, *ACM Transactions on Modeling and Computer Simulation* and *Mathematical Methods of Operations Research*. He has written or co-written two books devoted to simulation: *Rare event simulation using Monte Carlo methods* published by John Wiley & Sons in 2009, and *La simulation de Monte Carlo* (in French), published by Hermes Editions in 2010. His email address is bruno.tuffin@inria.fr.