Introduction to Rare-Event Simulation

Chapter · January 2011

DOI: 10.1002/9780470400531.eorms0006

2 authors:

John F. Shortle
George Mason University
133 PUBLICATIONS **1,826** CITATIONS

SEE PROFILE

Pierre L'Ecuyer
Université de Montréal
345 PUBLICATIONS **14,048** CITATIONS

SEE PROFILE

DRAFT: Introduction to Rare-event Simulation

John F. Shortle
Department of Systems Engineering & Operations Research, George Mason University, 4400 University Dr., MSN 4A6, Fairfax, VA 22030 {jshortle@gmu.edu}

Pierre L'Ecuyer
Département d'Informatique et de Recherche Opérationnelle
Université de Montreal, C.P. 6128, Succ. Centre-Ville
Montréal (Québec), H3C 3J7, Canada, http://www.iro.umontreal.ca/~lecuyer

Abstract: Without special techniques, simulation of rare events can be prohibitive due to the large amount of computer time needed. This article gives an introduction to methods that can be used to substantially reduce the variance associated with simulating rare events. Two broad classes of methods are discussed—importance sampling and splitting.

1. Introduction

The evaluation of rare-event probabilities can pose significant challenges for simulation. To illustrate, let γ be the probability of a rare event. The standard way to estimate γ is to conduct n i.i.d. replications. Assuming that the outcome of each replication is either a 1 (denoting the occurrence of the rare event) or a 0 (denoting the non-occurrence of the event), then the standard estimator $\hat{\gamma}$ is the number of observed occurrences divided by the number of replications. This is a binomial random variable with mean γ and variance

$$\mathrm{Var}[\hat{\gamma}] = \frac{\gamma(1-\gamma)}{n} \approx \frac{\gamma}{n}.$$

Although $\mathrm{Var}[\hat{\gamma}]$ gets smaller as the rare-event probability γ decreases, it is more useful to consider the *relative error* of the estimator, defined here as the standard deviation divided by the actual value:

$$\text{Relative Error} \equiv \frac{\sqrt{\mathrm{Var}[\hat{\gamma}]}}{\gamma} \approx \frac{\sqrt{\gamma/n}}{\gamma} = \frac{1}{\sqrt{\gamma n}}. \qquad (1)$$

The relative error can be interpreted as the root mean square error of the estimator $\hat{\gamma}$ divided by the root mean square error of the trivial estimator that always returns the value zero. If $\hat{\gamma}$ is approximately normally distributed, the relative error provides a measure of the size of the confidence interval relative to the value being estimated (though "approximately normal" is not typical in the context of rare events). The basic problem is that the relative error increases as the rare-event probability gets smaller. More generally, the required number of replications grows inversely with γ. It also grows inversely with the square of the relative error. Table 1 shows examples of the required time to achieve a given relative error for a specified rare-event probability (assuming simulation speeds of 1 and 1,000 replications per second). This illustrates that standard simulation may be impractical for many rare-event problems.

Table 1: Computer time needed to achieve a specified relative error (standard simulation).

	1 replication per second		1,000 replications per second	
	Desired Relative Error		Desired Relative Error	
γ	100%	10%	100%	10%
10^{-3}	17 min	1 day	1 sec	2 min
10^{-5}	1 day	4 months	2 min	3 hr
10^{-7}	4 months	32 years	3 hr	12 days
10^{-9}	32 years	3,169 years	12 days	3 years

Rare-event simulation is used in many application areas, including finance and insurance where the rare event could be a large financial gain or loss, communication systems where the rare event could be the loss of important information, power systems where the rare event could be a major power outage, network reliability where the rare event could mean that certain nodes in the network are disconnected, nuclear physics where the rare event could be that particles cross a given protection shield, and aviation safety where the rare event could be an aircraft collision. It is also encountered in computer graphics (image synthesis by Monte Carlo methods), computational statistics, computational physics, computational biology, military applications, and so on. The goal is not always to estimate the probability of a rare event. It can be to estimate the mathematical expectation of a random variable that is strongly affected by rare events. But the main ideas and methods are basically the same as when the goal is to estimate the rare event probability itself.

There are two main approaches for improving the efficiency of rare-event simulations – importance sampling (IS) and splitting. The idea of IS is to change the underlying sampling distribution so that rare events are more likely. The idea of splitting is to create separate copies of the simulation whenever the simulation gets “close” to the rare event of interest, effectively multiplying promising runs that are more likely to reach the rare event. Splitting is useful for systems that tend to take many incremental steps on the path to the rare event. Importance sampling is also useful for systems that tend to take a small number of "catastrophic jumps" to the rare event.

This article does not give a comprehensive literature review but simply offers some suggestions for further reading: The book edited by Rubino and Tuffin (2009) contains survey chapters on a wide range of topics and applications in rare-event simulation. Much of the material in this article can be found in more elaborate form in the first three chapters of that book. More extensive coverage and lists of references on IS can be found in Heidelberger (1995), Bucklew (2004), Juneja and Shahabuddin (2006), and Asmussen and Glynn (2007). Good introductory references on splitting include Garvels (2000) and L'Ecuyer et al. (2006).

2. Importance Sampling

We first consider IS as applied to a static problem. Let X be a random variable in d-dimensional space with density function f. Let $\mathcal{R}$ denote a rare-event set in $\mathbb{R}^d$. Let $\gamma = \Pr\{X \in \mathcal{R}\}$. The standard way to estimate γ using Monte Carlo simulation is to generate independent samples $X_1, \ldots, X_n$ from the density f and to estimate γ by

$$\hat{\gamma} = \frac{1}{n} \sum_{i=1}^{n} I_{\mathcal{R}}(X_i), \tag{2}$$

where $I_{\mathcal{R}}(x) = 1$ if $x \in \mathcal{R}$ and 0 otherwise. If γ is very small, the relative error $(\gamma n)^{-1/2}$ of this estimator can be very large. In fact, $\hat{\gamma}$ may turn out to be 0, in which case a confidence interval on γ computed in a standard way would also have zero width.

The idea of IS is to sample values from a different density function g rather than from the original density f. Presumably, the new density g is chosen so that the rare event is more likely to occur. Naturally, some adjustment must be made to correct for sampling from a different density function. The procedure is to generate samples $Y_1, \ldots, Y_n$ from the density g and to estimate γ by

$$\hat{\gamma} = \frac{1}{n} \sum_{i=1}^{n} \frac{f(Y_i)}{g(Y_i)} I_{\mathcal{R}}(Y_i) = \frac{1}{n} \sum_{i=1}^{n} L(Y_i) I_{\mathcal{R}}(Y_i), \tag{3}$$

where $L(y) \equiv f(y)/g(y)$ is the *likelihood ratio* of the two densities. To show that $\hat{\gamma}$ is unbiased, observe that

$$\mathrm{E}\left[\frac{f(Y_i)}{g(Y_i)} I_{\mathcal{R}}(Y_i)\right] = \int \frac{f(y)}{g(y)} I_{\mathcal{R}}(y)\, g(y) dy = \int I_{\mathcal{R}}(x)\, f(x) dx = \gamma.$$

It is assumed that $g(y) > 0$ whenever $f(y)I_{\mathcal{R}}(y) > 0$.

A key objective is to find a change of measure that achieves significantly lower variance for the IS estimator in (3) compared with the standard estimator in (2). That is, we seek to minimize the variance of $I_{\mathcal{R}}(Y_i)L(Y_i)$. In fact, it is possible to find a change of measure such that $I_{\mathcal{R}}(Y_i)L(Y_i)$ has *zero variance*. This is achieved by the *optimal change of measure*:

$$g(y) = \frac{1}{\gamma}f(y)I_{\mathcal{R}}(y) \qquad \text{and} \qquad L(y) = \frac{\gamma}{I_{\mathcal{R}}(y)}.$$

The optimal change of measure g has two key properties: (a) It is zero outside of $\mathcal{R}$, and (b) it is proportional to the original density f inside of $\mathcal{R}$. The first property implies that $I_{\mathcal{R}}(Y_i)$ (sampled from the density g) is always 1. The second property implies that $L(Y_i) = f(Y_i)/g(Y_i)$ is always the same constant. Together, these two properties imply that $L(Y_i)I_{\mathcal{R}}(Y_i)$ has zero variance.

Practically speaking, finding the optimal change of measure requires knowing the rare-event probability γ, which eliminates the need for simulation in the first place. But, the optimal change of measure is still useful in the sense that it provides a guideline for choosing a good change of measure. A good change of measure is mostly concentrated on the rare-event set $\mathcal{R}$ and is roughly proportional to the original density f on this set. Alternatively, $L(y)$ is small and roughly constant when $y \in \mathcal{R}$.

In the next examples, γ is known a priori, so there is no need for simulation. We use these small examples just to illustrate the basic ideas.

Example (L'Ecuyer et al., 2009b): Let X be an exponential random variable with density $f(x) = \lambda e^{-\lambda x}$, and $\mathcal{R} = \{x | x \geq T\}$, so $\gamma = e^{-\lambda T}$. The optimal change of measure is $g(y) = \lambda e^{-\lambda y}/e^{-\lambda T}$ for $y \geq T$ and $g(y) = 0$ elsewhere. But suppose for the sake of example that we only allow changes of measure that are linear scalings of the original density. That is, the new sampling density is restricted to be exponential with a different parameter μ, namely $g(y) = \mu e^{-\mu y}$. The likelihood ratio is

$$L(y) = \frac{\lambda e^{-\lambda y}}{\mu e^{-\mu y}} = (\lambda/\mu)e^{-(\lambda-\mu)y}.$$

Minimizing the variance of the IS estimator $\hat{\gamma}$ in (3) is equivalent to minimizing the second moment of $L(Y_i)I_{\mathcal{R}}(Y_i)$:

$$\begin{aligned}
\mathrm{E}[[L(Y_i)I_{\mathcal{R}}(Y_i)]^2] &= \mathrm{E}[L^2(Y_i)I_{\mathcal{R}}(Y_i)] \\
&= \int_T^\infty (\lambda/\mu)^2 e^{-2(\lambda-\mu)y}\, g(y)dy = (\lambda^2/\mu)\int_T^\infty e^{-(2\lambda-\mu)y}dy \\
&= \frac{\lambda^2}{\mu(2\lambda-\mu)}e^{-(2\lambda-\mu)T}. \qquad (4)
\end{aligned}$$

The integral is finite only when $2\lambda - \mu > 0$, so $\mathrm{Var}[\hat{\gamma}]$ is finite only for $0 < \mu < 2\lambda$. In particular, $\mathrm{Var}[\hat{\gamma}] \to \infty$ as $\mu \to 0$. This shows that simply forcing the samples to take on large values (i.e., by letting μ be a small number) is not necessarily a good idea. Although a small value of μ makes $I_{\mathcal{R}}(Y_i)$ more likely to equal one, it may drastically increase the likelihood function $L(Y_i)$ for some $Y_i \in \mathcal{R}$ by making its denominator $g(Y_i)$ (the new density) much too small, thus increasing the overall variance of $I_{\mathcal{R}}(Y_i)L(Y_i)$. This highlights a key difficulty of IS in general: one must make sure that the probabilities (or densities) of sample realizations that contribute to the expectation being estimated are never reduced too much, because otherwise the estimator may take huge values occasionally and will then have a large variance. In typical multivariate situations, this can be very difficult to take care of.

Figure 1 shows the ratio of the IS-estimator variance and the standard-estimator variance as a function of the change-of-measure parameter μ. The figure shows results for two different problems ($T = 3$ and $T = 5$) with $\lambda = 1$. Intuitively, as might be expected, choosing a value of μ larger than λ (i.e., forcing the samples to take on smaller values) makes the IS estimator worse than standard simulation. Choosing a value of μ smaller than λ (i.e., forcing the samples to take larger values) typically makes the IS estimator better than standard simulation. However, μ cannot be too small, otherwise the variance can be much worse than standard simulation. This illustrates the fact that without being sufficiently careful, it is easy to choose an IS estimator that is much worse than the standard estimator.

It is easily verified that the value of μ that minimizes the second moment in (4) is $\mu^* = \lambda + 1/T - (\lambda^2 + 1/T^2)^{1/2}$. This value is always smaller than λ and $1/T$. Also, $\mu^* T \to 1$ when $T \to \infty$. The squared relative error, namely $((4) - \gamma^2)/\gamma^2$, with the optimal μ is

$$\frac{\lambda^2}{\mu^*(2\lambda-\mu^*)}e^{\mu^* T} - 1 \;\sim\; \frac{e}{2}\lambda T - 1 = \frac{e}{2}\ln(1/\gamma) - 1,$$

where $u(\cdot) \sim v(\cdot)$ if $u(T)/v(T) \to 1$ as $T \to \infty$. Although the squared relative error increases when $\gamma \to 0$ (or $T \to \infty$), it only increases at a logarithmic speed in $1/\gamma$ (or linearly in T). In contrast, with crude Monte Carlo, it increases linearly in $1/\gamma$ (or exponentially in T). For

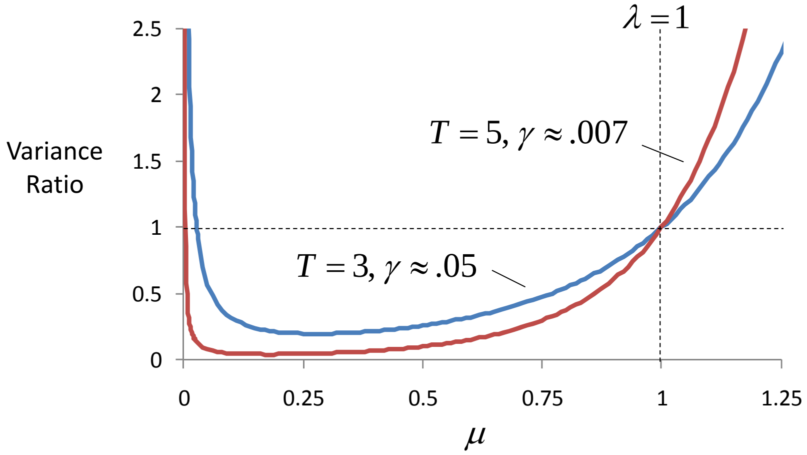

Figure 1: Dependence of variance ratio (Var$[L(Y_i)I_{\mathcal{R}}(Y_i)]$ / Var$[I_{\mathcal{R}}(X_i)]$) as a function of μ.

example, suppose that $\lambda = 1$ and $T = 20$, so $\gamma = e^{-20}$. With crude Monte Carlo, it would take about $n \geq 100 \cdot (1/\gamma) \approx 5 \times 10^{10}$ replications to get a relative error below 10%. With the given IS scheme, it would take about $n \geq 100 \cdot [(e/2)\ln(1/\gamma) - 1] \approx 2,618$ replications.

Very similar examples can be constructed with other distributions, for example a normal or a Poisson distribution for which we want to estimate the (small) probability that the random variable exceeds a given threshold.

With the IS scheme in the previous example, the relative error was still increasing in T, but more slowly than with crude Monte Carlo. Ideally, we would like to have *bounded relative error* meaning that the relative error remains bounded as $T \to \infty$, or more generally as $\gamma \to 0$. In this case, a fixed number of runs permits one to achieve a desired relative error regardless of how rare the event is. Here, we have a slightly weaker condition called *logarithmic efficiency* (or sometimes *asymptotic optimality*), namely that

$$\lim_{\gamma \to 0} \frac{\ln \mathrm{E}[\hat{\gamma}^2]}{\ln \gamma^2} = 1. \tag{5}$$

This means that the second moment of the estimator and the square of the mean go to zero at the same exponential rate when $\gamma \to 0$. This is almost as good as bounded relative error, and it can be obtained in a much broader range of situations. There are also situations where one can achieve *vanishing relative error* under IS; this means that the relative error converges to 0 when $\gamma \to 0$, so rarer events are easier to estimate (L'Ecuyer and Tuffin, 2009; L'Ecuyer et al., 2010). These asymptotic properties of the second (relative) moment also generalize to moments of any order (L'Ecuyer et al., 2010). For example, *logarithmic efficiency of order k* means that (5) holds with 2 replaced by k.

Now we consider IS applied to discrete-time Markov chains (DTMCs). Let $\mathcal{R}$ denote a set of rare-event states and let $\mathcal{O}$ denote a set of "other" terminating states. The problem is to estimate the probability γ that a Markov chain enters $\mathcal{R}$ before $\mathcal{O}$. The idea of IS is to change the transition probabilities to reduce the variance of the rare-event estimator. Specifically, let $p(x, y)$ be the transition probability from x to y in the original chain, and let $p'(x, y)$ be the transition probability from x to y in a modified chain. The only restriction is that $p'(x, y)$ must be positive whenever $p(x, y)$ has a positive contribution to γ. Let $X_0, X_1, X_2, \ldots$ and $Y_0, Y_1, Y_2, \ldots$ be sets of states sampled from the original and modified chains. Let M and N be stopping times when the original and modified chains first enter either $\mathcal{R}$ or $\mathcal{O}$. The rare-event probability is $\gamma = \Pr\{X_M \in \mathcal{R}\}$.

Standard simulation estimates γ by simulating the original Markov chain to generate multiple samples of $X = I_{\mathcal{R}}(X_M)$. Importance sampling simulates the modified Markov chain to generate samples of

$$Y = I_{\mathcal{R}}(Y_N)L(Y_0, Y_1, \ldots, Y_N),$$

where the likelihood ratio $L(\cdot)$ of a sample path is:

$$L(Y_0, Y_1, \ldots, Y_N) = \frac{\prod_{n=1}^{N} p(Y_{n-1}, Y_n)}{\prod_{n=1}^{N} p'(Y_{n-1}, Y_n)}. \tag{6}$$

As in the static case, the modified sampling is unbiased: $\mathrm{E}[Y] = \gamma$. The main challenge is to find a change of measure so that $\mathrm{Var}[Y] \ll \mathrm{Var}[X]$. As before, it is possible (in theory) to find a change of measure such that $\mathrm{Var}[Y] = 0$. Specifically, the optimal change of measure is

$$p'(x, y) = \frac{\gamma(y)}{\gamma(x)} p(x, y), \tag{7}$$

where $\gamma(x) = \Pr\{X_M \in \mathcal{R} | X_0 = x\}$ is the probability that the original chain enters $\mathcal{R}$ before $\mathcal{O}$, starting in state x. Note that $p'(x, y) = 0$ when $y \in \mathcal{O}$, since $\gamma(y) = 0$ when $y \in \mathcal{O}$. Thus, the optimal change of measure cuts all links to terminating states that are not rare-event states. This forces the chain to eventually hit the rare-event set with probability 1. To verify that $\mathrm{Var}[Y] = 0$:

$$\begin{aligned}
Y &= I_{\mathcal{R}}(Y_N)L(Y_0, Y_1, \ldots, Y_N) \\
&= I_{\mathcal{R}}(Y_N)\frac{\prod_{n=1}^{N} p(Y_{n-1}, Y_n)}{\prod_{n=1}^{N} p'(Y_{n-1}, Y_n)} \\
&= I_{\mathcal{R}}(Y_N)\frac{\prod_{n=1}^{N} p(Y_{n-1}, Y_n)\gamma(Y_{n-1})}{\prod_{n=1}^{N} p(Y_{n-1}, Y_n)\gamma(Y_n)} \\
&= I_{\mathcal{R}}(Y_N)\frac{\gamma(Y_0)}{\gamma(Y_N)} \\
&= \gamma(Y_0).
\end{aligned}$$

The last equality follows because $\gamma(Y_N) = I_{\mathcal{R}}(Y_N)$, since Y_N is in a terminating state in either $\mathcal{R}$ or $\mathcal{O}$. Without loss of generality, we can assume that Y_0 is a constant, so $\mathrm{Var}[Y] = \mathrm{Var}[\gamma(Y_0)] = 0$. (If Y_0 is random, define a pseudo-initial state at time -1 that transitions to one of the random initial states at time 0.)

Example: Consider an $M/M/1/K$ queue with arrival rate λ, service rate μ, and maximum system capacity K. The problem is to estimate the probability γ of reaching K customers in the system before returning to the empty state, starting from the empty state. This is equivalent to the gambler's ruin problem in which a gambler makes i.i.d. bets, winning \$1 with probability $p = \lambda/(\lambda+\mu)$ and losing \$1 with probability $q = \mu/(\lambda+\mu)$. The probability $\gamma(n)$ of reaching K in the system before returning to the empty state (or getting K dollars before loosing everything), starting with n in the system, has a simple analytic expression (e.g., Ross, 2007):

$$\gamma(n) = \frac{1-(q/p)^n}{1-(q/p)^K} = \frac{1-(1/\rho)^n}{1-(1/\rho)^K}, \tag{8}$$

where $\rho = \lambda/\mu$. Figure 2 shows the transition probabilities of the original chain and a modified chain. From (7), the optimal change of measure is

$$p'_n = \frac{\gamma(n+1)}{\gamma(n)}p = \frac{(1/\rho)^{n+1}-1}{(1/\rho)^n-1}p, \qquad n = 1, 2, \ldots, K-1. \tag{9}$$

As $n \to \infty$, $p'_n \to (1/\rho)p = q$ (assuming $\rho < 1$). Thus, for large n, the optimal change of measure approximately swaps the arrival and service rates.

Since the optimal change of measure requires knowing the rare-event function $\gamma(\cdot)$, we may wish to consider a restricted set of measures in which the modified transition probabilities are *state independent*, namely $p'_n \equiv p'$ for some value p'. Intuitively, a larger value for p'

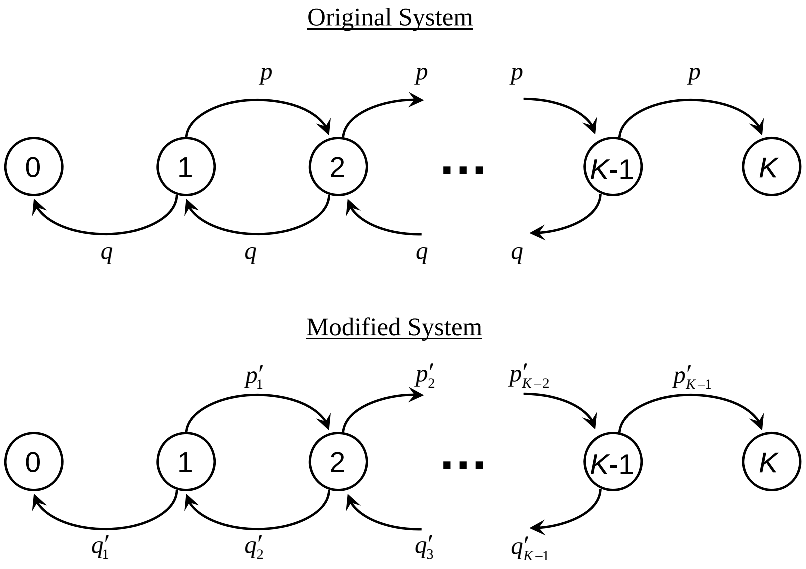

Figure 2: $M/M/1/K$ transition probabilities.

makes the rare event more likely, but if it is too large this can be detrimental. To illustrate, suppose that $p' = 1 - \epsilon$, where $\epsilon > 0$ is some small number. The most likely path to $\mathcal{R}$ is a sequence of upward jumps $Y_0 = 1, Y_1 = 2, \ldots, Y_{K-1} = K$. This path occurs with probability $(1-\epsilon)^{K-1} \approx 1$. The likelihood ratio of this path is $[p/(1-\epsilon)]^{K-1}$. Occasionally, a sample path may backtrack one state before continuing with upward jumps (e.g., $1, 2, 3, 2, 3, 4, \ldots, K$). The likelihood ratio of this path is $qp^K/\epsilon(1-\epsilon)^K$. That is, the loop from state 3 to 2 to 3 increases the likelihood ratio by the factor $qp/\epsilon(1-\epsilon)$. This path occurs with probability $\epsilon(1-\epsilon)^K$, so it contributes the following term to the *second moment* of the likelihood ratio:

$$\left(\frac{qp^K}{\epsilon(1-\epsilon)^K}\right)^2 \epsilon(1-\epsilon)^K \approx \frac{q^2p^{2K}}{\epsilon}.$$

This implies that the variance of the overall estimator $Y = I_{\mathcal{R}}(Y_N)L(Y_0, \ldots Y_N)$ can be made arbitrarily large for small enough ϵ. In summary, although $p' = 1-\epsilon$ has the positive effect of making the rare event very likely ($I_{\mathcal{R}}(Y_N)$ is usually 1), it has the negative effect of creating the potential for very large likelihood ratios, which acts to increase the variance of the overall estimator $I_{\mathcal{R}}(Y_N)L(Y_0, \ldots Y_N)$.

A better change of measure makes the rare event very likely but also keeps the likelihood ratio roughly constant over these sample paths. This illustrates why switching the arrival and service rates (i.e., $p' = q$ and $q' = p$) is close to optimal. A loop from n to $n-1$ to n modifies the likelihood ratio by a factor $qp/q'p' = 1$, leaving the likelihood ratio unchanged.

The true optimal change of measure in (9) additionally cuts the link to state 0, and also keeps the property that the likelihood ratio remains unchanged over loops (that is, $qp/q'_n p'_{n-1} = 1$).

The previous one-dimensional examples are relatively simple in that the rare-event probabilities are known and the optimal change of measure can be found exactly. In practical problems, the optimal change of measure is not known and the Markov chains typically have multidimensional states, which increases the difficulty. To some extent, the one-dimensional examples are deceiving because the sample paths that lead to the rare event are readily identified, which is hardly true in higher-dimensional settings. We briefly discuss a few approaches for handling more realistic problems. For more detailed summaries of these methods, see, for example, Juneja and Shahabuddin (2006) and L'Ecuyer et al. (2009b).

A first idea is to make heuristic adjustments to push the model toward the rare event. For instance, in the first example, if one were to make an "educated guess" for μ without the aid of Figure 1, one could probably choose an IS estimator with a smaller variance than standard simulation, but there is no guarantee. If the system is pushed too hard toward the rare event, then the IS estimator may be much worse than standard simulation. In multivariate state spaces, the variety of trajectories that lead to the rare event increases very fast with the dimension and the variance may explode if the densities of those trajectories are not inflated evenly between them. This is hard to achieve without a good understanding of the system behavior.

A more systematic approach, which dates back to the early days of IS in the 1950's, is to approximate the optimal change of measure via an approximation of $\gamma(\cdot)$ by learning it in an adaptive fashion. For example, the state space of the Markov chain can be discretized into a finite number of regions, and the system can be simulated for a period of time to get an initial estimate for $\gamma(\cdot)$ by assuming a simple form (perhaps a constant function) in each region. Then, a change of measure can be applied based on (7). The system can be simulated again using the new change of measure, and this process can be repeated iteratively. More generally, one can define a parameterized approximation of $\gamma(\cdot)$ with a few parameters and learn a good set of parameter values, for example by stochastic approximation. These approaches require learning and storing an approximation of $\gamma(\cdot)$ over the entire state space, and they hit the curse of dimensionality as soon as the dimension of the state space exceeds a few units. They are also impractical when the state space is already discrete but too large.

Asymptotic approximations of $\gamma(\cdot)$ are sometimes readily available in asymptotic regimes where the rare-event probability converges to 0. Plugging these approximations directly into (7) often works nicely when the rare-event probability is very small. For distributions with an exponentially decaying tail, the resulting IS density $g(x)$ is often the original density $f(x)$ multiplied by an exponential factor $e^{\theta x}/K(\theta)$ for some parameter θ, where $K(\theta)$ is a normalizing factor. This is called *exponential twisting* (Juneja and Shahabuddin, 2006; Asmussen and Glynn, 2007).

Consider for example a sequence of i.i.d. continuous random variables $X_1, X_2, \ldots$ with $\mathrm{E}[X_i] < 0$ and finite moment generating function around zero so that X_i is light tailed. Define a random walk $\{S_i,\ i \geq 0\}$ by $S_0 = 0$ and $S_i = S_{i-1} + X_i$. We want to estimate $\gamma = \Pr\{\sup_{i>0} S_i \geq \ell\}$, the probability that the walk ever exceeds a given constant $\ell > 0$, by simulating it as a Markov chain. This model has several applications. For example, it can be used to obtain the steady-state waiting-time distribution of a $GI/G/1$ queue (e.g., Gross et al., 2008, Section 7.2.3). Large-deviation theory tells us that when γ is small, $\gamma \approx e^{-\theta\ell}$ for some constant θ that depends on the distribution of X_i. Plugging this approximation into (7) (with probabilities replaced by densities) gives an exponentially-twisted density that is typically very effective for small γ. The old density $f(x)$ is multiplied by $e^{\theta x}$ for all x and then renormalized.

Another approach to take when the state space is large is to restrict the change of measure to a specific parametric class. Naturally, the parametric class should be chosen so that it includes good measures to begin with. The problem is to estimate the vector of parameters θ that minimizes the variance (or equivalently the second moment $\mathrm{E}[\hat{\gamma}^2]$) within the parametric class. For a given parameter set θ, estimates of $\mathrm{E}[\hat{\gamma}^2]$ are obtained by sample averages from simulation. Minimization of $\mathrm{E}[\hat{\gamma}^2]$ over θ can be conducted using methods from stochastic optimization. A variant of this is the cross-entropy method which, instead of minimizing the variance, minimizes the cross-entropy distance between the selected change of measure and the zero-variance measure (Rubinstein and Kroese, 2004).

3. Splitting

The idea of splitting is to “split” (or clone) a simulation run into separate runs whenever it gets “near” the rare event of interest (Figure 3). These runs share a common history up to the splitting point, but conditional on that history, they evolve independently of each other. In this way, more computer time is spent on promising runs that are close to the

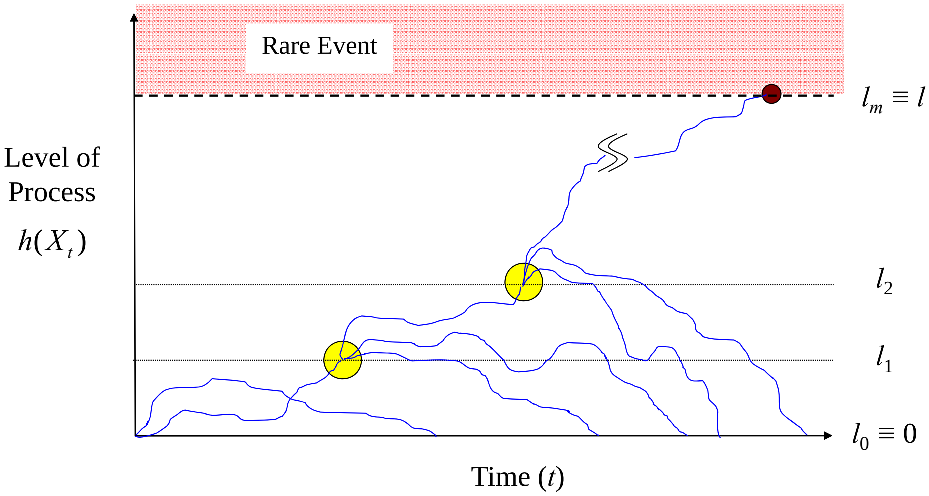

Figure 3: Level splitting.

rare event. To make this more concrete, let $X \equiv \{X_t, t \geq 0\}$ be a Markov process (possibly multidimensional) with state space χ. Let $h : \chi \to \mathbb{R}^+$ be a map of the state space to the "level" of the process; $h(\cdot)$ is sometimes called the *importance function*. Figure 3 shows sample paths of $h(X_t)$. The rare-event set $\mathcal{R}$ is defined as the set of states whose level is at least as large as some constant $l > 0$. That is, $\mathcal{R} \equiv \{x \in \chi : h(x) \geq l\}$. Let T_R be the time the process first enters $\mathcal{R}$, and let T_S be the time the process first returns to level 0. Specifically,

$$T_R \equiv \inf\{t > 0 : h(X_t) \geq l\}, \qquad T_S \equiv \inf\{t > 0 : h(X_t) = 0\},$$

where it is assumed that the process has just left level 0 at time $t = 0$ ($h(X_0) > 0$ and $h(X_{0-}) = 0$). The probability to estimate is $\gamma \equiv \Pr\{T_R < T_S\}$. For example, if $h(X_t)$ denotes the number of customers in a queueing system at time t, then $\{T_R < T_S\}$ is the event that the number in the system reaches some critical threshold (say, a buffer overflow) before the system empties.

To implement level splitting, define a sequence of m levels $0 < l_1 < \cdots < l_m$, where $l_m \equiv l$. Let $T_j \equiv \inf\{t > 0 : h(X_t) \geq l_j\}$ be the time the process first crosses l_j ($j = 1, \ldots, m$). Let $D_j \equiv \{T_j < T_S\}$ be the event that the process crosses l_j before returning to 0 ($j = 1, \ldots, m$). Let $p_j \equiv \Pr\{D_j | D_{j-1}\}$ be the probability that the process crosses l_j (before returning to 0) given that the process has crossed l_{j-1} (before returning to 0) ($j = 2, \ldots, m$). Also, let $p_1 \equiv \Pr\{D_1\}$. Since $D_m \subset D_{m-1} \subset \cdots \subset D_1$,

$$\gamma = \Pr\{D_m\} = \Pr\{D_1\}\Pr\{D_2|D_1\} \cdots \Pr\{D_m|D_{m-1}\} = p_1 p_2 \cdots p_m.$$

The basic idea is to estimate each p_j separately, rather than to estimate γ on its own. There are many variations of level splitting and we describe some of the key ideas here.

Fixed splitting: In this approach, any simulation run that reaches a new level is split into a fixed number R of independent runs. (The splitting factor can also be generalized so that it is level-dependent, R_i.) For example, in Figure 3, the splitting factor is $R = 3$. A run that down-crosses and up-crosses a level is only split at the first up-crossing. Each run that reaches the rare event comes from a sequence of m splits each by a factor R. Thus, each hit to the rare event is weighted by the factor $1/R^m$ to produce an unbiased estimator.

An advantage of fixed splitting is that it can be implemented recursively in a depth-first manner. This means that the computer only needs to store, at most, a single system state per level, corresponding to the simulation history of the current run. To implement fixed splitting, the simulation starts at an initial state and proceeds until it reaches either level 1 or level 0. If level 0 is reached, the replication terminates; if level 1 is reached, the simulation is split into R clones. The first clone is simulated until it reaches either level 2 or level 0. If level 0 is reached, the clone terminates, and the next clone at level 1 is simulated; if level 2 is reached, the simulation is split again, and so forth. This process continues so that all offspring of a clone are simulated to conclusion before proceeding to the next clone.

One drawback with fixed splitting is that it is sensitive to the choice of the splitting factor R. If R is too high, the total number of runs and the amount of work explodes. If R is too low, then few simulations reach the rare event.

Fixed-effort splitting: In this approach, a predetermined total number of runs n_i are made at each level. This is different from fixed splitting where the *total* number of runs at each level is random based on the outcomes of runs at lower levels. The main advantage is that there is no need to know a good splitting factor R in advance, and because the total number of runs at each level is fixed, the overall variance is often lower. The drawback is that the computer must store all of the entrance states to a level to serve as the candidate starting states for simulation of the next level. This can lead to memory problems for models with large state spaces.

One way to implement fixed-effort splitting is the following: Conduct n_1 runs starting from the initial state and simulate each until the system reaches level 1 or returns to level 0. The end states of the runs that reach level 1 are collected into a set A_1. These states become the starting states for the next stage of simulation. (A_1 may include duplicate copies of the same state.) Simulation at stage 2 draws n_2 starting states at random, with replacement, from the set A_1 and simulates each run until the system reaches level 2 or returns to level 0. The method proceeds in an analogous manner at higher stages. After that, an unbiased estimator for the rare-event probability is $\hat{\gamma} \equiv \hat{p}_1\hat{p}_2\cdots\hat{p}_m$, where $\hat{p}_i$ is the number of runs that reach level i divided by n_i.

Fixed-success splitting and fixed probability of success (L'Ecuyer et al., 2009a): In *fixed-success splitting*, at each level, the total number of trajectories that must reach the next level is fixed; independent replications at the current level continue until the fixed number of successes is achieved. According to Amrein and Künsch (2011), this approach is often superior to fixed-splitting and fixed-effort. In *fixed probability of success*, the levels are learned adaptively so that the probability of reaching the next level is approximately the same at all levels.

By far the most difficult issue in splitting, when the state space has more than one dimension, is the choice of the importance function $h(\cdot)$. In one dimension, the choice is typically straightforward. For example, for an $M/M/1$ queue, a natural choice is $h(x) = x$, where x is the number of customers in the system. In fact, all increasing functions in x are equivalent, provided one is free to select the location of the levels. The problem becomes more complicated when the state space is multidimensional. To illustrate, consider a two-stage Markovian tandem queue example, taken from Glasserman et al. (1998). Let (x_1, x_2) represent the numbers of customers at each station, and let γ be the probability of a buffer overflow at the *second* queue (prior to returning to an empty system). An intuitive choice for the level function is $h(x_1, x_2) = x_2$, since $\mathcal{R}$ is defined purely in terms of x_2. This turns out to be a bad choice when the first queue is a bottleneck. The reason is that h does not adequately capture *how* the system gets to the rare event. The system typically gets to $\mathcal{R}$ by building up customers at the first queue and then transferring them to the second queue. Thus, a system in state $(x_1 = \text{large}, x_2 = \text{small})$ is "close" to the rare event even though $h(x_1, x_2) = x_2$ is small indicating that the system is far away from $\mathcal{R}$. Thus, h does not adequately favor sample paths that are likely to reach the rare event. A better choice is $h(x_1, x_2) = [x_2 + \min(0, x_2 + x_1 - l)]/2$, where l is the overflow level of the second queue (L'Ecuyer et al., 2007, 2009a). Intuitively, this function is proportional to the minimal number of steps to the rare event (arrivals to the first queue and transfers to the second queue). In summary, a good choice for the importance function is not necessarily obvious, and a poor choice can increase the variance over crude Monte Carlo. The same type of difficulty occurs when IS is applied to multidimensional systems: the change of measure must favor the sample paths that are representative of how the model reaches the rare event; otherwise the variance can blow up.

One practical issue in splitting is that it can take a long time to simulate from a high level down to level 0. This is inefficient, since a lot of time is spent simulating descending runs that are not likely to rise back up to hit higher levels. The idea of *truncation* is to terminate runs that sink below some threshold under the assumption that they will eventually sink to 0. This reduces computation time but introduces a bias. *Probabilistic truncation* implements this idea in an unbiased way: Consider a run that starts at level k and down-crosses level $k-j$, where $j \geq 1$. The run continues with probability $1/r_{kj}$ and is terminated with probability $1-1/r_{kj}$, where $r_{kj} \geq 1$ are pre-chosen constants. A run that continues has its weights multiplied by r_{kj} to eliminate the bias. The general idea of killing some trajectories with some probability and inflating their weight when they survive is called *Russian roulette* in the Monte Carlo literature. Other variations include periodic truncation and tag-based truncation. A well-known variant of splitting that incorporates truncation is the RESTART method (Villén-Altamirano and Villén-Altamirano, 1994).

To see the kind of improvements that are possible with splitting, consider a problem in which the levels are chosen so that the probabilities p_i are equal, namely $p_i = \gamma^{1/m}$. For example, in the $M/M/1/K$ model, choosing evenly-spaced levels l_i (e.g., $l_i = ki$ for some constant $k > 0$) yields approximately equal values for p_i (at least for large i). Under a fixed-effort implementation with $n_i \equiv n$, it is relatively straightforward to show (e.g., L'Ecuyer et al., 2006) that splitting reduces $\mathrm{Var}[\hat{\gamma}]$ by a factor $m^2\gamma^{1-1/m}$ compared with standard simulation. Taking the derivative with respect to m shows that $m \approx -(\ln\gamma)/2$ achieves the best improvement. For example, if $\gamma = 10^{-9}$, then it is roughly optimal to select $m = 10$ levels yielding a variance reduction of approximately 6 orders of magnitude. However, it is not always easy to choose levels so that the p_i are equal. One adaptive approach is to conduct an initial set of simulations to approximate the appropriate levels, e.g., Garvels, 2000, section 3.3. Also, this analysis implicitly ignores the actual time to simulate each level. In practice, it typically takes longer to simulate runs that start at higher levels. This reduces the variance reduction achieved for a fixed computing budget.

Splitting can also be applied without predefining any levels for the importance function. A splitting or Russian roulette decision can be made at any step of the chain depending on its current weight and current value of the importance function (L'Ecuyer et al., 2006). Finally, particle filters and sequential Monte Carlo methods, popular in computational statistics, are closely related to fixed-effort splitting; see Andrieu et al. (2010) and the references given there.

References

Amrein, M., H. Künsch. 2011. A variant of importance splitting for rare event estimation: Fixed number of successes. *ACM Transactions on Modeling and Computer Simulation* **21** Article 12.

Andrieu, C., A. Doucet, R. Holenstein. 2010. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series B* **72** 1–33.

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

Bucklew, J. A., ed. 2004. *Introduction to Rare Event Simulation*. Springer, New York.

Garvels, M. 2000. The splitting method in rare event simulation. Ph.D. thesis, University of Twente, The Netherlands.

Glasserman, P., P. Heidelberger, P. Shahabuddin, T. Zajic. 1998. A large deviations perspective on the efficiency of multilevel splitting. *IEEE Transactions on Automatic Control* **43** 1666–1679.

Gross, D., J. Shortle, J. Thompson, C. Harris. 2008. *Fundamentals of Queueing Theory*. 4th ed. Wiley, Hoboken, NJ.

Heidelberger, P. 1995. Fast simulation of rare events in queueing and reliability models. *ACM Transactions on Modeling and Computer Simulation* **5** 43–85.

Juneja, S., P. Shahabuddin. 2006. Rare event simulation techniques: An introduction and recent advances. S. G. Henderson, B. L. Nelson, eds., *Handbook in Operations Research and Management Science: Simulation*. Elsevier, 291–350.

L'Ecuyer, P., J. H. Blanchet, B. Tuffin, P. W. Glynn. 2010. Asymptotic robustness of estimators in rare-event simulation. *ACM Transactions on Modeling and Computer Simulation* **20** Article 6.

L'Ecuyer, P., V. Demers, B. Tuffin. 2006. Splitting for rare-event simulation. L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, R. M. Fujimoto, eds., *Proceedings of 2006 Winter Simulation Conference*. IEEE, Piscataway, NJ, 137–148.

L'Ecuyer, P., V. Demers, B. Tuffin. 2007. Rare-events, splitting, and quasi-Monte Carlo. *ACM Transactions on Modeling and Computer Simulation* **17** Article 9.

L'Ecuyer, P., F. LeGland, P. Lezaud, B. Tuffin. 2009a. Splitting techniques. G. Rubino, B. Tuffin, eds., *Rare Event Simulation Using Monte Carlo Methods*. Wiley, 39–62. Chapter 3.

L'Ecuyer, P., M. Mandjes, B. Tuffin. 2009b. Importance sampling and rare event simulation. G. Rubino, B. Tuffin, eds., *Rare Event Simulation Using Monte Carlo Methods*. Wiley, 17–38. Chapter 2.

L'Ecuyer, P., B. Tuffin. 2009. Approximating zero-variance importance sampling in a reliability setting. *Annals of Operations Research* DOI 10.1007/s10479-009-0532-5.

Ross, S. M. 2007. *An Introduction to Probability Models*. 9th ed. Academic Press, New York.

Rubino, G., B. Tuffin, eds. 2009. *Rare Event Simulation using Monte Carlo Methods*. Wiley, Chichester, U.K.

Rubinstein, R., D. P. Kroese. 2004. *A Unified Approach to Combinatorial Optimization, Monte Carlo Simulation, and Machine Learning*. Springer Verlag, Berlin.

Villén-Altamirano, M., J. Villén-Altamirano. 1994. RESTART: A straightforward method for fast simulation of rare events. *Proceedings of the 1994 Winter Simulation Conference*. IEEE, Piscataway, NJ, 282–289.