A Monte Carlo Method Based on $\Lambda\Pi_\tau$ Sequences

Ivan Dimov
(in collaboration with Rayna Georgieva)

Bulgarian Academy of Sciences (BAS)
e-mail: ivdimov@bas.bg

National Science Fund Grant DTK 02/44
"Efficient Monte Carlo Methods for Large-Scale Scientific Problems"

Eighth IMACS Seminar on
"Monte Carlo Methods"

- Introduction
  - deterministic algorithms
  - randomized (Monte Carlo) algorithms
- Problem setting
- Mathematical background
  - Uniformly distributed sequences
  - (t, m, s)-nets and (t, s)-sequences in base b
  - $\Lambda\Pi_\tau$ sequences
  - Randomized Quasi-Monte Carlo (RQMC)
- Description of the Monte Carlo algorithm
- Numerical experiments
- Discussion and applicability
- Concluding remarks

- Consider the following problem of integration:

$$S(f) := I = \int_{E^s} f(x)dx, \quad \text{where}$$

$$E^s \equiv [0,1]^s, \quad x \equiv (x_1, \ldots, x_s) \in E^s \subset \mathbb{R}^s, \quad f \in C(E^s).$$

- Assume for a given r.v. θ one can prove that $E\theta = I$.
- Monte Carlo approximation to the solution: $\bar{\theta}_n = \frac{1}{n}\sum_{i=1}^{n} \theta^{(i)} \approx I$.

Definition.

If I is the exact solution of the problem, then the probability error is the least possible real number R_n, for which $P = Pr\left\{|\bar{\theta}_n - I| \leq R_n\right\}$, where $0 < P < 1$.

The computational problem is to approximate $S(f)$:

$$S(f) : f \to \mathbb{R}, \quad \text{where}$$

$$S(f) = \int_{E^s} f(x)dx \quad \text{and} \quad f \in F_0 \subset C(E^s).$$

- Quadrature formula $A = \sum_{i=1}^{n} c_i f(x^{(i)})$.
- Randomized quadrature formula $A^R = \sum_{i=1}^{n} \sigma_i f(\xi^{(i)})$.

Definition.

Consider the set $\mathcal{A}$ of algorithms A: $\mathcal{A} = \{A : Pr(R_n \leq \varepsilon) \geq c\}$ that solve a given problem with a probability error R_n such that the probability that R_n is less than *a priori* given constant ε is bigger than a constant $c < 1$.

Definition (▸ H. Weyl, 1916).

The sequence $x_1, x_2, \ldots$ is called an uniformly distributed sequence (u.d.s.) if, for an arbitrary region $\Omega \subset \mathbf{E}^s$,

$$\lim_{n\to\infty} [S_n(\Omega)/n] = V(\Omega),$$

where $S_n(\Omega)$ is the number of points with $1 \leq i \leq n$ that lie inside Ω and $V(\Omega)$ is the s-dimensional volume of Ω.

Theorem (▸ H. Weyl, 1916; Sobol', 1990).

The relation

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} f(\xi_j) = \int_{\mathbf{E}^s} f(x)dx$$

holds for all Riemann integrable functions f if and only if the sequence $x_1, x_2, \ldots$ is u.d.s.

Definition (▸ H. Weyl, 1916).

The sequence $x_1, x_2, \ldots$ is called an uniformly distributed sequence (u.d.s.) if, for an arbitrary region $\Omega \subset \mathbf{E}^s$,

$$\lim_{n\to\infty} [S_n(\Omega)/n] = V(\Omega),$$

where $S_n(\Omega)$ is the number of points with $1 \leq i \leq n$ that lie inside Ω and $V(\Omega)$ is the s-dimensional volume of Ω.

A u.d.s. should satisfy three additional requirements:

(i) the best asymptote as $n \to \infty$;
(ii) well distributed points for small n;
(iii) a computationally inexpensive algorithm.

- An elementary s-interval in base b is a subset of $\mathbf{E}^s$ of the form

$$\prod_{j=0}^{s} \left[\frac{a_j}{b^{d_j}}, \frac{a_j + 1}{b^{d_j}} \right],$$

where a_j, d_j are integers and $a_j < d_j$ for all $j \in \{1, ..., s\}$.

Definition (▸ H. Niederreiter, 1988).

Given two integers $0 \leq t \leq m$, a (t, m, s)-net in base b is a sequence x_n of b^m points of $\mathbf{E}^s$ such that $Card\ P \cap \{x_1, \ldots, x_{b^m}\} = b^t$ for any elementary interval P in base b of hypervolume $\lambda(P) = b^{t-m}$.

Definition (▸ H. Niederreiter, 1988).

Given two integers $0 \leq t \leq m$, a (t, m, s)-net in base b is a sequence x_n of b^m points of $\mathbf{E}^s$ such that $Card\ P \cap \{x_1, \ldots, x_{b^m}\} = b^t$ for any elementary interval P in base b of hypervolume $\lambda(P) = b^{t-m}$.

Definition (▸ H. Niederreiter, 1988).

Given a non-negative integer t, a (t, s)-sequence in base b is an infinite sequence of points x_n such that for all integers $k \geq 0, m \geq t$, the sequence $\{x^{kb^m}, \ldots, x^{(k+1)b^m-1}\}$ is a (t, m, s)-net in base b.

- Choose a primitive polynomial of degree s_j over the Galois field

$$P_j = x^{s_j} + a_{1,j}x^{s_j-1} + a_{2,j}x^{s_j-2} + \ldots + a_{s_j-1,j}x + 1,$$

where the coefficients $a_{1,j}, \ldots, a_{s_j-1,j} \in \{0, 1\}$.

- A sequence of positive integers $\{m_{1,j}, m_{2,j}, \ldots\}$ is defined by

$$m_{k,j} = 2a_{1,j}m_{k-1,j} \oplus 2^2 a_{2,j}m_{k-2,j} \oplus \cdots \oplus 2^{s_j} m_{k-s_j,j} \oplus m_{k-s_j,j},$$

where the initial values $m_{k,j}, 1 \leq k \leq s_j$ are odd and less than 2^k.

- The direction numbers $\{v_{1,j}, v_{2,j}, \ldots\}$: $\quad v_{k,j} = \dfrac{m_{k,j}}{2^k}$.

- The j-th component of the i-th point in a Sobol' sequence

$$x_{i,j} = i_1 v_{1,j} \oplus i_2 v_{2,j} \oplus \ldots,$$

where i_k is the k-th binary digit of $i = (\ldots i_3 i_2 i_1)_2$.

▸ I. Sobol' (1979), P. Bradley, B. Fox (1988)

- Randomized Quasi-Monte Carlo turns QMC into a variance reduction method by carefully randomizing well distributed points $x_i \equiv (x_{i,1}, x_{i,2} \ldots x_{i,s})$.
- Examples of RQMC point sets include
  - randomly shifted lattice rules,
  - scrambled digital nets,
  - digital nets with a random digital shift,
  - a Latin hypercube sample or a stratified sample followed by a random permutation of the points.
- Array-RQMC.
  - P. L'Ecuer, C. Lecot, B. Tuffin (2008)

- If

$$x_i = (x_{i,1}, x_{i,2} \dots x_{i,s}) \in \mathbf{E}^s$$

- the *i*-th $\Lambda\Pi_\tau$ point,

then

the *i*-th random point $\xi_i(\rho)$ with a p.d.f. $p(x)$:

$$\xi_i(\rho) = x_i + \rho\omega_i,$$

where ω_i is a unique uniformly distributed vector in $\mathbf{E}^s$ and ρ is the "shaking radius".

Example:

$$\omega_i = \{\cos\phi_i, \sin\phi_i\} \in \mathbf{E}^2$$

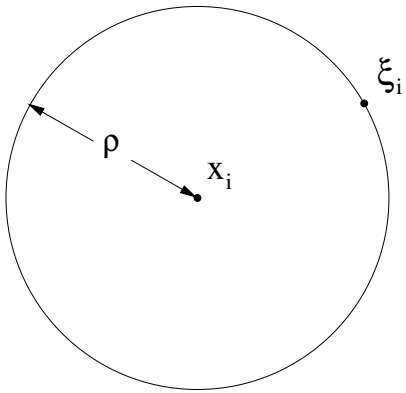

Figure: Generation of a random point $\xi_i \in \mathbf{E}^2$.

Theorem.

The mathematical expectation of the random variable $\theta = f(\xi)$ is equal to the value of the integral, that is

$$E\{\theta\} = S(f) = \int_{E^s} f(x)dx.$$

Sketch of proof.

Assume $\xi(\rho) = x + \rho\omega \in \mathbf{E}^s$, where ρ is relatively small $\rho << \left[\frac{a_j}{2^{d_j}}, \frac{a_j+1}{2^{d_j}}\right]$. $\xi_i(\rho)$ is still in the same elementary s-interval $\mathbf{E}_i^s = \prod_{j=0}^{s} \left[\frac{a_j}{2^{d_j}}, \frac{a_j+1}{2^{d_j}}\right]$, where the pattern $\Lambda\Pi_\tau$ point x_i is. Since $\int_{E^s} p(x)dx = 1$; $p(x) = 1$ for $x \in E^s$. That's why for the mathematical expectation of $\theta = f(\xi)$ we have

$$E\{\theta\} = \int_{E^s} f(x)p(x)dx = \int_{E^s} f(x)dx.$$

Example of
a non-smooth integrand:

$$f_1(x_1, x_2, x_3, x_4) = \sum_{i=1}^{4} |(x_i - 0.5)^{-1/3}|,$$

$$S(f_1) \approx 7.55953.$$

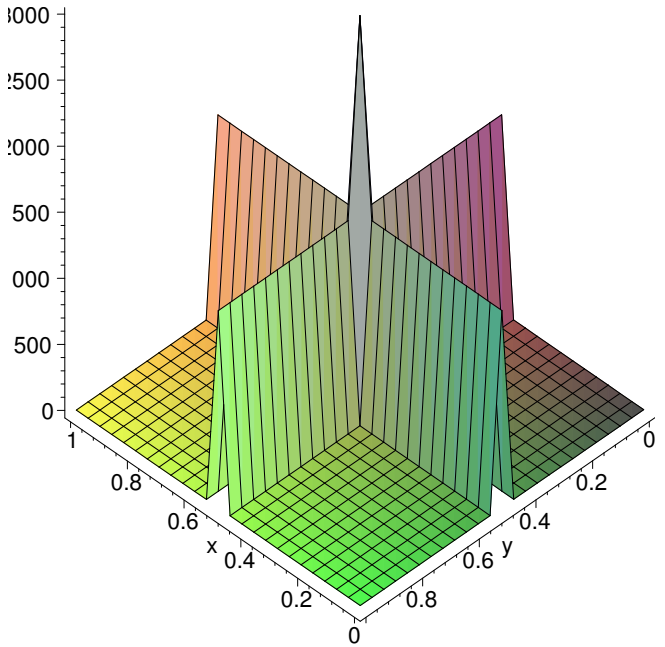

Figure: The integrand function in two-dimensional case.

Table: Radius ρ of spheres of the random points (*radius coefficient* $\kappa = \rho/\delta$).

n	Min. dist., δ	κ	ρ	κ	ρ	κ	ρ
10	0.43301	0.001	0.00043	0.09	0.03897	0.4	0.17321
10^2	0.13166	0.001	0.00013	0.09	0.01185	0.4	0.05266
10^3	0.06392	0.001	0.00006	0.09	0.00575	0.4	0.02557
10^4	0.02812	0.001	0.00003	0.09	0.00253	0.4	0.01125
50.10^3	0.01400	0.001	0.00001	0.09	0.00126	0.4	0.00560

Table: Relative error and computational time for numerical integration.

n	SFMT		Sobol'		MCA				
	Rel. err.	Time (s)	Rel. err.	Time (s)	δ	κ	$\rho \times 10^3$	Rel. err.	Time (s)
10	0.0001	< 0.01	0.2813	< 0.01	0.433	0.03	13	0.0438	< 0.01
						0.45	195	0.0509	< 0.01
10^2	0.0114	0.01	0.0565	< 0.01	0.132	0.03	3.9	0.0038	0.01
						0.45	59	0.0050	0.01
10^3	0.0023	0.06	0.0114	0.01	0.064	0.03	1.9	0.0016	0.10
						0.45	29	0.0004	0.11
10^4	0.0006	0.53	0.0023	0.06	0.028	0.03	0.8	4**e**-05	3.56
						0.45	12.7	0.0002	3.58
30.10^3	0.0002	1.63	0.0011	0.19	0.019	0.03	0.6	0.0002	28.5
						0.45	8.3	0.0003	28.8
50.10^3	0.0009	2.67	0.0008	0.29	0.014	0.03	0.4	0.0002	74.8
						0.45	6.3	2**e**-05	75.7

Calculations have been carried out on a PC with Intel(R) Pentium(R) 4 Processor.

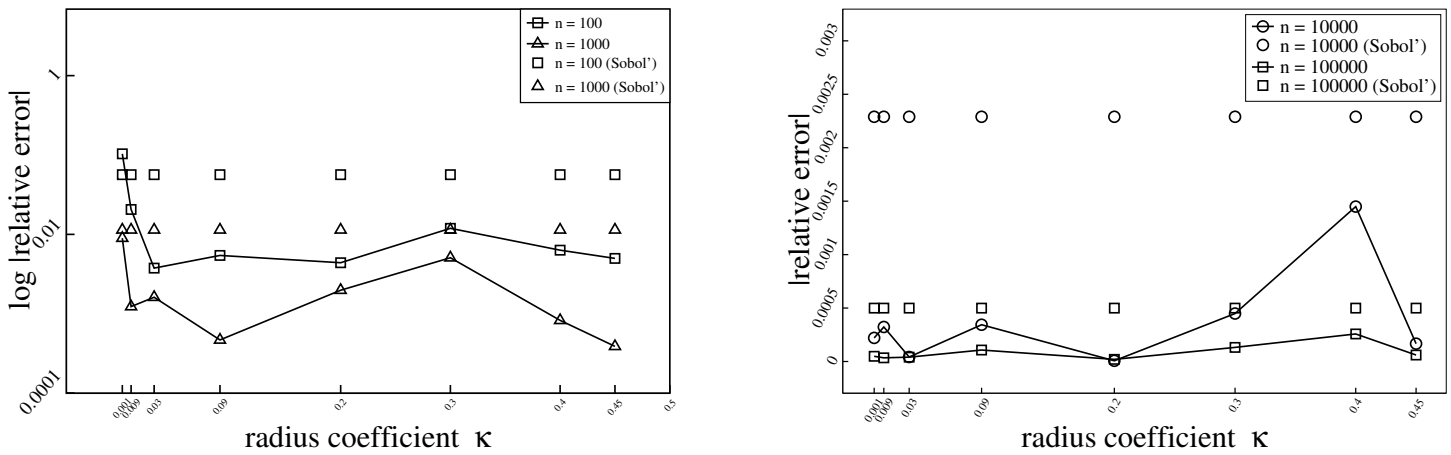

Figure: Relative error according to the "shaking radius".

Table: Difference of relative errors for Sobol' algorithm and the proposed Monte Carlo algorithm.

n \ κ	0.009	0.03	0.2	0.45
10	0.07709	0.23746	0.20639	0.23037
10^2	0.03594	0.05277	0.05214	0.05155
10^3	0.01014	0.00976	0.00940	0.01099
10^4	0.00197	0.00225	0.00228	0.00212
30.10^3	0.00102	0.00094	0.00084	0.00079
50.10^3	0.00077	0.00062	0.00077	0.00078

Example of
a smooth integrand:

$$f_2(x_1, x_2, x_3, x_4) = \mathbf{e}^{x_1+2x_2} \frac{cos(x_3)}{1 + x_2 + x_3 + x_4},$$

$$S(f_2) \approx 1.83690.$$

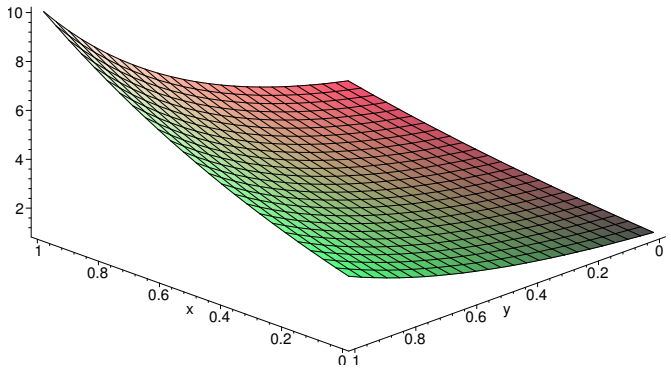

Figure: The integrand function in two-dimensional case.

Table: Relative error and computational time for numerical integration.

n	SFMT		Sobol'		MCA				
	Rel. err.	Time (s)	Rel. err.	Time (s)	δ	κ	$\rho \times 10^3$	Rel. err.	Time (s)
10^2	0.0350	< 0.01	0.0155	< 0.01	0.132	0.03	3.9	0.0160	0.01
						0.45	59	0.0264	0.01
10^3	0.0045	0.01	0.0023	< 0.01	0.064	0.03	1.9	0.0025	0.06
						0.45	29	0.0058	0.06
10^4	0.0016	0.10	0.0002	0.02	0.028	0.03	0.8	0.0003	3.29
						0.45	12.7	0.0016	3.28
30.10^3	0.0006	0.28	0.0001	0.04	0.019	0.03	0.6	0.0002	28.5
						0.45	8.3	0.0011	28.4
50.10^3	0.0004	0.46	6**e**-05	0.07	0.014	0.03	0.4	0.0001	76.0
						0.45	6.3	0.0008	76.1

Calculations have been carried out on a PC with Intel(R) Pentium(R) 4 Processor.

- *The proposed algorithm improves the error estimates for non-smooth integrands when the radius ρ is smaller than the minimal distance between $\Lambda\Pi_\tau$ points δ.* Strongly speaking the proposed approach is applicable if ρ is much smaller than δ. The implementation of the algorithm shows that this requirement is not very strong. Even for relatively large radiuses ρ the results are good. The reason is that centers of spheres are very well uniformly distributed by definition. So that, even for large values of radiuses of *shaking* the generated random points continue to be well distributed.

- For relatively low number of points (< 1000) the proposed algorithm gives results with a high accuracy. The relative error is approximately equal to 0.0038 for $n = 100$. For the same sample size the Sobol' algorithm gives more than 10 times higher error. For $n = 1000$ our algorithm gives relative error $0.0004 - 0.0016$ depending on the parameter κ while the Sobol' algorithm gives 0.0114. This is an important fact because *one has a possibility to estimate the value of the integral with a relatively high accuracy using a small number of random points*.

- *The proposed algorithm improves the error estimates for non-smooth integrands when the radius ρ is smaller than the minimal distance between $\Lambda\Pi_\tau$ points δ.* Strongly speaking the proposed approach is applicable if ρ is much smaller than δ. The implementation of the algorithm shows that this requirement is not very strong. Even for relatively large radiuses ρ the results are good. The reason is that centers of spheres are very well uniformly distributed by definition. So that, even for large values of radiuses of *shaking* the generated random points continue to be well distributed.

- For relatively low number of points (< 1000) the proposed algorithm gives results with a high accuracy. The relative error is approximately equal to 0.0038 for $n = 100$. For the same sample size the Sobol' algorithm gives more than 10 times higher error. For $n = 1000$ our algorithm gives relative error $0.0004 - 0.0016$ depending on the parameter κ while the Sobol' algorithm gives 0.0114. This is an important fact because *one has a possibility to estimate the value of the integral with a relatively high accuracy using a small number of random points*.

- The proposed algorithm combines properties of *two of the best available approaches* - Sobol's quasi-Monte Carlo integration and a high quality pseudorandom number SIMD-oriented Fast Mersenne Twister (SFMT) generator.
- The Monte Carlo algorithm has advantages against quasi-Monte Carlo and SFMT for non-smooth integrands. *For relatively small number of points the proposed approach gives much better results than Sobol's quasi-Monte Carlo integration.*
- *In case of smooth functions the proposed algorithm has significant advantage against plain Monte Carlo that uses SFMT generator with respect to the relative error.*

- The proposed algorithm combines properties of *two of the best available approaches* - Sobol's quasi-Monte Carlo integration and a high quality pseudorandom number SIMD-oriented Fast Mersenne Twister (SFMT) generator.
- The Monte Carlo algorithm has advantages against quasi-Monte Carlo and SFMT for non-smooth integrands. *For relatively small number of points the proposed approach gives much better results than Sobol's quasi-Monte Carlo integration*.
- *In case of smooth functions the proposed algorithm has significant advantage against plain Monte Carlo that uses SFMT generator with respect to the relative error.*

- The proposed algorithm combines properties of *two of the best available approaches* - Sobol's quasi-Monte Carlo integration and a high quality pseudorandom number SIMD-oriented Fast Mersenne Twister (SFMT) generator.
- The Monte Carlo algorithm has advantages against quasi-Monte Carlo and SFMT for non-smooth integrands. *For relatively small number of points the proposed approach gives much better results than Sobol's quasi-Monte Carlo integration*.
- *In case of smooth functions the proposed algorithm has significant advantage against plain Monte Carlo that uses SFMT generator with respect to the relative error*.

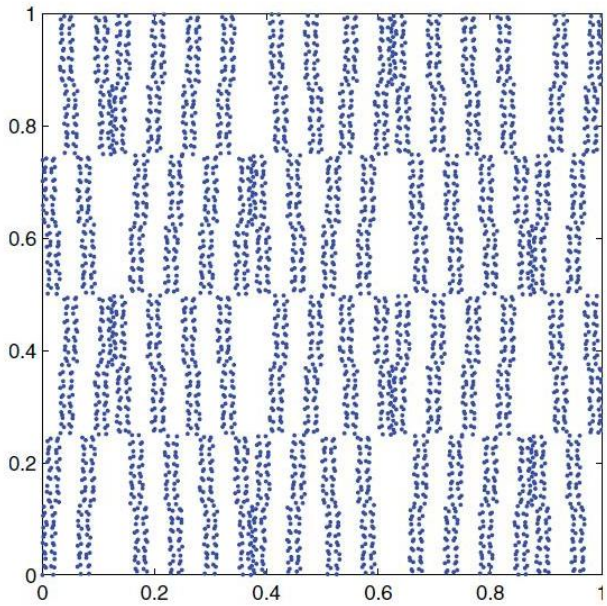

Figure: Poor projections of $\Lambda\Pi_\tau$ sequences for high dimensions

- Bradley, P., Fox, B.: Algorithm 659: Implementing Sobol's Quasi Random Sequence Generator. ACM Trans. Math. Software 14(1), 88–100 (1988)
- Joe, S., Kuo, F.Y.: Constructing Sobol Sequences with Better Two-dimensional Projections. SIAM J. Sci. Comput. 30, 2635–2654 (2008).
- L'Ecuyer, P., Lemieux, C.: Recent Advances in Randomized Quasi-Monte Carlo Methods. In: Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications, pp. 419–474. Kluwer Academic Publishers, Boston (2002)
- P. L'Ecuer, C. Lecot, B. Tuffin (2008): A Randomized Quasi-Monte Carlo Simulation Method for Markov Chains. Operations Research, 56, 4 (2008), 958-975.
- Niederreiter, H.: Low-Discrepancy and Low-Dispersion Sequences. Journal of Number Theory 30, 51–70 (1988)
- Sobol', I.: On the Systematic Search in a Hypercube. SIAM J. Numerical Analysis 16, 790–793 (1979)
- Sobol', I.: Quasi - Monte Carlo Methods. In: Sendov, Bl., Dimov, I.T. (eds.) International Youth Workshop on Monte Carlo Methods and Parallel Algorithms 1989, pp. 75–81. World Scientific, Singapore (1990)
- Weyl, H.: Ueber die Gleichverteilung von Zahlen mod Eins. Math. Ann. 77(3), 313–352 (1916)

- **More information about** ▸ SFMT **generator:**
  http://www.math.sci.hiroshima-u.ac.jp/ m-mat/MT/SFMT/index.html