

## VARIANCE REDUCTION'S GREATEST HITS

Pierre L'Ecuyer

Département d'Informatique et de Recherche Opérationnelle  
Université de Montréal, C.P. 6128, Succ. Centre-Ville  
Montréal (Québec), H3C 3J7, CANADA

### ABSTRACT

Monte Carlo simulation is an incredibly versatile tool for studying complex stochastic systems. By replicating the simulation several times independently, one can in principle estimate performances measures of the system to arbitrary accuracy. Decisions and operating rules can also be optimized via simulation. A major drawback, however, is that the method converges very slowly and often requires an excessive amount of computing time.

Efficiency improvement methods provide ways of either reducing the required computing time for a given target accuracy, or of obtaining an estimator with better accuracy for a given computational budget. Variance reduction is the primary way of improving efficiency. Key ideas for variance reduction were already introduced in the early days of the Monte Carlo method, in the late forties, at Los Alamos. Since then, enormous progress has been made in our understanding of these methods.

This paper is a guided tour of five different methods that can make a huge difference in the accuracy of simulation estimators. They can reduce the variance (or improve the efficiency) by an arbitrary large factor. In some situations, this type of efficiency improvement is essential for the simulation approach to be viable. We discuss *common random numbers* and their synchronization for comparing similar systems, for derivative estimation, and for optimization, *importance sampling* for rare-event simulation, exploiting auxiliary information via *control variates*, smoothing estimators via *conditional Monte Carlo*, and reducing the noise via *generalized antithetic variates or quasi-Monte Carlo*. We give examples where a clever use of these methods can make a huge difference in the required computing time for a given target accuracy.

### INTRODUCTION

Suppose we want to estimate  $\mu = \mathbb{E}[X]$  for some random variable  $X$  having a very complicated distribution that we can only simulate. With the Monte Carlo method, we generate  $n$  independent realizations of  $X$ , take their average  $\bar{X}_n$  and sample variance  $S_n^2$ , and use this information to compute a confidence interval on  $\mu$ . For a given  $n$ , the accuracy of the method depends on  $\sigma^2 = \text{Var}[X]$ . In this paper, we focus on ways of constructing an alternative estimator having the same expectation as  $X$ , but with much smaller variance. Other factors can also be taken into account to measure the quality of an estimator; for example the bias (when it differs from zero) and the computational cost of the estimator (e.g., in terms of CPU time) (Glynn and Whitt 1992). Here, we assume that there is no bias, we disregard the computational cost (or assume that it does change much when we change the estimator), and we focus on the variance. The *efficiency* of an estimator is often defined as the inverse of the product of the mean square error by the expected computing cost. With our simplifications, it becomes equivalent to one over the variance.

A selection of five variance reduction methods are discussed here: common random numbers (CRN), importance sampling (IS), control variates (CV), conditional Monte Carlo (CMC), and randomized quasi-Monte Carlo (RQMC). For each of these methods, the variance can be reduced by an arbitrary large factor, and one can construct examples where the variance can be reduced to zero. Each of them has also been shown to provide a very large efficiency improvement (by a factor in the thousands or millions or more) in at least one realistic application. There are also situations (e.g., in rare-event simulation) where no meaningful estimator could be obtained without these techniques. Additional numerical illustrations will be given in the talk.

There are of course other useful variance reduction (or efficiency improvement) methods than these five. To probe further, we refer the reader to Fishman (1996), Glasserman (2004), Glynn (1994), L'Ecuyer (1994), and L'Ecuyer (2008).

## COMMON RANDOM NUMBERS

Simulation is used frequently to compare the performance of similar systems, often for the purpose of optimization. In that context, we often want to estimate differences of the form  $\mu_2 - \mu_1$  by  $\Delta = X_2 - X_1$ , where  $\mu_k = \mathbb{E}[X_k]$  for  $k = 1, 2$ . This estimator has variance

$$\text{Var}[\Delta] = \text{Var}[X_1] + \text{Var}[X_2] - 2\text{Cov}[X_1, X_2],$$

which is minimized by maximizing the covariance between  $X_1$  and  $X_2$ . If  $X_k$  has (fixed) distribution function  $F_k$  for  $k = 1, 2$ , then a classical result of Fréchet (1951) tells us that the covariance is maximized by taking  $X_k = F_k^{-1}(U)$  where  $U \sim U(0, 1)$  is a *common* random variable, uniformly distributed over the interval  $(0, 1)$ . For complex simulations, generating  $X_k$  by inversion from a single uniform like this is typically impractical, because  $F_k$  is much too complicated. What can be done, however, is to use exactly the same streams of uniforms to drive the simulation for the two systems, try to use them for the same purpose in both systems (this is called *synchronization*), and generate all random variates by inversion. This technique is known as *common random numbers* (CRN). An intuitive interpretation is that by sharing the same “random noise”, the observed differences in performance will be due mainly to real differences between the two systems, and not to the different choice of random numbers. If for each uniform  $U_j$  used in the simulation, both  $X_1$  and  $X_2$  are monotone function of  $U_j$  in the same direction (increasing or decreasing), and they are not independent, then one can prove that  $\text{Cov}[X_1, X_2] > 0$ . The variance is then guaranteed to be reduced. There are situations where we want to compare not only two, but hundreds or even thousands or millions of similar systems. These systems usually differ by the values taken by some decision variables.

An interesting setting is when the performance of interest is a mathematical expectation which is a smooth function of some parameter  $\theta$ , say  $\mu(\theta)$ , and we want to estimate the derivative  $\mu'(\theta) = \partial\mu(\theta)/\partial\theta$  (or the gradient, if  $\theta$  is a vector). This occurs frequently for sensitivity analysis with respect to  $\theta$  if we have doubts about the exact value of  $\theta$ , or within an optimization procedure in which  $\theta$  is a continuous decision variable (L'Ecuyer 1991, L'Ecuyer and Perron 1994, L'Ecuyer and Yin 1998, Fu 2006)). Gradient estimates are also required for the implementation of hedging strategies of a financial portfolio by trading the underlying assets (Glasserman 2004; Chapter 7).

Suppose  $\mu(\theta) = \mathbb{E}[X(\theta, \mathbf{U})]$  for some random variable  $X(\theta) = X(\theta, \mathbf{U})$  that depends on  $\theta$  and on a vector  $\mathbf{U}$  of independent uniform random variables, and which can be generated by simulation. We estimate  $\mu'(\theta)$  by the finite

difference

$$D(\delta) = \frac{X(\theta + \delta, \mathbf{U}_2) - X(\theta, \mathbf{U}_1)}{\delta}$$

for some small  $\delta > 0$ , where  $\mathbf{U}_1$  and  $\mathbf{U}_2$  are two vectors of uniforms.

**Proposition 1** *The following is proved in L'Ecuyer and Perron (1994) and L'Ecuyer (2008):*

(i) *If  $\mathbf{U}_1$  and  $\mathbf{U}_2$  are independent, then*

$$\delta^2 \text{Var}[D(\delta)] \rightarrow 2\text{Var}[X(\theta)]$$

*when  $\delta \rightarrow 0$ . This means that the variance of  $D(\theta)$  increases to infinity at rate  $1/\delta^2$ .*

(ii) *If  $\mathbf{U}_1 = \mathbf{U}_2 = \mathbf{U}$  (CRNs), if  $X(\theta, \mathbf{U})$  is continuous in  $\theta$  and differentiable almost everywhere, and if  $D(\delta)$  is uniformly integrable (uniformly in  $\theta$ ), then  $\text{Var}[D(\delta)]$  remains bounded when  $\delta \rightarrow 0$ .*

(iii) *Suppose now that  $\mathbf{U}_1 = \mathbf{U}_2 = \mathbf{U}$  but that  $X(\theta, \mathbf{U})$  is not continuous in  $\theta$ . If the probability that  $X(\cdot, \mathbf{U})$  is discontinuous in the interval  $(\theta, \theta + \delta)$  converges to 0 as  $O(\delta^\beta)$  when  $\delta \rightarrow 0$ , and if  $X^{2+\varepsilon}(\theta)$  is uniformly integrable for some  $\varepsilon > 0$ , then  $\text{Var}[D(\delta)] = O(\delta^{\beta-2-\varepsilon})$ , for any  $\varepsilon > 0$ , when  $\delta \rightarrow 0$ .*

Note that the ratio of variances (and the efficiency improvement factor) between any two of these cases can become arbitrary large when  $\delta \rightarrow 0$ . For example, if the bound in (ii) and  $\text{Var}[X(\theta)]$  are both equal to 1, and if we take  $\delta = 10^{-4}$ , then  $\text{Var}[D(\delta)]$  is  $2 \times 10^8$  (200 millions) times larger with (i) than with (ii). This means that we would need 200 millions times more simulation runs to reach the same accuracy with independent random numbers (case i) than with CRNs (case ii).

When case (ii) holds, we can sometimes take the stochastic derivative  $X'(\theta) = \lim_{\delta \rightarrow 0} D(\delta)$  as an (unbiased) estimator of  $\mu'(\theta)$  (L'Ecuyer 1990, Glasserman 1991; 2004). This is the best possible situation, provided that this  $X'(\theta)$  is not too difficult to compute. Glasserman (1991, 2004) provides several examples. Sometimes, we may intentionally change the definition of  $X(\theta)$  to make it continuous and benefit from (ii), e.g., by replacing some of its components by conditional expectations (see the section on CMC). For example, if  $X(\theta)$  counts the customer abandonments in a queueing system, we may replace each indicator of abandonment (which takes the value 0 or 1) by its conditional expectation (the probability that this customer abandons) given its waiting time.

Case (iii) shows that using common random numbers can still provide very substantial benefit even we cannot make  $X(\theta)$

continuous. L'Ecuyer (2008) studies further refinements of case (iii) and gives examples. L'Ecuyer and Buist (2006) study a model of a large telephone call center for which  $X(\theta)$  is the number of incoming calls who had to wait less than 20 seconds on a given day, and  $\theta$  is a parameter (e.g., the mean) of the service time distribution. In that case,  $X(\theta)$  can only take integer values, so it cannot be continuous unless it is a constant, in which case  $D(\theta) = 0$ . L'Ecuyer and Buist (2006) prove that with well-synchronized common random numbers, (iii) applies with  $\beta = 1$ . In their empirical results with  $\delta = 10^{-3}$ , the variance is reduced by a factor of about 30,000 compared with independent random numbers.

Their example also illustrates the importance and difficulties of synchronization. There are four types of random numbers in their model; they are used to generate the following random variables: (a) the busyness factor of the day, in the morning (this factor determines the time-varying arrival rate for the day); (b) the interarrival times between successive calls; (c) the call durations, or service times; and (d) the patience times (as soon as the waiting time of a call reaches its patience time, the call abandons). It is important to underline that when simulating the model for two different values of  $\theta$ , the random numbers are not always used in the same order. For example, a given service time may be generated before a given arrival in one case, and after that arrival in the other case. Moreover, some customers may abandon in one case (so there is no need to generate their service time) and not in the other case. And a call may start its service immediately upon arrival in one case (so there is no need to generate its patience time) and not in the other case. To maintain the synchronization, we use four different streams of random numbers, one stream for each type. Modern simulation tools like Arena, Simul8, Witness, or SSJ (for example) provide such multiple streams of random numbers, with multiple substreams for each stream, and tools to rewind back to the start of the current stream or substream, or to jump ahead to the next substream (L'Ecuyer and Buist 2005, L'Ecuyer 2004b). It was also found empirically in L'Ecuyer and Buist (2006) that it was best to generate service times and patience times for *all* customers, even when it was not needed; this gave a better synchronization. For comparison, if patience times were generated only for the customers who did not start their service immediately upon arrival, the variance was about 40 times larger! Also, if a single stream was used for all types of random numbers (with no attempt to maintain synchronization), the variance was about 400 times larger (which is still better than independent random numbers by a factor of 75).

Another important situation where CRNs are crucial is for solving an optimization problem where the objective

function and/or the constraints contain mathematical expectations that cannot be computed exactly for each setting of the decision variables, but can only be estimated by simulation. Performing independent simulations to compare the performance at different settings of interest is typically much too noisy to be effective. One class of methods, called *sample-average optimization*, operates as follows. Suppose (conceptually) that we simulate the model  $n$  times at all possible parameter settings (which is usually an infinite set), using CRNs across the different parameter settings, and that we estimate all expectations by the sample average over these  $n$  runs. After fixing all the (common) random numbers of the  $n$  runs, the sample averages become deterministic functions of the decision variables, which means that we have obtained a *deterministic* optimization problem. This *sample-average problem* can be solved (at least in principle) by our favorite optimization method. To evaluate the sample-average functions (and perhaps their gradients) at any given setting, we must run the simulation at that setting, using well-synchronized CRNs across these settings. Under certain conditions on the model, one can prove that both the optimal value and the optimal solution of the sample-average problem converge to those of the original problem when  $n \rightarrow \infty$ . This convergence can actually be characterized by central-limit theorems and other useful properties (Rubinstein and Shapiro 1993, Ruszczyński and Shapiro 2003).

This type of approach has been used successfully for example to optimize the staffing and scheduling of agents in a large call center with multiple types of agents (Cezik and L'Ecuyer 2006, Avramidis et al. 2007). In that setting there can be thousands of integer-valued decision variables, and the sample-average problem is solved via linear or integer programming. This requires simulating the system at millions of configurations, with CRNs. There is no chance that doing this with independent random numbers would provide a good solution in reasonable time.

## IMPORTANCE SAMPLING

*Importance sampling* (IS) consists in changing the probability laws of the input random variables of a simulation, usually with the aim of concentrating the sampling effort in the most important areas of the sample space. It is the primary way of dealing with *rare event simulation*, i.e., situations where certain types of rare events have an important impact on the performance of interest (Juneja and Shahabuddin 2006). To be specific, suppose we want to estimate  $\mu = \mathbb{E}[h(\mathbf{Y})]$  for some function  $h: \mathbb{R}^d \rightarrow \mathbb{R}$ , where  $\mathbf{Y}$  is a continuous random vector with density  $\pi(\mathbf{y})$  over the  $d$ -dimensional real space  $\mathbb{R}^d$ . If  $g$  is any other density

such that  $g(\mathbf{y}) > 0$  whenever  $h(\mathbf{y})\pi(\mathbf{y}) \neq 0$ , then we can write

$$\begin{aligned}\mu &= \mathbb{E}_\pi[h(\mathbf{Y})] = \int_{\mathbb{R}^d} h(\mathbf{y})\pi(\mathbf{y})d\mathbf{y} \\ &= \int_{\mathbb{R}^d} [h(\mathbf{y})\pi(\mathbf{y})/g(\mathbf{y})]g(\mathbf{y})d\mathbf{y} \\ &= \mathbb{E}_g[h(\mathbf{Y})\pi(\mathbf{Y})/g(\mathbf{Y})]\end{aligned}\quad (1)$$

where  $\mathbb{E}_\pi$  denotes the mathematical expectation under the density  $\pi$  and similarly for  $g$ . This means that if  $\mathbf{Y}$  is generated from density  $g$ , then

$$X_{\text{is}} = h(\mathbf{Y})\pi(\mathbf{Y})/g(\mathbf{Y}) \quad (2)$$

is an unbiased estimator of  $\mu$ . Here, the original estimator  $X = h(\mathbf{Y})$  is multiplied by the *likelihood ratio*  $L(\mathbf{Y}) = \pi(\mathbf{Y})/g(\mathbf{Y})$ . For discrete random variables, just replace the densities by probability mass functions.

If  $h$  cannot take negative values, it is always possible (in theory) to reduce the variance to zero by taking  $g(\mathbf{y})$  proportional to  $h(\mathbf{y})\pi(\mathbf{y})$ . It turns out that the right proportionality constant, so that  $g$  integrates to 1, is  $1/\mu$ . Then the estimator is always equal to  $\mu$ . Implementing this in practice is usually more difficult than estimating  $\mu$  in the first place, but it provides a guideline on how to select a  $g$  that could reduce the variance, by approximating the optimal one. In a nutshell, we want to inflate the density by a factor that is approximately proportional to  $h(\mathbf{y})$ . As a special case, when estimating the probability of a rare event, the optimal  $g$  is the original density conditional on the occurrence of the rare event. This idea generalizes to the simulation of Markov chains in general, with arbitrary state spaces. This is studied in another paper by L'Ecuyer and Tuffin (2007), in these proceedings.

There are several types of real-life situations where the probability that an event of interest (that has an impact on the performance) occurs in the simulation is extremely small, for example smaller than  $10^{-9}$  or even less. Then, with standard Monte Carlo, we would need to perform an excessively large number of runs to obtain any kind of meaningful estimator. In this rare-event simulation context, the main viable approaches are IS, together with another class of methods called *splitting* (Ermakov and Melas 1995, Glasserman et al. 1999, Garvels 2000, L'Ecuyer et al. 2007). Concrete examples where arbitrary small probabilities can be estimated very easily under the appropriate change of density can be found in Glasserman (2004), Juneja and Shahabuddin (2006), L'Ecuyer (2008), elsewhere in these proceedings, and in many other places. The variance reduction factor can be arbitrary large.

For a specific numerical illustration, suppose an insurance firm receives money continuously at rate  $c > 0$ , and receives claims (that they must pay) according to a Poisson process with rate  $\lambda > 0$ . The claim sizes are i.i.d. random variables with density  $f$ . The firm starts with an initial amount  $R(0)$ , and we want to estimate the probability that its amount in hand (the reserve) eventually becomes negative. This is the *ruin probability* and it should be very small. Besides the fact that ruin rarely occurs, another important difficulty with standard Monte Carlo is that we can never be sure that ruin will not occur unless we simulate over an infinite time horizon. This is of course totally unpractical.

It turns out that a good IS scheme in this case is to replace the density  $f(x)$  of  $C_j$  by  $f_\theta(x) = f(x)e^{\theta x}/M_f(\theta)$  and to increase the rate of the Poisson process to  $\lambda_\theta = \lambda + \theta c$ , where  $M_f(\theta) = \int_{-\infty}^{\infty} f(x)e^{\theta x}dx$  and  $\theta$  is the largest solution to the equation  $M_f(\theta) = (\lambda + \theta c)/\lambda$ . Under this change of densities, it turns out that the ruin occurs with probability 1, and the corresponding (unbiased) estimator of  $\mu$  simplifies to

$$X_{\text{is}} = \exp[\theta(R - R(0))],$$

where  $R$  is the (negative) reserve when the ruin occurs.

Suppose, for example, that  $R(0) = 200$ ,  $\lambda = 1$ , the claim sizes are exponential with mean 2, and  $c = 3$ . Then the ruin probability is  $\mu \approx 2.2 \times 10^{-15}$ . Even if we were able to simulate over an infinite horizon at unit cost, estimating such a small probability by standard Monte Carlo would be too costly. The variance per run would be  $\mu(1 - \mu) \approx \mu$ , so the relative error would be  $1/\sqrt{\mu}$ , which increases to infinity when  $\mu \rightarrow 0$ . With the IS scheme described above, the variance is reduced to  $6.3 \times 10^{-31}$ , a reduction by a factor of about  $3 \times 10^{15}$ . If we take  $c = 10$  instead, then we have  $\mu \approx 3.6 \times 10^{-36}$  whereas the variance with IS is  $2.3 \times 10^{-71}$ , a reduction by a factor of about  $1.5 \times 10^{35}$  compared with standard Monte Carlo. Estimating this small probability of  $3.6 \times 10^{-36}$  with 10% relative error would require a sample size of approximately  $n = 2.8 \times 10^{37}$  without IS, and  $n = 183$  with IS.

## CONTROL VARIATES

With *control variates* (CV), we exploit auxiliary information to improve our estimate of the average performance of interest. Suppose  $X$  is the basic estimator and  $\mathbf{C} = (C^{(1)}, \dots, C^{(q)})^t$  is a  $q$ -dimensional vector of control variates, correlated with  $X$ , with known expectation  $\mathbb{E}[\mathbf{C}] = \mathbf{v} = (v^{(1)}, \dots, v^{(q)})^t$  (the  $^t$  is for vector and matrix

transposition). The *controlled* estimator is

$$X_c = X - \beta^t(C - v),$$

where  $\beta = (\beta_1, \dots, \beta_q)^t$  is a vector of constants. We have  $\mathbb{E}[X_c] = \mathbb{E}[X] = \mu$ .

Let  $\Sigma_C = \text{Cov}[C]$ , a matrix whose element  $(i, j)$  is  $\text{Cov}[C^{(i)}, C^{(j)}]$ , and let  $\Sigma_{CX} = (\text{Cov}(X, C^{(1)}), \dots, \text{Cov}(X, C^{(q)}))^t$ . We assume that  $\text{Var}[X] = \sigma^2 < \infty$ ,  $\Sigma_C$  and  $\Sigma_{CX}$  are finite, and  $\Sigma_C$  is positive definite. Then, we have

$$\text{Var}[X_c] = \text{Var}[X] + \beta^t \Sigma_C \beta - 2\beta^t \Sigma_{CX}.$$

This variance is minimized by taking

$$\beta = \beta^* = \Sigma_C^{-1} \Sigma_{CX},$$

and this gives

$$\text{Var}[X_c] = (1 - R_{CX}^2) \text{Var}[X]$$

where

$$R_{CX}^2 = \frac{\Sigma_{CX}^t \Sigma_C^{-1} \Sigma_{CX}}{\text{Var}[X]}.$$

This squared correlation coefficient indicates by what fraction the variance is reduced when we take the optimal  $\beta$ . With a perfect correlation of  $\pm 1$ , the variance is reduced to zero. This means that the variance reduction factor can be arbitrary large. The variance and covariance terms that define  $\beta^*$  are usually estimated from the same simulation runs; the CV scheme is then equivalent to a linear regression. For theoretical analysis and other variants and recommendations, one can consult Nelson (1990), Glynn and Szechtman (2002), Glynn (1994), for example. A related class of methods called *moment matching*, widely used in finance applications (Boyle et al. 1997), turns out to be asymptotically dominated by linear CVs.

For a concrete illustration, we consider the pricing of an Asian call option on a single asset whose price evolves as a geometric Brownian motion (the exponential of a Brownian process)  $\{S(t), t \geq 0\}$ . The process is observed at the fixed times  $0 = t_0 < t_1 < \dots < t_c = T$ . We want to estimate the option value given by  $\mathbb{E}[X]$  where

$$X = e^{-rT} \max \left( 0, \frac{1}{t} \sum_{j=1}^c S(t_j) - K \right),$$

and  $r$  and  $K$  are given positive constants. This can be done by generating  $n$  independent realizations of the same path, and averaging the  $n$  realizations of  $X$ . Interestingly, if we replace the arithmetic average in the definition of  $X$  by a

geometric average, we obtain a random variable

$$C = e^{-rT} \max \left( 0, \prod_{j=1}^c (S(t_j))^{1/c} - K \right),$$

whose expectation  $v = \mathbb{E}[C]$  can be computed exactly by exploiting the famous Black-Scholes formula. Kemna and Vorst (1990) proposed using this  $C$  as a CV. Lemieux and L'Ecuyer (1998) have observed variance reductions by factors of up to a million for some examples, with this technique.

## CONDITIONAL MONTE CARLO

The idea is to *hide information* and replace the basic estimator  $X$  by its expectation conditional on the information that remains available. Suppose  $\mathcal{G}$  is a sigma-field that contains not enough information to compute the realization of  $X$ , but enough to compute its expectation conditional on  $\mathcal{G}$ . The *conditional Monte Carlo* (CMC) estimator is then

$$X_e = \mathbb{E}[X \mid \mathcal{G}].$$

We have  $\mathbb{E}[X_e] = \mathbb{E}[\mathbb{E}[X \mid \mathcal{G}]] = \mathbb{E}[X]$ , so this estimator is unbiased, and an application of the Rao-Blackwell theorem gives

$$\text{Var}[X_e] = \text{Var}[X] - \mathbb{E}[\text{Var}[X \mid \mathcal{G}]] \leq \text{Var}[X].$$

The choice of  $\mathcal{G}$  is a matter of compromise. The less information it contains, the more the variance is reduced. But if it contains too little information,  $X_e$  becomes too difficult to compute.

In many situations,  $X$  is defined as a sum of random variables, say  $C_1, C_2, \dots$ , and it may be much more convenient to apply CMC to each  $C_j$  separately, with a different sigma field  $\mathcal{G}_j$ . This *extended* CMC methodology is no longer guaranteed to reduce the variance, but it can be very convenient and effective in certain cases.

CMC by itself rarely reduces the variance by a huge factor such as those mentioned in the previous examples. Typical gains are quite modest. However, CMC can make a big hit when combined with a finite difference estimator with CRNs. There are indeed many situations where Proposition 1 (ii) does not apply because  $X(\theta)$  is not continuous in  $\theta$ , but where  $X_e(\theta) = \mathbb{E}[X(\theta) \mid \mathcal{G}]$  is continuous in  $\theta$  for an appropriate choice of conditioning  $\mathcal{G}$ . The role of CMC in that context is to boost the efficiency of the derivative estimator, and it can make a huge difference, as can be seen by comparing Proposition 1 (ii) and (iii). Several examples

of this are given by L'Ecuyer and Perron (1994) and Fu and Hu (1997). A frequent situation where this applies is when the estimator  $X$  involves indicator functions.

As an illustration, suppose that in the call center example discussed earlier, we want to estimate the expected number of abandonments in a day, and the derivative of this expectation with respect to a service time parameter  $\theta$ . The basic estimator  $X(\theta)$  here is just the number of abandonments in the day. It is piecewise constant in  $\theta$  for any fixed set of underlying uniform random numbers. Suppose now that we erase from the simulation's sample path all the calls that have abandoned and all traces of these calls. Let  $\mathcal{G}$  be the information that remains. If the calls arrive according to a (non-stationary) Poisson process and if the patience times are all exponential, then it is possible to compute the expected number of abandonments conditional on  $\mathcal{G}$ . The idea is to multiply the arrival rate at time  $t$  by the probability that a call arriving at time  $t$  would have abandoned before starting its service, and integrate this with respect to  $t$  for the entire day. Under mild conditions, this conditional expectation would be continuous with respect to  $\theta$ , so it would provide a much more accurate derivative estimator.

## GENERALIZED ANTITHETIC VARIATES AND RANDOMIZED QUASI-MONTE CARLO

These two classes of techniques are essentially the same idea viewed from slightly different angles. We estimate the expectation of  $X$  by the average of  $k$  realizations of  $X$ , say  $X^{(1)}, \dots, X^{(k)}$ , each having the same distribution as  $X$ , and we want to induce *negative dependence* between these  $k$  copies of  $X$ . The (unbiased) estimator based on *generalized antithetic variates* (GAV) is

$$X_a = \frac{1}{k} \sum_{i=1}^k X^{(i)}$$

with variance

$$\begin{aligned} \text{Var}[X_a] &= \frac{1}{k^2} \sum_{j=1}^k \sum_{\ell=1}^k \text{Cov}[X^{(j)}, X^{(\ell)}] \\ &= \frac{\text{Var}[X]}{k} + \frac{2}{k^2} \sum_{j < \ell} \text{Cov}[X^{(j)}, X^{(\ell)}]. \end{aligned}$$

The variance is reduced in comparison with independent runs if and only if the last sum is negative. The goal is to make it as negative as possible. One way of doing this is via antithetic pairs of random variables (with  $k = 2$ ). Another way is Latin hypercube sampling. But these methods are generally dominated by randomized quasi-Monte Carlo (RQMC).

RQMC is based on the following intuition. Suppose that generating  $X$  requires  $s$  independent uniform random variables over  $(0, 1)$ , so that its expectation can be written as the integral of some function  $f$  over the  $s$ -dimensional unit hypercube. Ordinary Monte Carlo then corresponds to generating independent random points in this hypercube, evaluating  $f$  at each point, and taking the average. The idea of RQMC is to start with a set  $P_k = \{\mathbf{u}_0, \dots, \mathbf{u}_{k-1}\}$  of  $k$  points that cover the unit hypercube  $[0, 1]^s$  in a very uniform way, and randomize  $P_k$  so that after the randomization:

- (a)  $P_k$  retains its high uniformity when taken as a set and
- (b) each individual point of  $P_k$  has the uniform distribution over  $[0, 1]^s$ .

If  $X^{(i)}$  represents the value taken by  $f$  at the  $i$ th randomized point, then we are back to the GAV setting. To estimate the variance and compute confidence intervals, it is usually necessary to repeat the randomization  $m$  times, independently, for some positive integer  $m$ , to obtain  $m$  independent copies of  $X_a$ . The variance is then estimated from the sample variance of these  $m$  independent copies.

Specific ways of constructing highly-uniform point sets  $P_k$  are detailed in Niederreiter (1992), for example. They are usually known under the name of *low-discrepancy* or *quasi-Monte Carlo* point sets. The main classes of constructions are digital nets and lattice rules (Niederreiter 1992, L'Ecuyer and Lemieux 2002, Glasserman 2004). Randomization methods that satisfy (a) and (b) above are examined in L'Ecuyer and Lemieux (2002), for example. Software implementations are available in SSJ (L'Ecuyer 2004b). Under the assumption that the integrand  $f$  is smooth enough, these methods provide estimators whose *worst-case* square error converge as  $O(n^{-2}(\log n)^{2s})$ , which is asymptotically better than the  $O(n^{-1})$  convergence rate for the *mean* square error in the standard Monte Carlo method. One could rightfully argue about the practical significance of this asymptotic result when  $s$  is larger than 10 or so. But there are many important applications, e.g., in computational finance, computer graphics, and computational statistics, where RQMC is really effective (Keller 2006, L'Ecuyer 2004a; 2008).

What happens is that even if  $s$  is large, it is often possible to redefine the function  $f$  in a way that  $f$  is well-approximated by a sum of low-dimensional functions, and it suffices that these low-dimensional functions are integrated with good accuracy by the RQMC method. Concrete examples of this can be found in L'Ecuyer (2004a), Glasserman (2004), for example. In numerical examples of option pricing, L'Ecuyer (2004a) obtains variance reductions by factors of up one million in certain cases.

## ACKNOWLEDGMENTS

This work has been supported by Grant OGP-0110050 and a Canada Research Chair to the author.

## REFERENCES

- Avramidis A.N.; Gendreau M.; L'Ecuyer P.; and Pisacane O., 2007. *Simulation-Based Optimization of Agent Scheduling in Multiskill Call Centers*. In *Proceedings of the 2007 Industrial Simulation Conference*. Eurosis, 255–263.
- Boyle P.; Broadie M.; and Glasserman P., 1997. *Monte Carlo methods for Security Pricing*. *Journal of Economic Dynamics and Control*, 21, 1267–1321.
- Cezik M.T. and L'Ecuyer P., 2006. *Staffing Multiskill Call Centers via Linear Programming and Simulation*. *Management Science*. To appear.
- Ermakov S.M. and Melas V.B., 1995. *Design and Analysis of Simulation Experiments*. Kluwer Academic, Dordrecht, The Netherlands.
- Fishman G.S., 1996. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer Series in Operations Research. Springer-Verlag, New York, NY.
- Fréchet M., 1951. *Sur les Tableaux de Corrélation dont les Marges sont Données*. *Annales de L'Université de Lyon, Sciences Mathématiques et Astronomie*, 14, 53–77.
- Fu M. and Hu J.Q., 1997. *Conditional Monte Carlo*. Kluwer Academic, Boston.
- Fu M.C., 2006. *Gradient Estimation*. In S.G. Henderson and B.L. Nelson (Eds.), *Simulation*, Elsevier, Amsterdam, The Netherlands, Handbooks in Operations Research and Management Science. 575–616. Chapter 19.
- Garvels M.J.J., 2000. *The Splitting Method in Rare Event Simulation*. Ph.D. thesis, Faculty of mathematical Science, University of Twente, The Netherlands.
- Glasserman P., 1991. *Gradient Estimation Via Perturbation Analysis*. Kluwer Academic, Norwell, MA.
- Glasserman P., 2004. *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, no. 4, 585–600.
- Glynn P.W., 1994. *Efficiency Improvement Techniques*. *Annals of Operations Research*, 53, 175–197.
- Glynn P.W. and Szechtman R., 2002. *Some New Perspectives on the Method of Control Variates*. In K.T. Fang; F.J. Hickernell; and H. Niederreiter (Eds.), *Monte Carlo and Quasi-Monte Carlo Methods 2000*. Springer-Verlag, Berlin, 27–49.
- Glynn P.W. and Whitt W., 1992. *The Asymptotic Efficiency of Simulation Estimators*. *Operations Research*, 40, 505–520.
- Juneja S. and Shahabuddin P., 2006. *Rare Event Simulation Techniques: An Introduction and Recent Advances*. In S.G. Henderson and B.L. Nelson (Eds.), *Simulation*, Elsevier, Amsterdam, The Netherlands, Handbooks in Operations Research and Management Science. 291–350. Chapter 11.
- Keller A., 2006. *Myths of Computer Graphics*. In H. Niederreiter and D. Talay (Eds.), *Monte Carlo and Quasi-Monte Carlo Methods 2004*. 217–243.
- Kemna A.G.Z. and Vorst A.C.F., 1990. *A Pricing Method for Options Based on Average Asset Values*. *Journal of Banking and Finance*, 14, 113–129.
- L'Ecuyer P., 1990. *A Unified View of the IPA, SF, and LR Gradient Estimation Techniques*. *Management Science*, 36, no. 11, 1364–1383.
- L'Ecuyer P., 1991. *An Overview of Derivative Estimation*. In B.L. Nelson; W.D. Kelton; and G.M. Clark (Eds.), *Proceedings of the 1991 Winter Simulation Conference*. IEEE Press, Piscataway, NJ, 207–217.
- L'Ecuyer P., 1994. *Efficiency Improvement via Variance Reduction*. In *Proceedings of the 1994 Winter Simulation Conference*. IEEE Press, 122–132.
- L'Ecuyer P., 2004a. *Quasi-Monte Carlo Methods in Finance*. In R.G. Ingalls; M.D. Rossetti; J.S. Smith; and B.A. Peters (Eds.), *Proceedings of the 2004 Winter Simulation Conference*. IEEE Press, Piscataway, New Jersey.
- L'Ecuyer P., 2004b. *SSJ: A Java Library for Stochastic Simulation*. Software user's guide, Available at <http://www.iro.umontreal.ca/~lecuyer>.
- L'Ecuyer P., 2008. *Stochastic Simulation*. Notes for a graduate simulation course, textbook in preparation.
- L'Ecuyer P. and Buist E., 2005. *Simulation in Java with SSJ*. In *Proceedings of the 2005 Winter Simulation Conference*. IEEE Press, 611–620.
- L'Ecuyer P. and Buist E., 2006. *Variance Reduction in the Simulation of Call Centers*. In *Proceedings of the 2006 Winter Simulation Conference*. IEEE Press, 604–613.
- L'Ecuyer P.; Demers V.; and Tuffin B., 2007. *Rare-Events, Splitting, and Quasi-Monte Carlo*. *ACM Transactions on Modeling and Computer Simulation*, 17, no. 2, Article 9.
- L'Ecuyer P. and Lemieux C., 2002. *Recent Advances in Randomized Quasi-Monte Carlo Methods*. In M. Dror; P. L'Ecuyer; and F. Szidarovszky (Eds.), *Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications*, Kluwer Academic, Boston. 419–474.
- L'Ecuyer P. and Perron G., 1994. *On the Convergence Rates of IPA and FDC Derivative Estimators*. *Operations Research*, 42, no. 4, 643–656.
- L'Ecuyer P. and Yin G., 1998. *Budget-Dependent Convergence Rate for Stochastic Approximation*. *SIAM Journal on Optimization*, 8, no. 1, 217–247.

- Lemieux C. and L'Ecuyer P., 1998. *Efficiency Improvement by Lattice Rules for Pricing Asian Options*. In D.J. Medeiros; E.F. Watson; J.S. Carson; and M.S. Manivanan (Eds.), *Proceedings of the 1998 Winter Simulation Conference*. IEEE Press, Piscataway, NJ, 579–586.
- Nelson B.L., 1990. *Control-Variate Remedies*. *Operations Research*, 38, 974–992.
- Niederreiter H., 1992. *Random Number Generation and Quasi-Monte Carlo Methods*, *SIAM CBMS-NSF Regional Conference Series in Applied Mathematics*, vol. 63. SIAM, Philadelphia, PA.
- Rubinstein R.Y. and Shapiro A., 1993. *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*. Wiley, New York.
- Ruszczynski A. and Shapiro A. (Eds.), 2003. *Stochastic Programming*. Handbooks in Operations Research and Management Science. Elsevier, Amsterdam, The Netherlands.

## AUTHOR BIOGRAPHY

**PIERRE L'ECUYER** is Professor in the Département d'Informatique et de Recherche Opérationnelle, at the Université de Montréal, Canada, and a member of the GERAD and CIRRELT research centers. He holds the Canada Research Chair in Stochastic Simulation and Optimization. His main research interests are random number generation, quasi-Monte Carlo methods, efficiency improvement via variance reduction, sensitivity analysis and optimization of discrete-event stochastic systems, and discrete-event simulation in general. He is currently Associate/Area Editor for ACM Transactions on Modeling and Computer Simulation, ACM Transactions on Mathematical Software, Statistical Computing, International Transactions in Operational Research, The Open Applied Mathematics Journal, and Cryptography and Communications. He obtained the E. W. R. Steacie fellowship in 1995-97, a Killam fellowship in 2001-03, and he was elected INFORMS Fellow in 2006. His recent research articles are available on-line from his web page: <http://www.iro.umontreal.ca/~lecuyer>.