Proceedings of the 17th IASTED International Conference
MODELLING AND SIMULATION
May 24-26, 2006, Montreal, QC, Canada

SIMULATING OPTION PRICES AND SENSITIVITIES BY HIGHER RANK LATTICE RULES

Yongzeng Lai
Department of Mathematics
Wilfrid Laurier University
75 University Ave. W.
Waterloo, ON, N2L 3C5, Canada
email: ylai@wlu.ca

ABSTRACT
In this paper we introduce the intermediate rank or higher rank lattice rule for the general case when the number of quadrature points is $n^t m$, where m is a composite integer, t is the rank of the rule, n is an integer such that $(n, m) = 1$. Our emphasis is the applications of higher rank lattice rules to a class of option pricing problems. The higher rank lattice rules are good candidates for applications to finance based on the following reasons: the higher rank lattice rule has better asymptotic convergence rate than the conventional good lattice rule does and searching higher rank lattice points is much faster than that of good lattice points for the same number of quadrature points; furthermore, numerical tests for application to option pricing problems showed that the higher rank lattice rules are not worse than the conventional good lattice rule on average.

KEY WORDS
Simulation of multivariate integrations; Monte Carlo and Quasi-Monte Carlo methods; Lattice rules; Option Pricing.

1 Introduction

It is well known in scientific computation that Monte Carlo (MC) simulation method is the main method to deal with high dimensional (≥ 4) problems. The main drawback for this method is that it converges slowly with convergence rate $O(\frac{1}{\sqrt{N}})$, where N is the number of points (or samples or simulations), even after using various variance reduction methods. To speed it up, researchers use quasi-random or low-discrepancy point sets, instead of using pseudo-random point sets. This is the so called quasi-Monte Carlo (QMC) method.

There are two classes of low-discrepancy sequences (LDS). The first one is constructive LDS, such as Halton's sequence, Sobol's sequence, Faure's sequence, and Niederreiter's $(t, m, s)-$nets and $(t, s)-$sequence. This kind of LDS has convergence rate $O(\frac{(\log N)^s}{N})$, where s is the dimension of the problem, N is, again, the number of points. The second class is the integration lattice points, for example, good lattice points (GLP). This type of LDS has convergence rate $O(\frac{(\log N)^{\alpha s}}{N^\alpha})$, where $\alpha > 1$ is a parameter related to the smoothness of the integrand, s and N are the same as above. The monograph by Niederreiter [1] gives very detailed information on constructive LDS and good lattice points, while the monograph by Hua and Wang [2] and Sloan and Joe [3] describe good lattice rules in detail.

Unlike the constructive sequences, the construction of good lattice points is not constructive in the sense that they could be found only by computer searches (except in the $2-$ dimensional case, where good lattice points can be constructed by using the Fibonacci numbers). Such searches are usually very time consuming, especially when the number of points is large or the dimension is high, or both. Therefore, to develop algorithms which can be used in finding good lattice points fast is of practical importance.

This paper discusses the applications of the intermediate rank or higher rank lattice rules (HRLR) to option pricing problems. The motivations of using higher rank lattice points are as follows. For a class of finance problems, we found that using the randomized good lattice points (GLP) can reach much better convergence than the randomized constructive quasi-random sequences (such as, Sobol sequence), let alone the pseudo-random point sets, see [4] about this (in that paper, the lattice points were taken from [2]). The theory given in Section 2 shows that the error bound of a higher rank lattice rule is smaller than that of a good lattice rule, at least asymptotically. And searching higher rank lattice points is much faster than searching good lattice points. Our extensive numerical results confirmed this fact. Some results are listed in Section 2. Furthermore, the results in Section 3 showed that the standard errors of the randomized higher rank lattice points are smaller than those of the randomized good lattice points (most of the times), which are much smaller than the standard errors of the randomized Sobol sequence, when these quasi-random point sets are applied to some financial derivative pricing problems in simulating option values and sensitivities.

2 Higher Rank Lattice Rules

Detailed information about lattice rules can be found in the literature, such as [1], [2] and [3]. We start to introduce lattice rules briefly by considering an integral

$$If = \int_{\mathbf{C}^s} f(\mathbf{x})d\mathbf{x}, \tag{1}$$

where $\mathbf{C}^s = [0,\ 1]^s$ is the s-dimensional unit hypercube, $f(\mathbf{x})$ is one-periodic in each component of $\mathbf{x}$, i.e. $f(\mathbf{x}) = f(\mathbf{x}+\mathbf{z}), \forall\ \mathbf{z} \in Z^s$ (the set of $s-$ dimensional integer points), $\mathbf{x} \in R^s$ (the $s-$dimensional real space). An $s-$dimensional integration lattice $\mathcal{L}$ is a discrete subset of R^s that is closed under addition and subtraction and contains Z^s as a subset. A lattice rule for (1) is a rule of the form

$$Qf = \frac{1}{N}\sum_{j=0}^{N-1} f(\mathbf{x}_j), \tag{2}$$

where $\{\mathbf{x}_0, \cdots, \mathbf{x}_{N-1}\} \subset \mathcal{L} \cap \mathbf{U}^s$ with $\mathbf{U}^s = [0,1)^s$, N is called the order of the rule.

Now we consider the intermediate rank or higher rank lattice rules, i.e. rules of the form

$$Q_t f = \frac{1}{n^t m}\sum_{k_t=0}^{n-1}\cdots\sum_{k_1=0}^{n-1}\sum_{j=0}^{m-1} f(\{\frac{j}{m}\mathbf{g}+\frac{k_1}{n}\mathbf{y}_1+\cdots+\frac{k_t}{n}\mathbf{y}_t\}) \tag{3}$$

for $1 \le t \le s$, where $(m,n) = 1$ and $\mathbf{g}, \mathbf{y}_1, \cdots, \mathbf{y}_t \in Z^s$. Notice that $t = 0$ or $t = 1$ and $n = 1$ in (3) is just the conventional good lattice points rule (we refer it to the rank-1 rule in this paper). Under some conditions (see, for example, *Theorem 7.1*, [3]) on $\mathbf{g}, \mathbf{y}_1, \cdots, \mathbf{y}_t$, the points in (3) are distinct, so that Q_t is a lattice rule of order $N = n^t m$, and it has rank t.

Korobov (1959) gave the first existence of good lattice points in the case where N is a prime number. Niederreiter (1978) extended the existence to general number N. Disney and Sloan proved the existence and obtained the best asymptotic convergence rate for general N in good lattice points case. The existence of good rank t rules can be established, but much more complicated. We introduce

Definition 1. *For any integer $N \ge 2$, let $G = G(N) = \{g = (g_1, \cdots, g_s) \in Z^s, (g_j, N) = 1$ and $-N/2 < g_j \le N/2, 1 \le j \le s\}$. Let $y_1, \cdots, y_t \in Z^s$ be fixed. The mean of $P_\alpha(Q_t)$ over G is*

$$M_{\alpha,t}^{(n)}(m) = \frac{1}{Card(G)}\sum_{\mathbf{g}\in G} P_\alpha(Q_t), \alpha > 1 \tag{4}$$

For the sake of simplicity, Sloan et al chose the special form of $\mathbf{y}_j$ with all the components 0 except the jth which is 1 - the so-called copying rule. Thus (3) becomes

$$Q_t f = \frac{1}{n^t m}\sum_{k_t=0}^{n-1}\cdots\sum_{k_1=0}^{n-1}\sum_{j=0}^{m-1}$$

$$f(\{\frac{j}{m}\mathbf{g}+\frac{(k_1,\cdots,k_t,0,\cdots,0)}{n}\}). \tag{5}$$

With this choice, $P_\alpha(Q_t)$ is easily calculated as follows. For $\alpha > 1, 1 \le t \le s$ and $n \ge 2$, define

$$f_{\alpha,t}^{(n)}(\mathbf{x}) = (\prod_{j=1}^{t} F_\alpha^{(n)}(x_j))\prod_{k=t+1}^{s} F_\alpha(x_k), \tag{6}$$

where

$$F_\alpha^{(n)}(x) = 1 + \frac{1}{n^\alpha}\sum_{h\in Z^*} |\ h\ |^{-\alpha}\ e(hx),$$

and

$$F_\alpha(x) = 1 + \sum_{h\in Z^*} |\ h\ |^{-\alpha}\ e(hx).$$

If $Q_t^{(n)} f$ is the m-point lattice rule defined by

$$Q_t^{(n)} f = \frac{1}{m}\sum_{j=0}^{m-1} f(\{\frac{jn}{m}g_1, \cdots, \frac{jn}{m}g_t, \frac{j}{m}g_{t+1}, \cdots, \frac{j}{m}g_s\}), \tag{7}$$

then

$$P_\alpha(Q_t) = Q_t^{(n)} f_{\alpha,t}^{(n)} - 1. \tag{8}$$

For $1 \le t \le s$ and $\mathbf{g} = (g_1, \cdots, g_s) \in G$, denote $\mathbf{w} = (ng_1, \cdots, ng_t, g_{t+1}, \cdots, g_s)$, and $r_t(\mathbf{h}) = (\prod_{j=1}^{t} r(nh_j)) \cdot \prod_{k=t+1}^{s} r(h_k)$, $\forall \mathbf{h} = (h_1, \cdots, h_s)$. Then applying the rank-1 lattice rule with generating vector $\mathbf{w}$, we have

$$P_\alpha(Q_t) = \sum_{\mathbf{h}\cdot\mathbf{w}\equiv\mathbf{0}\ (mod\ \mathbf{m})} r_t(\mathbf{h})^{-\alpha} - 1. \tag{9}$$

The existence of good rank$-t$ rules and the error bounds for prime m was established by Joe and Sloan (*Theorem* 7.4, [3]). The corresponding results for general m were discovered and proved in [5], and is stated below.

Theorem 1. *For $\alpha > 1, 1 \le t \le s, n \ge 1$ integer, $m > 0$ any integer with $(n,m) = 1$, then*

$$M_{\alpha,t}^{(n)}(m) = \frac{1}{m}\sum_{k=0}^{t}\sum_{l=0}^{s-t}\binom{t}{k}\binom{s-t}{l}\frac{(2\zeta(\alpha))^{k+l}}{n^{k\alpha}}$$

$$\cdot\prod_{p|m} F_{\alpha,k+l}(p^\epsilon) - 1, \tag{10}$$

where the product is over all prime factors p of m, p^ϵ is the highest power of p dividing m, $\prod_{p|m} F_{\alpha,0}(p^\epsilon) = m$, and for $k \ge 1$, $F_{\alpha,k}(p^\epsilon)$ is given by

$$F_{\alpha,k}(p^\epsilon) = 1 + (-1)^k\frac{(1-1/p^{\alpha-1})^k(1-1/p^{(\alpha k-1)\epsilon})}{(p-1)^{k-1}(1-1/p^{\alpha k-1})}. \tag{11}$$

Remark: Using the Binomial Theorem, we can obtain the result of Theorem 7.4 in [3] (the case when m is prime) from Theorem 1, since the assumption that m is prime and n is not a multiple of m implies that $(n,m) = 1$. Moreover, the result of Theorem 1 also holds for $n = 1$ or $t = 0$. In either case, the right hand side of (10) is just

the case of rank-1 in [3], and if $n \geq 2$ and $t = s$, then we obtain the result of maximal rank case in [3].

Now as in the case of rank-1, we give an upper bound for $M_{\alpha,t}^{(n)}(m)$ and hence $P_\alpha(Q_t)$.

Corollary 1. *Under the conditions of Theorem 1, we have*

$$M_{\alpha,t}^{(n)}(m) \leq \frac{4\zeta(\alpha)^2}{\varphi(m)}[\binom{s-t}{2} + \frac{1}{n^{2\alpha}}\binom{t}{2} + \frac{(s-t)t}{n^\alpha}]$$
$$+\frac{1}{m}\{[a(1+2\zeta(\alpha))^{s-t} + b(1-2\zeta(\alpha))^{s-t}]$$
$$+[a(1+\frac{2\zeta(\alpha)}{n^\alpha})^t + b(1-\frac{2\zeta(\alpha)}{n^\alpha})^t]$$
$$+\frac{1}{n^\alpha}[a(1+2\zeta(\alpha))^s + b(1-2\zeta(\alpha))^s]\}, \qquad (12)$$

where $\binom{s-t}{2} = 1$ *for* $s - t < 2, \binom{t}{2} = 1$, *for* $t < 2$; $a = \frac{\zeta(3)}{\zeta(6)} + \frac{1}{2} \approx 1.68$, *and* $b = a - 1$. *Hence*

$$M_{\alpha,t}^{(n)}(m) = O(\frac{\log\log m}{m}), as m \to \infty. \qquad (13)$$

Theorem 2. *Let* $\beta(m) = (1 - s/\log m)^{-1}$. *If* $m > e^{\alpha s/(\alpha-1)}$ *then there is a* $\mathbf{g} \in G$ *such that*

$$P_\alpha(Q_t) \leq M_{\beta(m),t}^{(n)}(m)^{\alpha/\beta(m)}. \qquad (14)$$

If $s \geq 3$, *then*

$$M_{\beta(m),t}^{(n)}(N)^{\alpha/\beta(m)} \sim \frac{1}{n^{\alpha t}}(\frac{2e}{s})^{\alpha s}\frac{(\log m)^{\alpha s}}{m^\alpha} \qquad (15)$$

as $m \to \infty$, *where* $f(x) \sim h(x)$ *as* $x \to \infty$ *means* $lim_{x\to\infty}\frac{f(x)}{h(x)} = 1$.

It is hard to obtain a precise comparison result between the mean for the case of $t = 0$, i.e., $M_{\alpha,t}^{(n)}(m)$, and the corresponding mean for the case of $t = 0$ rule, i.e., $M_\alpha(n^t m)$, even when m is prime, as pointed out in [3]. Notice that the number of points for rank t rule is $n^t m$, and we should use the same number of points when comparing efficiency or convergence rate among different methods. We give an approximate result on this direction based on (15) and a result in [3] similar to (15).

Corollary 2. *For* $\alpha > 1, 1 \leq t \leq s$, $n \geq 1$ *integer,* $m > 0$ *any integer with* $(n, m) = 1$, *let* $\beta_1(m) = (1 - s/\log m)^{-1}$, $\beta_2(n^t m) = (1 - s/\log(n^t m))^{-1}$. *If* $m > e^{\alpha s/(\alpha-1)}$ *then*

$$\frac{M_{\beta_1(m),t}^{(n)}(m)^{\alpha/\beta_1(m)}}{M_{\beta_2(n^t m)}(n^t m)^{\alpha/\beta_2(n^t m)}} \sim \left(\frac{\log m}{\log m + t\log n}\right)^{\alpha s} < 1. \qquad (16)$$

From Corollary 2, we can roughly see that $P_\alpha(Q_t) < P_\alpha(Q_1)$ for $t > 1$ with the same order (number of points) for both rules, at least asymptotically. Our numerical tests showed that it is true even for small number of points. Furthermore, higher rank good lattice points can also be found by computer search via minimizing $P_\alpha(Q_t)$ based on (8), but using m instead of using $n^t m$ (as in the case of good lattice points). Therefore, searching higher rank lattice points is much faster than searching rank-1 lattice points.

Usually, the good lattice points were found by searching Korobov type $\mathbf{g} = (1, b, b^2, ..., b^{s-1})$ mod m (componentwise) with $(b, m) = 1$. Sloan and Reztsov proposed a new searching algorithm - the component-by-component method. We searched extensively for both types of points. Based on the search results we found that these two types of lattice points are comparable in both errors and searching times for the same rank and the same number of points. We only report the Korobov type lattice points here limited to space.

So far as we know, the theory of copying higher rank lattice rule is valid under the assumption that $(n, m) = 1$. We conjecture that this restriction can be relaxed. We are unable to prove this yet so far. But our numerical results strongly support our conjecture, see Table 1 (only partial results are listed). In this table, the comparison is based on the same number of points, where Kor_t0 stands for the Korobov type lattice points with $t = 0$ (i.e., rank-1 case), similarly for Kor_t4. Time is measured in seconds. Whenever time is zero, it just means that the time used in searching is less than 0.5 seconds. The CPU times used in searching may be machine dependent. Dev-C++ was used as our programming language (run on a laptop under Windows system). In order to measure CPU time as precise as possible, all the programs were run on the same machine and only one program, no any other programs, was run one at a time on the machine. Our searching results showed that within the same type of lattice points, the higher the rank, the smaller the P_2, and the faster the search. The search time for rank=4 in the case of number of point =32768 is about 1 second, those for all the other cases are less than 0.5 seconds.

Table 1: Computer search results of $t = 0$ and $t = 4$, with $(n, m) \neq 1$, $n = 2$, $m =$ a power of 2, dimension = 5.

	Kor_t0			Kor_t4	
2^tm	b	P_2	Time	b	P_2
1024	189	0.735	0	5	0.373
2048	453	0.264	1	27	0.164
4096	1595	0.121	3	21	0.067
8192	2099	0.048	10	61	0.026
16384	2959	0.018	43	35	0.010
32768	1975	0.007	169	131	0.004

3 Applications to Option Pricing

Under the Black-Scholes framework, many European options can be expressed in terms of multivariate normal distributions. Examples are options on maximum and minimum of n assets, discrete lookback options, discrete shout options, discrete partial barrier options, reset options, etc., see [6] and the references therein.

In this section, we apply both the Monte Carlo and the quasi-Monte Carlo methods to applied finance area-

option pricing, and compare the efficiencies among different methods. For the quasi-Monte Carlo methods, we use Sobol sequence and both rank-1 and higher rank lattice points. The Sobol sequence is usually the best among the constructive LDS based on our tests.

To compare the efficiencies of different methods, we need a benchmark for fair comparisons. If the exact value of the quantity to be estimated can be found, then we use the absolute error or relative error for comparison. Otherwise, we use the standard error (*stderr*) for comparison. Here $stderr = \frac{\overline{\sigma}}{\sqrt{N}}$, where $\overline{\sigma}^2$ is the unbiased sample variance, N is the sample size. For LDS sequences, we define the standard error by introducing random shift as follows. Assume that we estimate $\mu = E[f(\mathbf{x})]$, where $\mathbf{x}$ is an $s-$*dimensional* random vector. Let $\{\mathbf{x}_i\}_{i=1}^m \subset \mathbf{C}^s$ be a finite LDS sequence, $\{\mathbf{r}_j\}_{j=1}^n \subset \mathbf{C}^s$ be a finite sequence of random vectors. For each fixed j, we have a sequence $\{\mathbf{y}_i^{(j)}\}_{i=1}^m$ with $\mathbf{y}_i^{(j)} = \mathbf{x}_i + \mathbf{r}_j$. It can be shown that such a sequence still has the same convergence rate as the original one. Denote $\mu_j = \frac{1}{m}\sum_{i=1}^m f(\mathbf{y}_i^{(j)})$ and $\overline{\mu} = \frac{1}{n}\sum_{j=1}^n \mu_j$.The unbiased sample variance is $\overline{\sigma}^2 = \frac{\sum_{j=1}^n (\mu_j - \overline{\mu})^2}{n-1} = \frac{n\sum_{j=1}^n \mu_j^2 - \left(\sum_{j=1}^n \mu_j\right)^2}{(n-1)n}$. Then the standard error is defined by $stderr = \frac{\overline{\sigma}}{\sqrt{n}}$. The efficiency of a QMC method (after randomization) over the MC method is defined as the ratio of the standard error of the MC method to the standard error of a QMC method (both methods have the same number of points, otherwise the comparison is not fair).

As an example, let us consider the computation of call options on maximum of s assets. Using martingale method, Dufresne et al derived in [7] that the value of a call option can be expressed in terms of multivariate normal distributions:

$$V = V_{\max}^s(\{S_i\}, \{\sigma_i\}, \Sigma_0, r, q, \phi) = \sum_{i=1}^s \phi S_i e^{-q_i T}\cdot$$

$$N_s(e_{i1}, ..., e_{i,i-1}, d_i^{(i)}(K,T), e_{i,i+1}, e_{is}; \Sigma_i) -$$

$$Ke^{-rT}\left[1 - N_s(-d_1^Q(K,T), ..., -d_s^Q(K,T); \Sigma_0)\right] \quad (17)$$

where

$$e_{ik} = \frac{\log(S_i/S_k) + T\Omega_{ik}^2/2}{\Omega_{ik}\sqrt{T}},$$

$$\Omega_{ik} = \sqrt{\sigma_i^2 - 2\rho\sigma_i\sigma_k + \sigma_k^2},$$

$$d_i^{(i)}(K,T) = \frac{\log(S_i/K) + (r + \sigma_i^2/2)T}{\sigma_i\sqrt{T}},$$

$$d_i^Q(K,T) = \frac{\log(S_i/K) + (r - \sigma_i^2/2)T}{\sigma_i\sqrt{T}},$$

$\Sigma_0 = (\rho_{jk})_{s\times s}$ and for $i = 1, ..., s$, $\Sigma_i = (\rho_{jk}^{(i)})_{s\times s}$ with

$$\rho_{jk}^{(i)} = \frac{\sigma_i^2 + \rho_{jk}\sigma_j\sigma_k - \rho_{ij}\sigma_i\sigma_j - \rho_{ik}\sigma_i\sigma_k}{\Omega_{ij}\Omega_{ik}}, j, k \neq i;$$

$$\rho_{ik}^{(i)} = \frac{\sigma i - \rho_{ik}\sigma_k}{\Omega_{ik}}, i \neq k; \rho_{ii} = 1.$$

Thus, in order to estimate the option values, we need to estimate the following $s-$variate normal distribution $H(\mathbf{a}, \Sigma) = \frac{1}{\sqrt{\det(\Sigma)(2\pi)^s}}\int_{-\infty}^{a_1}\cdot\cdot\cdot\int_{-\infty}^{a_s}\exp(-\frac{1}{2}\mathbf{x}^t\Sigma^{-1}\mathbf{x})d\mathbf{x}$, where $\mathbf{a} = (a_1, a_2, ..., a_s)$, $-\infty \leq a_i \leq +\infty$ ($i = 1, 2, ..., s$), $\mathbf{x} \in R^s$, $d\mathbf{x} = dx_1...dx_s$, $\Sigma = (\sigma_{ij})_{s\times s}$ is a positive definite correlation matrix. Details about the computation of multivariate normal distributions can be found in [8]. Notice that after the transformation, the $s-$dimensional integral for $H(\mathbf{a}, \Sigma)$ is transformed into an $s-1$ dimensional integral.

For the numerical demonstration, we consider a call option on maximum of 6 stocks. In our simulations, each method was randomly shifted, including the MC method, so that each method has the same number of points. We took the number of random shifts to be 10, other parameters are $s = 6$, $K \in \{\$90, \$100, \$110\}$, $r = 10\%$, $S_i = \$100$, $\sigma_i = 0.2$, $\rho_{ij} = 0.5$, $i \neq j$, $i, j = 1, ..., 6$. Besides the option values, the option sensitivities or Greek letters $\Delta_i = \frac{\partial V}{\partial S_i}$, $\Gamma_{ij} = \frac{\partial^2 V}{\partial S_i \partial S_j}$, $\mathcal{V}_i = \frac{\partial V}{\partial \sigma_i}$, $\theta = \frac{\partial V}{\partial T}$ and $\rho = \frac{\partial V}{\partial r}$ are very important quantities in financial risk management and trading. They are usually harder to obtain than the option values themselves. The results where $K = \$100$ are listed in the following Tables 2, 3 and 4, and the results where $K = \$90$ and $K = \$110$ are similar and are omitted here.

In these tables, column 1 contains the numbers of points, numbers in the MC column are the standard errors, those in the columns of quasi-Monte Carlo methods are efficiencies of the corresponding methods over the Monte Carlo method. Here we do not include the CPU times for different methods since these programs were run on a mainframe using UNIX system, and there were many other programs were also running at the time I ran these programs. And I think that the CPU times measured in this way are not precise.

Table 2: Comparison of estimated call option values and efficiencies, the option value is \$28.81 with standard error 1.5099e-06 obtained by higher rank lattice rule (rank=4) using 2^{14} =16384 points with 10 random shifts. The standard error is zero in my simulation by the same rule using 2^{15} =32768 points with 10 random shifts.

N	MC	Sobol	Kor_t0	Kor_t4
2^{10}	0.6832	10.8	219.9	200.5
2^{11}	0.4834	27.8	666.8	1194.9
2^{12}	0.3413	39.6	2104.7	371.8
2^{13}	0.2409	41.2	18129.2	22758.4
2^{14}	0.1703	110.2	30156.0	112804.8
2^{15}	0.1206	101.3	253540.7	*

From Table 2, we observed that the randomized lattice rules achieve much better results than the randomized Sobol's sequence does, the latter is about 10 to 110 time more efficient than the MC method. The randomized Korobov type higher rank lattice points beat the randomized

rank-1 lattice points, except when $N = 2^{10} = 1024$ and $2^{12} = 4096$.

Table 3: Comparison of estimated option sensitivity (Greek letter, Δ_1 in this table) values and efficiencies, the value of Δ_1 is 0.1898 with standard error 4.4427E-09 obtained by higher rank lattice rule (rank=4) using 2^{15} =32768 points with 10 random shifts.

N	MC	Sobol	Kor_t0	Kor_t4
2^{10}	0.0151	7.3	135.5	98.1
2^{11}	0.0107	10.7	330.8	1226.6
2^{12}	0.0077	9.9	900.7	110.7
2^{13}	0.0054	19.7	8861.8	9833.5
2^{14}	0.0038	30.7	24742.9	70692.9
2^{15}	0.0027	42.8	137025.0	605473.8

Again, the randomized lattice rules are much more efficient than the randomized Sobol's sequence, the latter is about 8 to 43 time more efficient than the MC method. Kor_t4 is more efficient than Kor_t0 except when $N = 2^{10} = 1024$ and $2^{12} = 4096$.

Table 4: Comparison of estimated gamma ($\Gamma_{11} = \frac{\partial^2 V}{\partial S_1^2}$) values and efficiencies, the value of Γ_{11} is 0.01631 with standard error 1.2418e-09 obtained by higher rank lattice rule (rank=4) using 2^{15} =32768 points with 10 random shifts.

N	MC	Sobol	Kor_t0	Kor_t4
2^{10}	4.6E-4	8.0	107.6	12.8
2^{11}	3.3E-4	14.0	118.6	223.7
2^{12}	2.3E-4	19.6	484.3	30.3
2^{13}	1.7E-4	22.6	4217.8	4457.5
2^{14}	1.2E-4	24.1	3108.7	28713.5
2^{15}	8.2E-5	40.0	108047.3	66385.4

The conclusion is similar to that of Table 3. The randomized Kor_t4 is more efficient than the randomized Kor_t0 except when $N = 2^{10} = 1024$, $2^{12} = 4096$ and $2^{15} = 32768$.

In our simulations, the pseudo random number generator we used is $ran2()$ in [9]. The periodizing function used is $\psi(x) = \frac{1}{2\pi}\left(2\pi x - \sin(2\pi x)\right)$.

4 Conclusion

In this paper, We introduced the higher rank lattice rules and gave a general expression for the average of $P_\alpha(Q_t)$ for higher rank lattice rule over a subset of Z^s, an upper bound and an asymptotic rate for higher rank lattice rule. The results recovered the cases of good lattice rule and maximal rank rule. Computer search results showed that $P_2's$ by the higher rank lattice rule were smaller than those by good lattice rule, while searching higher rank lattice points was much faster than that of good lattice points for the same number of quadrature points. Numerical tests for applications to an option pricing problem showed that the higher rank lattice rules ($t > 0$) usually beat the conventional good lattice rule ($t = 0$ case). Both of these rules showed significant superiority over the Sobol sequence. Our tests (not listed here) on other types of options showed similar efficiency gains of higher rank lattice rules over good lattice rules, though the gains may vary.

Since searching higher rank lattice points is much faster than that of $rank - 1$ lattice points (say the rank is larger than 2); the search algorithm is simple; and the values of P_2 for higher rank lattice points are smaller than that for the $rank - 1$ points; furthermore, (standard) errors obtained by higher rank lattice rules to practical problems are not worse than those by the $rank - 1$ rules on average, the higher rank lattice rules are good candidates for applications. One unsolved problem in lattice rules (whether high rank or not) is the periodizing seems not work well in high dimensions. It needs futher exploration.

Acknowledgements

This research was partially supported by an Natural Sciences and Engineering Research Council of Canada (NSERC) grant.

References

[1] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, Philadelphia, 1992.

[2] L. Hua and Y. Wang, *Applications of Number Theory in Numerical Analysis*, Springer-Verlag, 1980.

[3] I. H. Sloan and S. Joe, *Lattice Methods for Multiple Integration*, Oxford University Press, New York, 1994.

[4] P. Boyle, Y. Lai and K. S. Tan, Pricing Options Using lattice rules, *North American Actuarial Journal*, 9(3), 2005, 50-76.

[5] Y. Lai, Monte Carlo and Quasi-Monte Carlo Methods and Their Applications, Ph. D Dissertation, Department of Mathematics, Claremont Graduate University, California, USA, 2000.

[6] P. Zhang, *Exotic Options*, 2nd edition, World Scientific, 1998.

[7] P.C. Dufresne, W. Keirstead and M. P. Ross, Pricing Derivatives the Martingale Way, working paper, 1996.

[8] Y. Lai, Effcient Computations of Multivariate Normal Distributions with Applications to Finance, working paper, Departmetn of Mathematics, Wilfrid Laurier University, Waterloo, Ontario, Canada, 2005.

[9] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical recipes in C: The Art of Scientific Computing*, Cambridge University Press, 1992.