High dimensional Sobol' Sequences and their applications

S. Kucherenko

Email: s.kucherenko@broda.co.uk

BRODA Ltd. www.broda.co.uk

1. Evaluation of high dimensional integrals using MC and QMC methods

There are three main applications of Monte Carlo methods. First, the generation of (quasi or pseudo) random numbers is used to generate representative samples (scenarios), to describe the uncertainties of the problem at hand through probabilities distributions (Niederreiter H., 1992). Second, random number sampling is used for random search in stochastic optimization (Kucherenko S., 2005). Third, the simulation problem can be presented as evaluation of integrals. In this section we consider this application.

Many problems in numerical analysis and physics are concerned with high dimensional integrals. While the classical grid methods are very efficient for low dimensional integrands, they become computationally impractical when the number of dimensions d increases and thus the number of required integrand evaluations grows exponentially. This effect is known as "the curse of dimensionality". In contrast, the convergence rate of Monte Carlo (MC) integration methods does not depend on the number of dimensions d. However, the rate of convergence $O(1/N^{1/2})$, where N is the number of sampled points, attained by MC is rather slow. A higher rate of convergence can be obtained by using deterministic uniformly distributed sequences instead of pseudo-random numbers. Methods based on the usage of such sequences are known as Quasi Monte Carlo (QMC) methods. Asymptotically, QMC can provide the rate of convergence $O(1/N)$.

Consider evaluation of an integral

$$I[f] = \int_{H^n} f(\vec{x}) d\vec{x} ,$$

where the function $f(\vec{x})$ is integrable in the d-dimensional unit hypercube H^d. Classical grid methods have an integration error decreasing as $O(N^{p/d})$, where p is the order of an integration method. The Monte Carlo quadrature formula is based on the probabilistic interpretation of an integral. For a random variable that is uniformly distributed in H^d

$$I[f] = E[f(\vec{x})] ,$$

where $E[f(\vec{x})]$ is the mathematical expectation. An approximation to this expectation is

$$I_N[f] = \frac{1}{N}\sum_{i=1}^{N} f(\vec{x}_i),$$

where $\{\vec{x}_i\}$ is a sequence of random points in of length N. The approximation $I_N[f]$ converges to $I[f]$ with probability **1**. Consider an integration error ε defined as

$$\varepsilon = \left| I[f] - I_N[f] \right|.$$

It follows from the Central Limit Theorem that the expectation of ε^2 is

$$E(\varepsilon^2) = \frac{\sigma^2(f)}{N},$$

where $\sigma^2(f)$ is the variance given by

$$\sigma^2(f) = \int_{H^n} f^2(\vec{x})d\vec{x} - (\int_{H^n} f(\vec{x})d\vec{x})^2$$

Then the expression for the root mean square error of the MC method is

$$\varepsilon_{MC} = (E(\varepsilon^2))^{1/2} = \frac{\sigma(f)}{N^{1/2}}.$$

In contrast to grid methods, the convergence rate of MC methods does not depend on the number of variables *d* although it is rather slow.

2. Regular grid, MC and QMC sampling methods

The efficiency of MC methods is determined by the properties of the random numbers. It is known that random number sampling is prone to clustering: for any sampling there are always empty areas as well as regions in which random points are wasted due to clustering. As new points are added randomly, they do not necessarily fill the gaps between already sampled points. Low-discrepancy sequences (LDS) are specifically designed to place sample points as uniformly as possible. Unlike random numbers, successive low discrepancy points “know" about the position of previously sampled points and fill the gaps between them (Fig 2.1). LDS are also known as quasi random numbers.

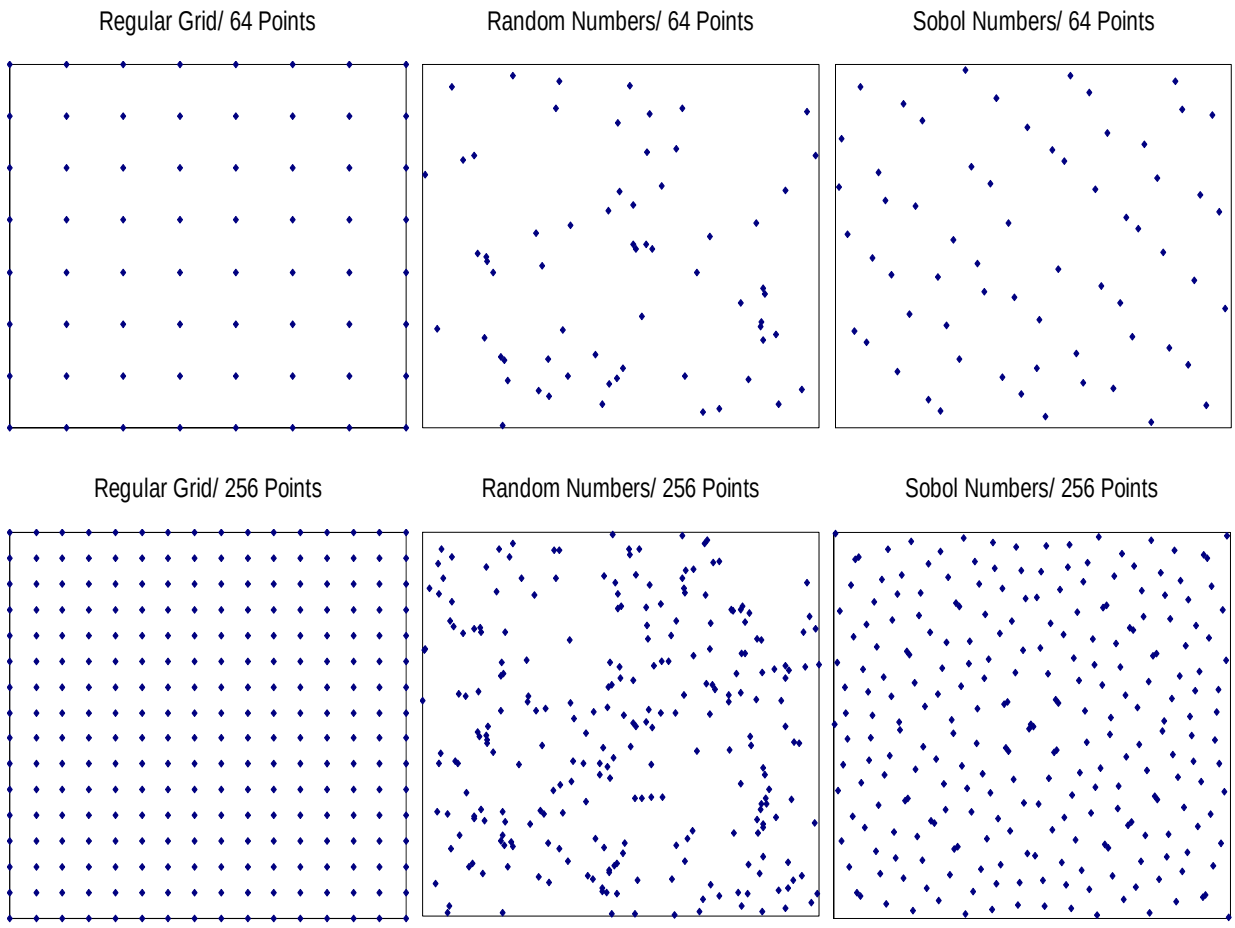

Fig. 2.1. Regular grid, MC and QMC distributions of N=64 (upper row) and N=256 (lower row) points in two dimensions

The regular grid of points (or lattice methods) seems be an efficient way for the integral evaluation. For up to 4 dimensions it works better or not worse than random sampling. For dimensions higher than 4, regular grid is not practical.

The points in the regular grid are centred into each cell of the grid. Fig 2.2 illustrates the difference between the regular grid and quasi Monte Carlo sampling. It also illustrates a very important property of LDS: the projection of the d-dimensional LDS on the s-dimensional subspace forms the s-dimensional LDS. It is important to note that in practice low dimensional projections have good uniform distributions, while in high dimensions LDS may not be particularly well equidistributed for feasible N. In particular, it explains the efficiency QMC methods in high dimensions for many practical problems although formal measures like discrepancy show that in high dimensions LDS sampling may be not as good as random number sampling (see next paragraph). The regular grid of points does not possess this property. This is why it is obvious that LDS sampling gives much better way of arranging N points in d–dimensions.

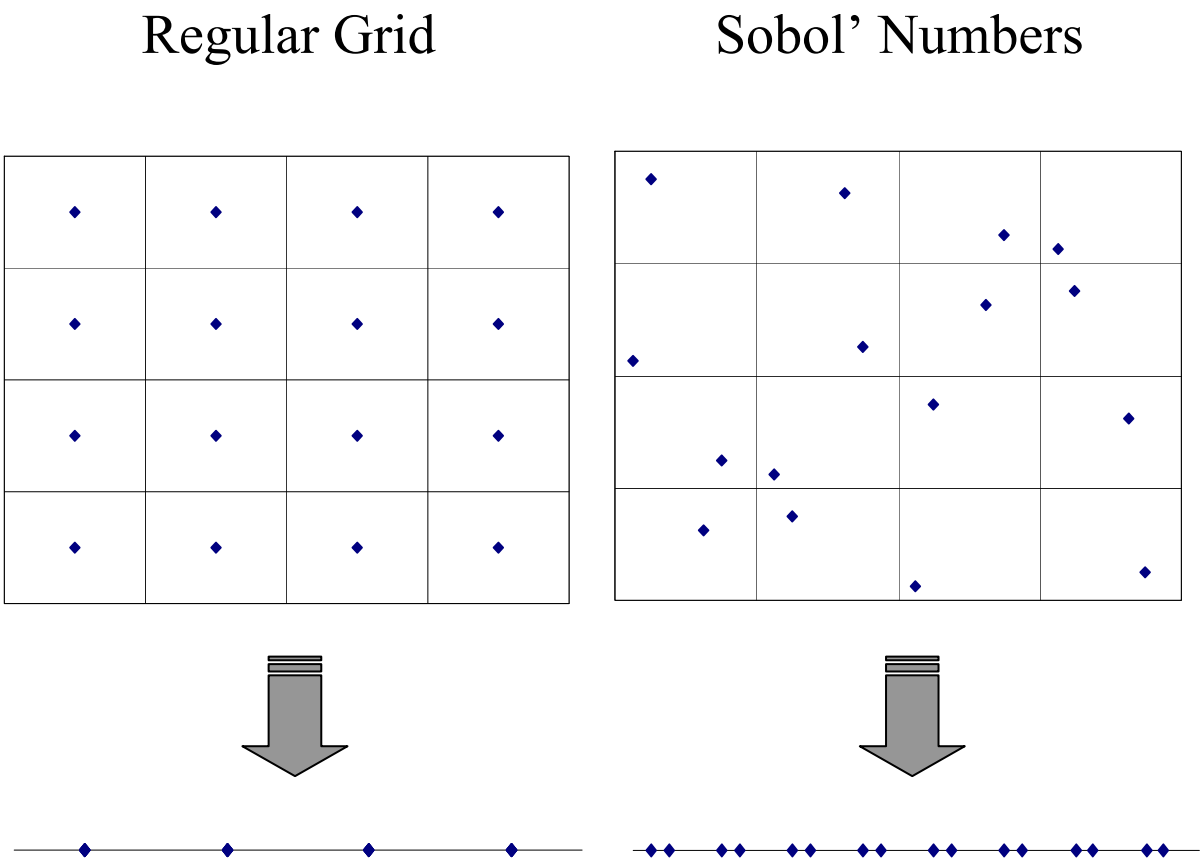

Fig. 2.2 Two different ways of arranging N=16 points in two dimensions

There are three problems for the use of regular grid for evaluation of integrals:

1. The problem of dimensionality ("the curse of dimensionality"). It's been discussed in the previous section.
2. It is not possible to incrementally enlarge the size of the grid and at the same time keep the grid uniform. This means that with a uniform grid approach it is not possible to have a termination criterion that can be invoked incrementally.
3. The concavity bias. The regular grid generates small errors that add up, whereas random sampling generates big errors that cancel on average. The details can be found in Dupire & Savine (1998).

The random and quasi-random sampling methods do not have this kind of problems; however, there are significant differences between them. A very important difference is that quasi random numbers are more uniformly distributed. One the useful quantitative measures uniformity is the discrepancy.

3. Discrepancy

Discrepancy measures the extent to which the points are evenly dispersed in the unit hypercube. Consider a number of points N from a sequence $\{\vec{x}\}$ in an d - dimensional rectangle Q whose sides are parallel to the coordinate axes, $Q \in H^{\mathrm{d}}$. Then, the discrepancy is defined as

$$D_N = \sup_{Q \in H^d} \left| \frac{N_Q}{N} - m(Q) \right|,$$

where $m(Q)$ is a volume of Q and N_Q is the number of points of the sequence $\{\vec{x}\}$ that are contained in Q.

A low discrepancy sequence is one satisfying the upper bound condition:

$$D_N \le c(d) \frac{(\ln\ N)^n}{N}$$

Constant $c(d)$ depends on the sequence and even on the dimension, but does not depend on N. For random numbers the expected discrepancy is

$$D_N = O((\ln\ \ln\ N) / N^{1/2})$$

4. The Koksma- Hlawka bound

The Koksma- Hlawka inequality gives an upper bound for the QMC integration error:

$$\varepsilon \le V(f) D_N\ .$$

Here, $V(f)$ is the variation of $f(\vec{x})$. It is assumed that $f(\vec{x})$ is a function of bounded variation in the sense of Hardy and Krause (Niederreiter, 1992). Apparently, the smaller the discrepancy D_N, the better the convergence of the QMC integration.

It was shown experimentally in Morokoff & Caflisch (1995) that the QMC integration error is determined by the variance and not by the variation of the integrand. It is generally accepted that the rate of the discrepancy determines the expected rate of the accuracy, so one can use an estimate of the QMC convergence rate

$$\varepsilon_{QMC} = \frac{O(\ln\ N)^n}{N}$$

4. Construction of low-discrepancy sequences

There are a few well-known and commonly used LDSs. Different principles were used for their construction by Halton, Faure, Sobol, Niederreiter and others. The LDS developed by Niederreiter has the best theoretical asymptotic properties (Niederreiter, 1992). However, many practical studies have proven that the Sobol' LDS is in many aspects superior to other LDS (Bratley et al, 1992, Caflisch, 1997).

4.1 Halton's sequence

Halton's sequence is the easiest to construct. Consider an integer n in a base b, where b is a prime number: $n = (\cdots a_4 a_3 a_2 a_1 a_0)_b$ In the decimal system $n = \sum_{j=1}^{m} a_j b^j$. Reverse the digits and add a radix point to obtain a number within the unit interval

$$y = (0.\ a_0 a_1 a_2 a_3 a_4 \cdots)_b$$

The n-th number in the Halton's sequence with base b is

$$h(n;b) = \sum_{j=1}^{m} a_j b^{-j-1}.$$

Here $m = \log_b n$. d – dimensional Halton's sequence $\{(h(n;2), h(n;3), ..., h(n;b_d)\}$, b_i is the prime number.

The Halton sequence uses one different prime base for each dimension. For the first dimension it uses base 2, for the second dimension it uses base 3, for the third dimension it uses base 5, and so on. Higher base means higher cycle and higher computational time.

The major problem for the quasi-random sequences is their degradation when the dimension is large. The generation process of uniformly distributed points in $[0, 1)^d$ becomes increasingly harder as d increases because the space to fill becomes too large.

The high-dimensional Halton sequences exhibit long cycle lengths, due the large prime number base. For example, in the dimension 50, is used the base 229, the 50th prime number. The long cycle length means that the high-dimensional sequence needs several numbers for an entire walk in the interval [0, 1). Halton's sequence becomes unsatisfactory after dimension 14. In practice, due the correlation, practitioners prefer to avoid the use of Halton's sequence for more than 6 or 8 dimensions.

4.2 Sobol' sequence

The construction of Sobol' sequence is rather complex. The Sobol' sequence numbers $x_n = (x_n^1, x_n^2, ..., x_n^d)$ are generated from a set of binary fractions of length *b* bits, $v^j = (0.v_1^j v_2^j v_3^j ... v_b^j)_2$, $j=1,\ldots,d$. Obviously, that $v^j{}_i \in \{0,1\}$.

As above *n*- th number in a sequence is considered in a binary form

$$n = (\cdots b_4 b_3 b_2 b_1 b_0)_2$$

To produce the Sobol' integer x_n^j the following formula is used

$$x_n^j = b_1 v_1^j \oplus b_2 v_2^j \oplus ... \oplus v_b^j ,$$

where $\oplus$ is an addition modulo 2 operator: $0 \oplus 0 = 0$, $1 \oplus 1 = 0$, $0 \oplus 1 = 1$, $1 \oplus 0 = 1$. $\oplus$ can also be seen as bit wise XOR. This result is obtained by performing the bitwise exclusive XOR of the direction numbers v_i, for which $b_i \neq 0$. Direction numbers are defined below.

The final conversion to a uniform variate y_n^j is performed by dividing x_n^j by 2^{b_j} :

$$y_n^j = x_n^j / 2^{b_j} .$$

To generate direction numbers, we first need a primitive polynomial, which is irreducible polynomial with binary coefficients over the field G2.

$$P_l = x^q + a_1 x^{q-1} + ... + a_{q-1} x + 1, \; a_k \in \{0,1\} .$$

It is irreducible (can't be factored) and the smallest power *p* for which polynomial divides $x^p + 1$ is $p = 2^q - 1$. Examples of primitive polynomials: $x+1$, $x^2 + x + 1$, $x^3 + x + 1$, $x^3 + x^2 + 1$.

A different primitive polynomial is used in each dimension. Given a primitive polynomial of degree *q*, m_i's are defined by a recurrence relation in integer arithmetic

$$m_i = 2a_1 m_{i-1} \oplus 2^2 a_2 m_{i-2} \oplus \cdots \oplus 2^q m_{i-q} \oplus m_{i-q}$$

The recurrence relation defined by the degree 0 polynomials is $m_i = 1$. To fully define the direction numbers initial numbers $m_1,\ldots, m_d$ are required. An algorithms which provides initial numbers for an efficient Sobol' sequence generation is presented in Sobol' *et al* (1992). Once m_i are generated, direction numbers are defined as $v_i = m_i / 2^i$, where $m_i < 2^i$ is an odd integer.

The Sobol' LDS satisfies the following the three main requirements (Sobol', 1976):

1. Best uniformity of distribution as $N \to \infty$.
2. Good distribution for fairly small initial sets.
3. A very fast computational algorithm.

As a result, points generated by the Sobol' LDS produce a very uniform filling of the space even for a rather small number of points *N*, which is a very important point in practice.

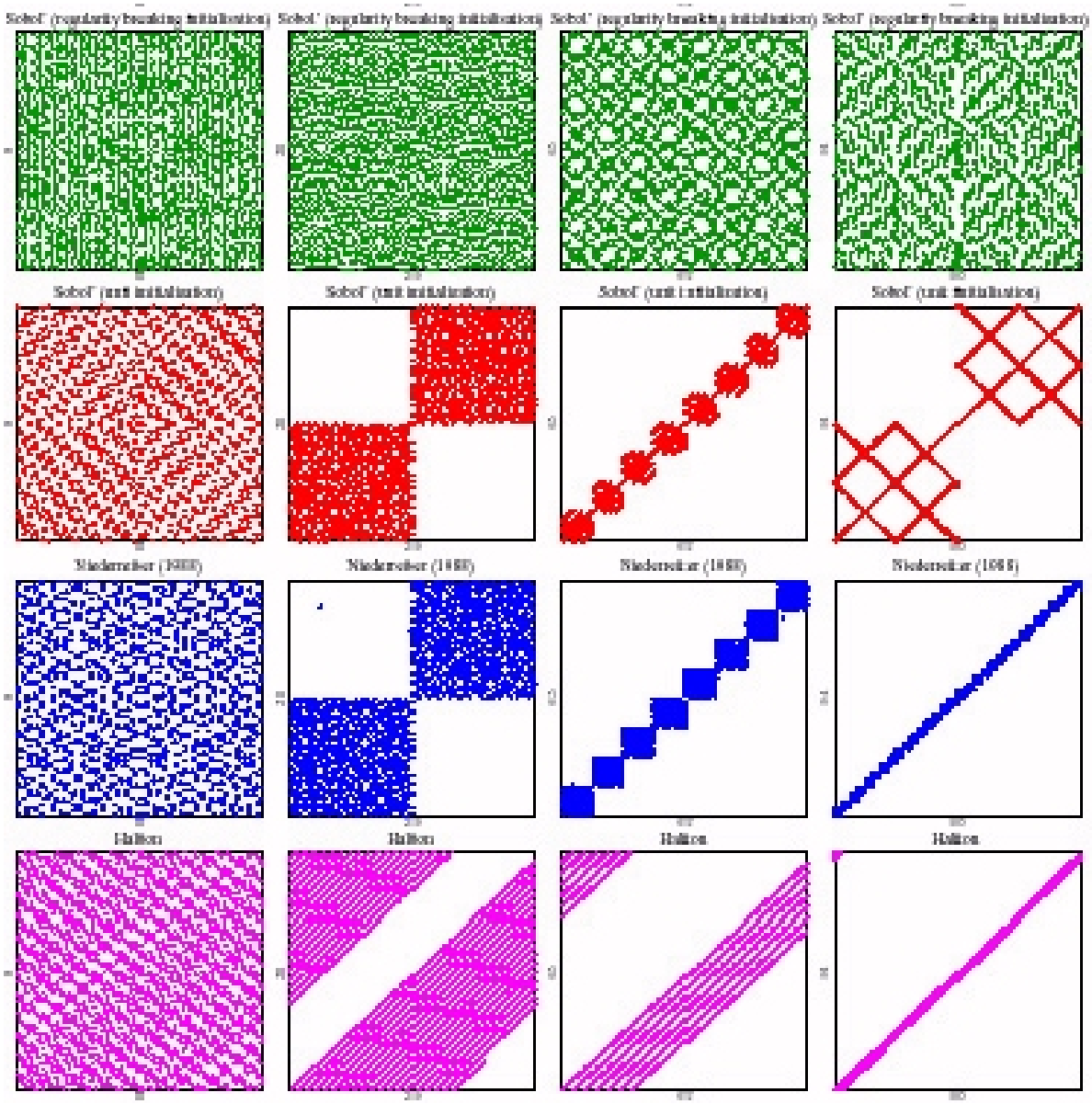

Fig. 4.1 2D rojections from adjacent dimensions for Sobol' , Sobol' with unit initialization, Niederreiter and Holton LDS (Jackel, 2002).

In Sobol's algorithm direction numbers is a key component to its efficiency. In some implementations (f.e. Tezuka, 1995) this critical issue was overlooked. As a result, constructed LDS did not satisfy above-mentioned criteria and did not perform well in tests (Fig. 4.1).

We conclude this chapter by a quote "Preponderance of the experimental evidence amassed to date points to Sobol' sequences as the most effective quasi-Monte Carlo method for application in financial engineering." (Glassserman, 2003).

5. Discrepancy. Numerical results

To compare uniformity properties of randon numbers and quasi random sequences the L_2 discrepancy was calculated for various dimensions (Fig. 5.1 – Fig. 5.3). T_N^n discrepancy is defined in Tezuka (1995).

Notations for the figures:

“Random” is random number generator *Urand* taken from Forsythe *et al* (1977).

“Sobol” is Sobol’ LDS (*Broda, 2008)*

“Halton ” is Halton LDS

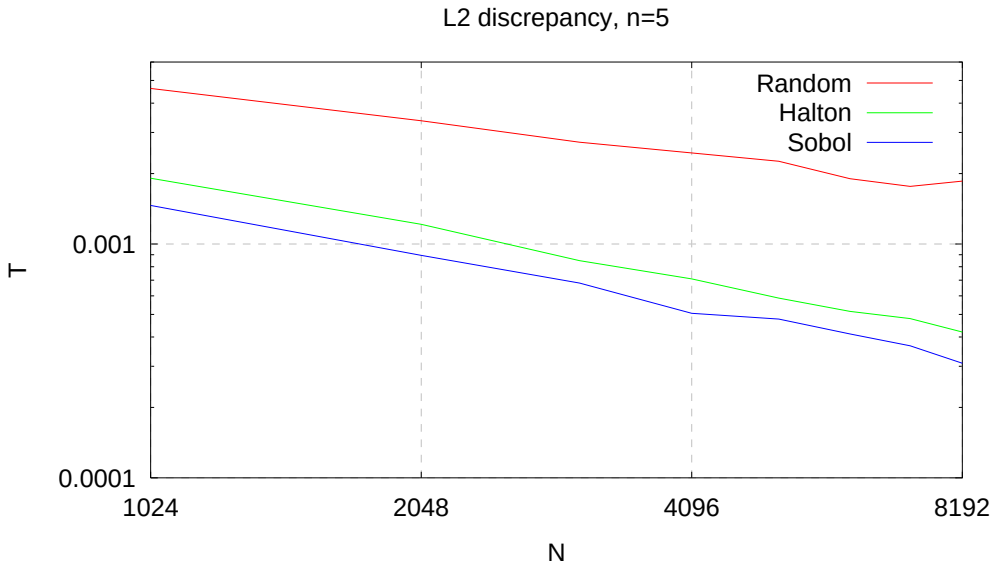

Fig. 5.1 Comparison of T_N^n discrepancy in 5 dimensions for the first 2^{13} points.

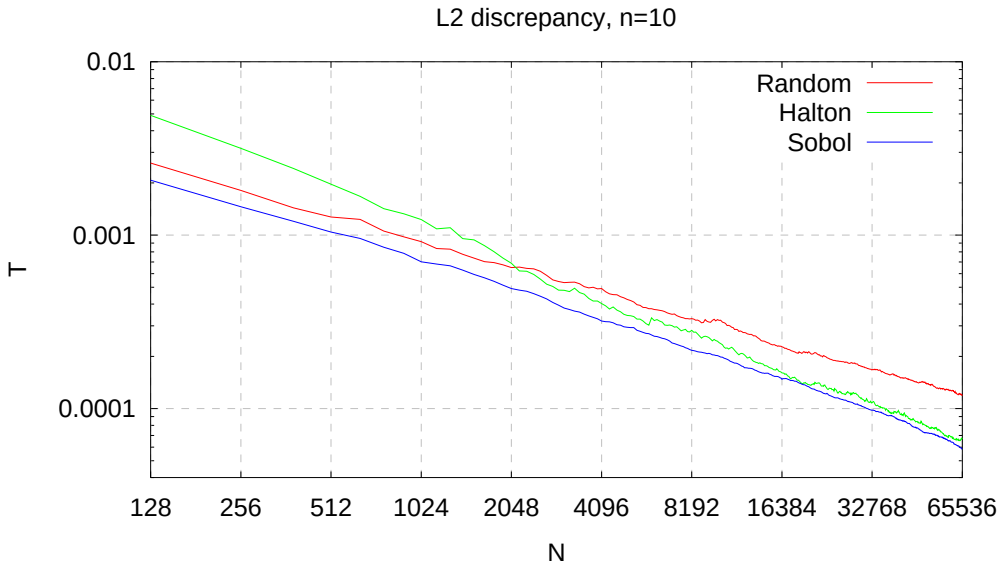

Fig. 5.2 Comparison of T_N^n discrepancy in 10 dimensions for the first 2^{16} points.

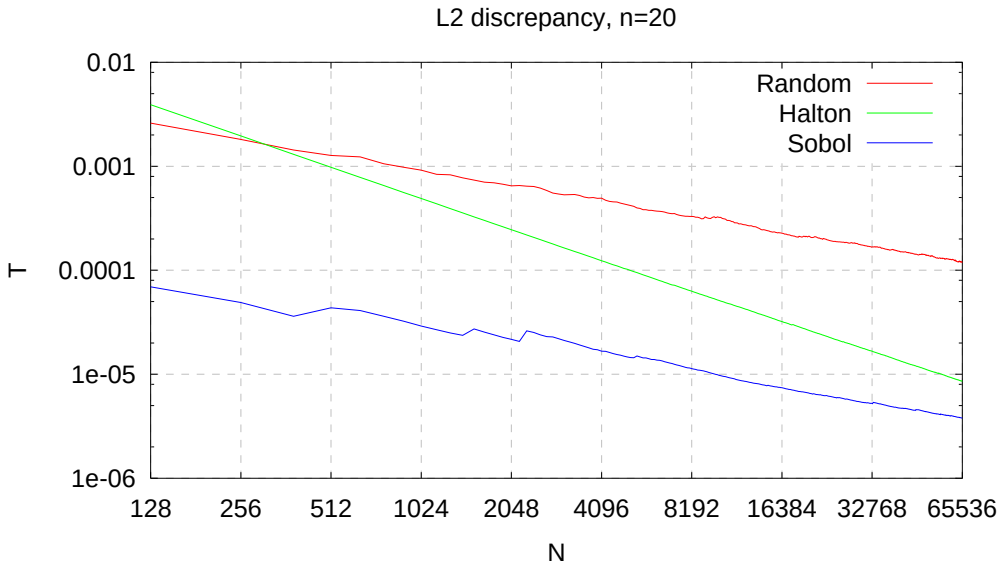

Fig. 5.3 Comparison of T_N^n discrepancy in 20 dimensions for the first 2^{16} points.

It can be seen that for very low dimensions (up to 5) both LDS are superior to random numbers. In higher dimensions ($d > 50$) the performance of Halton LDS becomes inferior to that of Sobol' LDS and random numbers.

6. Normally distributed variates

Many practical simulation problems require generation of normally distributed variates. For variate *y* with the probability distribution *p*(*y*) the fundamental transformation law of probabilities defines relationship between p(y) and a known distribution *p(x)* is

$$\left|p(y)\,dy\right| = \left|p(x)\,dx\right| \tag{6.1}$$

From (6.1)

$$p(y) = p(x)\left|\frac{dx}{dy}\right| \tag{6.2}$$

If p(x) is a uniform probability distribution then we can solve the differential equation (6.2) to obtain

$$x = \int_{-\infty}^{y} p(y')dy'$$

Or

$$x = F(y), \tag{6.3}$$

where $F(y)$ is the cumulative distribution function. Equation (6.3) can be solved with regard to variate *y*:

$$y(x) = F^{-1}(x). \tag{6.4}$$

Here *x* is uniformly distributed variate, has the required probability distribution *p*(*y*). Method (6.4) is known as the inverse transformation method. One of the difficulties associated with this method is for many important distributions the inverse function $F^{-1}(x)$ does not exist in analytical form.

In particular, the cumulative normal distribution function

$F(y) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{y} e^{-w^2/2}dw = \frac{1}{2}[1 + erf(\frac{y}{\sqrt{2}})]$ can not be expressed in terms of simple arithmetic operations (additions, subtractions, multiplications, and root extractions), and must be either computed numerically or otherwise approximated. The most efficient method based on the inverse transformation method is a method by Moro.

Transformation method (6.2) can be straightforwardly generalised to the multidimensional case.

One the simple and popular transformation methods for producing normal variates is the Box- Muller method. We generate two uniformly distributed variates γ_1, γ_2. Then using the following transformation

$$\zeta_1 = \sqrt{-2\ln\gamma_1}\cos 2\pi\gamma_2$$
$$\zeta_2 = \sqrt{-2\ln\gamma_1}\sin 2\pi\gamma_2$$

we obtain two normally distributed variates $\zeta_1,\ \zeta_2$. However, the Box- Muller method is not efficient in combination with LDS because it mixes well distributed low dimensions with less well distributed high dimensions. Therefore, the inverse transformation method should be used for practical purposes.

7. Tests for normality

Quality of sampling can be verified by various statistical analysis techniques: histogram, the normal probability plot. Smignov-Kolmogorov test and some other methods.

The histogram graphically shows the following:

1. center (i.e., the location) of the data;
2. spread (i.e., the scale) of the data;
3. skewness of the data;
4. presence of outliers; and
5. presence of multiple modes in the data.

These features provide strong indications of the proper distributional model for the data. The histogram is obtained by splitting the range of the data into equal-sized bins (called classes). Then for each bin, the number of points from the data set that fall into each bin is counted. That is

1. Vertical axis: Frequency (i.e., counts for each bin)
2. Horizontal axis: Response variable

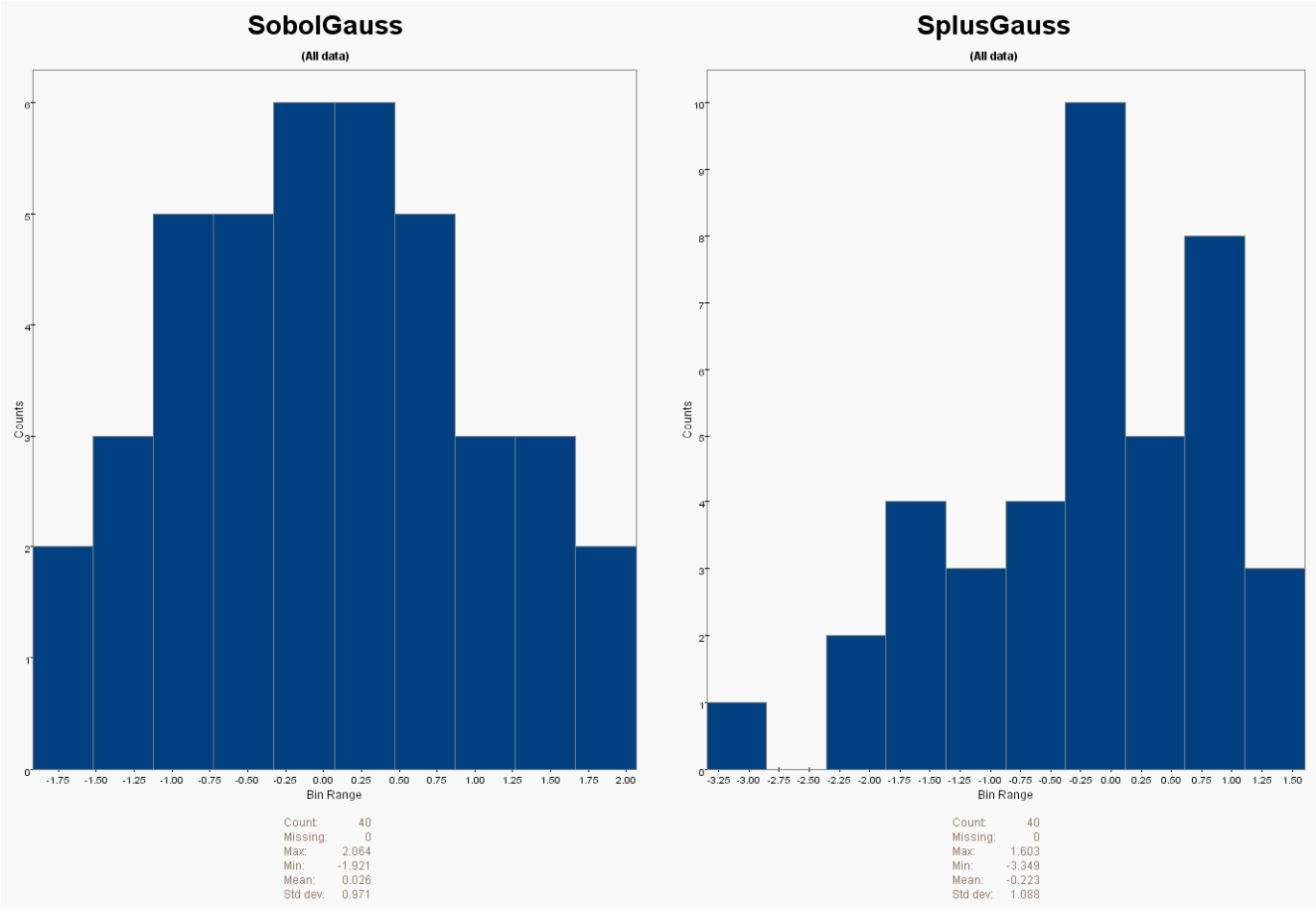

Fig. 7.1 The histograms obtained by using transformed Sobol' sequences (left figure) (BRODA, 2008) and S-Plus normal number generator rnorm(N,0,1) (right figure), (Insightful (2008, N=40

Fig. 7.1 shows that normal distribution generated by using Sobol' sequences has much better statistical properties than normal distribution based on random numbers.

The table 7.1 quantitatively illustrates the statistical properties of random numbers and Sobol' sequences.

Table 7.1. Comparison of normally distributed random variables using Sobol' Sequences and S-Plus rnorm(*N*,0,1) function.

Sobol – SobolNorm1D	S_Plus – normal(0,1)	S_Plus – normal(0,1)
Min: -3.297190e+000	Min: -2.93674364	Min: -3.708886e+000
1st Qu.: -6.752590e-001	1st Qu.: -0.68085430	1st Qu.: -6.825863e-001
Mean: -3.218717e-003	Mean: -0.02383565	Mean: 1.282197e-003
Median: -6.119700e-004	Median: -0.04290438	Median: -1.114428e-003
3rd Qu.: 6.721875e-001	3rd Qu.: 0.63453304	3rd Qu.: 6.803170e-001
Max: 3.097270e+000	Max: 2.71813014	Max: 3.814796e+000
Total N: 1024.000	Total N: 1024.000	Total N: 1.000000e+004
NA's : 0.000000e+000	NA's : 0.00000000	NA's : 0.000000e+000
Std Dev.: 9.994393e-001	Std Dev.: 0.98017813	Std Dev.: 1.001605e+000

Comparison of statistical measures shows that to achieve the same accuracy of generated distributions S-Plus rnorm(*N*,0,1) requires approximately 1000 times more points than the normal variate generator SobolNorm1D (see Appendix)

The normal probability plot is a graphical technique for assessing whether or not a data set is approximately normally distributed (Chambers, 1983). The data are plotted against a theoretical normal distribution in such a way that the points should form an approximate straight line. Departures from this straight line indicate departures from normality.
The normal probability plot is formed by:

1. Vertical axis: Ordered response values
2. Horizontal axis: Normal order statistic medians

The observations are plotted as a function of the corresponding normal order statistic medians which are defined as:

$$N(i) = G(U(i))$$

where $U(i)$ are the uniform order statistic medians (defined below) and G is the percent point function of the normal distribution. The percent point function is the inverse of the cumulative distribution function. That is, given a probability, we want the corresponding x of the cumulative distribution function..

The uniform order statistic medians are defined as follows:

$m(i) = 1 - m(n)$ for $i = 1$

$m(i) = (i - 0.3175)/(n + 0.365)$ for $i = 2, 3, ..., n\text{-}1$

$m(i) = 0.5^{(1/n)}$ for $i = n$

The further the points vary from this line, the greater the indication of departures from normality

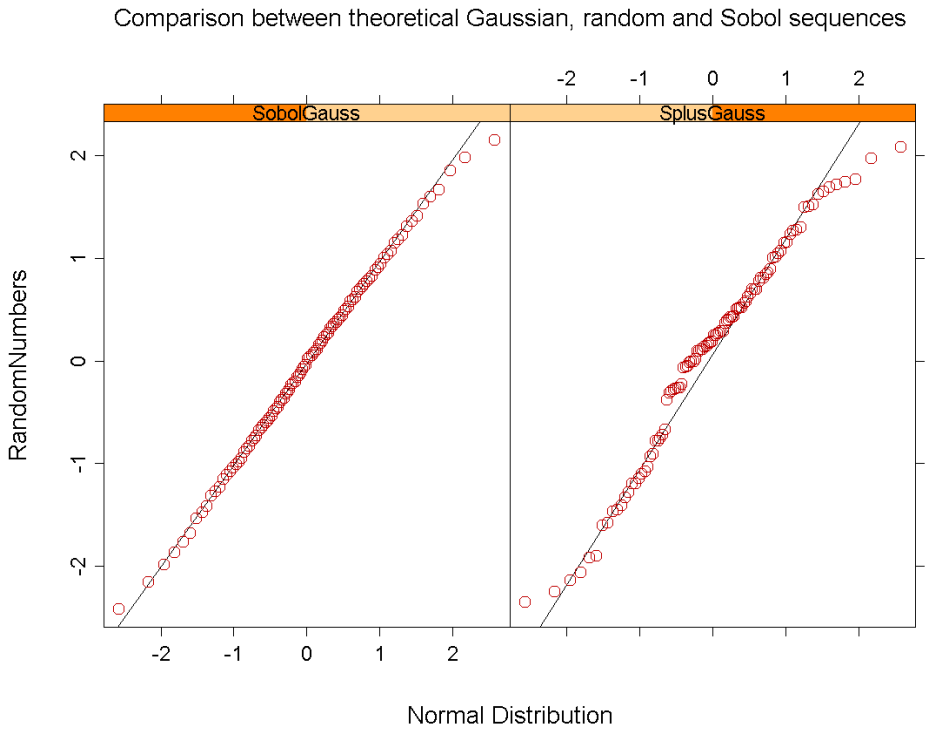

Fig. 7.2 Normal probability plots obtained by using SobolNorm1D generator (left figure) and S-Plus rnorm(*N*,0,1) generator (right figure).

Fig. 7.2 presents a comparison between normal probability plots generated using Sobol sequences and random number generators. Results clearly show that normal distribution obtained by using Sobol sequences is much closer to a straight line (we recall, that a straight line is a theoretical limit).

8.1. Evaluation of high dimensional integrals. Uniformly distributed sequences

The integrals of several popular tests functions in high dimensions ($d = 360$) are approximated by using random and Sobol' sequences (Fig. 8.1 – Fig. 8.3). For each example the root mean square error is approximated by the formula $cN^{-\alpha}$, $0 < \alpha < 1$.

The root mean square error is defined as

$$\varepsilon = \left(\frac{1}{K} \sum_{k=1}^{K} (I_d - I_N^k)^2 \right)^{1/2},$$

where K is a number of independent runs versus number of sampled points N. For the MC method all runs were statistically independent. For QMC integration for each run a different part of the Sobol' LDS was used. For all tests K=50.

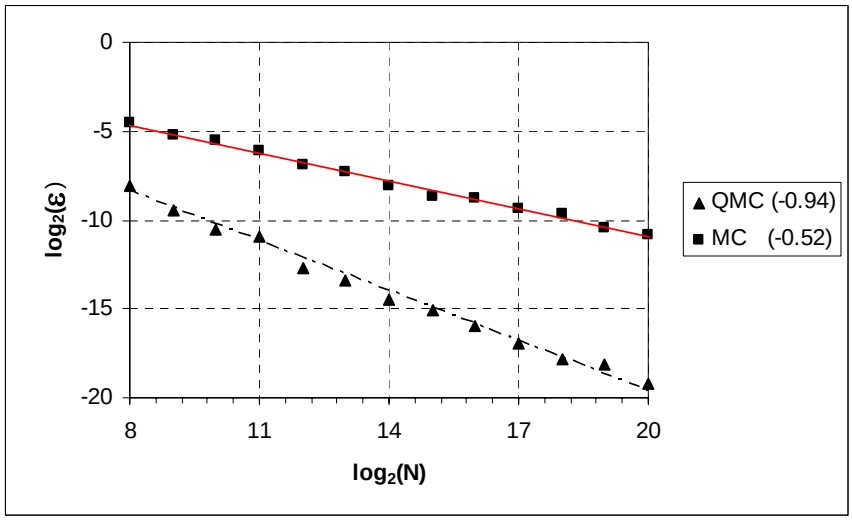

Fig. 8.1 Root mean square integration error for $f(x) = \sum_{j=1}^{d} (-1)^j \prod_{i=1}^{j} x_i$, where $d = 360$.

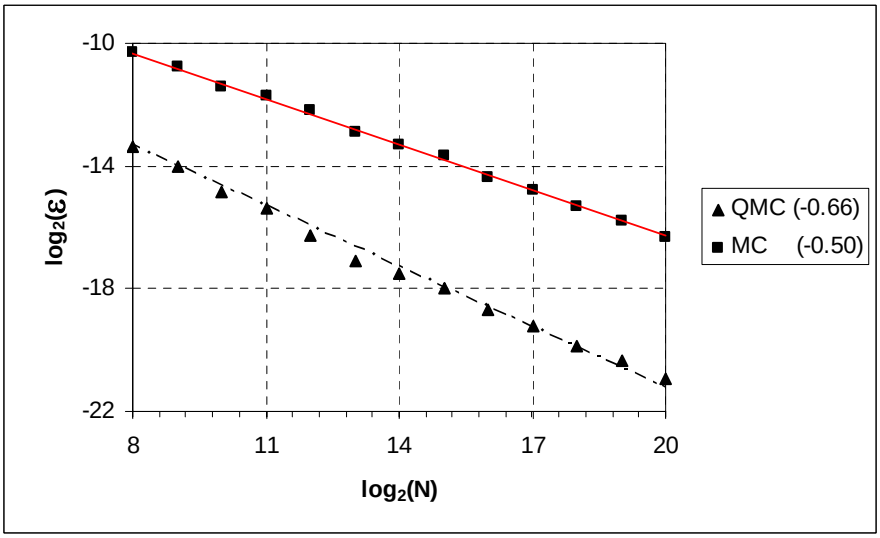

Fig. 8.2 Root mean square integration error for $f(x) = \prod_{i=1}^{d} (d - x_i)/(d - 0.5)$, where $d = 360$.

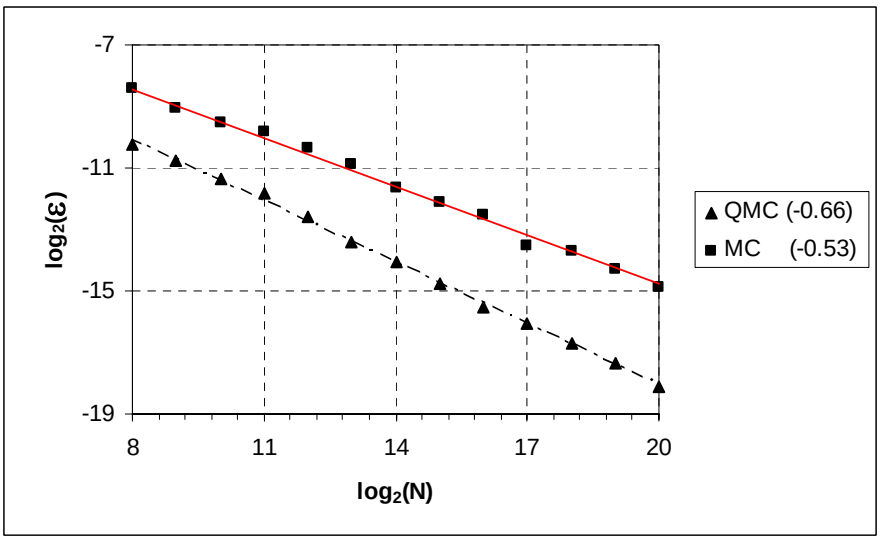

Fig. 8.3 Root mean square integration error for $f(x)=\prod_{i=1}^{d}(1+1/d)x_i^{1/d}$, where $d=360$.

Presented results show that QMC integration is superior to that of MC both in terms of the rate of convergence (larger α) and in the absolute value of the integration error (constant c).

8.2 Evaluation of high dimensional integrals. Normally distributed sequences

Consider Keister's test integral:

$$I_d = \int_{\mathbb{R}^d} \cos(\|x\|) e^{-\|x\|^2} dx ,$$

where $\|x\|$ denotes the Euclidian norm in $\mathbb{R}^d$. Integral I_d can be further transformed

$$\begin{aligned} I_d &= 2^{-d/2} \int_{\mathbb{R}^d} \cos(\|y\|/\sqrt{2}) e^{-\|y\|^2/2} dy \\ &= \pi^{d/2} \int_{\mathbb{R}^d} \cos(\|y\|/\sqrt{2}) \frac{e^{-\|y\|^2/2}}{(2\pi)^{d/2}} dy \\ &= \pi^{d/2} \int_{[0,1]^d} \cos\left(\sqrt{\sum_{i=1}^{d} (\phi^{-1})^2(t_i)/2}\right) dt, \end{aligned} \tag{8.1}$$

Here ϕ is the cumulative normal distribution function with mean 0 and variance 1:

$$\phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{u} e^{-s^2/2} ds, u \in [-\infty, \infty].$$

We consider case of d=25, the exact value of I_d (d=25) is -1356914. Evaluation of (8.1) involves generations of 25-dimensional vector of normal variates. The root mean square error defined in chapter 8.1 is presented in Fig. 8.4.

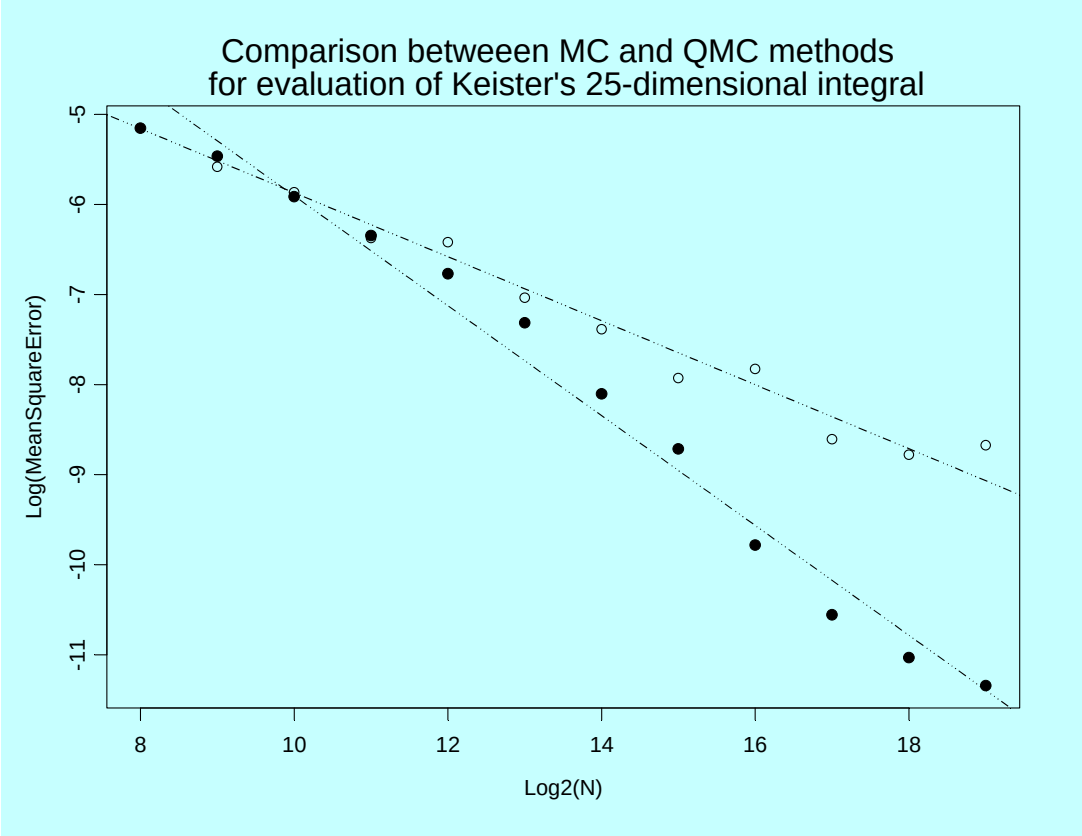

Fig. 8.4 Root mean square integration error for Keister's 25-dimentional integral. QMC – solid circles, MC – clear circles.

Results show superior performance of the QMC approach based on Sobol sequences. The convergence rate for the QMC method decreases as $1/N$, while for the MC method this rate is only $1/N^{1/2}$.

Results were obtained with S-Plus 8.0 for Windows (Insightful, 2008). BRODA's SobolSeq generators were interfaced with S-Plus.

8.3 Comparison between MC and QMC methods for evaluation of quantiles

Consider evaluation of quantiles for distibution $f(x) = \sum_{i=1}^{n} x_i^2$, where x_i are independent standard normal variates, n is the dimension.

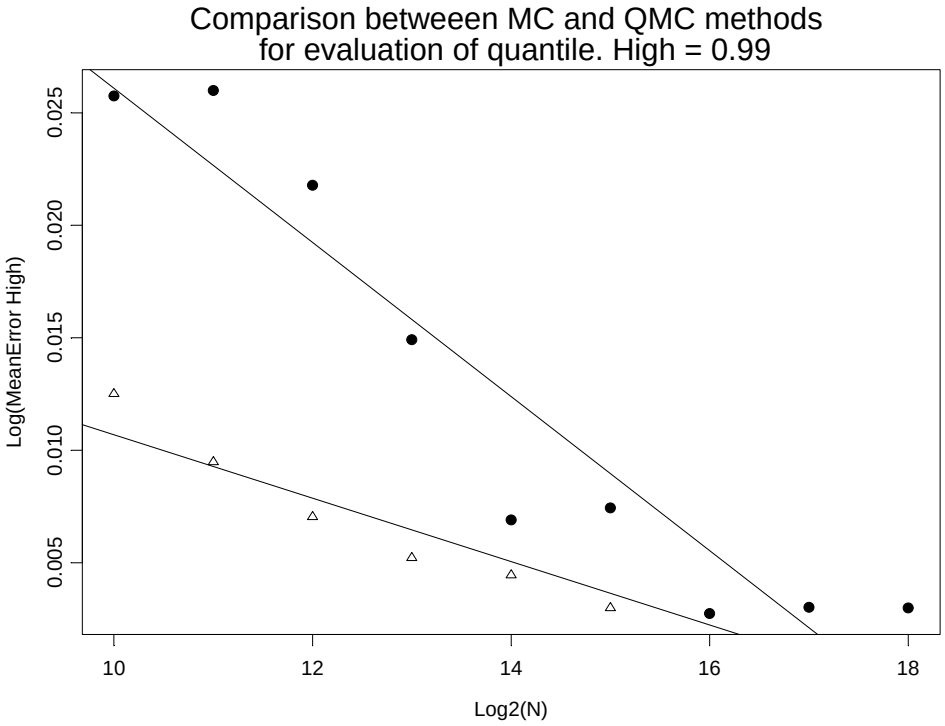

Fig. 8.5 Comparison betweeen MC and QMC methods for evaluation of quantiles. Low quantile = 0.01, dimension n = 10. QMC – triangles, MC – solid circles

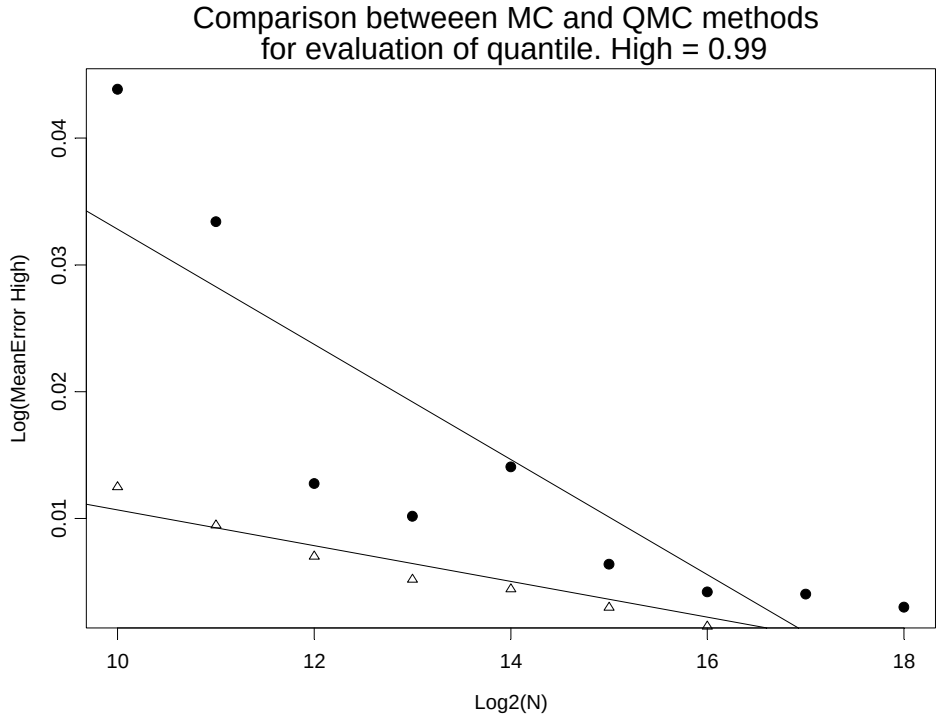

Fig. 8.6 Comparison betweeen MC and QMC methods for evaluation of quantiles. High quantile = 0.99, dimension n = 10. QMC – triangles, MC – solid circles

Fig 8.5 and 8.6 present RMSE of evaluation of quantiles versus the number of sampled point for dimension n = 10. RMSE is calculated by

$$\varepsilon = \left(\frac{1}{L} \sum_{l=1}^{L} (Q_{est} - Q_N^{(l)})^2 \right)^{1/2} .$$

The number of independent runs L = 10, Q_{est} are estimates obtained with $N = 10^6$ points: Q_{est} low =2.55, Q_{est} high =23.20. $Q_N^{(l)}$ is an estimated value of quantile on l-th run for N-points. Results clearly show a superior convergence of the QMC method.

It is clear from Fig 8.5 and 8.6 that the QMC approach gives a higher convergence rate than the MC method.

Results were obtained with S-Plus 8.0 for Windows (Insightful, 2008). BRODA's SobolSeq generators were interfaced with S-Plus.

References

Bratley P., Fox B.L., Niederreiter H., Implementation and tests of low- discrepancy sequences, *ACM Transactions on Modeling and Computer Simulation* 2 (3) (1992) 195-213.

BRODA: http://www.broda.co.uk (2009)

Chambers, John, William Cleveland, Beat Kleiner, and Paul Tukey, Graphical Methods for Data Analysis, Wadsworth (1983)

Dupire B., Savine A., Dimension Reduction and Other Ways of Speeding Monte Carlo Simulation" in Dupire (Eds.), Monte Carlo - Methodologies and Applications for Pricing and Risk Management, Risk Books, 1998, pp.51-63 (1998)

Forsythe G.E., Malcolm M.A., Moler C.B. *Computer Methods for Mathematical Calculations*. Englewoods Cliffs, NJ:Prentice-Hall (1977).

Glassserman P., Monte Carlo methods in Financial Engineering, Springer (2003)

Hung W.Y., Kucherenko S., Samsatli N., Shah N. A Flexible and Generic Approach to Dynamic Modelling of Supply Chains. *J. of the Operational Research Society*, 55, 801-813 (2004).

Insightful. http://www.insightful.com/support/documentation.asp (2008)

Jackel P., Monte Carlo methods in finance, John Wiley & Sons (2002)

Kucherenko S., Sytsko Yu. Application of deterministic low-discrepancy sequences to nonlinear global optimization problems. *Comp Optimization and Applications,* 30, 3, 287-318 (2005).

Morokoff W.J., Caflisch R.E., Quasi Monte-Carlo integration, *J. Comput. Physics* 122, 218-230 (1995).

Niederreiter, H. , Random number generation and Quasi-Monte Carlo methods, SIAM (1992).

Sobol' I.M. Uniformly distributed sequences with additional uniformity properties. USSR *Computational Mathematics and Mathematical Physics*, 16, 5, 236-242 (1976).

Sobol' I.M., Turchaninov S, Levitan V, Shukhman B.V. Quasirandom sequence generators. Keldysh Inst. Appl. Maths RAS Acad. Sci., Moscow, 24 p. (1992).

Sobol' I.M., On quasi- Monte-Carlo integrations, *Mathematics and Computers in Simulation* 47 103-112. (1998)