Sampling Techniques

Sergei Kucherenko

Imperial College London, SW7 2AZ, UK

Outline

Part I.

1. What is the optimal way to arrange N points in D-dimensions

2. Low discrepancy sequences and their properties

3. Stratified Sampling: Latin Hypercube sampling

4. Monte Carlo and Quasi Monte Carlo integration methods

Part II

1. ANOVA decomposition

2. Sobol’ Sensitivity Indices

3. Effective dimensions

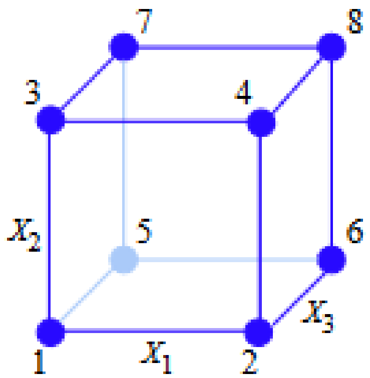
4. Classification of functions

Factorial Designs

Objective: to develop a good understanding of a model with *n* parameters (dimensions)

The simplest factorial design: $N=2^n$ (two levels for each factor)

E1: Dimension $n=3$, $N=2^3= 8$



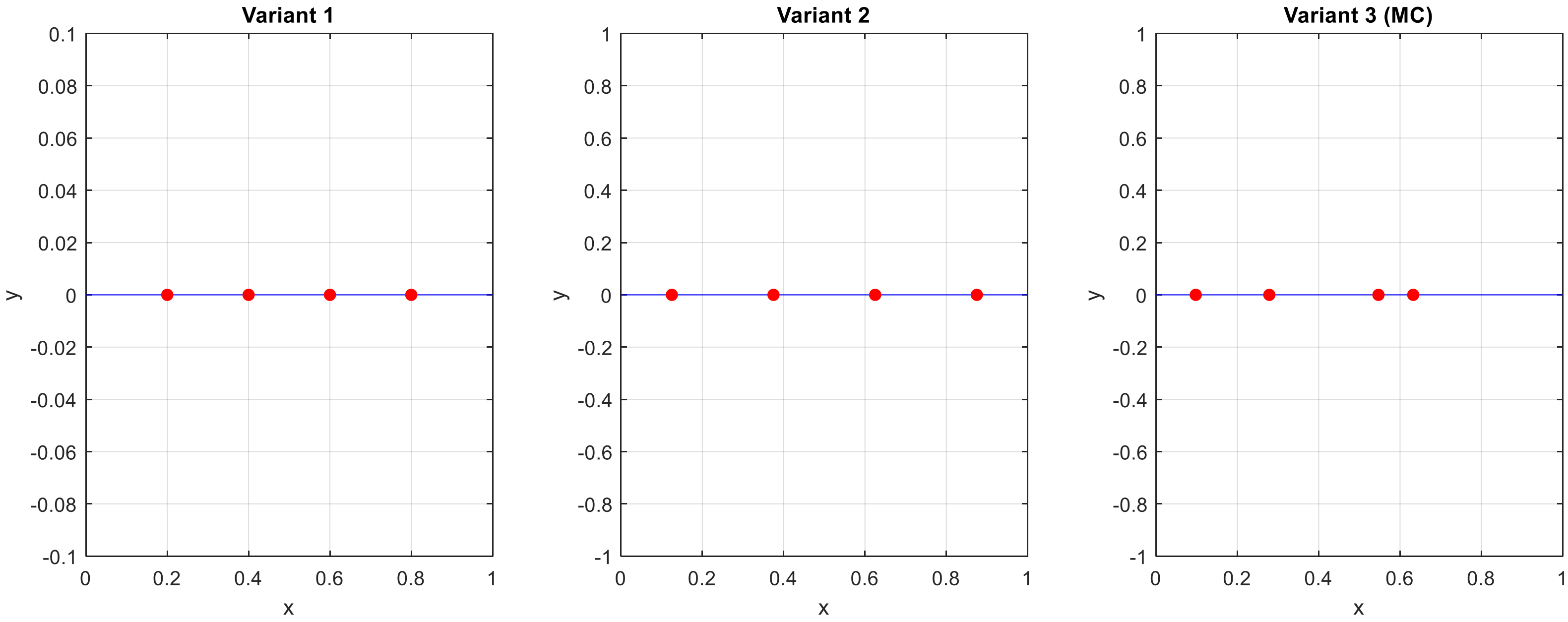
Design Point	Factor settings		
	X_1	X_2	X_3
1	-1	-1	-1
2	+1	-1	-1
3	-1	+1	-1
4	+1	+1	-1
5	-1	-1	+1
6	+1	-1	+1
7	-1	+1	+1
8	+1	+1	+1

E2: Simulation with $n=100$ parameters. A single replication of this experiment would take over 40 million years on the "Roadrunner" (the fastest computer until recently), even if each of the $N=2^{100} \approx 10^{30}$ simulation runs consisted of a single machine instruction!

"The curse of Dimensionality"

What is the optimal way to arrange 4 points in 1D?

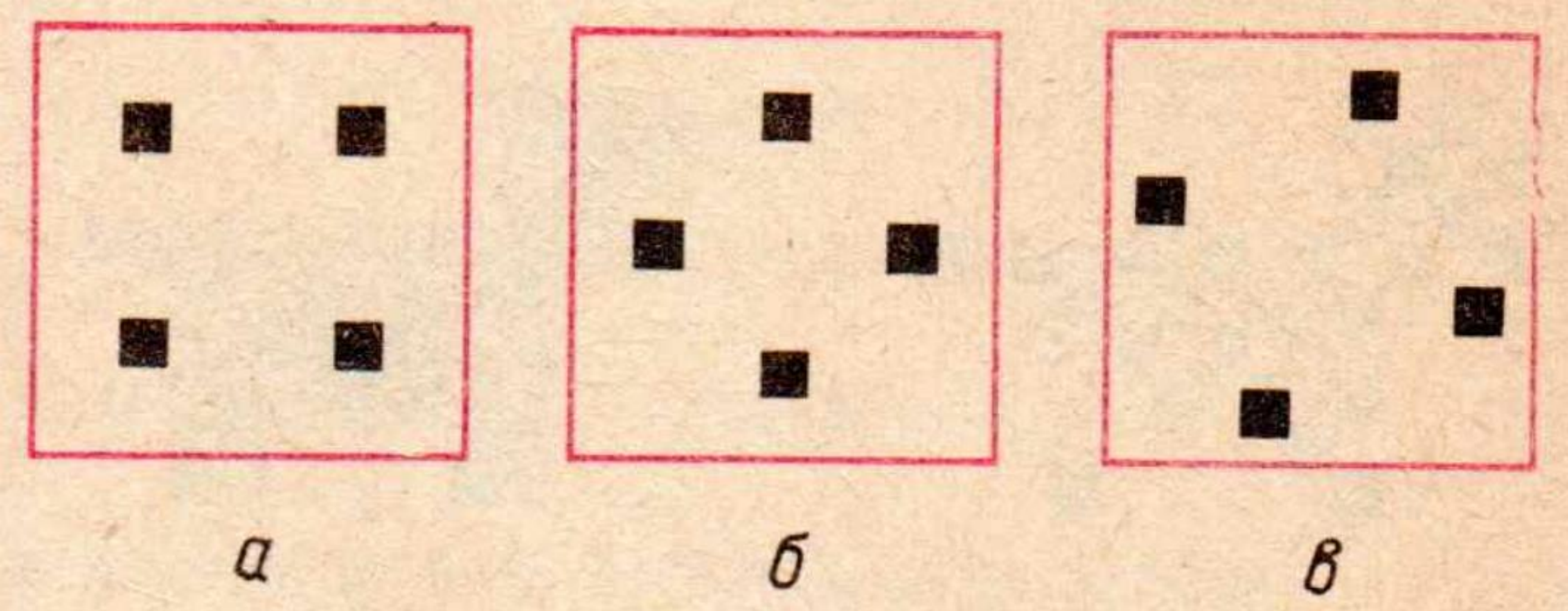
Objective: to develop a good understanding of a model with 1 parameter



?

What is the optimal way to arrange 4 points in 2D?

Objective: to develop a good understanding of a model with 2 parameters

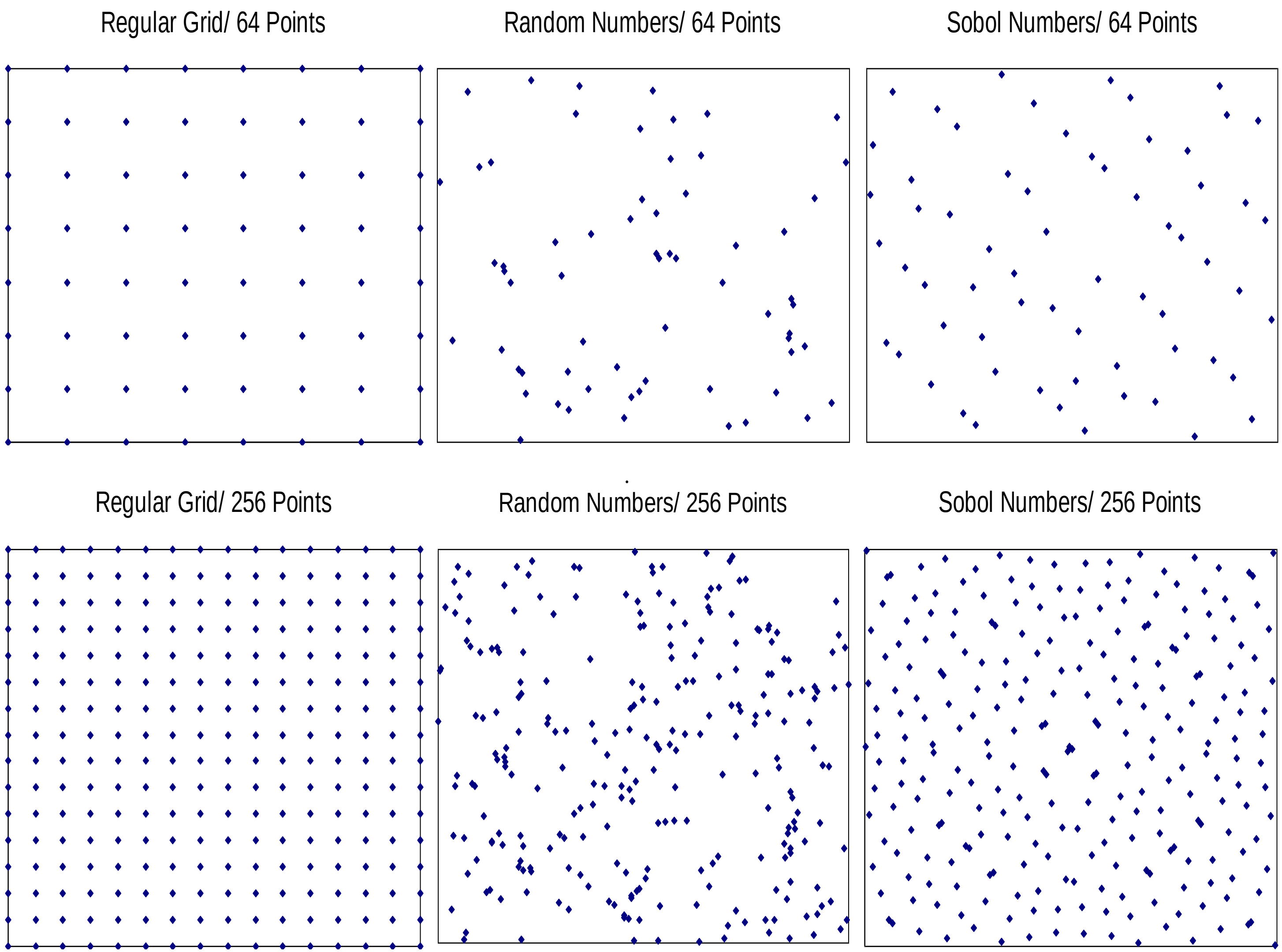


I. Sobol. Uniform samplings in high dimensional cubes. 1985

?

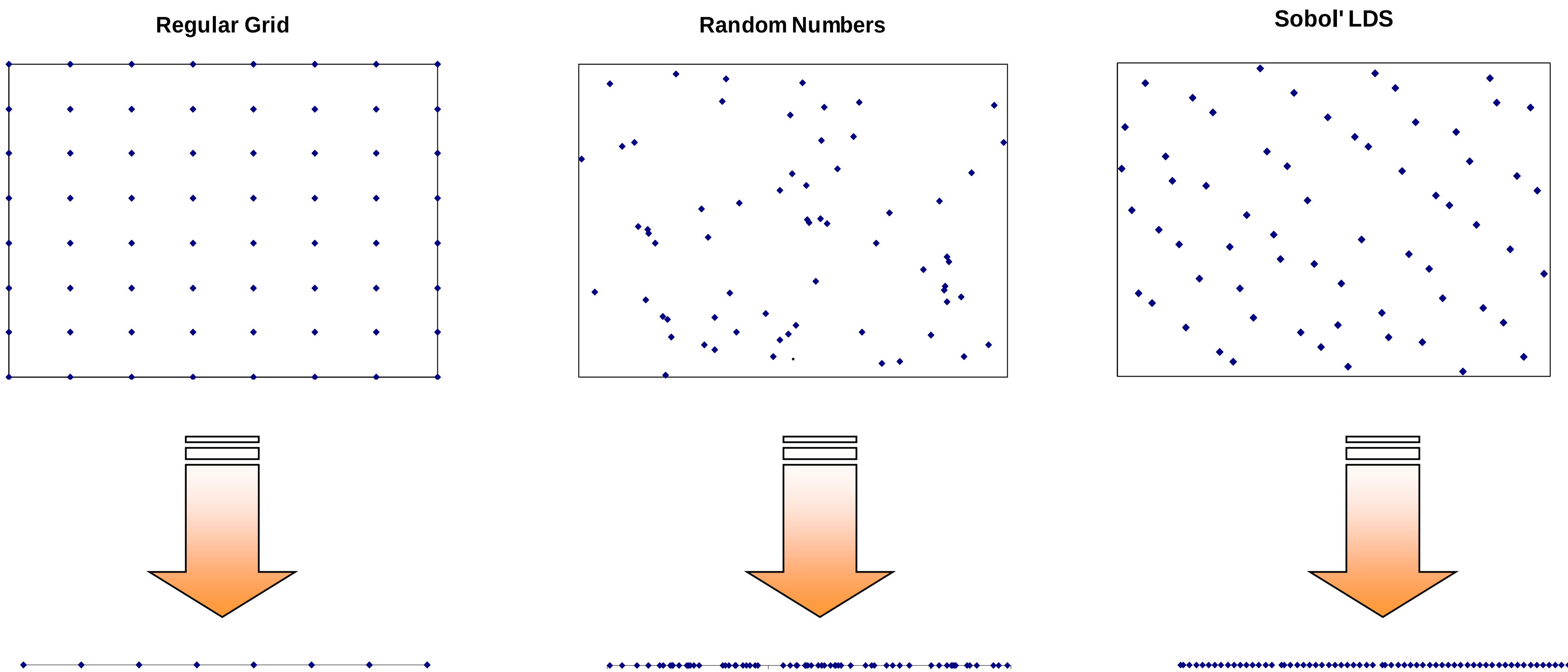
One million $ question: What is the optimal way to arrange *N* points in *D*-dimensions ?

Sobol' Sequences vrs Random numbers and regular grid



Unlike random numbers, successive Sobol' points "know" about the position of previously sampled points and fill the gaps between them

Projections of Different 2D sequences to 1D

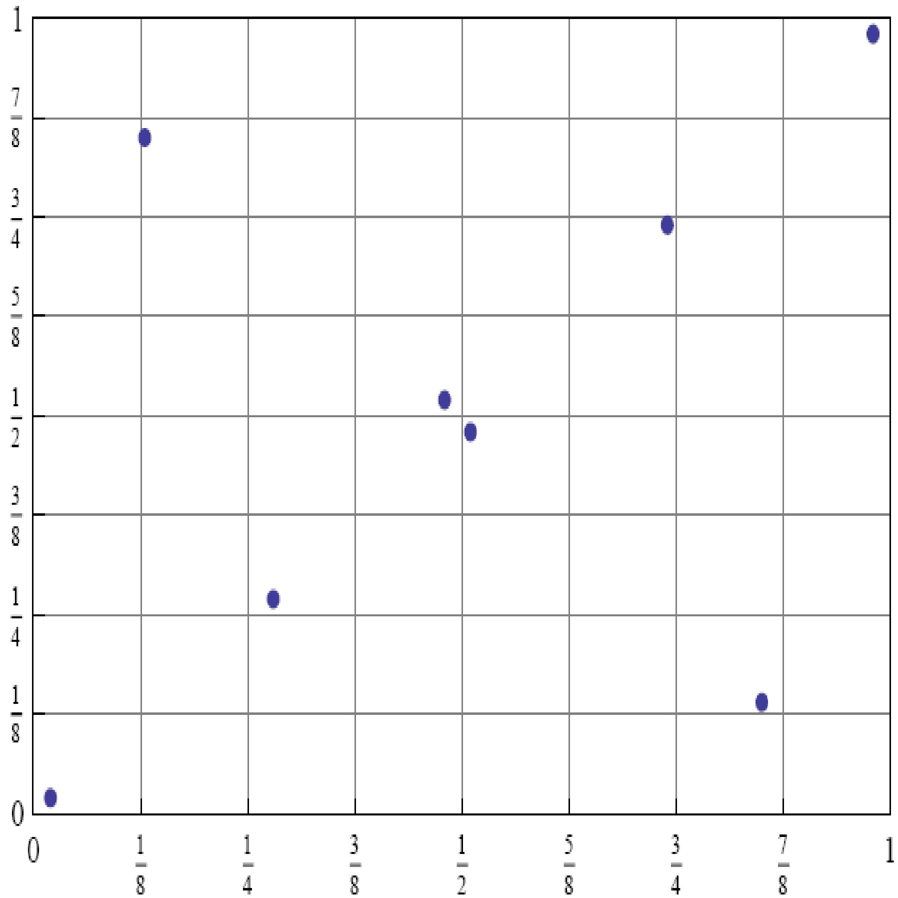


Shadow effect
(64 -> 8 points)

Clustering

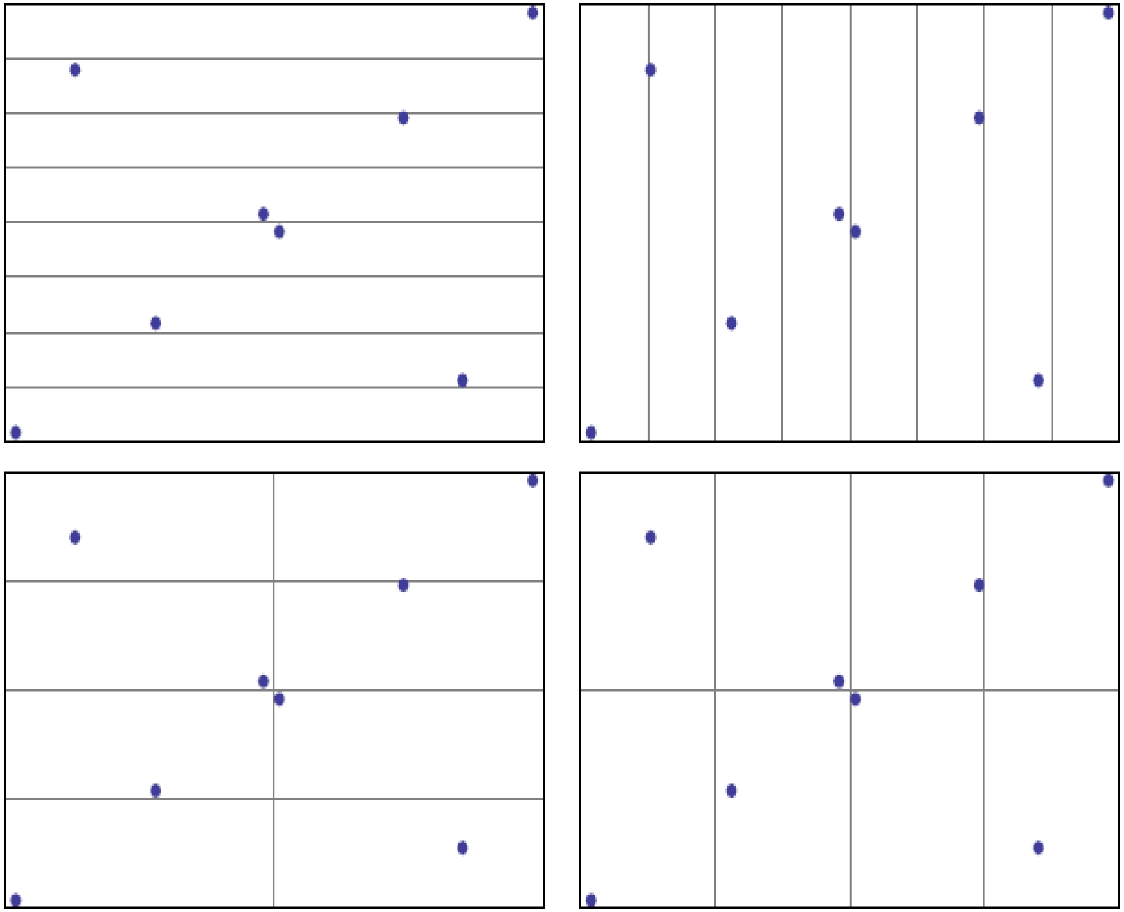
Uniform distribution

Sobol’ numbers: the most uniform way to allocate points. I.



Sample 8 points. If we divide the unit square into elementary subrectangles with area 1/8, then each subrectangle will have exactly one point of the sequence. This gives us the most uniform way to allocate 8 points to 8 rectangles.

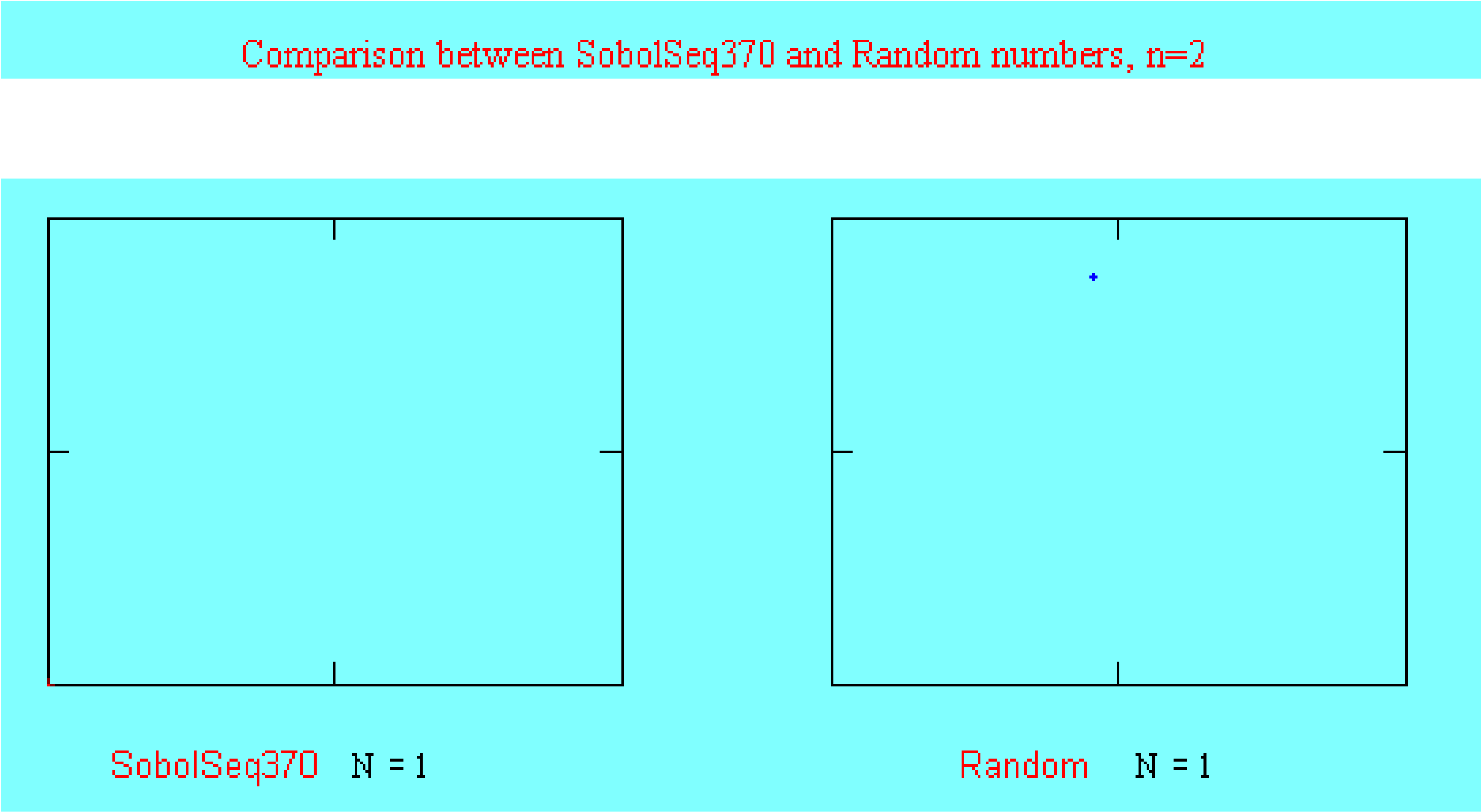
Sobol' numbers : the most uniform way to allocate points. II.



Sample N=8 points: divide the unit square into elementary subrectangles with area 1/8, then each subrectangle will have exactly one point of the sequence ->

The most uniform way to allocate 8 points to 8 rectangles.

Comparison between Sobol sequences and random numbers



https://www.youtube.com/watch?v=QnJQpXrOs34

https://www.youtube.com/watch?v=TmrobpYC8Bs

How to construct Sobol' sequence ?
Van der Corput sequence

Van der Corput sequence:

Number i written in base b: $i = (\cdots a_4 a_3 a_2 a_1 a_0)_b$

In the decimal system: $i = \sum_{j=0}^{m} a_j \, b^j$, $0 \leq a_j \leq b - 1$

Reverse the digits and add a radix point to obtain a number within the unit interval:
$y = (0.a_0 a_1 a_2 a_3 a_4 \cdots)_b$

In the decimal system: $h(i;b) = \sum_{j=0}^{m} \frac{a_j}{b^{j+1}}$

Example:
$i = 4$, base $b = 2$

$4 = 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = (100)_2 = (a_2 a_1 a_0)_2$

$(0.001)_2 \rightarrow 0 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = 1/8$

Van der Corput sequence

Reverse the digits and add a radix point to obtain a number within the unit interval:

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

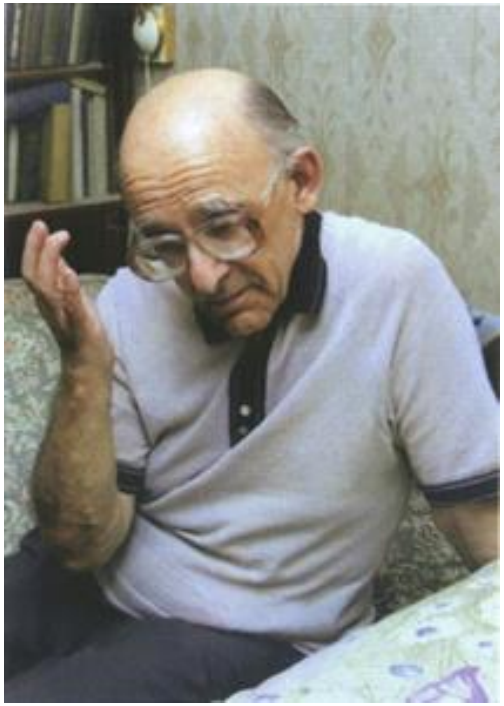
In the decimal system: $h(i;b) = \sum_{j=0}^{m} \frac{a_j}{b^{j+1}}$

i	i Binary	$\mathrm{h}_2(i)$ Binary	$\mathrm{h}_2(i)$
0	0	0	0
1	1	0.1	$1/2$
2	10	0.01	$1/4$
3	11	0.11	$3/4$
4	100	0.001	$1/8$
5	101	0.101	$5/8$

Sobol' (LP_tau) Sequences

Sobol' sequence−A permutation of Van der Corput sequence in each dimension.
There are many degrees of freedom.
Sobol imposed some constraints -> Sobol' Sequences have:

1. Best uniformity of distribution as N goes to infinity.
2. Good distribution for fairly small initial sets of points (low N).
3. A very fast computational algorithm.

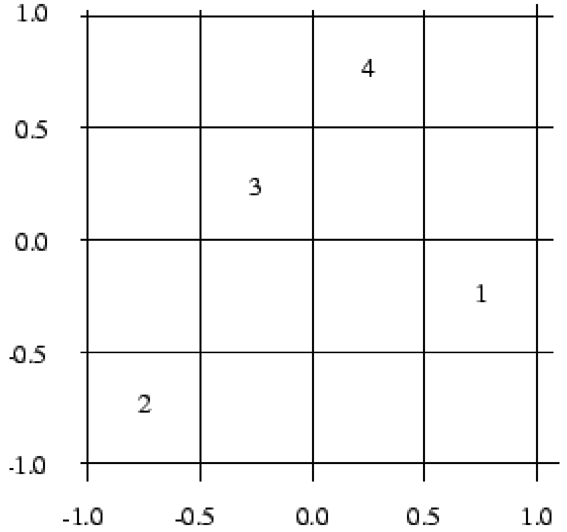


Ilya Sobol'
Moscow ,
1926 –

Latin Hypercube sampling

A square grid containing sample positions is a Latin square if (and only if) there is only one sample in each row and each column

In n-dimensions: Generate N points, dimension-by-dimension, using 1D stratified sampling with 1 value per stratum, assigning them randomly to get precisely one point in each stratum



Latin Hypercube sampling. Definitions

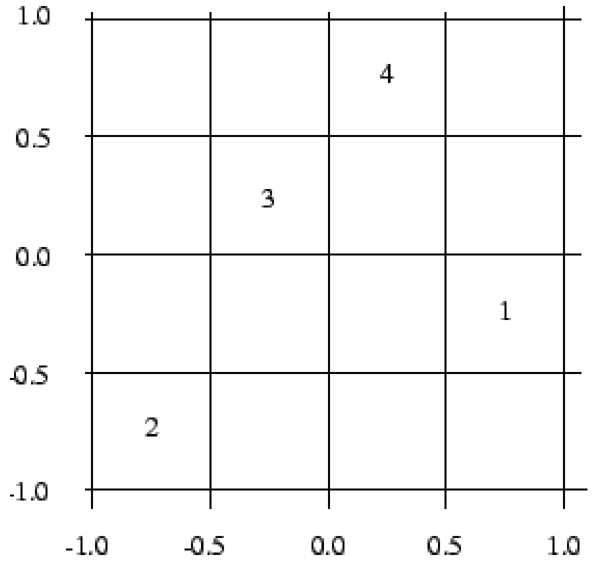
$\{\pi_k\}, k = 1, \ldots, n$ – independent random permutations of a set $\{1, \ldots, N\}$

(N! possible permutations)

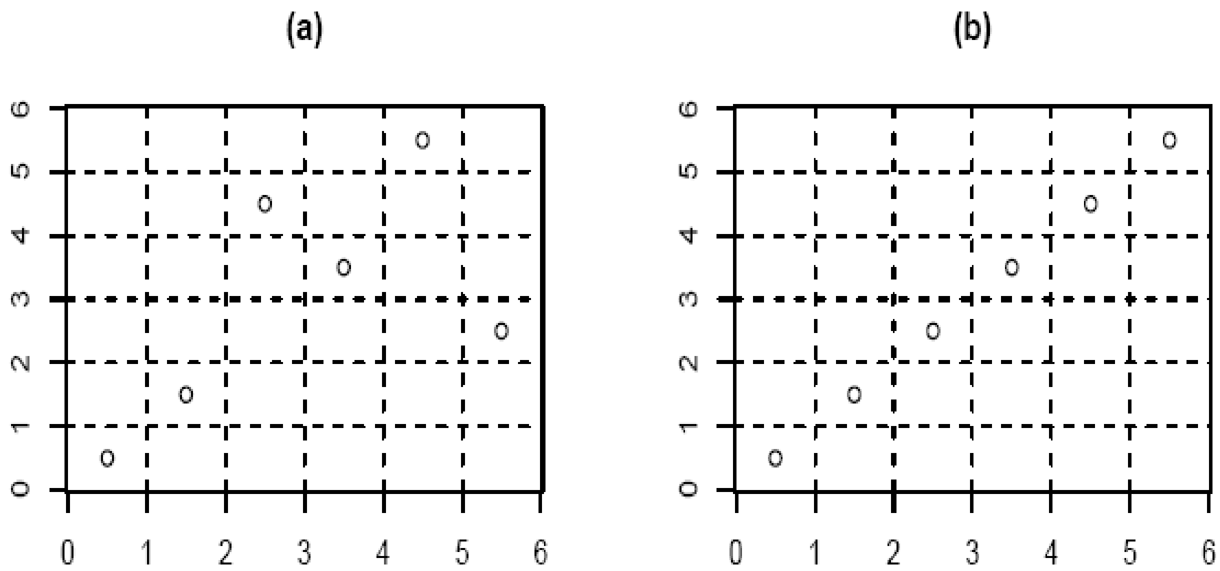
LHS coordinates: $x_i^k = \dfrac{\pi_k(i) - 1 + \xi_i^k}{N}, i = 1, \ldots, N, k = 1, \ldots, n$

$\xi_i^k \sim \mathrm{U}(0,1)$

LHS is built by superimposing well stratified one−dimensional samples.



Deficiencies of LHS sampling



1) Space is badly explored (a)

2) Possible correlation between variables (b)

3) Points can not be sampled sequentially

$\Rightarrow$Not suited for integration

LHS does not provide good uniformity properties in the whole volume of a n−dimensional unit hypercube.

Sobol’ Sequences have “LHS property” built in (plus many other good properties)

Deterministic integration methods in high dimensions

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

Deterministic integration method of *p*–order,

k points in each direction: $N_{\mathrm{n}} = k^{\mathrm{n}}$

Error: $\varepsilon = \mathrm{O}(\mathrm{k}^{-\mathrm{p}})$, $N_{\mathrm{n}} = \mathrm{O}(1/\varepsilon)^{\mathrm{n/p}}$.

Estimate: $\varepsilon = 10^{-2}, p = 2, \mathrm{dimension}\ n = 50 \rightarrow$

$N_{\mathrm{n}} = 10^{50} \approx$ the total number of particles in the universe

$\rightarrow \mathrm{I[f]}$ is impossible to evaluate !

"The curse of Dimensionality"

Monte Carlo integration methods

$$I[f] = E[f(\vec{x})],\ \vec{x} \text{ - } R.V.$$

Monte Carlo estimator : $I_N[f] = \dfrac{1}{N}\sum_{i=1}^{N} f(\vec{\xi}_i)$

$\{\vec{\xi}_i\}$ – is a sequence of random points in H^n

Error: $\varepsilon_N = |I[f] - I_N[f]|$

$\varepsilon_N = (\,E(\varepsilon^2))^{1/2} = \dfrac{\sigma(f)}{N^{1/2}} \rightarrow$

Convergence does not depend on dimensionality *n* but it is slow:

If we want to increase accuracy 10 times,

we need to increase *N* - 100 times

Quasi Monte Carlo methods

Replace random numbers with "low discrepancy points" (quasi-random numbers) $P_1,\ldots,P_k,\ldots$, which are deterministic and uniformly distributed (u.d.) in H^n

$$\lim_{N\to\infty}\frac{1}{N}\sum_{k=1}^{N} f(P_k) = \int_{H^n} f(x)dx$$

.

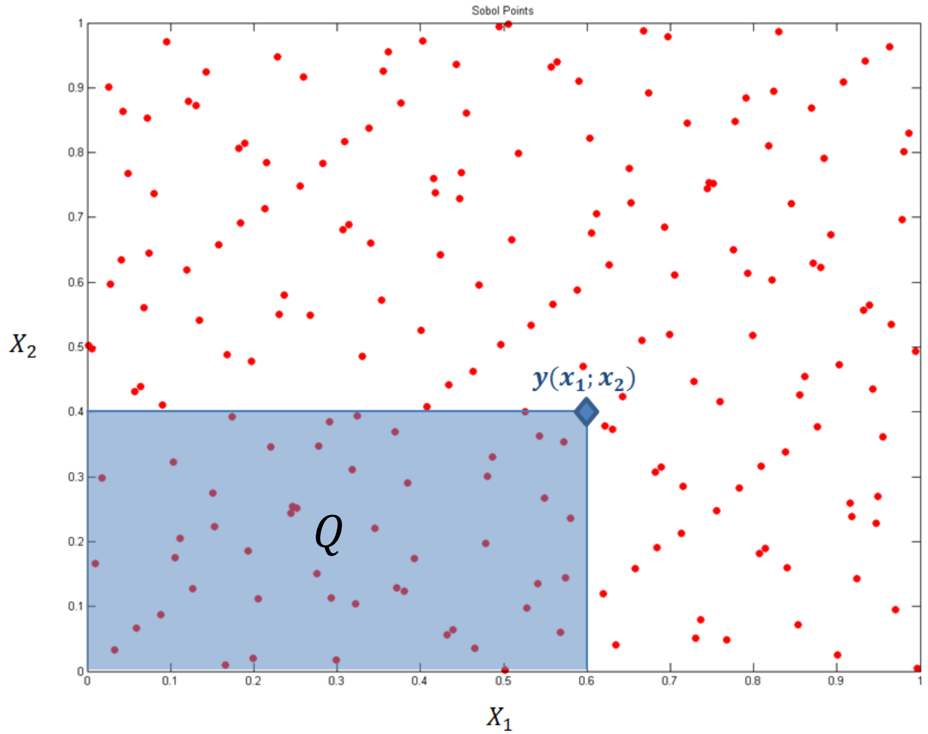
Deterministic version of Monte Carlo is called Quasi Monte Carlo (QMC).

QMC error:

$$\varepsilon_{QMC} = |I[f] - I_N[f]| = \left|\frac{1}{N}\sum_{k=1}^{N} f(P_k) - \int_{H^n} f(x)dx\right|$$

How to estimate ?

How to quantify "optimal" (or uniform) sampling?



Discrepancy is a measure of deviation from uniformity

Consider a subcube $Q(\vec{y}) \in H^{\mathrm{n}}$, $Q(\vec{y}) = [0, y_1) \times [0, y_2) \times \ldots \times [0, y_n)$,
$m(Q)$ – volume of Q

$$D_N = \sup_{Q(\vec{y}) \in H^n} \left| \frac{N_{Q(\vec{y})}}{N} - m(Q) \right|$$

$D_N \leq c(n) \dfrac{(\ln \ N)^n}{N}$ – Low discrepancy sequences (LDS)

Discrepancy for random and LDS

$$D_N = \sup_{Q(\vec{y}) \in H^n} \left| \frac{N_{Q(\vec{y})}}{N} - m(Q) \right|$$

Random sequences: $D_N \to (\ln\ \ln\ N)/N^{1/2} \sim 1/N^{1/2}$

Low discrepancy sequences (LDS): $D_N \leq c(n) \dfrac{(\ln\ N)^n}{N}$

QMC Convergence:

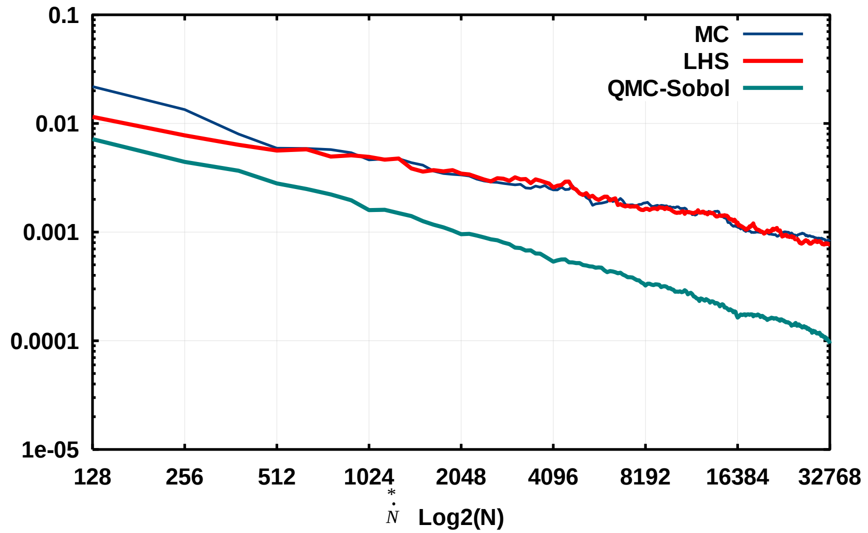
$$\varepsilon_{QMC} = |I[f] - I_N[f]| \leq V(f) D_N,$$

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

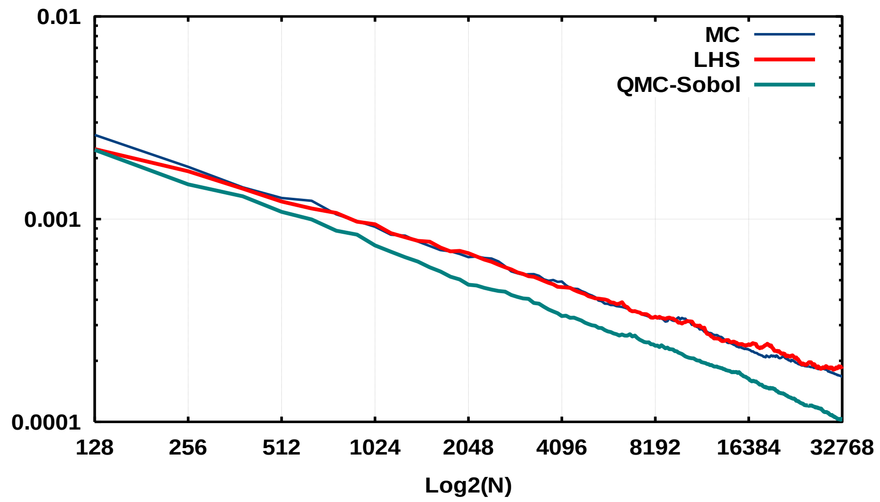
Assymptotically $\varepsilon_{QMC} \sim O(1/N)$ → much higher than $\varepsilon_{MC} \sim O(1/\sqrt{N})$

Comparison of Discrepancy

n=10

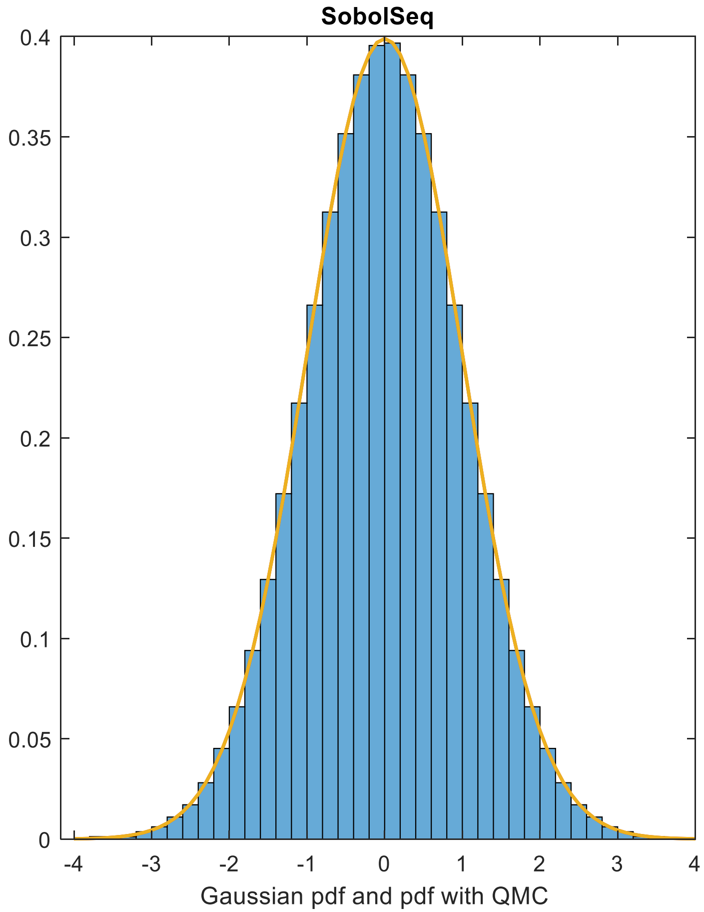


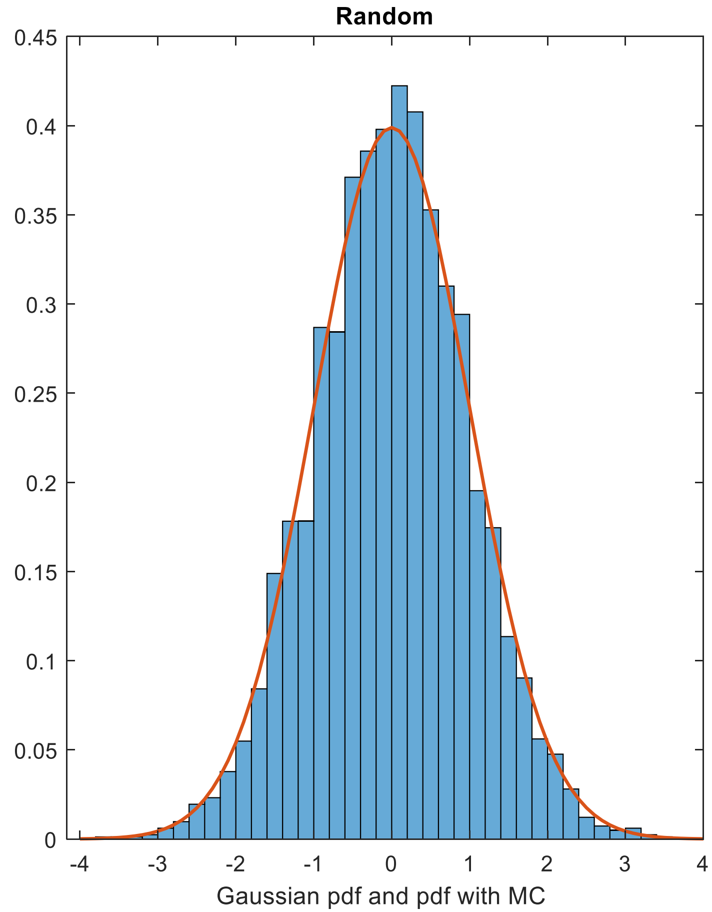
n=50



QMC shows much smaller discrepancy than MC and LHS

Normal distributions obtained using Sobol' Sequences (QMC) and random numbers (MC)





Gaussian pdf (solid lines) and pdf with QMC sampling (left) and MC sampling (right), *N*=4096 points

Sobol ' sequences are widely used in finance

PATENTING

Patently Ridiculous?

The subject of patenting applications of mathematical algorithms raises a number of questions, philosophical, ethical and also basic issues of good business sense. **Dan Tudball** reports.

There was a joke doing the rounds on the internet a few years ago, regarding Microsoft's announcement that it intended to patent '1' and '0', on the basis that the firm had been using the binary system since the mid 1970s. Of course the convoluted gag tapped into attitudes about Microsoft, which haven't changed noticeably, and was less a commentary on the workings of the US Trademark and Patent Office. After all, the word 'patent' can still conjure up images of daffy old inventors visiting dusty oak-lined offices to get a stamp of approval from desk-bound boffins peering over their pince-nez. The world of patenting seems, to the general population, a distant bureaucratic domain, only encroaching upon the day to day as part of some seriously antiquated sales pitch. But that rather cozy image has been under intensified scrutiny as patent offices around the world have attempted to drag themselves into the 21st century. As an adjunct to the controversy that still rages over software patenting, the point at which math meets the business of finance has also produced its own debate.

D. Tudball, Patently Ridiculous? Wilmott, 2003

US Pat. No. 5,940,810, Columbia University. Estimation Method And System For Complex Securities Using Low Discrepancy Deterministic Sequences. 1997

Columbia University: FINDER software (Sobol' Sequence generator for maximum dimension 360).

Cost – 100,000 USD

References

Kucherenko S., Albrecht D., Saltelli A. , Exploring multi-dimensional spaces: a Comparison of Latin Hypercube and Quasi Monte Carlo Sampling Techniques, 2015 arXiv:1505.02350

I.M. Sobol', D. Asotsky, A. Kreinin, S. Kucherenko. ''Construction and Comparison of High-Dimensional Sobol' Generators'', 2011, Wilmott Journal, Nov, pp. 64-79

Part II

ANOVA decomposition

Sobol' Sensitivity Indices

Effective dimensions

Classification of functions

ANOVA decomposition and Sobol' Sensitivity Indices

$$Y = f(X)$$
$$X = (X_1, X_2, \dots, X_n) \in H^n$$
$$0 \le X_i \le 1$$

$f(X)$ is L2 integrable

ANOVA decomposition is unique:

$$Y = f(X) = f_0 + \sum_{i=1}^{n} f_i(X_i) + \sum_i \sum_{j>i} f_{ij}\left(X_i, X_j\right) + \dots + f_{1,2,\dots,k}(X_1, X_2, \dots, X_n),$$

$$\int_0^1 f_{i_1 \dots i_s}(X_{i_1},, \dots, X_{i_s})\, dX_{i_p} = 0, \forall p, 1 \le p \le s, \rightarrow \int_0^1 f_{i_1 \dots i_s} f_{i_1 \dots i_l}\, dX_{i_p} dX_{i_l} = 0, \forall i_p \ne i_l$$

Variance decomposition:

$$D = \sum_{i=1}^{n} D_i + \sum_i \sum_{j>i} D_{ij} + \cdots + D_{1,2,\dots,n}$$

Sobol' SI:

$$1 = \sum_{i=1}^{n} S_i + \sum_{i<j} S_{ij} + \sum_{i<j<l} S_{ijl} + \dots + S_{1,2,\dots,n}$$

Sobol' Sensitivity Indices

Consider two sets of variables : $x = (y, z)$

ANOVA decomposition:

$$f(\boldsymbol{x}) = f(y,z) = f_0 + g_1(y) + g_2(z) + g_{12}(y,z) \qquad (1)$$

$$\int g_1(y)dy = \int g_2(z)dz = \int g_{12}(y,z)dy = \int g_{12}(y,z)dz = 0 \quad (2)$$

We square and integrate (1) and because of (2), we obtain decomposition of the total variance D:

$$D = D_y + D_z + D_{yz}$$

Define

$$D_y^{tot} = D_y + D_{yz} = D - D_z$$
$$D_z^{tot} = D_z + D_{yz} = D - D_y$$
$$S_y = \frac{D_y}{D}, \qquad S_y^{tot} = \frac{D_y^{tot}}{D}$$

Evaluation of Sobol' Sensitivity Indices

Straightforward use of ANOVA decomposition requires
2^n integral evaluations – not practical !
There are efficient formulas for evaluation of Sobol' SI (Sobol' 2001):

$$S_y = \frac{1}{D}\left[\int_0^1 f(y,z')^2\, dydzdz' - f_0^2\right]$$

$$S_y^{tot} = \frac{1}{2D}\int_0^1 [f(y,z) - f(y',z)]^2\, dydzdz'$$

$$D = \int_0^1 f^2(y,z)\, dydz - f_0^2$$

Evaluation is reduced to high-dimensional integration by MC/QMC methods.

III. Classification of functions

Effective dimensions

Let $|u|$ be a cardinality of a set of variables u.

Define Sobol' indices $S_u = D_u / D$.

The effective dimension of $f(x)$ in the superposition sense is the smallest integer d_S such that $\sum_{0<|u|\le d_S} S_u \approx 1$

It means that $f(x)$ is almost a sum of d_S-dimensional functions.

The function $f(x)$ has the effective dimension in the truncation sense d_T if $\sum_{u\subseteq\{1,2,\dots,d_T\}} S_u \approx 1$

Important property: $d_S \le d_T$

Example 1 : $f(x) = \sum_{i=1}^{n} x_i$, $x_i \sim U[0,1] \to d_S = 1,\ d_T = n$

Example 2 : $f(x) = f_0 + \sum_{i=1}^{n} f_i(x_i) + \sum_i \sum_{j>i} f_{ij}(x_i, x_j) \to d_S = 2,\ d_T = n$

Classification of functions

Type A. Variables are not equally important

$$\frac{S_y^T}{n_y} >> \frac{S_z^T}{n_z} \leftrightarrow d_T << n$$

Type B,C. Variables are equally important

$$S_i \approx S_j \leftrightarrow d_T \approx n$$

Type B.
Dominant low order terms

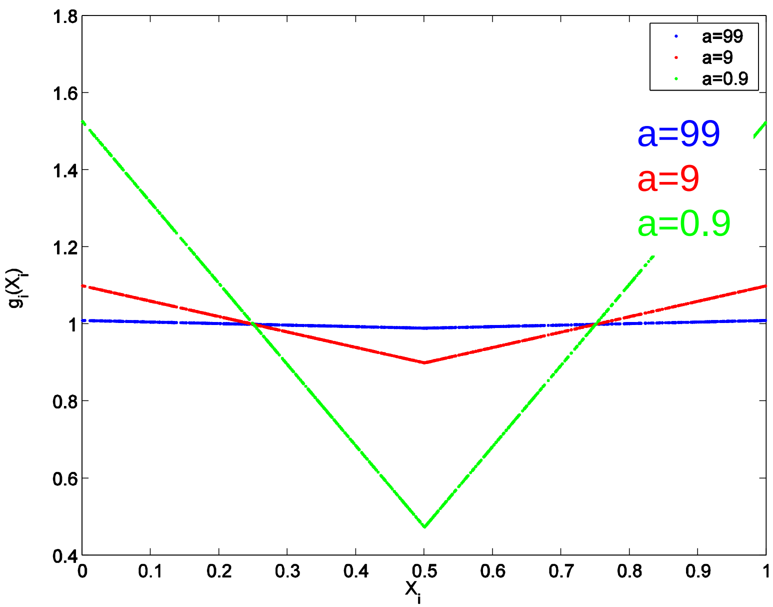
$$\sum_{i=1}^{n} S_i \approx 1 \leftrightarrow d_S << n$$

Type C. Dominant higher order terms

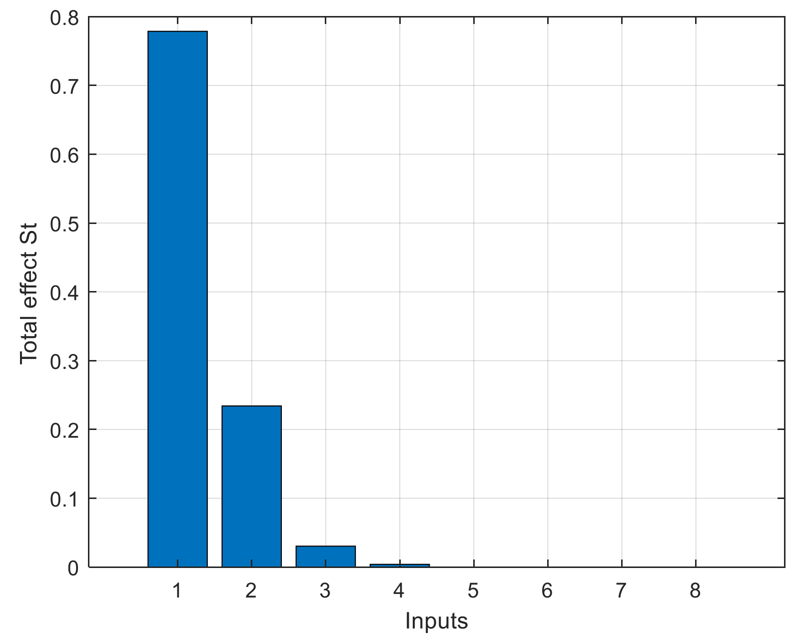
$$\sum_{i=1}^{n} S_i << 1 \leftrightarrow d_S \approx n$$

Type A function. Example

$$f(x) = \prod_{i=1}^{n} g_i\,(X_i), \qquad g_i(X_i) = \frac{|4X_i - 2| + a_i}{1 + a_i}\,,\, a_i \geq 0\,,\, X_i \sim U[0{,}1]$$



Function $g(X, a)$ at different values of *a*



Values of Sobol' indices, *n*=8,

a_1 =0, a_2 =1, a_3 =4.5, a_4 =9, $a_{5,..,8}$ =99.

Only tree first inputs are important.

Function depends only a small number of leading variables: $d_T = 3 \;<< n = 8$

Classification of functions. Efficiencies of MC/QMC/LHS

Function type	Description	Relationship between S_i and S_i^{tot}	d_T	d_S	QMC is more efficient than MC	LHS is more efficient than MC
A	A few dominant variables	$S_y^{tot}/n_y >> S_z^{to} / n_z$	$<< n$	$<< n$	Yes	No
B	No unimportant subsets; only low-order interaction terms are present	$S_i \approx S_j, \forall\, i, j$ $S_i / S_i^{tot} \approx 1, \forall$ i	$\approx n$	$<< n$	Yes	Yes
C	No unimportant subsets; high-order interaction terms are present	$S_i \approx S_j, \forall\, i, j$ $S_i / S_i^{tot} << 1, \forall\, i$	$\approx n$	$\approx n$	No	No

How to monitor convergence of MC and QMC calculations ?

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

The root mean square error (RMSE) is defined as

$$\varepsilon_N = \left(\frac{1}{K} \sum_{k=1}^{K} (I - I_N^k)^2 \right)^{1/2}$$

K is a number of independent runs, I_N^k - is the MC/QMC approximation of I at N on k-th iteration.

The root mean square error empirically can be approximated by the formula:

$$\varepsilon_N \sim cN^{-\alpha}, 0 < \alpha < 1$$

MC: $\alpha = 1/2$

QMC: $1/2 \leq \alpha \leq 1$,

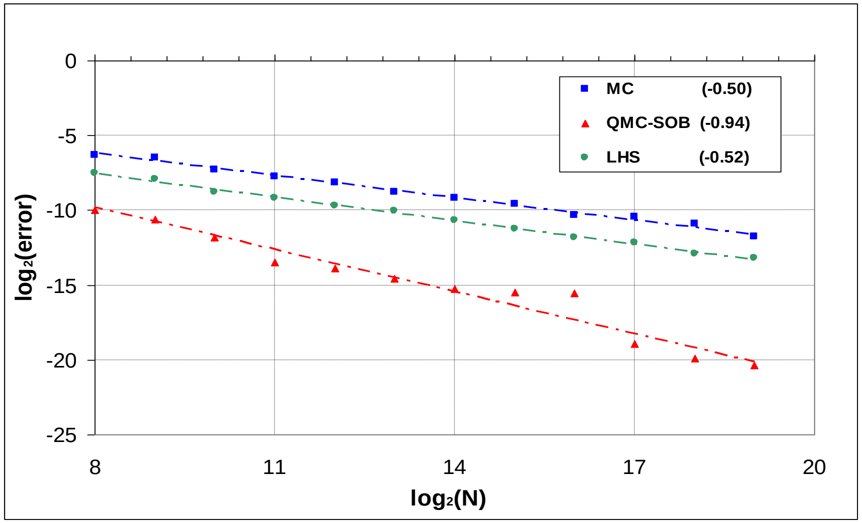
α close to 1 for problems with low effective dimensions !

Integration error vs. N. Type A

$d_T << n$

(a)

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

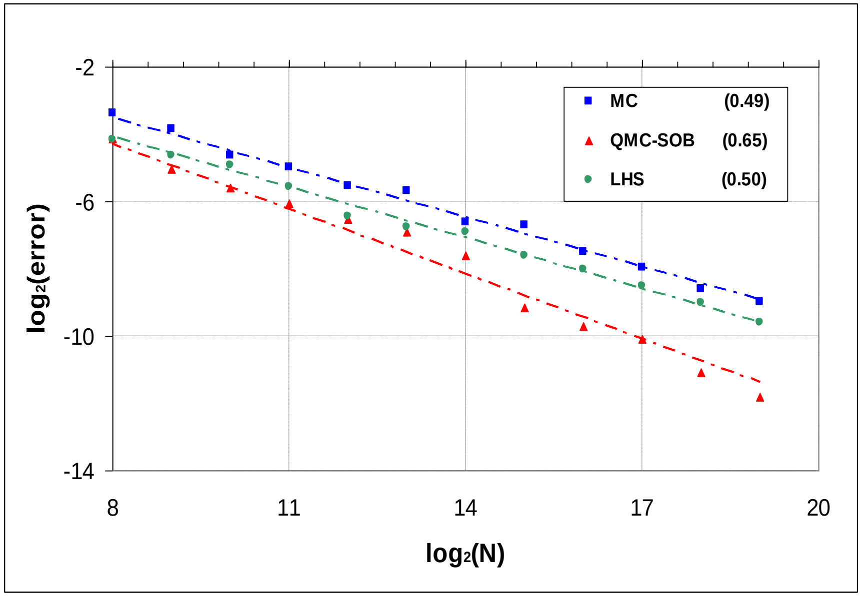


$$\sum_{i=1}^{n} (-1)^i \prod_{j=1}^{i} x_j$$

$$n = 360$$

$$\varepsilon \sim N^{-\alpha}, \ 0 < \alpha < 1$$

(b)



$$\prod_{i=1}^{n} \frac{|4x_i - 2| + a_i}{1 + a_i},$$

$$a_1 = a_2 = 0,$$

$$a_3 = \ \ldots \ = a_{100} = 6.52$$

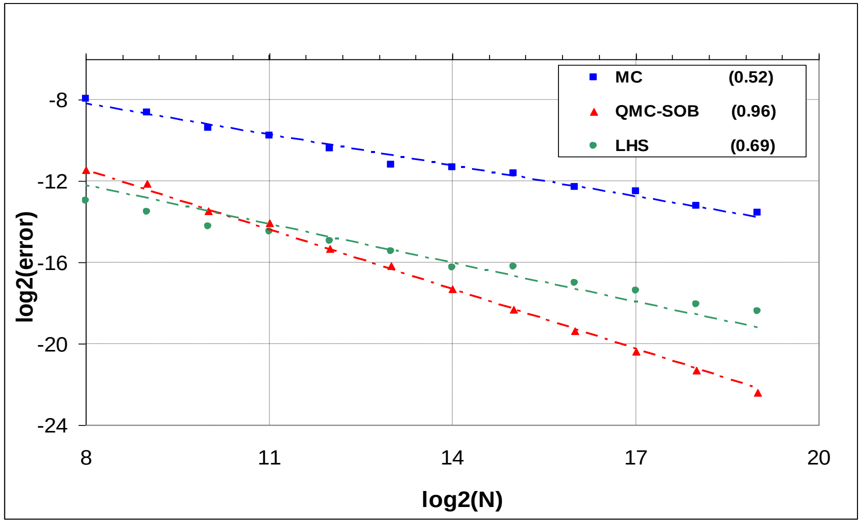
$$n = 100$$

Integration error vs. N. Type B

$$\sum_{i=1}^{n} S_i \approx 1 \leftrightarrow d_S << n$$

Dominant low order indices

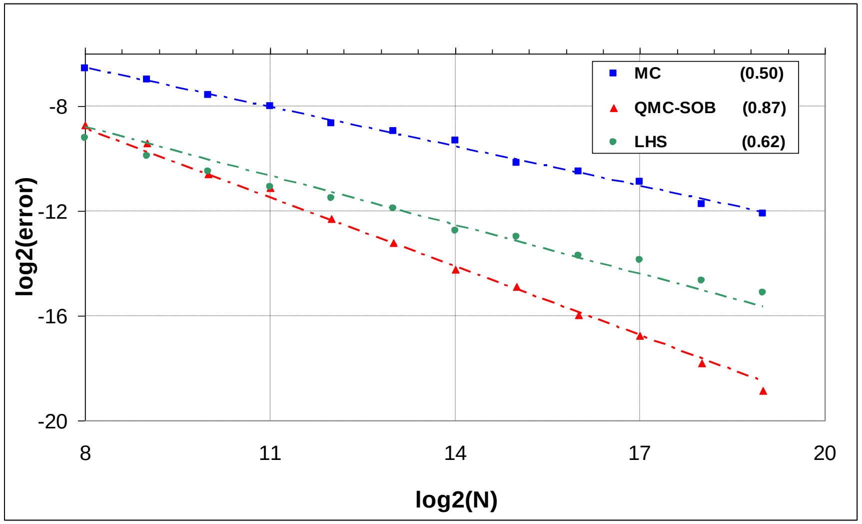
(a)



$$f(x) = \prod_{i=1}^{n} \frac{n - x_i}{n - 0.5}$$

$$n = 360$$

(b)



$$f(x) = \prod_{i=1}^{n} (1 + 1/n) x_i^{1/n}$$

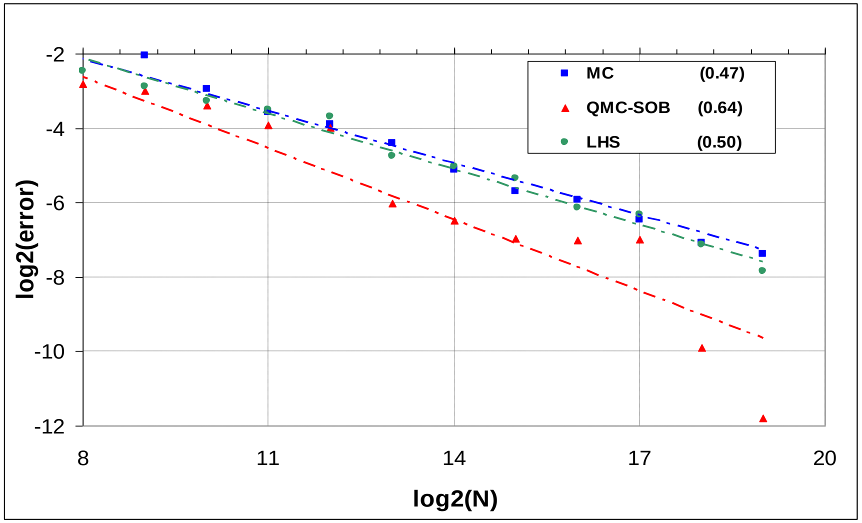
$$n = 360$$

The integration error vs. N. Type C

Dominant higher order indices:

$$\sum_{i=1}^{n} S_i << 1 \leftrightarrow d_S \approx n$$

(a)

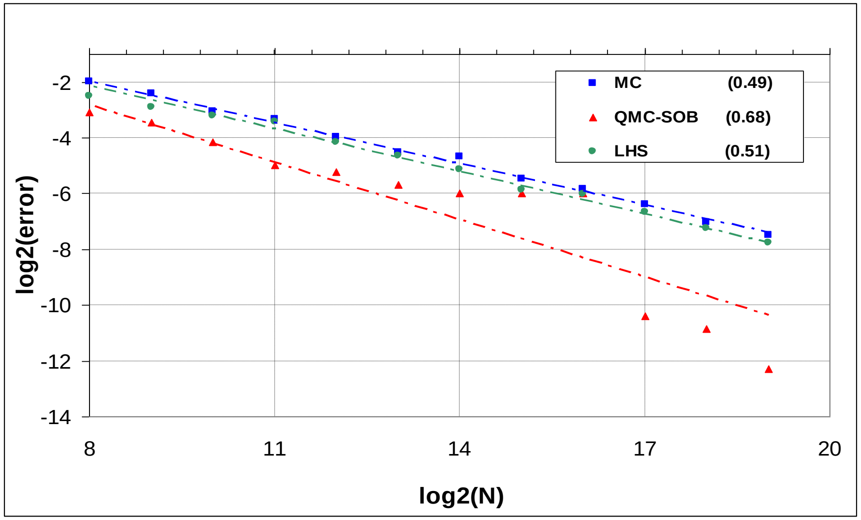


$$f(x) = \prod_{i=1}^{n} \frac{|4x_i - 2| + a_i}{1 + a_i}, a_i = 0$$

$$\rightarrow \prod_{i=1}^{n} |4x_i - 2|$$

$$n = 10$$

(b)



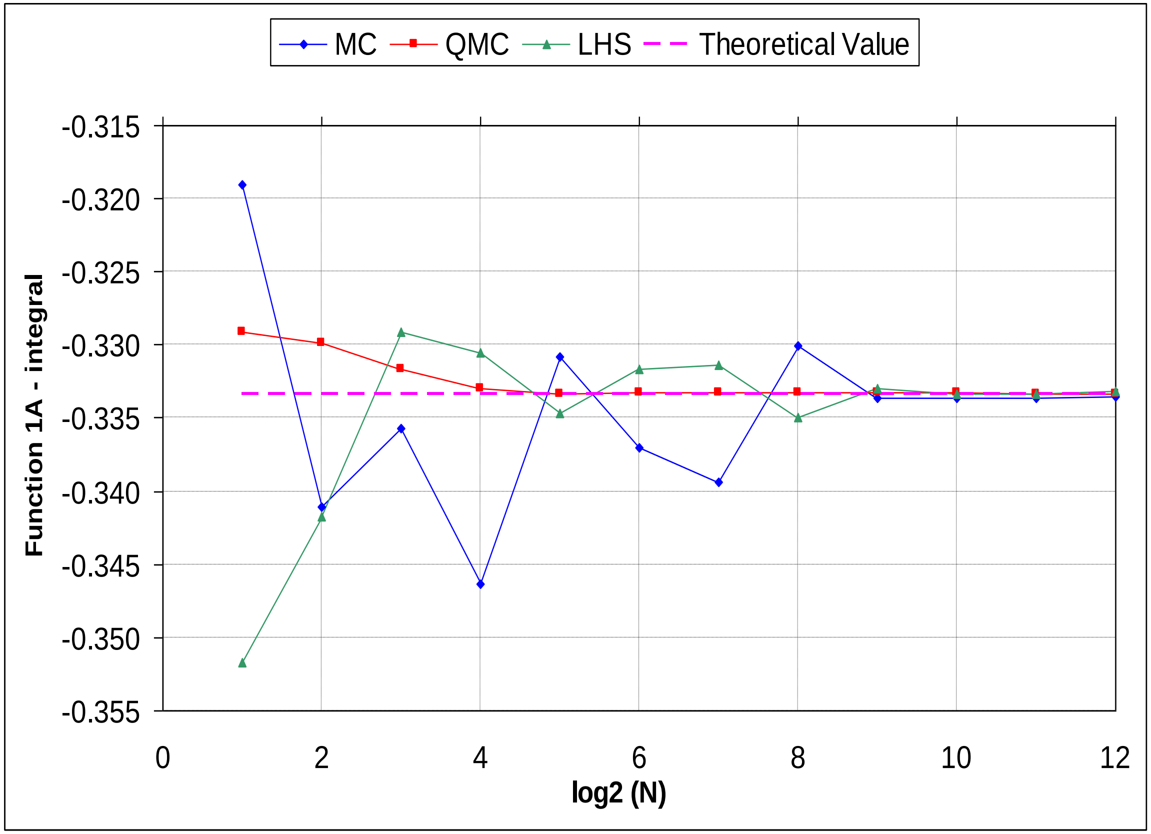
$$f(x) = (1/2)^{1/n} \prod_{i=1}^{n} x_i$$

$$n = 10$$

The integration error vs. N. Function 1A

$$\sum_{i=1}^{n}(-1)^i \prod_{j=1}^{i} x_j,$$

$$n = 360$$



QMC: convergence is monotonic

MC and LHS: convergence curves are oscillating

QMC is 30 much faster than MC and LHS (speed up is problem dependent)

References

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D. Saisana, M., and Tarantola, S., 2008, Global Sensitivity Analysis. The Primer, John Wiley & Sons.

Sobol' (2001) Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates, *Mathematics and Computers in Simulation,* 55, 271–280

Sobol' I., Kucherenko S. Global Sensitivity Indices for Nonlinear Mathematical Models. Review, *Wilmott*, 56-61, 1, 2005.

Kucherenko S., Albrecht D., Saltelli A. , Exploring multi-dimensional spaces: a Comparison of Latin Hypercube and Quasi Monte Carlo Sampling Techniques, 2015 arXiv:1505.02350

Kucherenko S., Feil B., Shah N., Mauntz W. (2011) The identification of model effective dimensions using global sensitivity analysis Reliability Engineering and System Safety 96, 440–449