Notes on the application of randomized quasi-Monte Carlo methods to financial engineering problems

Hozumi Morohosi[1] Masanori Fushimi[2]

[1] National Graduate Institute for Policy Studies
7-22-1 Roppongi, Minato-ku, Tokyo 106-8677, Japan
Email: morohosi@grips.ac.jp
[2] Department of Mathematical Sciences, Nanzan University
27 Seirei-cho, Seto-shi, Aichi 489-0863, Japan
Email: fushimi@nanzan-u.ac.jp

1 Introduction

An outstanding performance of randomized quasi-Monte Carlo methods for multidimensional integration problems in finance are widely appreciated. Many financial option pricing problems use quasirandom (vector) sequences to generate sample paths of the underlying asset price by summing up the transformed (usually by inverse normal distribution) components of each vector. In this paper we consider the distribution of the summation of the transformed components of each vector in the randomized quasirandom sequence. Numerical experiments for financial option pricing problems are shown to compare randomized quasirandom sequence and another Monte Carlo method. In Sec. 2 we introduce several notions used in the rest part of the paper. Sec. 3 gives the investigation on the distribution of the sum of point coordinates in scrambled nets. We present several numerical experimental results in Sec. 5. In the final section we summarize our result.

2 Preliminaries

In this section we give a brief introduction to the numerical algorithm of option pricing in order to show readers why multidimensional integration is necessary in financial problems. Then we review several definitions and results on quasi-Monte Carlo method.

2.1 Numerical method for option pricing

Assume that, under the risk-neutral measure, the underlying asset price S_t satisfies the stochastic differential equation

$$\mathrm{d}S_t = rS_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}B_t, \quad 0 \le t \le T, \tag{1}$$

where B_t is the standard Wiener process, r is the risk-free interest rate, and σ is the volatility. The price of an option on $S = \{S_t; 0 \le t \le T\}$ is the expected value of its payoff function $p(S)$. We approximate this option price by the expectation over the asset prices S_j at the discrete time points $\{t_j = j\Delta t; 1 \le j \le s\}$, $\Delta t = T/s$. Using a notational convention S_j for $S_{j\Delta t}$, the asset price S_j, as the solution of (1), follows the recurrence relation

$$\begin{aligned} S_j &= S_{j-1}\exp\left((r-\frac{1}{2}\sigma^2)\Delta t + \sigma\sqrt{\Delta t}Z_j\right) \\ &= S_0\exp\left((r-\frac{1}{2}\sigma^2)t_j + \sigma\sqrt{\Delta t}\sum_{k=1}^{j} Z_k\right), \quad j = 1,\ldots,s. \end{aligned} \tag{2}$$

where S_0 is the given initial price, and Z_i's are mutually independent standard normal random variables. The option price is given by evaluating the following expectation:

$$\mathsf{E}\,[p(S_1,\ldots,S_s)] = \mathsf{E}\,[\tilde{p}(Z_1, Z_1+Z_2,\ldots,\sum_{k=1}^{s} Z_k)]. \tag{3}$$

We approximate this expectation by the average of the payoff function over N discrete sample paths $(z_1^{(i)},\ldots,z_s^{(i)})$, $i = 1,\ldots,N$.

$$\mathsf{E}\,[\tilde{p}(Z_1, Z_1+Z_2,\ldots,\sum_{k=1}^{s} Z_k)] \sim \frac{1}{N}\sum_{i=1}^{N}\tilde{p}(z_1^{(i)}, z_1^{(i)}+z_2^{(i)},\ldots,\sum_{k=1}^{s} z_k^{(i)}), \tag{4}$$

where $(z_1,\ldots,z_s)$ are generated from uniform random vector $(x_1,\ldots,x_s)$ over s-dimensional unit cube $[0,1)^s$ by inverse normal distribution function Ψ:

$$z_j = \Psi(x_j), \qquad j = 1,\ldots,s. \tag{5}$$

This inverse transform method is verified by considering the integration form of the expectation and applying the rule of change of variables.

$$\begin{aligned} \mathsf{E}\,[\tilde{p}(Z_1,\ldots,\sum_{k=1}^{s} Z_k)] &= \int_{\Re^s}\tilde{p}(z_1,\ldots,\sum_{k=1}^{s} z_k)\varphi(z_1)\cdots\varphi(z_s)\,\mathrm{d}z_1\cdots\mathrm{d}z_s \\ &= \int_{[0,1]^s}\tilde{p}(\Psi(x_1),\ldots,\sum_{k=1}^{s}\Psi(x_k))\,\mathrm{d}x_1\cdots\mathrm{d}x_s, \end{aligned} \tag{6}$$

where φ is the density of the standard normal distribution.

2.2 Quasi-Monte Carlo method

As shown in (6) all integral problems can be transformed into the one over the unit cube by appropriate change of variables.

$$I = \int_{[0,1]^s} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}. \tag{7}$$

Quasi-Monte Carlo (QMC) methods use low-discrepancy point set $\{\boldsymbol{x}_i\} \in [0,1)^s$ and compute the approximate value of (7) by

$$\hat{I} = \frac{1}{N}\sum_{i=1}^{N} f(\boldsymbol{x}_i). \tag{8}$$

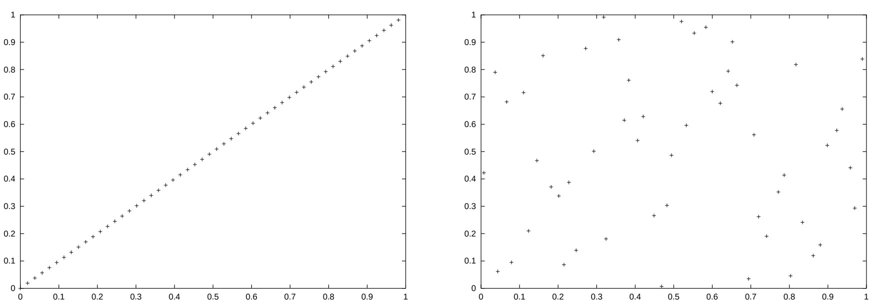
Our investigation focuses on (t,m,s)-net, which is a widely used low-discrepancy point set. The definition of (t,m,s)-net [4] is given below. A subset E of $[0,1)^s$ of the form

$$E = \prod_{j=1}^{s}\left[\frac{a_j}{b^{d_j}}, \frac{a_j+1}{b^{d_j}}\right) \tag{9}$$

where a_j, d_j, $1 \le j \le s$, are integers with $d_j > 0$, $0 \le a_j < b^{d_j}$, is called an *elementary interval in base b.*

Definition 1 *Let t and m be nonnegative integers and $t \le m$. A (t,m,s)-net in base b is a set of b^m points in $[0,1)^s$ such that every elementary interval of volume b^{t-m} contains exactly b^t points of the point set.*

Generally speaking, the smaller t value is, the more uniformly distributed a (t,m,s)-net is.



(a) (b)

Figure 1: A projection of $(0,1,50)$-net in base 53. (a) First 53 points of original Faure sequence. (b) its scrambled result.

In practice, however, a (t,m,s)-net sometimes shows undesirable distribution. See Fig. 1(a), which is the projection to first and second coordinates of (0,1,50)-net in base 53. This set is the first 53 points of Faure sequence in dimension $s = 50$. This point set actually satisfies (0,1,50)-net property defined above, but seems very different from uniform distribution in the unit square. Randomized quasi-Monte Carlo (RQMC) methods are introduced to reduce this "diagonal" distribution on the one hand, and RQMC is used to obtain an error estimation of QMC integration on the other hand. Among several randomized methods, most powerful device may be the following one.

Scrambling method [6].

Let $\{\boldsymbol{y}_i\}$ be a (t, m, s)-net in base b. Suppose $\boldsymbol{y}_i = (y_1^{(i)}, \ldots, y_s^{(i)})$ and $y_j^{(i)} = \sum_{k=1}^{\infty} y_{jk}^{(i)} b^{-k}$ for integers $0 \leq y_{jk}^{(i)} < b$. A scrambled net $\{\boldsymbol{x}_i\}$, $\boldsymbol{x}_i = (x_1^{(i)}, \ldots, x_s^{(i)})$ is defined as $x_j^{(i)} = \sum_{k=1}^{\infty} x_{jk}^{(i)} b^{-k}$, where $x_{jk}^{(i)}$ is obtained by applying a random permutation to $y_{jk}^{(i)}$. Specifically $x_{jk}^{(i)}$ are determined as follows.

$$\begin{aligned} x_{j1}^{(i)} &= \pi_j(y_{j1}^{(i)}), \\ x_{j2}^{(i)} &= \pi_{j y_{j1}^{(i)}}(y_{j2}^{(i)}), \\ &\vdots \\ x_{jk}^{(i)} &= \pi_{j y_{j1}^{(i)} \cdots y_{j,k-1}^{(i)}}(y_{jk}^{(i)}). \\ &\vdots \end{aligned} \tag{10}$$

Here each π is a random permutation over $\{0, 1, \ldots, b-1\}$. In the second line the subscript $y_{j1}^{(i)}$ means that the permutation depends on the value of $y_{j1}^{(i)}$. In the same way $\pi_{j y_{j1}^{(i)} y_{j2}^{(i)} \cdots y_{j,k-1}^{(i)}}$ is a permutation depending on the values of $y_{j1}^{(i)}, \ldots, y_{j,k-1}^{(i)}$. All permutations are mutually independent. A scrambled net thus derived is also a (t, m, s)-net in base b with the same parameter value t as the original net. For the details of scrambling method, the reader is referred to Owen [6].

Fig. 1(b) shows the result of scrambling Fig. 1(a). One can see points are scattered out in the unit square.

Randomly shifting method.

Let $\{\boldsymbol{y}_i\}$ be a (t, m, s)-net in base b and $\boldsymbol{u}$ be a random vector uniformly distributed over the unit cube $[0, 1)^s$. A randomly shifted net $\{\boldsymbol{x}_i\}$ is given by

$$\boldsymbol{x}_i = \boldsymbol{y}_i + \boldsymbol{u} \mod \mathbf{1}, \tag{11}$$

where (mod $\mathbf{1}$) means the componentwise (mod 1) operation.

In [2, 3] we applied these two methods to several test functions and practical problems, and showed the both methods give reliable error estimates.

2.3 Computational notes on unbiasedness

Although randomly shifting method always gives an unbiased estimate of the integral, scrambling method must be implemented carefully to obtain unbiased result. (Owen's first paper on scrambling [5] pointed out this issue.)

Suppose we perform the scrambling procedure on leading k digits and further digits are left untouched. This may be the usual case because we cannot make scrambling on infinitely many digits. Then in the resulting scrambled net, each coordinate is given by

$$x_j^{(i)} = 0.x_{j1}^{(i)} \dots x_{jk}^{(i)} y_{j,k+1}^{(i)} \dots .$$

Since $x_{j1}^{(i)}, \dots, x_{jk}^{(i)}$ are independent and uniformly distributed on values $\{0, 1, \dots, b-1\}$ but $y_{j,k+1}^{(i)}, y_{j,k+2}^{(i)}, \dots$ are still deterministic, the expectation of $x_j^{(i)}$ is given as follow.

$$\begin{aligned} \mathsf{E}\,[x_j^{(i)}] &= \frac{\frac{1}{b}\sum_{i=0}^{b-1} i}{b} + \dots + \frac{\frac{1}{b}\sum_{i=0}^{b-1} i}{b^k} + \frac{y_{j,k+1}^{(i)}}{b^{k+1}} + \dots \\ &= \frac{1}{2}\left(1 - \frac{1}{b^k}\right) + \frac{y_{j,k+1}^{(i)}}{b^{k+1}} + \dots \\ &\neq \frac{1}{2} \end{aligned} \tag{12}$$

This means that $x_j^{(i)}$ is not uniformly distributed on $[0, 1)$ and may give a biased result in numerical integration.

A simple way to remove this bias is to combine scrambling method with shifting method, i.e., add a uniform random vector $\boldsymbol{u}$ to the scrambled-net $\{\boldsymbol{x}_i\}$ with modulo 1:

$$\boldsymbol{x}_i' = \boldsymbol{x}_i + \boldsymbol{u} \mod \mathbf{1}. \tag{13}$$

Then each coordinate $x_j^{(i)\prime}$ in the point set $\{\boldsymbol{x}_i'\}$ satisfies $\mathsf{E}\,[x_j^{(i)\prime}] = \frac{1}{2}$.

3 Distribution of the sum of point coordinates in scrambled net

We investigate the distribution of the sample points in the numerical approximation (4) which are given by the sum of components of the point in scrambled $(0, m, s)$-net in base b.

$$\sum_{j=1}^{s} \Psi(x_j) \tag{14}$$

Since the sample points given by the transform $\Psi : x \mapsto \Psi(x)$ concentrate around $\Psi(\frac{1}{2})$, we apply Taylor expansion around $x = \frac{1}{2}$ to each term and use the fact that

Ψ and even order derivatives of Ψ are zero at $x = \frac{1}{2}$ to get

$$\Psi(x_j) = \sum_{k=0}^{\infty} \frac{1}{(2k+1)!} \Psi^{(2k+1)}(\frac{1}{2}) \left(x_j - \frac{1}{2}\right)^{2k+1}, \tag{15}$$

where $\Psi^{(k)}$ is the k-th order derivative of Ψ.

3.1 The case $b = 2$, first order approximation

We assume $(x_1, \ldots, x_m)$ be a point of Owen's scrambled $(0, m, s)$-net in base 2 in this subsection. First we are concerned with the sum arising from the first term of (15):

$$\sum_{j=1}^{s} \left(x_j - \frac{1}{2}\right) = \sum_{j=1}^{s} x_j - \frac{s}{2}, \tag{16}$$

in the special but interesting case $b = 2$. Let x_j be denoted in base $b = 2$,

$$x_j = 0.x_{j1}x_{j2}\ldots x_{jm}. \tag{17}$$

By the construction, each x_{jk} takes the value 0 or 1 with the same probability $\frac{1}{2}$ independently. Hence the sum of k-th bit $\sum_{j=1}^{s} x_{jk}$ follows the binomial distribution:

$$\mathsf{P}\{\sum_{j=1}^{s} x_{jk} = l\} = \frac{1}{2^s}\binom{s}{l}. \tag{18}$$

Since each bit is independent of the others, the joint distribution of the sum of each digit of x_j appearing in (16) is obtained by the product of binomial distributions.

$$\mathsf{P}\{\sum_{j=1}^{s} x_{j1} = l_1, \ldots, \sum_{j=1}^{s} x_{jm} = l_m\} = \prod_{k=1}^{m} \frac{1}{2^s}\binom{s}{l_k}. \tag{19}$$

If the dimension s is large enough to apply normal approximation

$$\frac{1}{2^s}\binom{s}{l_k} \sim \sqrt{\frac{2}{\pi s}} \exp\left(-\frac{2(l_k - \frac{s}{2})^2}{s}\right), \tag{20}$$

the distribution (19) can be approximated as

$$\mathsf{P}\{\sum_{j=1}^{s} x_{j1} = l_1, \ldots, \sum_{j=1}^{s} x_{jm} = l_m\} \sim \left(\frac{2}{\pi s}\right)^{m/2} \prod_{k=1}^{m} \exp\left(-\frac{2(l_k - \frac{s}{2})^2}{s}\right) \tag{21}$$

Our object is to find the distribution of the form $\mathsf{P}\{\sum_{j=1}^{s} 0.x_{j1}\ldots x_{jm} = l\}$, which can be obtained by summing up all the probabilities for which the tuple of numbers $(l_1, \ldots, l_m)$ satisfies the condition

$$\frac{l_1}{2} + \frac{l_2}{2^2} + \ldots + \frac{l_m}{2^m} = l. \tag{22}$$

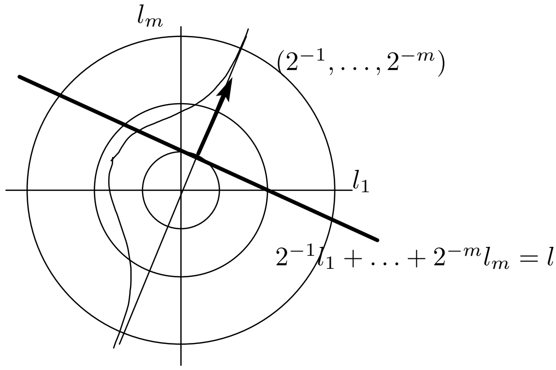


Figure 2: The marginal density of multivariate normal density along the direction $(2^{-1}, \ldots, 2^{-m})$

In case the normal approximation (20) is valid, the probability $\mathsf{P}\{\sum_{j=1}^{s} 0.x_{j1}\ldots x_{jm} = l\}$ is approximated by $\mathsf{P}\{l \le \frac{l_1}{2} + \frac{l_2}{2^2} + \ldots + \frac{l_m}{2^m} \le l + \mathrm{d}l\}$ (see Fig. 2), in other words $\mathsf{P}\{\sum_{j=1}^{s} 0.x_{j1}\ldots x_{jm} = l\}$ is approximated by the marginal density of multivariate normal density (21) along the direction $(2^{-1}, \ldots, 2^{-m})$. Since a marginal density of multivariate normal density is also a normal density, we have

$$\mathsf{P}\{\sum_{j=1}^{s} 0.x_{j1}\ldots x_{jm} = l\} \sim \frac{\sqrt{6}}{\sqrt{\pi s}} \exp\left(-\frac{6(l-\frac{s}{2})^2}{s}\right). \tag{23}$$

Hence we obtain the approximation to the distribution of (16) as

$$\mathsf{P}\left\{\sum_{j=1}^{s}\left(x_j - \frac{1}{2}\right) = l\right\} \sim \frac{\sqrt{6}}{\sqrt{\pi s}} \exp\left(-\frac{6l^2}{s}\right). \tag{24}$$

This result shows that the transformed sample points (14) by scrambling method are approximately normally distributed, while original net can have quite different distribution.

We note that the same argument holds for any partial sum of components, because the projection of $(0, m, s)$-net to any subspace of dimension $s' < s$ becomes also $(0, m, s')$-net by the property of (t, m, s)-net.

3.2 General b case

When $b > 2$, we can continue to investigate the distribution by replacing binomial coefficients in Sec. 3.1 with multinomial coefficients $C_b(s, k)$ which are defined by the following expansion.

$$(1+x+\cdots+x^{b-1})^s = \sum_{k=0}^{s(b-1)} C_b(s,k)x^k. \tag{25}$$

Note that if $b=2$, the coefficient $C_b(s,k)$ is reduced to binomial coefficient $\binom{s}{k}$.

$$\mathsf{P}\{\sum_{j=1}^{s} 0.x_{j1}\dots x_{jm} = \frac{l_1}{b} + \dots + \frac{l_m}{b^m}\} = \prod_{k=1}^{m} \frac{1}{b^s} C_b(s,l_k) \tag{26}$$

Normal approximation can also be applied to the RHS of (26).

4 Numerical experiments

We compare the convergence speed of numerical integration by scrambled nets with stratified sampling method by using option pricing problems. The sum of point coordinates in a scrambled (t,m,s)-net, as shown in Sec. 3, is not only approximately normally distributed in the neighborhood of the mean, but also distributed on the discrete points with interval b^{-m}, since each point coordinate is given by m-digit number in base b, i.e. $0.x_{j1}\dots x_{jm}$.

Stratified sampling in Monte Carlo method has similar mechanism that the samples are drawn from disjoint subsets (strata) of the sample space. More specifically, let X be a random variable on the space Ω, and $A_1,\dots,A_K$ be disjoint subsets of Ω such that $\cup_i A_i = \Omega$ and $\mathsf{P}\{X\in A_i\} = p_i$. To estimate $\mathsf{E}[X]$ stratified sampling generates $n_i = np_i$ samples from A_i, where the total sample size is assumed to be n. The samples from A_i, $X_{i1},\dots,X_{in_i}$, are generated independently from the conditional distribution of X given $X\in A_i$. An unbiased estimator of $\mathsf{E}[X]$ is given by

$$\hat{X} = \sum_{i=1}^{K} \frac{p_i}{n_i} \sum_{j=1}^{n_i} X_{ij}. \tag{27}$$

In the experiments we construct the geometric Brownian motion following "terminal stratification" by [1]. Recalling the notations in Sec. 2.1, let $U_1,\dots,U_K$ be independent uniform random variables on $[0,1)$ and set

$$V_i = \frac{i-1}{K} + \frac{U_i}{K}, \quad i=1,\dots,K. \tag{28}$$

Then $\sqrt{T}\Psi(V_1),\dots,\sqrt{T}\Psi(V_K)$ form a stratified sample from normal distribution $N(0,T)$, and

$$S_{s,i} = S_0 \exp\left((r-\frac{1}{2}\sigma^2)T + \sigma\sqrt{T}\Psi(V_i)\right), \quad i=1,\dots,K \tag{29}$$

form a stratified sample of the terminal price.

First we compare scrambled nets and stratified sampling in European call option pricing. European call option has the payoff function determined by only terminal asset price:

$$p(S) = (S_T - E)^+ \stackrel{\text{def}}{=} \max\{S_T - E, 0\}, \tag{30}$$

where E is the exercise price.

In the experiment we set $s = 50$, $S_0 = 100$, $T = 0.2$, $K = 100$, $r = 0.05$, and $\sigma = 0.3$. We use two kinds of (t, m, s)-nets, one is generated from Faure sequence, which is $(0, m, 50)$-net in base 53, and another is generated from Sobol' sequence, which is (t, m, s)-net in base 2 with $t \neq 0$. The number of strata K is chosen to be a power of 53 for the comparison with scrambled Faure sequence. For example, we use $K = 53^2$ strata to compare with scrambled (0,2,50)-net in base 53.

In case of scrambled net, we calculate the M estimates, $\hat{I}^{(1)}, \ldots, \hat{I}^{(M)}$ by applying independent scrambling to the original net, then we obtain the estimate $\hat{I}$ by arithmetic mean of those M estimates:

$$\hat{I} = \frac{1}{M} \sum_{i=1}^{M} \hat{I}^{(i)}, \tag{31}$$

and the error estimate of $\hat{I}$ by the standard deviation of M estimates:

$$\hat{\sigma} = \left(\frac{1}{M(M-1)} \sum_{i=1}^{M} (\hat{I}^{(i)} - \hat{I})^2 \right)^{1/2}. \tag{32}$$

In stratified sampling, for the comparison with scrambled nets we compute the estimate by using K sample asset prices $S_{s,j}^{(1)}$

$$\hat{I}^{(1)} = \frac{1}{K} \sum_{j=1}^{K} \exp(-rT) p(S_{s,j}^{(1)}). \tag{33}$$

Then generating another independent stratified sample of K prices, we compute another estimate $\hat{I}^{(2)}$, and so on. After repeating this procedure M times, we compute the final estimate $\hat{I}$ and estimated error $\hat{\sigma}$.

Fig. 3(a) shows the relative error $|\hat{I}/I|$ of each method for $M = 10$ repetitions, here I is the true price of the option. Fig. 3(b) shows the estimated standard error $\hat{\sigma}/I$. In this experiment stratified sampling method is superior to the scrambled net method. This example is essentially one-dimensional problem, because the integrand (i.e. payoff function) depends only on the terminal asset price. This results implies stratified sampling method gives better sample distribution than scrambled net methods in one-dimensional case. The case of scrambled Faure sequence does not show a monotone convergence of relative errors at $N = 53^4 \approx 8 \times 10^6$. We do not have good explanation for this result, although the estimated standard error is monotonically decreasing and the 95% confidence interval $[\hat{I} - 1.96\hat{\sigma}, \hat{I} + 1.96\hat{\sigma}]$ includes the true value I.

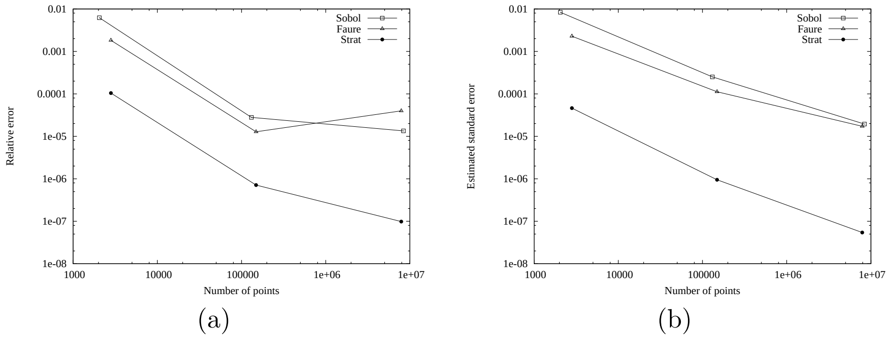


Figure 3: Experimental results for European call option: (a) Relative error by scrambled Sobol', scrambled Faure, and stratified sampling. (b) Estimated relative error of each method.

Stratified sampling method approaches a trapezoidal rule in (deterministic) numerical integration, when the number of strata becomes large. In one-dimensional numerical integration trapezoidal rule is efficient especially for exponential integrands. We consider that this rapid convergence of stratified sampling is due to its close relation to trapezoidal rule and that scrambled method also partially takes over the trapezoidal property because of its sample points' discreteness.

Second we apply the two methods to the pricing of barrier option. Barrier options have several forms of payoff function. Here we use a down-and-out call option, which has a barrier level H below the exercise price E and gives the same payoff as European call option if the asset price is always above the barrier H from the initial time 0 to the maturity T. Hence the payoff function of the down-and-out call option with a barrier level H and exercise price E is given by

$$p(S) = (S_T - K)^+ \times \mathbf{1}\{\min_{0\le t\le T} S_t > H\}, \tag{34}$$

where $\mathbf{1}\{A\}$ is the indicator function that has the value 1 if A is true, and 0 if A is false. Pricing a barrier option requires not only the terminal asset price but also full path of geometric Brownian motion. After generating terminal stratified values V_i, the following algorithm in [1] generates K geometric Brownian paths stratified by terminal price. Let $\Delta t = T/s$ and $t_j = j\Delta t$.

for $i = 1, \ldots, K$

$W_{0,i} := 0$, $W_{s,i} := \Psi(V_i)$, and $S_{s,i} := S_0 \exp\left((r - \frac{1}{2}\sigma^2)T + \sigma\sqrt{T}W_{s,i}\right)$

for $j = 1, \ldots, s-1$

generate $Z \sim N(0,1)$

$W_{j,i} := \frac{t_m - t_j}{t_m - t_{j-1}} W_{j-1,i} + \frac{t_j - t_{j-1}}{t_m - t_{j-1}} W_{s,i} + \sqrt{\frac{(t_m - t_j)(t_j - t_{j-1})}{t_m - t_{j-1}}} Z$

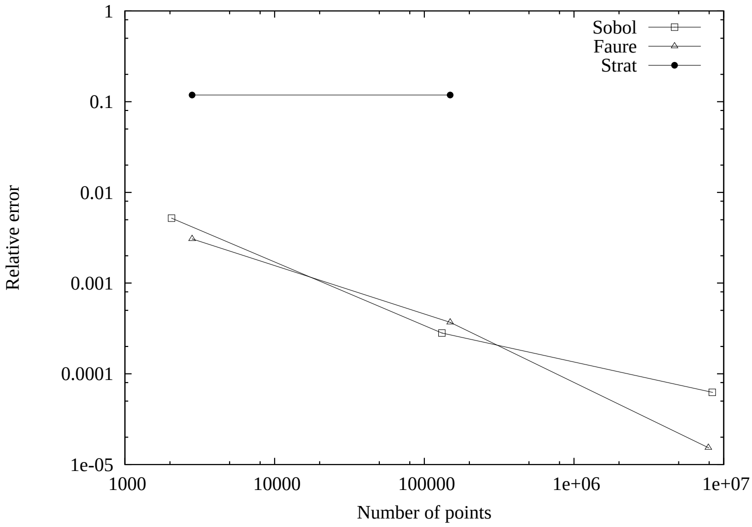


Figure 4: Experimental results for barrier option. Relative error by scrambled Sobol', scrambled Faure, and stratified sampling.

$$S_{j,i} := S_0 \exp\left((r - \tfrac{1}{2}\sigma^2)t_j + \sigma\sqrt{t_j}W_{j,i}\right)$$

In the experiment we use the barrier value $H = 95$ and other parameters are same as the previous ones. In Fig. 4, stratified sampling does not seem to work well in barrier option, i.e. it gives strongly biased result, while scrambled method shows stable convergence. The terminal stratification may induce biased distribution of Brownian sample paths, but scrambled net has good approximate distribution.

5 Concluding remarks

This paper investigated the distribution of the sum of coordinates of the points in a scrambled net, and compared the scrambled net integration and stratified sampling method by numerical experiments of financial option pricing problems. In European call option pricing, stratified sampling is superior to scrambling method in convergence speed, but in a more complicated example, barrier option pricing, stratified sampling fails to give the rapid convergence and seems to give a biased result. Scrambling method gives a reliable result with relatively fast convergence.

Tune-up of Monte Carlo method, such as stratified sampling method, sometimes shows remarkable improvement in convergence speed. But one technique which proved a great success in some case does not necessarily give good results in another case. Randomized quasi-Monte Carlo methods, such as scrambled nets, can be applicable to wide range of problems in high dimensional integration without special tune-up. The reason why RQMC works well is still unclear. Our investigation on the distribution of sum of coordinates may give a partial explanation on why RQMC works well but not-randomized QMC shows poor performance, although the consideration is on very limited cases. More detailed analysis on the distribution of sum will be a future work.

References

[1] Glasserman, P.: *Monte Carlo Methods in Financial Engineering*, Springer, 2003.

[2] Morohosi, H. and M. Fushimi: A Practical Approach to the Error Estimation of Quasi-Monte Carlo Integrations, in *Monte Carlo and Quasi-Monte Carlo Methods 1998*, H. Niederreiter and J. Spanier (eds.), pp. 377–390, Springer, 2000.

[3] Morohosi, H. and M. Fushimi: Experimental Studies on the Error Estimation of the Numerical Integration by Generalized Niederreiter Sequences, in *Operations Research and Its Applications*, Zhang, X.-S. and Liu, D. (eds.), World Publishing, pp. 46-53, 2002.

[4] Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS-NSF 63, SIAM, 1992.

[5] Owen, A. B.: Randomly Permuted (t, m, s)-nets and (t, s)-sequences, in *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, H. Niederreiter and P.J.-S.Shiue (eds.), LNS 106, pp. 299–317, Springer, 1995.

[6] Owen, A. B.: Monte Carlo Variance of Scrambled Net Quadrature, *SIAM J. Numerical Analysis*, Vol. 34(1997), No. 5, pp. 1884–1910.