Variance Reduction Techniques in Monte Carlo Methods

Jack P. C. Kleijnen[1], **Ad A. N. Ridder**[2] **and Reuven Y. Rubinstein**[3]

(1) Tilburg University, Tilburg, The Netherlands, kleijnen@uvt.nl
(2) Vrije University, Amsterdam, The Netherlands, aridder@feweb.vu.nl
(3) Technion, Haifa, Israel, ierrr01@ie.technion.ac.il

INTRODUCTION

Monte Carlo methods are simulation algorithms to estimate a numerical quantity in a statistical model of a real system. These algorithms are executed by computer programs. Variance reduction techniques (VRT) are needed, even though computer speed has been increasing dramatically, ever since the introduction of computers. This increased computer power has stimulated simulation analysts to develop ever more realistic models, so that the net result has not been faster execution of simulation experiments; e.g., some modern simulation models need hours or days for a single 'run' (one replication of one scenario or combination of simulation input values). Moreover there are some simulation models that represent rare events which have extremely small probabilities of occurrence), so even modern computer would take 'for ever' (centuries) to execute a single run—were it not that special VRT can reduce theses excessively long runtimes to practical magnitudes.

Preliminaries

In this contribution the focus is to estimate a quantity

$$\ell = E(H(\mathbf{Y})), \tag{1}$$

where $H(\mathbf{Y})$ is the performance function driven by an input vector $\mathbf{Y}$ with probability density function $f(\mathbf{y})$. To estimate ℓ through simulation, one generates a random sample $\mathbf{Y}_i$ with $i = 1, \ldots, N$ from $f(\mathbf{y})$, computes the sample function $H(\mathbf{Y}_i)$, and the sample-average estimator

$$\hat{\ell}_N = \frac{1}{N} \sum_{i=1}^{N} H(\mathbf{Y}_i).$$

This is called crude Monte Carlo sampling (CMC). The resulting sample-average estimator is an unbiased estimator for ℓ. Furthermore, as N gets large, laws of large numbers may be invoked (assuming simple conditions) to verify that the sample-average estimator stochastically converges to the actual quantity to be estimated. The efficiency of the estimator is captured by its relative error (RE), i.e., the standard error divided by the mean: $\text{RE} = \sqrt{\text{Var}(\hat{\ell}_N)}/E(\hat{\ell}_N)$. Applying the Central Limit Theorem, one easily gets that $z_{1-\alpha/2}\text{RE} < \varepsilon$, where $z_{1-\alpha/2}$ is the $(1-\alpha/2)^{th}$ quantile of the standard normal distribution (typically one takes $\alpha = 0.05$ so $z_{1-\alpha/2} = 1.96$) if and only if

$$P\left(\left|\frac{\hat{\ell}_N - \ell}{\ell}\right| < \varepsilon\right) > 1 - \alpha. \tag{2}$$

When (2) holds, the estimator is said to be $(1-\alpha,\varepsilon)$-efficient.

To illustrate, consider the one-dimensional version of (1):

$$\ell = \int h(y)\, f(y)\, dy.$$

Monte Carlo integration is a good way to estimate the value of the integral when the dimension is much higher than one, but the concept is still the same. Monte Carlo integration has become an important tool in financial engineering for pricing financial products such as options, futures, and swaps (Glasserman, 2003). This Monte Carlo estimate samples $Y_1,\ldots,Y_N$ independently from f and calculates

$$\hat{\ell}_N = \frac{1}{N}\sum_{i=1}^{N} h(Y_i).$$

Then $\hat{\ell}_N$ is an unbiased estimator for ℓ, and the standard error is

$$\sqrt{\mathrm{Var}\left(\hat{\ell}_N\right)} = \sqrt{\frac{1}{N}\mathrm{Var}\left(h(Y)\right)} = \sqrt{\frac{1}{N}E\left(h(Y)-\ell\right)^2} = \sqrt{\frac{1}{N}\int \left(h(y)-\ell\right)^2 f(y)\, dy}.$$

Hence, the relative error (or efficiency) of the estimator is proportional to $1/\sqrt{N}$. This is a poor efficiency in case of high-dimensional problems where the generation of a single output vector is costly and consumes large computing time and memory. VRT improve efficiency if they indeed require smaller sample sizes. To be more specific, consider again the performance measure (1), and assume that besides the CMC-estimator $\hat{\ell}_N$, a VRT results in another unbiased estimator, denoted $\hat{\ell}^*_N$, also based on a sample of N independent and identical observations. The VRT-estimator is said to be statistically more efficient than the CMC-estimator if

$$\mathrm{Var}(\hat{\ell}^*_N) < \mathrm{Var}(\hat{\ell}_N).$$

Then one usually computes the reduction factor for the variance:

$$\frac{\mathrm{Var}(\hat{\ell}_N) - \mathrm{Var}(\hat{\ell}^*_N)}{\mathrm{Var}(\hat{\ell}_N)} \times 100\%.$$

Notice that this factor does not depend on the sample size N. Suppose that the reduction factor is $100r\%$, so $r = 1-(\mathrm{Var}(\hat{\ell}^*)/\mathrm{Var}(\hat{\ell}))$, and suppose that $(1-\alpha,\varepsilon)$-efficiency is desired. The required sample size for the CMC-estimator is N, given by $z_{1-\alpha/2}\mathrm{RE} = \varepsilon$, which holds iff

$$\frac{\ell^2\varepsilon^2}{z^2_{1-\alpha/2}} = \mathrm{Var}(\hat{\ell}_N) = \frac{1}{N}\mathrm{Var}(\hat{\ell}_1) \;\Leftrightarrow\; N = \frac{z^2_{1-\alpha/2}}{\ell^2\varepsilon^2}\mathrm{Var}(\hat{\ell}_1).$$

The same reasoning holds for the VRT-estimator with a required sample size N^*. Consequently, the reduction in sample size becomes

$$\frac{N-N^*}{N} = \frac{\mathrm{Var}(\hat{\ell}_1) - \mathrm{Var}(\hat{\ell}^*_1)}{\mathrm{Var}(\hat{\ell}_1)} = r,$$

which is the same reduction as for the variance.

Generating samples under a VRT consumes generally more computer time (exceptions are antithetic and common random numbers; see next section). Thus to make a fair comparison with CMC, the computing time should be incorporated when assessing efficiency improvement. Therefore, denote the required time to compute $\hat{\ell}_N$ by $\text{TM}(\hat{\ell}_N)$. Then the effort of an estimator may be defined to be the product of its variance and its computing time: $\text{EFFORT} = \text{Var} \times \text{TM}(\hat{\ell}_N)$. Notice that the effort does not depend on the sample size, if the computing time of N samples equals N times the computing time of a single sample. Then the estimator $\hat{\ell}^*_N$ is called more efficient than estimator $\hat{\ell}_N$ if the former requires less effort:

$$\text{EFFORT}(\hat{\ell}^*_N) < \text{EFFORT}(\hat{\ell}_N).$$

Again, a reduction factor for the effort can be defined, and one can analyze the reduction in computer time needed to obtain $(1-\alpha,\varepsilon)$-efficiency.

Estimating the Probability of Rare Events

An important class of statistical problems assesses probabilities of risky or undesirable events. These problems have become an important issue in many fields; examples are found in reliability systems (system failure), risk management (value-at-risk), financial engineering (credit default), insurance (ruin), and telecommunication (packet loss); see Juneja and Shahabuddin (2006); Rubino and Tuffin (2009). These problems can be denoted in the format of this contribution by assuming that a set A contains all the risky or undesirable input vectors $\mathbf{y}$, so that (1) becomes

$$\ell = P(A) = P(\mathbf{Y} \in A) = E(I_A(\mathbf{Y})),$$

where I_A is the indicator function of the set A (and thus in (1) $H = I_A$). The standard error of the Monte Carlo estimator is easily computed as $\sqrt{\ell(1-\ell)/N}$. Hence, the relative error becomes

$$\text{RE} = \frac{\sqrt{\ell(1-\ell)}}{\ell\sqrt{N}} = \frac{\sqrt{(1-\ell)}}{\sqrt{\ell N}}. \tag{3}$$

This equation implies that the sample size is inverse proportional to the target probability ℓ when requiring a prespecified efficiency; for instance, to obtain (95%,10%)-efficiency, the sample size should be $N \geq 385(1-\ell)/\ell$. This leads immediately to the main issue of this contribution; namely $\ell << 1$ so A is called a rare event. To illustrate, suppose $\mathbf{Y} = (Y_1, \ldots, Y_n)$, where Y_j ($j = 1, \ldots, n$) are identically and independently distributed (IID) with finite mean $\mu = E(Y_1)$ and standard deviation $\sigma = \sqrt{\text{Var}(Y_1)}$. Denote their sum by $S(\mathbf{Y}) = Y_1 + \cdots + Y_n$, and let the rare event be $A = \{S(\mathbf{Y}) > n(\mu + \delta)\}$ for a positive δ. A normal approximation results for $n = 500, \delta = 0.5, \sigma = 1$ that $\ell \approx$ 2.5E-29. A (95%,10%)-efficient CMC-estimator would need sample size $N \approx$ 1.5E+31; which is impossible to realize. For example, the practical problem might require the daily simulation of a financial product for a period of two years in which a single normal variate needs to be generated per simulated day. Fast algorithms for normal variate generation on standard PCs require about 20 seconds for E+9 samples. This gives only E+5 vector samples $\mathbf{Y}$ per second. Note that the number of calls of the random number generator (RNG) is at least $N \times n$, which in our numerical example equals 7.5E+33; this number is large, but modern RNGs can meet this requirement (L'Ecuyer, 2006).

In conclusion, the desired level of efficiency of the CMC estimator for rare event problems requires sample sizes that go far beyond available resources. Hence, researchers have looked for ways to reduce the variance of the estimator as much as possible for the same amount of

sampling resources. Traditional VRTs are common random numbers, antithetic variates, control variates, conditioning, stratified sampling and importance sampling (Law, 2007; Rubinstein and Kroese, 2008). Modern VRTs include splitting techniques, and quasi-Mont Carlo sampling (Asmussen and Glynn, 2007; Glasserman, 2003).

ANTITHETIC AND COMMON RANDOM NUMBERS

Consider again the problem of estimating $\ell = E(H(\mathbf{Y}))$ defined in (1). Now let $\mathbf{Y}_1$ and $\mathbf{Y}_2$ be two input samples generated from $f(\mathbf{y})$. Denote $X_i = H(\mathbf{Y}_i)$ with $i = 1,2$. Then $\hat{\ell} = (X_1 + X_2)/2$ is an unbiased estimator of ℓ with variance

$$\mathrm{Var}(\hat{\ell}) = \frac{1}{4}\big(\mathrm{Var}(X_1) + \mathrm{Var}(X_2) + 2\mathrm{Cov}(X_1, X_2)\big).$$

If X_1, X_2 would be independent (as is the case in CMC), then $\mathrm{Var}(\hat{\ell})$ would be $\frac{1}{4}(\mathrm{Var}(X_1) + \mathrm{Var}(X_2))$. Obviously, variance reduction is obtained if $\mathrm{Cov}(X_1, X_2) < 0$. The usual way to make this covariance negative is as follows. Whenever the uniform random number U is used for a particular purpose (for example, the second service time) in generating $\mathbf{Y}_1$, use the antithetic number $1-U$ for the same purpose to generate $\mathbf{Y}_2$. Because U and $1-U$ have correlation coefficient -1, it is to be expected that $\mathrm{Cov}(X_1, X_2) < 0$. This can be formalized by the following technical conditions.

(a). The sample vector $\mathbf{Y} = (Y_1, \ldots, Y_n)$ has components Y_j that are one-dimensional, independent random variables with distribution functions F_j that are generated by the inverse transformation method; i.e., $Y_j = F_j^{-1}(U_j)$, for $j = 1, \ldots, n$.

(b). The performance function H is monotone.

Under these conditions, negative correlation can be proved (Rubinstein and Kroese, 2008). In condition (a) the inverse transformation requirement can be replaced by the assumption that all Y_j-components are Gaussian: when $Y \sim \mathrm{N}(\mu, \sigma^2)$, then $\tilde{Y} = 2\mu - Y \sim \mathrm{N}(\mu, \sigma^2)$, and clearly Y and $\tilde{Y}$ are negatively correlated. This alternative assumption is typically applied in financial engineering for option pricing (Glasserman, 2003).

The method of common random numbers (CRN) is often applied in practice, because simulationists find it natural to compare alternative systems under 'the same circumstances'; for example, they compare different queueing disciplines (such as First-In-First-Out or FIFO, Last-In-First-Out or LIFO, Shortest-Jobs-First or SJF) using the same sampled arrival and service times in the simulation.

To be more specific, let $\mathbf{Y}$ be an input vector for two system performances $E(H_1(\mathbf{Y}))$ and $E(H_2(\mathbf{Y}))$, and the performance quantity of interest is their difference

$$\ell = E(H_1(\mathbf{Y})) - E(H_2(\mathbf{Y})).$$

To estimate ℓ, two choices produce an unbiased estimator:

1. Generate one sequence of IID input vectors $\mathbf{Y}_1, \ldots, \mathbf{Y}_N$, and estimate ℓ by

$$\hat{\ell}_N^{(1)} = \frac{1}{N}\sum_{i=1}^{N} \left(H_1(\mathbf{Y}_i) - H_2(\mathbf{Y}_i)\right).$$

2. Generate two independent IID sequences of input vectors $\mathbf{Y}_1^{(1)},\ldots,\mathbf{Y}_N^{(1)}$, and $\mathbf{Y}_1^{(2)},\ldots,\mathbf{Y}_N^{(2)}$, and estimate ℓ by

$$\hat{\ell}_N^{(2)} = \frac{1}{N}\sum_{i=1}^{N} H_1(\mathbf{Y}_i^{(1)}) - \frac{1}{N}\sum_{i=1}^{N} H_2(\mathbf{Y}_i^{(2)}).$$

The first method is the CRN method, and is intuitively prefered because it reduces variability:

$$\text{Var}(\hat{\ell}_N^{(1)}) < \text{Var}(\hat{\ell}_N^{(2)}).$$

To prove this inequality, denote $X_i = H_i(\mathbf{Y}_i)$. Then $\hat{\ell} = X_1 - X_2$ is an unbiased estimator of ℓ with variance

$$\text{Var}(\hat{\ell}) = \text{Var}(X_1) + \text{Var}(X_2) - 2\text{Cov}(X_1, X_2). \tag{4}$$

If X_1 and X_2 are independent (as is the case in the second method), then (4) becomes $\text{Var}(X_1) + \text{Var}(X_2)$. Hence, variance reduction is obtained if $\text{Cov}(X_1, X_2) > 0$ in (4). This requirement is precisely the opposite of what was needed in antithetic variates. To force the covariance to become positive through CRN, the uniform random number U used for a particular purpose in generating $\mathbf{Y}_1$, is used for the same purpose to generate $\mathbf{Y}_2$. This can be formalized by the technical conditions completely analogous to those for antithetic variates.

CRN is often applied not only because it seems 'fair' but also because CRN is the default in many simulation software systems; e.g., Arena compares different scenarios using the same seed—unless, the programmer explicitly selects different seeds to initialize the various sampling processes (arrival process, service time at work station 1, etc.) for different scenarios. Detailed examples are given in Law (2007), pp. 582-594.

So while the simulation programmers need to invest little extra effort to implement CRN, the comparisons of various scenarios may be expected to be more accurate; i.e., the what-if or sensitivity analysis gives estimators with reduced variances. However, some applications may require estimates of the absolute (instead of the relative) responses; i.e., instead of sensitivity analysis the analysis aims at prediction or interpolation from the observed responses for the scenarios that have already been simulated. In these applications, CRN may give worse predictions; also see Chen, Ankenman, and Nelson (2010).

The analysis of simulation experiments with CRN should go beyond (4), which compares only two scenarios. The simplest extension is to compare a fixed set of (say) k scenarios using (4) combined with the Bonferroni inequality so that the type-I error rate does not exceed (say) α; i.e., in each comparison of two scenarios the value α is replaced by α/m where m denotes the number of comparisons (e.g., if all k scenarios are compared, then $m = k(k-1)/2$). Multiple comparison and ranking techniques are discussed in Chick and Gans (2009).

However, the number of interesting scenarios may be not fixed in advance; e.g., the scenarios differ in one or more quantitative inputs (e.g., arrival speed, number of servers) and the optimal input combination is wanted. In such situations, regression analysis is useful; i.e., the regression model is then a metamodel that enables validation, sensitivity analysis, and optimization of the simulation model; see Kleijnen (2008). The estimated regression coefficients (regression parameters) may have smaller variances if CRN is used—because of arguments based on (4)—except for the intercept (or the 'grand mean' in Analysis of Variance

or ANOVA terminology). Consequently, CRN is not attractive in prediction, but it is in sensitivity analysis and optimization.

A better metamodel for prediction may be a Kriging or Gaussian Process model, assuming the scenarios correspond with combinations of quantitative inputs; e.g., the scenarios represent different traffic rates in a queuing simulation. Kriging implies that the correlation between the responses of different scenarios decreases with the distance between the corresponding input combinations; i.e., the Gaussian process is stationary (Kleijnen, 2008). In random simulation (unlike deterministic simulation, which is popular in engineering) the Kriging metamodel also requires the estimation of the correlations between the 'intrinsic' noises of different scenarios caused by the use of random numbers U; see Chen, Ankenman, and Nelson (2010).

An important issue in the implementation of Antithetics and CRN is synchronization, which is a controlling mechanism to ensure that the same random variables are generated by the same random numbers from the random number generator. As an example, consider comparing a single-server queue $GI/GI/1$ with a two-server system $GI/GI/2$. The two systems have statistically similar arrivals and service times, but the single server works twice as fast. The performance measure is the expected waiting time per customer (which is conjectured to be less in the two-server system). In a simulation study, the two simulation models with CRN should have the same arrival variates, and the same service-time variates. Suppose that $A_1, A_2, \ldots$ are the consecutive interarrival times in a simulation run of the $GI/GI/1$ model, and $S_1, S_2, \ldots$ are their associated service-time requirements. Then, in the corresponding simulation run of the $GI/GI/2$ model, these same values are used for the consecutive interarrival times, and their associated service times; see Kelton, Sadowski, and Sturrock (2007); Law (2007).

Antithetic and common random numbers can be combined. Their optimal combination is the goal of the Schruben-Margolin strategy; i.e., some blocks of scenarios use CRN, whereas other blocks use antithetic variates, etc.; see Song and Chiu (2007).

CONTROL VARIATES

Suppose that $\hat{\ell}$ is an unbiased estimator of ℓ in the estimation problem (1); for example, C is the arrival time in a queueing simulation. A random variable C is called a control variate for $\hat{\ell}$ if it is correlated with $\hat{\ell}$ and its expectation γ is known. The linear control random variable $\hat{\ell}(\alpha)$ is defined as

$$\hat{\ell}(\alpha) = \hat{\ell} - \alpha(C - \gamma),$$

where α is a scalar parameter. It is easy to prove that the variance of $\hat{\ell}(\alpha)$ is minimized by

$$\alpha^* = -\frac{\text{Cov}(\hat{\ell}, C)}{\text{Var}(C)}.$$

The resulting minimal variance is

$$\text{Var}(\hat{\ell}(\alpha^*)) = \left(1 - \rho^2_{\hat{\ell}C}\right) \text{Var}(\hat{\ell}), \tag{5}$$

where $\rho_{\hat{\ell}C}$ denotes the correlation coefficient between $\hat{\ell}$ and C. Since $\text{Cov}(\hat{\ell}, C)$ is unknown, the optimal control coefficient α^* must be estimated from the simulation. Estimating both

$\mathrm{Cov}(\hat{\ell}, C)$ and $\mathrm{Var}(C)$ means that linear regression analysis is applied to estimate α^* . Estimation of α^* implies that the variance reduction becomes smaller than (5) suggests, and that the estimator may become biased. The method can be easily extended to multiple control variables (Rubinstein and Marcus, 1985).

A well-known application of control variates is pricing of Asian options. The payoff of an Asian call option is given by

$$H(\mathbf{Y}) = \max\Big(0,\ \frac{1}{n}\sum_{j=1}^{n} Y_j - K\Big),$$

where $Y_j = S_{jT/n}$, the expiration date T is discretized into n time units, K is the strike price, and S_t is the asset price at time t, which follows a geometric Brownian motion. Let r be the interest rate; then the price of the option becomes

$$\ell = E\left(e^{-rT} H(\mathbf{Y})\right).$$

As control variate may be $C = e^{-rT}\max(0, S_T - K)$ whose expectation is readily available from the Black-Scholes formula. Alternative control variates are S_T, or $\frac{1}{n}\sum_{j=1}^{n} S_{jT/n}$.

CONDITIONING

The method of conditional Monte-Carlo is based on the following basic probability formulas. Let X and Z be two arbitrary random variables, then

$$E(E(X|Z)) = E(X) \quad \text{and} \quad \mathrm{Var}(X) = E(\mathrm{Var}(X|Z)) + \mathrm{Var}(E(X|Z)). \tag{6}$$

Because the last two terms are both nonnegative, variance reduction is obvious:

$$\mathrm{Var}(E(X|Z)) \le \mathrm{Var}(X).$$

The same reasoning holds for the original problem (1), setting $X = H(\mathbf{Y})$. Also Z is allowed to be a vector variable. These formulas are used in a simulation experiment as follows. The vector $\mathbf{Z}$ is simulated, and the conditional expectation $C = E(H(\mathbf{Y})|\mathbf{Z})$ is computed. Repeating this N times gives the conditional Monte-Carlo estimator

$$\hat{\ell}_N^* = \frac{1}{N}\sum_{i=1}^{N} C_i.$$

A typical example is a level-crossing probability of a random number of variables:

$$\ell = P\Big(\sum_{j=1}^{R} Y_j > b\Big),$$

where $Y_1, Y_2, \ldots$ are IID positive random variables, R is a nonnegative integer-valued random variable, independent of the Y_j variables, and b is some specified constant. Such problems are of interest in insurance risk models for assessing aggregate claim distributions (Glasserman, 2003). CMC can be improved by conditioning on the value of R for which level crossing occurs. To be more specific, denote the event of interest by A, so $\ell = E(I_A(\mathbf{Y}))$. Define

$$M = \min\Big(r : \sum_{j=1}^{r} Y_j > b\Big).$$

Assume that the distribution of Y can be easily sampled, and that the distribution of R is known and numerically available (for instance, Poisson). Then it is easy to generate a value of M. Suppose that $M = m$. Then $E(I_A(\mathbf{Y})|M = m) = P(R \ge m)$, which can be easily computed.

STRATIFIED SAMPLING

Recall the original estimation problem $\ell = E(H(\mathbf{Y}))$, and its crude Monte Carlo estimator $\hat{\ell}_N$. Suppose now that there is some finite random variable Z taking values from $\{z_1, \ldots, z_m\}$, say, such that

(i). the probabilities $p_i = P(Z = z_i)$ are known;

(ii). for each $i = 1, \ldots, m$, it is easy to sample from the conditional distribution of $\mathbf{Y}$ given $Z = z_i$.

Because

$$\ell = E(E(H(\mathbf{Y}))) = \sum_{i=1}^{m} p_i E(H(\mathbf{Y})|Z = z_i),$$

the stratified sampling estimator of ℓ may be

$$\hat{\ell}_N^* = \sum_{i=1}^{m} p_i \frac{1}{N_i} \sum_{j=1}^{N_i} H(\mathbf{Y}_{ij}),$$

where N_i IID samples $\mathbf{Y}_{i1}, \ldots, \mathbf{Y}_{iN_i}$ are generated from the conditional distribution of $\mathbf{Y}$ given $Z = z_i$, such that $N_1 + \cdots + N_m = N$. Notice that the estimator is unbiased. To assess its variance, denote the conditional variance of the performance estimator by $\sigma_i^2 = \mathrm{Var}(H(\mathbf{Y})|Z = z_i)$. The variance of the stratified sampling estimator is then given by

$$\mathrm{Var}(\hat{\ell}_N^*) = \sum_{i=1}^{m} \frac{p_i^2 \sigma_i^2}{N_i}.$$

Because of (6)

$$\mathrm{Var}(H(\mathbf{Y})) \geq \mathrm{Var}(H(\mathbf{Y})|Z) = \sum_{i=1}^{m} p_i \sigma_i^2.$$

Selecting proportional strata sample sizes $N_i = p_i N$ gives variance reduction:

$$\mathrm{Var}(\hat{\ell}_N^*) = \sum_{i=1}^{m} \frac{p_i \sigma_i^2}{N} \leq \frac{1}{N} \mathrm{Var}(H(\mathbf{Y})) = \mathrm{Var}(\hat{\ell}_N).$$

It can be shown that the strata sample sizes N_i that minimize this variance are

$$N_i = N \frac{p_i \sigma_i}{\sum_{j=1}^{m} p_j \sigma_j};$$

see Rubinstein and Kroese (2008). A practical problem is that the standard deviations σ_i are usually unknown, so these variances are estimated by pilot runs. Stratified sampling is used in financial engineering to get variance reductions in problems such as value-at-risk, and pricing path-dependent options (Glasserman, 2003).

IMPORTANCE SAMPLING

The idea of importance sampling is explained best in case of estimating the probability of an event A. The underlying sample space is $(\Omega, \mathscr{F})$ for which $A \in \mathscr{F}$, and the probability

measure P on this space is given by the specific simulation model. In a simulation experiment for estimating $P(A)$, the CMC estimator would be $\hat{\ell}_N = \sum_{i=1}^N I_A^{(i)}$, where $I_A^{(1)}, \ldots, I_A^{(N)}$ are IID indicator functions of event A generated under P. On average in only one out of $1/P(A)$ generated samples the event A occurs, and thus for rare events (where $P(A)$ is extremely small) this procedure fails. Suppose that there is an alternative probability measure P^* on the same $(\Omega, \mathscr{F})$ such that (i) A occurs much more often, and (ii) P is absolutely continuous with respect to P^*, meaning

$$\forall F \in \mathscr{F} : P(F) > 0 \;\Rightarrow\; P^*(F) > 0.$$

Then according to the Radon-Nikodym theorem, it holds that there is a measurable function L on Ω such that $\int_F dP = \int_F L\, dP^*$ for all $F \in \mathscr{F}$. The function L is called likelihood ratio and usually written as $L = dP/dP^*$; the alternative probability measure P^* is said to be the importance sampling probability measure, or the change of measure. Thus, by weighting the occurrence I_A of event A with the associated likelihood ratio, simulation under the change of measure yields an unbiased importance sampling estimator

$$\hat{\ell}_N^* = \sum_{i=1}^N L^{(i)} I_A^{(i)}.$$

More importantly, variance reduction is obtained when the change of measure has been chosen properly, as will be explained below. Importance sampling has been applied successfully in a variety of simulation areas, such as stochastic operations research, statistics, Bayesian statistics, econometrics, finance, systems biology; see Rubino and Tuffin (2009). This section will show that the main issue in importance sampling simulation is the question which change of measure to consider. The choice is very much problem dependent, however, and unfortunately, it is difficult to prevent gross misspecification of the change of measure P^*, particularly in multiple dimensions.

Exponential change of measure

As an illustration, consider the problem of estimating the level-crossing probability

$$\ell_n = P(A_n) \quad \text{with} \quad A_n = \{Y_1 + \cdots + Y_n > na\}, \tag{7}$$

where $Y_1, \ldots, Y_n$ are IID random variables with finite mean $\mu = E(Y) < a$ and with a light-tailed PDF $f(y, \mathbf{v})$, in which $\mathbf{v}$ denotes a parameter vector, such as mean and variance of a normal density. It is well-known from Cramér's Theorem that $P(A_n) \to 0$ exponentially fast as $n \to \infty$. Suppose that under the importance sampling probability measure the random variables $Y_1, \ldots, Y_n$ remain IID, but with an exponentially tilted PDF (also called exponentially twisted), with tilting factor t:

$$f_t(y, \mathbf{v}) = \frac{f(y, \mathbf{v}) e^{ty}}{\int f(y, \mathbf{v}) e^{ty}\, dy}.$$

Thus, in the importance sampling simulations the Y_k-samples are generated from $f_t(y, \mathbf{v})$. Because of the IID assumption, the likelihood ratio becomes

$$L(Y_1, \ldots, Y_n) = \prod_{k=1}^n \frac{f(Y_k, \mathbf{v})}{f_t(Y_k, \mathbf{v})} = \exp\Big(n\psi(t) - t \sum_{k=1}^n Y_k\Big), \tag{8}$$

with $\psi(t) = \log \int f(y, \mathbf{v}) e^{ty}\, dy$. Variance reduction is obtained if

$$\begin{aligned} &\mathrm{Var}_t(\hat{\ell}_N^*) \le \mathrm{Var}(\hat{\ell}_N) \quad \Leftrightarrow \quad \mathrm{Var}_t(\hat{\ell}_1^*) \le \mathrm{Var}(\hat{\ell}_1) \\ &\Leftrightarrow \quad E_t[(\hat{\ell}_1^*)^2] \le E[(\hat{\ell}_1)^2] \quad \Leftrightarrow \quad E_t[(I_A L(Y_1, \ldots, Y_n))^2] \le E[(I_A)^2]. \end{aligned}$$

Because of (8), it is easy to show that the variance is minimized for $t = (\psi')^{-1}(a)$. In that case the importance sampling estimator is logarithmically efficient (also called asymptotically optimal; see Rubino and Tuffin (2009; Chapter 4)):

$$\lim_{n\to\infty} \frac{\log E_t[(\hat{\ell}_N^*)^2]}{\log E_t[\hat{\ell}_N^*]} = 2,$$

where the subscript t means that the underlying probability is the change of measure. Asymptotic optimality implies that $\mathrm{RE}(\hat{\ell}_N^*)$ grows subexponentially as $n \to \infty$, whereas for CMC the relative error grows exponentially (see (3)).

The cross-entropy method

A general heuristic for constructing an importance sampling algorithm is to consider only a parameterized family of changes of measures. Consider again problem (1), with PDF $f = f(\mathbf{y}, \mathbf{v})$ where $\mathbf{v}$ is the parameter vector. Thus, let Θ be all feasible parameter vectors for f. For any $\theta \in \Theta$, the change of measure P_θ induces the (single-run) importance sampling estimator

$$\hat{\ell}_\theta^* = H(\mathbf{Y}) \frac{dP}{dP_\theta}(\mathbf{Y}) = H(\mathbf{Y}) \frac{f(\mathbf{Y}, \mathbf{v})}{f(\mathbf{Y}, \theta)}.$$

The optimal change of measure is found by variance minimization. Since the estimators are unbiased, it suffices to minimize the second moment:

$$\min_{\theta\in\Theta} E_\theta \left[\left(H(\mathbf{Y}) \frac{f(\mathbf{Y}, \mathbf{v})}{f(\mathbf{Y}, \theta)} \right)^2 \right].$$

Generally, this problem is hard. A successful approach is based on cross-entropy minimization as explained in Rubinstein and Kroese (2004). First, consider the optimal change of measure, resulting in a zero-variance estimator:

$$dP^{\mathrm{opt}}(\mathbf{Y}) = \frac{H(\mathbf{Y}) dP(\mathbf{Y})}{\ell}. \tag{9}$$

This change of measure is not implementable as it requires knowledge of the unknown quantity ℓ. The cross-entropy method finds P_θ by minimizing the Kullback-Leibler distance (or cross-entropy) within the class of feasible changes of measure:

$$\min_{\theta\in\Theta} \mathscr{D}(dP^{\mathrm{opt}}, dP_\theta),$$

where the cross-entropy is defined by

$$\mathscr{D}(dP^{\mathrm{opt}}, dP_\theta) = E^{\mathrm{opt}} \left[\log \left(\frac{dP^{\mathrm{opt}}}{dP_\theta}(\mathbf{Y}) \right) \right] = E_\nu \left[\frac{dP^{\mathrm{opt}}}{dP}(\mathbf{Y}) \, \log \left(\frac{dP^{\mathrm{opt}}}{dP_\theta}(\mathbf{Y}) \right) \right].$$

Substituting expression (9), and canceling constant terms and factors, the equivalent cross-entropy problem becomes

$$\max_{\theta\in\Theta} E_\nu [H(\mathbf{Y}) \log dP_\theta(\mathbf{Y})].$$

There are several ways to solve this stochastic optimization problem. The original description of the cross-entropy method for such problems proposes to solve the stochastic counterpart iteratively, see Rubinstein and Kroese (2004). This approach has been applied successfully to a variety of estimation and rare-event problems.

State-dependent importance sampling

The importance sampling algorithms described above were based on a static change of measure; i.e, the samples are generated by a fixed alternative statistical law; see (8). In specific problems, such as (7), the static importance sampling algorithm yields an efficient estimator. However, for many problems it is known that efficient estimators require an adaptive or state-dependent importance sampling algorithm (Juneja and Shahabudding, 2006). To illustrate this concept, consider again the problem of estimating the level-crossing probability (7). The Y_k-variables are called jumps of a random walk $(S_k)_{k=0}^n$, defined by $S_0 = 0$, and for $k \geq 1$: $S_k = \sum_{j=1}^k Y_j = S_{k-1} + Y_k$. Under a state-dependent change of measure, the next jump Y_{k+1} might be generated from a PDF $f(y|k+1, S_k)$; i.e., it depends on jump time $k+1$ and current state S_k. Hence, under the change of measure, the process $(S_k)_{k=0}^n$ becomes an inhomogeneous Markov chain. Given a generated sequence $Y_1, \ldots, Y_n$, the associated likelihood ratio is

$$L(Y_1, \ldots, Y_n) = \prod_{k=1}^n \frac{f(Y_k, \mathbf{v})}{f(Y_k|k, S_{k-1})}.$$

The next question is: Which time-state dependent PDFs should be chosen for this kind of change of measure? The criterion could be (i) variance minimization, (ii) cross-entropy minimization, or (iii) efficiency.

(i). A small set of rare-event problems are suited to find so-called zero-variance approximate importance sampling algorithms, notably level-crossing problems with Gaussian jumps, reliability problems, and certain Markov chains problems; see L'Ecuyer et al. (2010).

(ii). A cross-entropy minimization is applied after each state S_k for determining the PDF of the next jump (Ridder and Taimre, 2009). The result is that when the level-crossing at time n can be reached from state S_k just by following the natural drift, no change of measure is applied. Otherwise, the next jump is drawn from an exponentially tilted PDF with tilting factor $t = (\psi')^{-1}((an - S_k)/(n-k))$. This would be the static solution given before when starting at time $k = 0$. This approach gives logarithmic efficiency.

(iii). The method developed by Dupuis and Wang (2007) considers the rare-event problem as an optimal control problem in a differential game. Applying dynamic programming techniques while using large-deviations expressions, the authors develop logarithmically efficient importance sampling algorithms. This approach works also for rare events in Jackson networks (Dupuis, Sezer, and Wang, 2007).

Markov chains

Many practical estimation problems in statistical systems (e.g., reliability, production, inventory, queueing, communications) can be reformulated as a Markov model to estimate a quantity $\ell = P(\mathbf{Y}_T \in \mathscr{F})$. Let $\{\mathbf{Y}_t : t = 0, 1, \ldots\}$ denote a discrete-time Markov chain with a state space $\mathscr{X}$ with transition probabilities $p(\mathbf{x}, \mathbf{y})$; $\mathscr{F} \subset \mathscr{X}$ is a subset of states, and T is a stopping time. A typical example is a system of highly reliable components where the response of interest is the probability of a break down of the system.

Assume that the importance sampling is restricted to alternative probability measures P^* such that the Markov chain property is preserved with transition probabilities $p^*(\mathbf{x}, \mathbf{y})$ satisfying

$$p(\mathbf{x}, \mathbf{y}) > 0 \;\Leftrightarrow\; p^*(\mathbf{x}, \mathbf{y}) > 0.$$

This constraint ensures the absolute continuity condition. Furthermore, assuming that the initial distribution remains unchanged, the likelihood ratio of a simulated path of the chain becomes simply

$$L = \prod_{t=0}^{T-1} \frac{p(\mathbf{Y}_t, \mathbf{Y}_{t+1})}{p^*(\mathbf{Y}_t, \mathbf{Y}_{t+1})}.$$

Thus, it suffices to find the importance-sampling transition-probabilities $p^*(\mathbf{x}, \mathbf{y})$. Considering these probabilities as parameters, the method of cross-entropy is most convenient; Ridder (2010) gives sufficient conditions to guarantee asymptotic optimality. However, many realistic systems are modeled by Markov chains with millions of transitions, which causes several difficulties: the dimensionality of the parameter space, the danger of degeneracy of the estimation, and numerical underflow in the computations. Several approaches are proposed to reduce the parameter space in the cross-entropy method (de Boer and Nicola, 2002; Kaynar and Ridder, 2010).

Another approach to importance sampling in Markov chains approximates the zero-variance probability measure P^{opt}. It is known that this P^{opt} implies transition probabilities of the form

$$p^{\text{opt}}(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}, \mathbf{y}) \frac{\gamma(\mathbf{y})}{\gamma(\mathbf{x})},$$

where $\gamma(\mathbf{x}) = P(\mathbf{Y}_T \in \mathscr{F} | \mathbf{Y}_0 = \mathbf{x})$. As these quantities are unknown (and in fact the subject of interest), these zero-variance transition probabilities cannot be implemented. However, approximations of the $\gamma(\mathbf{x})$ probabilities may be considered (L'Ecuyer et al., 2010). Under certain conditions this approach leads to strong efficiency of the importance sampling estimator.

SPLITTING

The splitting method may handle rare-event probability estimation. Unlike importance sampling, the probability laws remain unchanged, but a drift to the rare event is constructed by splitting (cloning) favorable trajectories, and terminating unfavorable trajectories. This idea may be explained as follows. Consider a discrete-time Markov chain $\{Y_t : t = 0, 1, \ldots\}$ on a state space $\mathscr{X}$. Suppose that the chain has a regeneration state or set $\mathbf{0}$, a set of failure states $\mathscr{F}$, and a starting state $\mathbf{y}_0$. The response of interest is the probability that the chain hits $\mathscr{F}$ before $\mathbf{0}$. More formally, if T denotes the stopping time

$$T = \inf\{t : \mathbf{Y}_t \in \mathbf{0} \cup \mathscr{F}\},$$

then

$$\ell = P(\mathbf{Y}_T \in \mathscr{F}).$$

The initial state $\mathbf{y}_0 \notin \mathbf{0} \cup \mathscr{F}$ may have either some initial distribution, or be fixed and known. The assumption is that ℓ is so small that CMC in impractical. Suppose that the state space is partitioned into sets according to

$$\mathscr{X} \supset \mathscr{X}_1 \supset \mathscr{X}_2 \supset \cdots \supset \mathscr{X}_m = \mathscr{F}, \tag{10}$$

with $\mathbf{0} \in \mathscr{X} \setminus \mathscr{X}_1$. Usually these sets are defined through an importance function $\phi : \mathscr{X} \to \mathbb{R}$, such that for each k, $\mathscr{X}_k = \{\mathbf{y} : \phi(\mathbf{y}) \geq L_k\}$ for certain levels $L_1 \leq L_2 \leq \cdots \leq L_m$, with $\phi(\mathbf{0}) = L_0 < L_1$. Now define stopping times T_k and associated events A_k by

$$T_k = \inf\{t : X(t) \in \mathbf{0} \cup \mathscr{X}_k\}; \quad A_k = \{\mathbf{Y}_{T_k} \in \mathscr{X}_k\}.$$

Because of (10), clearly $A_1 \supset A_2 \supset \cdots \supset A_m = A = \{\mathbf{Y}_T \in \mathscr{F}\}$. Thus the rare-event probability $\ell = P(A)$ can be decomposed as a telescoping product:

$$\ell = P(A_1) \prod_{k=2}^{m} P(A_k | A_{k-1}).$$

To estimate ℓ, one might estimate all conditional probabilities $P(A_k|A_{k-1})$ separately (say) by $\hat{\ell}_k$, which gives the product estimator

$$\hat{\ell}^* = \prod_{k=1}^{m} \hat{\ell}_k, \tag{11}$$

where $\hat{\ell}_1$ estimates $P(A_1)$. The splitting method implements the following algorithm for constructing the $\hat{\ell}_k$ estimators in a way that the product estimator is unbiased. In the initial stage ($k = 0$), run N_0 independent trajectories of the chain starting at the initial state $\mathbf{y}_0$. Each trajectory is run until either it enters $\mathscr{X}_1$ or it returns to $\mathbf{0}$, whatever come first. Let R_1 be the number of "successful" trajectories; i.e., trajectories that reach $\mathscr{X}_1$ before $\mathbf{0}$. Then set $\hat{\ell}_1 = R_1/N_0$. Consider stage $k \geq 1$, and suppose that R_k trajectories have entered set $\mathscr{X}_k$ in entrance states $\mathbf{Y}_1^{(k)}, \ldots, \mathbf{Y}_{R_k}^{(k)}$ (not necessarily distinct). Replicate (clone) these states, until a sample of size N_k has been obtained. From each of these states, run a trajectory of the chain, independently of the others. Each trajectory is run until either it enters $\mathscr{X}_{k+1}$ or it returns to $\mathbf{0}$, whatever come first. Let R_{k+1} be the number of successful trajectories, i.e., trajectories that reach $\mathscr{X}_{k+1}$ before $\mathbf{0}$. Then set $\hat{\ell}_{k+1} = R_{k+1}/N_k$. This procedure is continued until all trajectories have entered either $\mathscr{F}$ or returned to $\mathbf{0}$.

Recently this form of the splitting method has attracted a lot of interest (see the reference list in Rubino and Tuffin (2009; Chapter 3)), both from a theoretical point of view analyzing the efficiency, and from a practical point of view describing several applications. The analysis shows that the product estimator (11) is unbiased. Furthermore, the analysis of the efficiency of the splitting technique depends on the implementation of (a) selecting the levels, (b) the splitting (cloning) of successful trajectories, and (c) the termination of unsuccessful trajectories. Generally, the problem of solving these issues optimally is like choosing an optimal change of measure in importance sampling. In fact, Dean and Dupuis (2008) discusses this relationship when the model satisfies a large deviations principle.

Concerning issue (c), the standard splitting technique terminates a trajectory that returns to the regeneration state $\mathbf{0}$, or—in case of an importance function—when the trajectory falls back to level L_0. This approach, however, may be inefficient for trajectories that start already at a high level L_k. Therefore, there are several adaptations such as truncation (L'Ecuyer, Demers, and Tuffin, 2007), RESTART (Villen-Altamirano, and Villen-Altamirano, 1994), and Russian roulette principle (Melas, 1997).

Concerning issue (b), there are numerous ways to clone a trajectory that has entered the next level, but the two ways implemented mostly are (i) fixed effort, and (ii) fixed splitting. Fixed effort means that the sample sizes N_k are predetermined, and thus each of the R_k entrance states at set $\mathscr{X}_k$ is cloned $c_k = \lfloor N_k/R_k \rfloor$ times. The remaining $N_k \text{mod} R_k$ clones are selected randomly. An alternative is to draw N_k times at random (with replacement) from the R_k available entrance states. Fixed splitting means that the splitting factors c_k are predetermined, and each of the R_k entrance states at set $\mathscr{X}_k$ is cloned c_k times to give sample size $N_k = c_k R_k$.

For a certain class of models, Glasserman et al. (1999) has shown that fixed splitting gives asymptotic optimality (as $\ell \to 0$) when the number of levels $m \approx -\ln(\ell)/2$, with sets $\mathscr{X}_k$ such that $P(A_k|A_{k-1})$ are all equal (namely, roughly equal to e^{-2}) and splitting factors such that $c_k P(A_{k+1}|A_k) = 1$. However, since ℓ and the $P(A_{k+1}|A_k)$ are unknown in practice, this result can only be approximated. Moreover, one should take into account the amount of work or computing time in the analysis; for example, Lagnoux (2006) determines the optimal setting under a budget constraint of the expected total computing time.

Application to counting

Recently, counting problems have attracted the interest of the theoretical computer science and the operations research communities. A standard counting problem is model counting, or #SAT: how many assignments to boolean variables satisfy a given boolean formula consisting of a conjunction of clauses? The related classical decision problem is: does there exist a true assignment of the formula? Because exact counting is impracticable due to the exponential increase in memory and running times, attention shifted to approximate counting—notably by applying randomized algorithms. In this randomized setting, the counting problem is equivalent to rare event simulation: let $\mathscr{X}^*$ be the set of all solutions of the problem, whose number $|\mathscr{X}^*|$ is unknown and the subject of study. Assume that there is a larger set of points $\mathscr{X} \supset \mathscr{X}^*$ with two properties:

1. the number of points $|\mathscr{X}|$ is known;
2. it is easy to generate uniformly points $\mathbf{x} \in \mathscr{X}$.

Because

$$|\mathscr{X}^*| = \frac{|\mathscr{X}^*|}{|\mathscr{X}|}\,|\mathscr{X}|,$$

it suffices to estimate

$$\ell = \frac{|\mathscr{X}^*|}{|\mathscr{X}|} = P(\mathbf{U} \in \mathscr{X}^*),$$

where $\mathbf{U}$ is the uniform random vector on $\mathscr{X}$. Typically ℓ is extremely small, and thus rare event techniques are required. Splitting techniques with Markov chain Monte Carlo (MCMC) simulations have been developed in Botev and Kroese (2008) and Rubinstein (2010) to handle such counting problems.

QUASI MONTE-CARLO

Suppose that the performance function H in (1) is defined on the d-dimensional unit hypercube $[0,1)^d$, and the problem is to compute its expectation with respect to the uniform distribution:

$$\ell = E(H(\mathbf{U})) = \int_{[0,1)^d} H(\mathrm{u})\,d\mathrm{u}.$$

As was shown in the introduction, the variance of the CMC estimator $\hat{\ell}_{Nm}$ using a sample size $N \times m$ equals $\sigma^2/(N \times m)$, where

$$\sigma^2 = \int_{[0,1)^d} H^2(\mathrm{u})\,d\mathrm{u} - \ell^2.$$

Let $P_N = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\} \subset [0,1)^d$ be a deterministic point set that is constructed according to a quasi-Monte Carlo rule with low discrepancy, such as a lattice rule (Korobov), or a digital net (Sobol', Faure, Niederreiter); see Lemieux (2006). The quasi-Monte Carlo approximation of ℓ would be

$$\sum_{j=1}^{N} H(\mathbf{u}_j).$$

This deterministic approach is transformed into Monte Carlo simulation by applying a randomization of the point set. A simple randomization technique is the random shift: generate m IID random vectors $\mathbf{v}_i \in [0,1)^d$, $i = 1, \ldots, m$, and compute the quasi-Monte Carlo approximations

$$\hat{\ell}_i = \sum_{j=1}^{N} H(\mathbf{u}_j + \mathbf{v}_i \bmod 1).$$

Then the randomized quasi-Monte Carlo estimator using sample size $N \times m$ is defined by

$$\hat{\ell}^* = \frac{1}{m} \sum_{i=1}^{m} \hat{\ell}_i.$$

The scrambling technique is based on permuting the digits of the coordinates $u_{j\cdot}$. Other techniques of randomizing quasi-Monte Carlo point sets are less used. The main property is that when the performance function H is sufficiently smooth, these randomized quasi-Monte Carlo methods give considerable variance reduction (Lemieux, 2006).

REFERENCES

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

de Boer, P.T. and Nicola, V.F. (2002). Adaptive state-dependent importance sampling simulation of Markovian queueing networks, *European Transactions on Telecommunications*, 13, 303-315.

Botev, Z.I. and Kroese, D.P. (2008). An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting, *Methodology and Computing in Applied Probability*, 10, 471-505.

Chen, X., Ankenman, B. and Nelson, B.L. (2010). The effects of common random numbers on stochastic kriging metamodels. Working Paper, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL USA.

Chick, S.E. and Gans, N. (2009). Economic analysis of simulation selection problems, *Management Science*, 55, 421-437.

Dean, T. and Dupuis, P. (2008). Splitting for rare event simulation: a large deviations approach to design and analysis, *Stochastic Processes and their Applications*, 119, 562-587.

Dupuis, P., Sezer, D., and Wang, H. (2007). Dynamic importance sampling for queueing networks, *Annals of Applied Probability*, 17, 1306-1346.

Dupuis, P. and Wang, H. (2007). Subsolutions of an Isaacs equation and efficient schemes for importance sampling, *Mathematics of Operations Research*, 32, 723-757.

L'Ecuyer, P. (2006). Uniform random number generator. *Handbook in Operations Research and Management Science Vol 13: Simulation*; S.G. Henderson and B.L. Nelson (Eds.), Chapter 3, pp. 55-81.

L'Ecuyer, P., Blanchet, J.H., Tuffin, B., and Glynn, P.W. (2010). Asymptotic robustness of estimators in rare-event simulation, *ACM Transactions on Modeling and Computer Simulation*, 20, 1, article 6.

L'Ecuyer, P., Demers, V., and Tuffin B. (2007). Rare events, splitting, and quasi-Monte Carlo, *ACM Transactions on Modeling and Computer Simulation*, 17, 2, article 9.

Glasserman, P. (2003). *Monte Carlo Methods in Financial Engineering*, Springer-Verlag, New York.

Glasserman, P., Heidelberger, P., Shahabuddin, P., and Zajic, T. (1999). Multilevel splitting for estimating rare event probabilities, *Operations Research*, 47, 585-600.

Juneja, S. and Shahabuddin, P. (2006). Rare-event simulation techniques: an introduction and recent advances. *Handbook in Operations Research and Management Science Vol 13: Simulation*; S.G. Henderson and B.L. Nelson (Eds.), Elsevier, Amsterdam, Chapter 11, pp. 291-350.

Kaynar, B. and Ridder, A. (2010). The cross-entropy method with patching for rare-event simulation of large Markov chains, *European Journal of Operational Research*, 207, 1380-1397.

Kelton, W.D., Sadowski, R.P., and Sturrock D.T. (2007). *Simulation with Arena*, Fourth edition. Mc Graw-Hill, Boston.

Kleijnen, J.P.C. (2008). *Design and Analysis of Simulation Experiments*, Springer.

Lagnoux, A. (2006). Rare event simulation, *Probability in the Engineering and Informational Sciences*, 20, 45-66.

Law, A.M. (2007). *Simulation Modeling & Analysis*, Fourth edition, McGraw-Hill, Boston.

Lemieux, C. (2006). Quasi-random number techniques. *Handbook in Operations Research and Management Science Vol 13: Simulation*; S.G. Henderson and B.L. Nelson (Eds.), Chapter 12, pp. 351-379.

Melas, V.B. (1997). On the efficiency of the splitting and roulette approach for sensitivity analysis. *Proceedings of the 1997 Winter Simulation Conference*, pp. 269-274.

Ridder, A. (2010). Asymptotic optimality of the cross-entropy method for Markov chain problems, *Procedia Computer Science*, 1, 1565-1572.

Ridder, A. and Taimre, T. (2009). State-dependent importance sampling schemes via minimum cross-entropy. To appear in *Annals of Operations Research*.

Rubino G. and Tuffin, B. (Eds.). (2009). *Rare event simulation using Monte Carlo methods*, Wiley.

Rubinstein, R.Y. (2010). Randomized algorithms with splitting: Why the classic randomized algorithms do not work and how to make them work, *Methodology and Computing in Applied Probability*, 12, 1-50.

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.

Rubinstein, R.Y. and Kroese, D.P. (2008). *Simulation and the Monte Carlo method*, Wiley.

Rubinstein, R.Y. and Marcus, R. (1985). Efficiency of multivariate control variables in Monte Carlo simulation, *Operations Research*, 33, 661-667.

Song, W.T. and Chiu, W. (2007). A five-class variance swapping rule for simulation experiments: a correlated-blocks design, *IIE Transactions*, 39, 713-722.

Villen-Altamirano, M. and Villen-Altamirano, J. (1994). RESTART: a straightforward method for fast simulation of rare events. *Proceedings of the 1994 Winter Simulation Conference*, pp. 282-289.