Lattice Rules for the Simulation of Ruin Problems

Christiane Lemieux and Pierre L'Ecuyer

Département d'Informatique et de Recherche Opérationnelle
Université de Montréal, C.P. 6128, Succ. Centre-Ville
Montréal, H3C 3J7, CANADA
(514) 343-2143, (514) 343-5834 (fax)
lemieux@iro.umontreal.ca, lecuyer@iro.umontreal.ca

ABSTRACT

Lattice rules are a quasi-Monte Carlo method, an alternative to the Monte Carlo method for multidimensional numerical integration. The measures of performance that we estimate by simulation can be seen as the values of integrals in large dimensions. Therefore, instead of the usual Monte Carlo method, one can use lattice rules in simulation. In this paper, we compare these rules to the Monte Carlo method to simulate the surplus process of an insurance company, in order to estimate the ruin probability. We give results using different simulation techniques and also try different parameters, in order to see how this affects the relative performance of these two methods.

1. Introduction

Using straightforward simulation to estimate $\mathrm{E}(g(\boldsymbol{Y}))$, where $\boldsymbol{Y}$ is a vector of random variables describing the process we want to simulate, can be thought of as using the Monte Carlo (MC) method to estimate a multidimensional integral

$$If = \int_{[0,1)^s} f(\boldsymbol{x}) d\boldsymbol{x}, \tag{1}$$

where $\boldsymbol{x}$ is the vector of uniform random numbers driving the simulation and $f : [0,1)^s \to \mathbb{R}$ is such that $g(\boldsymbol{Y}(\boldsymbol{x})) = f(\boldsymbol{x})$, for all $\boldsymbol{x} \in [0,1)^s$.

From this point of view, one may be interested in using a quasi-Monte Carlo (QMC) method instead, to estimate If [14]. That is, in the estimator $Qf = \sum_{i=1}^{N} f(\boldsymbol{x}_i)/N$, instead of using a set of N i.i.d. random vectors uniformly distributed on $[0,1)^s$, take a low-discrepancy point set $P = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, with points more regularly distributed in $[0,1)^s$ than uniform random points.

In this paper, we compare MC and a QMC method to estimate the ruin probability for an insurance company, a well-known problem in risk theory. We investigate how the parameters of the model and the simulation techniques [9, 6] used in combination with each of these two methods affect their relative performance, which is measured by the estimated variance associated with each of these methods.

MC and QMC methods are usually the most appropriate to estimate (1). Using the central limit theorem, one can show that the probabilistic error associated with the MC method is in the order of $1/\sqrt{N}$. For QMC methods, by restricting f to be sufficiently regular, it can be shown that the deterministic error has an upper bound in the order of $N^{-1}(\log N)^s$. The usual way to show this is to choose a class of functions E for which an inequality of the form

$$|If - Qf| \leq D(P)V(f) \text{ for all } f \in E \tag{2}$$

can be derived [8, 14]. In (2), the discrepancy $D(P)$ measures the regularity of the point set P used to construct Qf and $V(f)$ measures the variability of the function f. A low-discrepancy point set P is in fact a set for which (roughly) $D(P) = O(N^{-1}(\log N)^s)$. The most useful property of this kind of upper bound is not to give an idea of the integration error, because the bound is usually too loose, but rather to derive criteria for choosing good sets P, by providing a quantity to minimize, namely $D(P)$.

So for fixed s, QMC is asymptotically better than MC. However, for fixed N, as s increases, MC may eventually get better than QMC. But numerical experiments that have been carried out for option valuation problems in finance seem to indicate that the advantage of QMC holds for large dimensions as well [15, 11, 4].

This paper is organized as follows: In the next section, we explain briefly how to construct lattice rules, which is the QMC method that we use here. In section 3, we present the ruin probability evaluation problem and its associated model, as well as how simulation can be used to solve it. Numerical results are given in the fourth section and a conclusion ends the paper in section 5.

2. Lattice rules

Lattice rules and (t, m, s)-nets are the two best-known QMC methods. They both succeed in constructing low-discrepancy point sets, but using different approaches.

We only describe one particular case of rules, the rank-1 lattices: They are obtained by choosing a generating vector $\boldsymbol{h} \in \mathbb{Z}^s$ and then defining $P = \{\{i\boldsymbol{h}/N\}, i = 0, \ldots, N-1\}$, where $\{\boldsymbol{x}\}$ means that we take each coordinate of $\boldsymbol{x}$ modulo 1. P is then the intersection of a certain lattice L with $[0,1)^s$. We just need one vector to generate all the points in P and this is the reason why these rules are called rank-1 (the rank of a lattice rule is the minimal number of vectors needed to generate all of its points).

When the generating vector is of the form $\boldsymbol{h} = (1, a, \ldots, a^{s-1})$, P is equal to the set of all successive overlapping s-tuples produced by a linear congruential generator (LCG) with modulo N and multiplier a, from all possible initial seeds [7, 13, 6, 14]. Thus, if the LCG has maximal period, the points in P may be generated quickly by just running the LCG over its full period, taking the successive s-tuples, and adding the origin.

How should one choose $\boldsymbol{h}$? As mentioned in the introduction, the usual way to do this is to use a particular definition of $D(P)$ and then try to find the set P that minimizes $D(P)$ among a certain category. For the search to be done in reasonable time, one should choose a discrepancy that can be calculated quickly for a given P. For example, if $E = \{f : |\hat{f}(\boldsymbol{h})| \leq c\|\boldsymbol{h}\|_\pi^{-\alpha},\ \forall \boldsymbol{h} \in \mathbb{Z}^s\}$, where α is an even integer and $\|\boldsymbol{h}\|_\pi = \prod_{j=1}^{s} \max(1, |h_j|)$, then $D(P)$ in (2) can be taken as $\sum_{\mathbf{h} \in L^\perp \setminus \{\mathbf{0}\}} \|\boldsymbol{h}\|_\pi$, where $L^\perp$ is the dual lattice of L [16, 8]. This particular case of $D(P)$ is denoted by P_α and has been often used as a figure of merit to choose lattice rules [16]. The properties of the lattice allow the derivation of a convenient formula for P_α, that can be calculated in $O(Ns)$ time.

One can also use the connection with LCGs to derive a selection criterion for P. For example, as suggested in [7], one might think of using the *spectral test*, a well-known selection criterion for LCGs, that can be calculated efficiently. In practice, the spectral test value can be computed more quickly than P_α and one can also construct a more global figure of merit that takes account of the quality of the lattice in several dimensions simultaneously, using this test. For these reasons, we adopt this approach to choose the rank-1 rules used in the numerical experiments of this paper. The reference giving tables of such “best LCGs” is [10]. For more on selection criteria, see [8, 7] for example.

The bound (2) is of little practical use to give an idea of the integration error. Therefore, one needs another way to estimate the error. For QMC methods, the usual way is to randomize the point set P, without destroying its regularity properties. By making i.i.d. randomizations of P, one can estimate the variance of Qf, as we normally do with the MC method. One way to do this is to generate m i.i.d. vectors $\Delta_1, \ldots, \Delta_m$, uniformly and randomly on $[0, 1)^s$, and use them to shift P, modulo 1. This is the *Cranley-Patterson randomization technique* [5], which we use in this paper.

3. The Model for the Surplus Process

An insurance company receives claims at random times and of random size. More precisely, let $S(t) = \sum_{k=1}^{N(t)} Y_k$ be the aggregate claims process, where $N(t)$ is the number of claims during the time interval $(0, t]$ and Y_k is the amount of the k^{th} claim. In exchange for the payment of the claims, the company receives premium at a certain rate $c(\cdot)$, that may depend on the surplus process $U(\cdot)$, where $U(t)$ denotes the amount of money available at time t. Therefore, one has $dU(t) = c(U(t))dt - dS(t)$. We are looking for the probability of ruin given an initial surplus u,

$$\psi(u) = P(U(t) < 0 \text{ for some } t \geq 0 \mid U(0) = u).$$

In this paper, we assume that $N(\cdot)$ is a Poisson process with rate λ and that the claims size $Y_k,\ k = 1, 2, \ldots$ are i.i.d. random variables of average β. Also, the premium rate function is assumed to be constant and the surplus process earns interest at a rate δ. These two assumptions amount to set $c(x) = c + \delta x$, where c and δ are two non-negative constants.

Typically, $\psi(u)$ is very small: The ruin is a *rare-event*. Therefore, using the straightforward approach of simulating the claims arrival process until ruin occurs might be very inefficient. This is a situation where *importance sampling* (IS) should be used: [1] explains how the probability laws should be modified in the case where $\delta = 0$. In the particular case of exponential claims size, the optimal change of measure is known. When $\delta > 0$, the problem is much harder, because we must locally change the distributional parameters during the simulation: see [3] for more details on this.

To estimate $\psi(u)$, one may also use the duality between $U(\cdot)$ and the *storage process* $X(\cdot)$ defined by

$$dX(t) = \begin{cases} -c(X(t))dt + dS(t) & \text{if } X(t) > 0 \\ dS(t) & \text{if } X(t) = 0 \end{cases}$$

and an initial value $X(0)$ which is set equal to u. We can then write [2]:

$$\psi(u) = 1 - \lim_{n \to \infty} \frac{D_n(u)}{\sum_{i=1}^{n} T_i},$$

where $D_n(u)$ is the total time spent by $X(\cdot)$ under level u before its n^{th} jump and T_i is the time between the $(i-1)^{th}$ jump and the i^{th} jump of $X(\cdot)$.

So, a second method to estimate $\psi(u)$ is to simulate l claims and use

$$\hat{\psi}(u) = 1 - \frac{D_l(u)}{\sum_{i=1}^{l} T_i}. \tag{3}$$

This approach is used in [12]. However, the properties of $X(\cdot)$ allow one to use regenerative simulation to estimate $\psi(u)$, as done in [17] for example. The regenerative times are the epochs when $X(t)$ hits u. With this third method, the i.i.d. replications are cycles of $X(\cdot)$ instead of l-claims simulations as for the second method, where l must be taken large if we want to remove most of the slowly decreasing bias in (3).

For each of these three different approaches, MC or QMC may be used. With QMC, we make m groups of N simulations each, where N is the number of points in the node set P and m is the number of random shifts used to randomize P. This can be compared with mN MC simulations.

If all random variables are generated by inversion, we need two $U(0,1)$ variates for each claim in the simulation; one for its size and one for the time elapsed since the last claim. Hence, the dimension of the problem is equal to twice the number of claims in a replication. With IS and the regenerative approach, the number of claims is random and increases (stochastically) with u, within a given replication. With the estimator (3), it is equal to $2l$, but the *effective dimension* [15, 8] is in some sense smaller than that, since the interactions that contribute the most to the variance are the ones among variables related to the same cycle. Notice that when the dimension is random, it is important to use a low-discrepancy point set that is uniformly good in several dimensions. The rank-1 lattices that we use have this property, because of the selection criterion of [10] and the particular form of the generating vector $\boldsymbol{h}$.

4. Numerical Results

We give some partial results of the numerical experiments conducted to see how lattice rules (LR) improves upon MC for this problem, depending on the approach used for the simulation and the parameters of the model.

We use an exponential claim size distribution in all of the following examples. This is not realistic, but in this case $\psi(u)$ has a closed-form expression, which permits one to compare simulation with theoretical results. DUAL stands for the method in which we simulate l claims of the dual process (see (3) above) and REG is for the regenerative simulation of X.

In all tables, $\hat{\mu}$ and $\hat{\sigma}$ are the estimates of $\psi(u)$ and $(\mathrm{Var}\ (\hat{\mu}))^{1/2}$, respectively. The numbers in parentheses below $\hat{\sigma}$ give the ratio of $\hat{\sigma}^2$ for the method considered to that of the MC estimator. We give in Table 1 the cpu time in seconds needed to compute each $\hat{\mu}$ and "eff." is an estimate of the *efficiency* $\mathrm{Eff}(\hat{\mu}) = (C(\hat{\mu})\mathrm{Var}\ (\hat{\mu}))^{-1}$, where $C(\hat{\mu})$ is the expected CPU time needed to compute $\hat{\mu}$.

In Table 1, LR reduces the variance in an important way: By a factor of 52 when $u = 0$ and by a factor of 5 when $u = 10$ (this factor is the inverse of the ratio given in parentheses). In terms of efficiency, LR has the additional advantage of requiring less CPU time than MC. When $u = 0$, LR improves the efficiency by a factor of 70. The improvement for the variance decreases with u, probably because the effective dimension is larger in this case. Notice that the dimension of the problem is equal to 8000 in this example. Therefore, it would probably be ineffective to use a (t, m, s)-net for this type of estimator, because (t, m, s)-nets are typically constructed so that the equidistribution properties of subsets of variables with indices in $u \subseteq S = \{1, \ldots, s\}$ deteriorate as u contains higher indices (near s), meaning that the last claims in the simulation would be generated by badly distributed points. The lattice rules we use do not have this problem. Their projections on a lower-dimensional space have regularity properties that depend only on the relative difference between indices in u.

Other numerical experiments indicate that if we use a larger l for the simulation, the variance of the LR estimator decreases at about the same rate as the MC estimator's variance. But when N increases, the variance of LR decreases a bit faster than MC.

Table 2 gives results with IS. We added two columns that give the theoretical μ and σ for the MC method, without or with IS. The ratios under $\hat{\sigma}$ are now taken relatively to the variance σ^2 of MC without IS. We set the interest rate δ to 0, so that the change of measure is easily done. The quantity r is the average number of claims per simulation. It gives an idea of the dimension of the problem formulated this way. For this simplified example (exponential claims size), IS reduces the variance by important factors. This makes it harder for LR to be able to bring an additional improvement. Still, it decreases the variance a little bit (by a factor of 1.8 when $\psi(u) = 0.5$) and, as expected, it is more useful when u is smaller. This is interesting, because IS has the opposite property: Its improvement upon standard MC (without IS) increases as $\psi(u)$ decreases, or equivalently, as u increases. Therefore, LR and IS are two complementary methods that seem to combine in an advantageous way.

Table 1: DUAL method, $\lambda = 1.0$, $\beta = 1.0$, $\delta = 0.05$, $N = 16381$, $m = 100$, $l = 4000$.

u		MC	LR
0	$\hat{\mu}$	0.841	0.841
	$\hat{\sigma}$	9.96e-6	1.38e-6
		(1.000)	(0.019)
	cpu	45214	33393
	eff.	2.23e 5	1.57e 7
10	$\hat{\mu}$	3.97e-2	3.98e-2
	$\hat{\sigma}$	8.40e-6	3.85e-6
		(1.000)	(0.210)
	cpu	45204	33050
	eff.	3.14e 5	2.05e 6

Table 2: IS method, $\lambda = 0.85$, $\beta = 1.0$, $\delta = 0$, $N = 16381$, $m = 100$.

u		MC (IS)	LR (IS)	Theo. MC no IS	Theo. MC with IS
3.54	$\hat{\mu}$	0.500	0.500	0.5	0.5
	$\hat{\sigma}$	5.93e-5	4.39e-5	3.91e-4	5.92e-5
		(2.30e-2)	(1.27e-2)	(1.000)	(2.30e-2)
	r	26.7	26.7		
29.6	$\hat{\mu}$	1.00e-2	1.00e-2	0.01	0.01
	$\hat{\sigma}$	1.19e-6	1.07e-6	7.77e-5	1.19e-6
		(2.34e-4)	(1.89e-4)	(1.000)	(2.34e-4)
	r	174.3	174.4		

Finally, we give results using regenerative simulation in Table 3. Again, LR reduces the variance compared to MC (by a factor of about 2.5). When $\delta = 0$, IS is known to give more precise estimators than the regenerative method [1]. However, when $\delta > 0$, IS is more difficult to apply, whereas the regenerative method works fine for all δ. Also, the bias

Table 3: REG method, $\lambda = 1.0$, $\beta = 1.0$, $\delta = 0.05$, $N = 16381$, $m = 100$.

u		MC	LR
0	$\hat{\mu}$	0.841	0.841
	$\hat{\sigma}$	2.51e-4	1.57e-4
		(1.000)	(0.392)
	r	7.3	7.3
10	$\hat{\mu}$	3.91e-2	3.92e-2
	$\hat{\sigma}$	8.18e-5	5.26e-5
		(1.000)	(0.414)
	r	42.7	42.7

with REG is not as persistent as with DUAL. Hence, when IS cannot be applied easily, the regenerative method seems to be the most appropriate.

One must be careful when comparing the estimator's variance across tables. Once the parameters of the model are fixed, the three methods do not use the same number of claims per replication. To make fair comparisons, one possibility is to multiply the variance by the number of claims per replication. To illustrate this, look at $\hat{\sigma}$ when $u = 10$ in Tables 1 and 3: Though DUAL seems to give a better estimator than REG, the adjusted empirical variance is approximately the same in both cases (2.82e-7 and 2.86e-7, respectively).

5. Conclusion

We used lattice rules to simulate the surplus process of an insurance company. This quasi-Monte Carlo method is fairly easy to apply and does not increase the computational cost. We looked at three different ways of conducting the simulation in our numerical experiments and in all cases, lattice rules reduce the variance compared to MC.

Two of these three methods have the property that the dimension of the associated function is random. Though this does not seem to prevent LR from improving upon MC, it would be interesting to further investigate how this affects the quality of the estimators and, if necessary, how could we reformulate the problem to flatten this difficulty. We think that the good performance of LR with these examples in large and/or random dimension is in part explained by the fact that the rules are chosen to be uniformly good over many dimensions.

References

[1] S. Asmussen. Conjugate processes and the simulation of ruin problems. *Stochastic Processes and their Applications*, 20:213–229, 1985.

[2] S. Asmussen. Ruin probabilities expressed in terms of storage process. *Advances in Applied Probability*, 20:913–916, 1988.

[3] S. Asmussen and H. M. Nielsen. Ruin probabilities via local adjustment coefficients. *Journal of Applied Probability*, 33:736–755, 1995.

[4] R. E. Caflish, W. Morokoff, and A. Owen. Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension. Manuscript, 1997.

[5] R. Cranley and T. N. L. Patterson. Randomization of number theoretic methods for multiple integration. *SIAM Journal on Numerical Analysis*, 13(6):904–914, 1976.

[6] G. S. Fishman. *Monte Carlo: Concepts, Algorithms, and Applications.* Springer Series in Operations Research. Springer-Verlag, New York, 1996.

[7] P. Hellekalek, K. Entacher, and P. L'Ecuyer. Quasi-Monte Carlo node sets from linear congruential generators. In *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, 1998. Submitted.

[8] F. J. Hickernell. Lattice rules: How well do they measure up? In P. Hellekalek and G. Larcher, editors, *Random and Quasi-Random Point Sets*, volume 138 of *Lecture Notes in Statistics*. Springer, New York, 1998. To appear.

[9] A. M. Law and W. D. Kelton. *Simulation Modeling and Analysis.* McGraw-Hill, New York, second edition, 1991.

[10] P. L'Ecuyer. A table of linear congruential generators of different sizes and good lattice structure. *Mathematics of Computation*, 1999. To appear.

[11] C. Lemieux and P. L'Ecuyer. Efficiency improvement by lattice rules for pricing asian options. In *Proceedings of the 1998 Winter Simulation Conference*, 1998. To appear.

[12] F. Michaud. Estimating the probability of ruin for variable premiums by simulation. *Astin Bulletin*, 26:93–105, 1996.

[13] H. Niederreiter. Multidimensional numerical integration using pseudorandom numbers. *Mathematical Programming Study*, 27:17–38, 1986.

[14] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*, volume 63 of *SIAM CBMS-NSF Regional Conference Series in Applied Mathematics.* SIAM, Philadelphia, 1992.

[15] S. Paskov and J. Traub. Faster valuation of financial derivatives. *Journal of Portfolio Management*, pages 113–120, Fall 1995.

[16] I. H. Sloan and S. Joe. *Lattice Methods for Multiple Integration.* Clarendon Press, Oxford, 1994.

[17] F.J. Vázquez-Abad. Pathwise derivative estimation of ruin probabilities. Manuscript, 1997.