

Optimized lattice rule and adaptive approach for multidimensional integrals with applications

Venelin Todorov *†, Ivan Dimov†, Stefka Fidanova †, Stoyan Poryazov *

*Institute of Mathematics and Informatics

Bulgarian Academy of Sciences

8 Acad. G. Bonchev Str., 1113 Sofia, Bulgaria

†Institute of Information and Communication Technologies

Bulgarian Academy of Sciences

25A Acad. G. Bonchev Str., 1113 Sofia, Bulgaria

Email: vtodorov@math.bas.bg, venelin@parallel.bas.bg, ivdimov@bas.bg, stefka@parallel.bas.bg, stoyan@math.bas.bg

Abstract—In this work we make a comparison between optimized lattice and adaptive stochastic approaches for multidimensional integrals with different dimensions. Some of the integrals has applications in environmental safety and control theory.

I. INTRODUCTION

MONTe Carlo methods are one of the most commonly used numerical methods. Their advantages are enhanced by increasing the dimensionality. For this reason, they are a major tool for numerically solving classes of problems in such important areas as particle physics, engineering chemistry, molecular dynamics, and financial mathematics. A major scientific challenge in the development of modern Monte Carlo methods is their relatively slow rate of convergence, which in many cases has the asymptotic $O(N^{-1/2})$, where N is the sample size. There are two approaches to improve convergence - reducing the variance of the estimated value and reducing the discrepancy of the sequence used. Adaptive strategy and lattice rules are two different ways to improve the convergence and has never been compared on this type of multidimensional integrals before.

II. THE STOCHASTIC APPROACHES

A. Adaptive approach

Adaptive strategy [1], [3], [4], [7] is well known method for evaluation of multidimensional integrals, especially when the integrand function has peculiarities and peaks. Let p_j and I_{Ω_j} be the following expressions: $p_j = \int_{\Omega_j} p(x) dx$ and $I_{\Omega_j} = \int_{\Omega_j} f(x)p(x) dx$. Consider now a random point $\xi^{(j)} \in \Omega_j$ with a density function $p(x)/p_j$. In this case $I_{\Omega_j} = E \left[\frac{p_j}{N} \sum_{i=1}^N f(\xi_i^{(j)}) \right] = E\theta_N$. This adaptive algorithm gives

Venelin Todorov is supported by the Bulgarian National Science Fund under Project KP-06-M32/2 - 17.12.2019 "Advanced Stochastic and Deterministic Approaches for Large-Scale Problems of Computational Mathematics" and by the National Scientific Program "Information and Communication Technologies for a Single Digital Market in Science, Education and Security (ICT in SES)", contract No DO1-205/23.11.2018, financed by the Ministry of Education and Science in Bulgaria. The work is also supported by Bulgarian National Science Fund under Project DN 12/5-2017 "Efficient Stochastic Methods and Algorithms for Large-Scale Problems" and by the Project KP-06-Russia/17 "New Highly Efficient Stochastic Simulation Methods and Applications" funded by the National Science Fund - Bulgaria.

an approximation with an error $\varepsilon \leq c N^{-1/2}$, where $c \leq 0.6745\sigma(\theta)$ ($\sigma(\theta)$ is the standard deviation).

Algorithm

1. **Input data:** total number of points N_1 , constant M (the initial number of subregions taken), constant ε (max value of the variance in each subregion), constant δ (maximal admissible number of subregions), d -dimensionality of the initial region/domain, f - the function of interest.
 - 1.1. **Calculate** the number of points to be taken in each subregion $N = N_1/\delta$.
2. **For** $j = 1, M^d$:
 - 2.1. **Calculate** the approximation of I_{Ω_j} and the variance D_{Ω_j} in subdomain Ω_j based on N independent realizations of random variable θ_N ;
 - 2.2. **If** $(D_{\Omega_j} \geq \varepsilon)$ **then**
 - 2.2.1. **Choose** the axis direction on which the partition will perform,
 - 2.2.2. **Divide** the current domain into two (G_{j_1}, G_{j_2}) along the chosen direction,
 - 2.2.3. **If** the length of obtained subinterval is less than δ **then go to** step 2.2.1 **else** $j = j_1$ G_{j_1} is the current domain right and **go to** step 2.1;
 - 2.3. **Else if** $(D_{\Omega_j} < \varepsilon)$ but an approximation of $I_{G_{j_2}}$ has not been calculated yet, **then** $j = j_2$ G_{j_2} is the current domain along the corresponding direction right and **go to** step 2.1;
 - 2.4. **Else if** $(D_{\Omega_j} < \varepsilon)$ but there are subdomains along the other axis directions, **then go to** step 2.1;
 - 2.5. **Else** Accumulation in the approximation I_N of I .

B. Lattice rules

We will use the following lattice point sets

$$x_n = \left\{ \frac{n}{N} z \right\}, \quad n = 0, \dots, N-1$$

where $z = (z_1, \dots, z_s)$ is the generating vector with dimensionality s . By a lattice rule we mean a rule of the form

$$I_N(f) = \frac{1}{N} \sum_{j=0}^{N-1} f(x_j), \quad (1)$$

where x_0, \dots, x_{N-1} are all points of a lattice L in the unit hypercube. Lets deal with the problem for approximate evaluation of the integral:

$$I(f) = \int_{[0,1]^s} f(x) dx,$$

where f is a real function in $[0,1]^s$. We consider the case when f has a periodic extension \tilde{f} in R^s ,

$$\begin{aligned} \tilde{f} &= f(x), x \in [0,1]^s, \\ \tilde{f}(x+z) &= \tilde{f}, x \in \mathbb{R}^s, z \in \mathbb{Z}^s. \end{aligned}$$

Let

$$I_N(f) = \frac{1}{N} \sum_{j=0}^{N-1} f(x_j), \quad (2)$$

where x_0, x_1, \dots, x_{N-1} are points from the lattice $L \in [0,1]^s$. We define the dual lattice L^\perp as

$$L^\perp = m \in \mathbb{R}^s : m.x \in \mathbb{Z}, x \in L.$$

In the case when the lattice has rank 1

$$L^\perp = m \in \mathbb{Z}^s : m.x \equiv 0 \pmod{N}.$$

Let f can be presented in Fourier series as:

$$f(x) = \sum_{m \in \mathbb{Z}^s} a(m) e^{2\pi i m \cdot x}, \quad x \in [0,1]^s,$$

where

$$a(m) = \int_{[0,1]^s} e^{-2\pi i m \cdot x} f(x) dx,$$

and the scalar product $m.u = m_1 x_1 + m_2 x_2 + \dots + m_s x_s$. Let E_s^α for $\alpha > 1$ and $c > 0$ is a class of functions f , for which the coefficients of Fourier [8] satisfies:

$$|a(m)| \leq \frac{c}{(\bar{m}_1 \dots \bar{m}_s)^\alpha}, \quad (3)$$

where

$$\bar{m} = |m|, \quad |m|, \quad m \geq 1, \quad \bar{m} = 1, \quad m = 0.$$

We define the Zaremba index [12] as

$$\rho = \min_{m \in L^\perp, m \neq 0} (\bar{m}_1 \dots \bar{m}_s).$$

The following theorems are key points in analysis the error of integration in the lattice rule:

Theorem 1: [10] Let L is a lattice with points x_0, x_1, \dots, x_{N-1} in $[0,1]^s$ and $m \in \mathbb{Z}^s$. Then

$$\frac{1}{N} \sum_{j=0}^{N-1} e^{2\pi i m \cdot x_j} = 1, \quad m \in L^\perp,$$

$$\frac{1}{N} \sum_{j=0}^{N-1} e^{2\pi i m \cdot x_j} = 0, \quad m \notin L^\perp.$$

Theorem 2: [12] Let L is a lattice with points x_0, x_1, \dots, x_{N-1} in $[0,1]^s$. Then for the error of integration is fulfilled

$$I_N(f) - I(f) = \sum_{m \in L^\perp, m \neq 0} a(m).$$

When we replace $f(x) = \sum_{m \in \mathbb{Z}^s} a(m) e^{2\pi i m \cdot x}$ in $I_N(f) = \frac{1}{N} \sum_{j=0}^{N-1} f(x_j)$:

$$\begin{aligned} I_N(f) &= \frac{1}{N} \sum_{j=0}^{N-1} f(x_j) = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{m \in \mathbb{Z}^s} a(m) e^{2\pi i m \cdot x_j} = \\ &\sum_{m \in \mathbb{Z}^s} a(m) \frac{1}{N} \sum_{j=0}^{N-1} e^{2\pi i m \cdot x_j} = \sum_{m \in L^\perp} a(m). \end{aligned}$$

and using the definition $I(f) = a(0)$ we will obtain

$$I_N(f) - I(f) = \sum_{m \in L^\perp, m \neq 0} a(m).$$

Theorem 3: [12] Let L is a lattice with points x_0, x_1, \dots, x_{N-1} in $[0,1]^s$ and let $f \in E_s^\alpha(c)$, $\alpha > 1$. Then

$$|I_N(f) - I(f)| \leq c \sum_{m \in L^\perp, m \neq 0} (\bar{m}_1 \dots \bar{m}_s)^{-\alpha}.$$

The proof of this fact directly leads from the previous theorem

$$\begin{aligned} |I_N(f) - I(f)| &= \sum_{m \in L^\perp, m \neq 0} |a(m)| \\ |a(m)| &\leq c \sum_{m \in L^\perp, m \neq 0} \frac{1}{(\bar{m}_1 \dots \bar{m}_s)^\alpha} \end{aligned}$$

Theorem 4: [12] Let L is a lattice with x_0, x_1, \dots, x_{N-1} in $[0,1]^s$ and let $f \in E_s^\alpha(c)$, $\alpha > 1$ and $\rho \geq 2$ is the Zaremba index. Then

$$|I_N(f) - I(f)| \leq cd(s, \alpha) \rho^{-\alpha} (\log \rho)^{s-1}.$$

Here we can define the number R_l , $l = 1, 2, \dots$, which will show the number of points $m \in L^\perp$ for which

$$\bar{m}_1 \dots \bar{m}_s < l\rho$$

Here we will use the lemma proven by Hua and Wang (1981) [6]:

$$R_l \leq e(s) l (\log 3l\rho)^{s-1}, \quad l = 1, 2, \dots,$$

where $e(s)$ depend only on s .

From Theorem 3

$$|I_N(f) - I(f)| \leq c \sum_{m \in L^\perp, m \neq 0} \frac{1}{(\bar{m}_1 \dots \bar{m}_s)^\alpha}$$

The sum on m can be broken down into a sum of collectibles by E_1, E_2, \dots , where E_l is defined by the inequalities

$$l\rho \leq \bar{m}_1 \dots \bar{m}_s < (l+1)\rho, l = 1, 2, \dots$$

By the definition of R_l we have the following inequalities:

$$\begin{aligned} \sum_{m \in L^\perp, m \neq 0} \frac{1}{(\bar{m}_1 \dots \bar{m}_s)^\alpha} &\leq \\ \sum_{l=1}^{\infty} \frac{R_{l+1} - R_l}{(l\rho)^\alpha} &\leq \\ \frac{1}{\rho^\alpha} \sum_{l=1}^{\infty} R_{l+1} \left(\frac{1}{l^\alpha} - \frac{1}{(l+1)^\alpha} \right). \end{aligned}$$

We have that

$$\frac{1}{l^\alpha} - \frac{1}{(l+1)^\alpha} = \alpha \int_l^{l+1} x^{-\alpha-1} dx \leq \frac{\alpha}{l^{\alpha+1}},$$

and using the Lemma of Hua and Wang

$$\begin{aligned} |I_N(f) - I(f)| &\leq \frac{c\alpha}{\rho^\alpha} \sum_{l=1}^{\infty} \frac{R_{l+1}}{l^{\alpha+1}} \leq \\ &\leq \frac{c\alpha e(s)}{\rho^\alpha} \sum_{l=1}^{\infty} \frac{(l+1)(\log 3(l+1)\rho)^{s-1}}{l^{\alpha+1}} \\ &\leq cd(s, \alpha)\rho^{-\alpha}(\log \rho)^{s-1}, \end{aligned}$$

where $d_1(s, \alpha)$ depend only on s and α . This proves theorem 4. In the theory of integration lattice rule a key point play the functions $f_\alpha, \alpha = 2, 4, \dots$. Every function f_α is the worst function [12] for appropriate class $E_s^\alpha(1)$. This functions are defined by

$$f_\alpha(x) = \sum_{m \in L^\perp} \frac{1}{(\bar{m}_1 \dots \bar{m}_s)^\alpha} e^{2\pi i m \cdot x}.$$

Furthermore $f_\alpha \in E_s^\alpha(1), I(f_\alpha) = 1$. Let $P_\alpha(z, N) = P_\alpha$ means the error in $I(f_\alpha)$. From Theorem 2

$$P_\alpha(z, N) = I_N(f_\alpha) - I(f_\alpha) = \sum_{m \in L^\perp, m \neq 0} \frac{1}{(\bar{m}_1 \dots \bar{m}_s)^\alpha}.$$

Now for $f \in E_s^\alpha(c)$ according Theorem 3 the error is defined by

$$|I_N(f) - I(f)| \leq cP_\alpha(N, z), \quad (4)$$

where $\alpha = 2, 4, \dots$ and the error is fulfilled when $f = f_\alpha$. The values of $P_\alpha(z, N)$ for fixed α are using as indication of the relative quality of the particular lattice. In the case of rank-1 lattice

$$P_\alpha(z, N) = \sum_{z, a \equiv 0 \pmod{N}, a \neq 0} \frac{1}{(\bar{m}_1 \dots \bar{m}_s)^\alpha}.$$

Bakhvalov proves that [12]:

Theorem 5: If P is a lattice point set, with an optimal generating vector z , for the error of integration we have

$$\left| \frac{1}{N} \sum_{k=0}^{N-1} f\left(\frac{k}{N}z\right) - \int_{[0,1]^s} f \right| \leq Cd(s, \alpha) \frac{(\log N)^{\beta(s, \alpha)}}{N^\alpha} \quad (5)$$

for $f \in E_s^\alpha(c), \alpha > 1$ and $d(s, \alpha), \beta(s, \alpha)$ does not depend on N .

Bakhvalov (1959) [12] prove that:

Theorem 6: If N is a prime number, there exists generating vector z , such that

$$\begin{aligned} D(N) &= O(N^{-1} \log^s N), \\ P_\alpha(z, N) &= O(N^{-\alpha} \log^{\alpha s} N). \end{aligned}$$

Niederreiter shows [11], if N is not a prime number, there exist lattice point sets for which:

$$P_\alpha(z, N) = O(N^{-\alpha} (\log N)^{\alpha(s-1)+1} \left(\frac{N}{\phi(N)} \right)), s \geq 2,$$

$$P_\alpha(z, N) = O(N^{-\alpha} (\log N)^\alpha \left(\frac{N}{\phi(N)} + \frac{\tau(N)}{\log(N)} \right)), s = 2,$$

$$P_\alpha(z, N) = O(N^{-\alpha} (\log N)^\alpha (s-1) \left(1 + \frac{\tau(N)}{\log^{s-1}(N)} \right)), s \geq 3,$$

where $\phi(N)$ is the Euler's totient function and $\tau(N)$ is the number of divisors of N . For prime number from this formulas leads that there exist z , for which

$$P_\alpha(z, N) = O(N^{-\alpha} \log^{\alpha(s-1)} N).$$

It is fulfilled the following theorem of Sharygin (1963) [9]:

Theorem 7: For a given lattice rule it is fulfilled that

$$P_\alpha(z, N) \geq O(N^{-\alpha} \log^{s-1} N). \quad (6)$$

When $s = 2$ there is an optimal construction. Bakhavalov (1959), Hua and Wang (1960) introduced construction, based on Fibonacci numbers, which are defined recursively by

$$F_0 = 0, F_1 = 1, F_l = F_{l-1} + F_{l-2}, l \geq 2.$$

Let $N = F_l$ and $z = (1, F_{l-1})$. For the obtained lattice Bakhavalov and Hua and Wang show that

$$P_\alpha((1, F_{l-1}), F_l) = O(F_l^{-\alpha} \log F_l),$$

which is optimal according to Sharygin. In 1966 Zaremba [12] shows that

$$D(F_l) = O(F_l^{-1} \log F_l),$$

which is optimal according to Schmidt (1972) [10]. It is important that for finding F_l are necessary only $O(\log F_l)$ elementary operations. There are different techniques for optimal constructions when $s \geq 2$. Let $s = \frac{p-1}{2}$, where $p \geq 5$ is a prime number. If we have the set $Q(2 \cos \frac{2\pi}{p})$, which is an algebraic field of degree s with basis functions

$2 \cos(2\pi j/p) | j = 1, \dots, s$, we construct the sequence $\eta_l, l = 1, 2, \dots$, which satisfies:

$$c_s^{-1}e^l < \eta_l < c_s e^l, c_s^{-1}e^{-l/(s-1)} \leq |\eta_l^{(j)}| \leq c_s^{-1}e^{-l/(s-1)}, \\ j = 2, \dots, s,$$

where c_s is a constant and $\eta^{(j)}$ is the conjugate of η . Define the generating vector by:

$$\eta_l = \sum_{j=1}^s \eta_l^{(j)}, h_j^{(l)} = [\eta_l 2 \cos(2\pi j/p)], j = 2, \dots, s,$$

where η_l is the number of points and $[.]$ is a function whole part. With such a choice of z Hua and Wang show that

$$D(\eta_l) = O(\eta_l^{-\frac{1}{2} - \frac{1}{2(s-1)+\varepsilon}}), P_\alpha(z, N) = O(\eta_l^{-\frac{\alpha}{2} - \frac{\alpha}{2(s-1)+\varepsilon}}),$$

where ε is a preliminary given positive number.

We will construct a lattice L with the following optimized generating vector. for positive number n :

$$z = (1, F_n(2), \dots, F_n(s)) \quad (7)$$

It is fulfilled that $F_n(j) = F_{n+j-1} - F_{n+j-2} - \dots - F_n$, where F_i are the generalized Fibonacci numbers with dimensionality s :

$$F_{l+s} = F_l + F_{l+1} + \dots + F_{l+s-1}, l = 0, 1, \dots \quad (8)$$

with initial condition:

$$F_0 = F_1 = \dots = F_{s-2} = 0, F_{s-1} = 1, \quad (9)$$

for $l = 0, 1, \dots$

After simplifying:

$$z = (1, F_{n-1} + F_{n-2} + \dots + F_{n-s+1}, \dots, F_{n-1} + F_{n-2}, F_{n-1}) \quad (10)$$

III. NUMERICAL EXAMPLES

We will test the optimized lattice rule into the following examples:

Example 1. $s=3$.

$$\int_{[0,1]^3} \exp(x_1 x_2 x_3) \approx 1.14649907. \quad (11)$$

Example 2. $s=4$.

$$\int_{[0,1]^4} x_1 x_2^2 e^{x_1 x_2} \sin(x_3) \cos(x_4) \approx 0.1089748630. \quad (12)$$

Example 3.

$$\int_{[0,1]^5} \exp(-100x_1 x_2 x_3)(\sin(x_4) + \cos(x_5)) \approx 0.1854297367. \quad (13)$$

Example 4. $s=7$.

$$\int_{[0,1]^7} e^{1 - \sum_{i=1}^3 \sin(\frac{\pi}{2} \cdot x_i)} \cdot \arcsin(\sin(1) + \frac{\sum_{j=1}^7 x_j}{200}) \approx 0.75151101. \quad (14)$$

Table I
RELATIVE ERROR FOR 3 DIMENSIONAL INTEGRAL

N	crude	t,s	adapt	t,s	lattice	t,s
10^3	3.62e-2	0.007	4.82e-3	0.17	1.21e-3	0.006
10^4	1.67e-3	0.07	1.07e-3	1.44	5.04e-4	0.07
10^5	8.60e-4	0.74	1.52e-4	10.9	5.34e-6	0.66
10^6	5.12e-4	6.12	5.11e-5	131	7.85e-7	7.02
10^7	3.15e-4	60.1	2.34e-5	1094	8.89e-8	79.7

Table II
RELATIVE ERROR FOR 3 DIMENSIONAL INTEGRAL FOR PRELIMINARY GIVEN TIME

time in sec.	crude	adapt	lattice
1	1.05e-3	7.96e-3	2.34e-6
5	6.84e-4	8.14e-4	8.47e-7
10	4.79e-4	1.82e-4	4.89e-7
100	1.57e-4	7.04e-5	6.53e-9

Example 5. $s=15$.

$$\int_{[0,1]^{15}} \left(\sum_{i=1}^{10} x_i^2 \right) (x_{11} - x_{12}^2 - x_{13}^3 - x_{14}^4 - x_{15}^5)^2 \approx 1.96440666. \quad (15)$$

Example 6. $s=25$.

$$\int_{[0,1]^{25}} \frac{4x_1 x_3^2 e^{2x_1 x_3}}{(1+x_2+x_4)^2} e^{x_5+\dots+x_{20}} x_{21} \dots x_{25} \approx 108.808. \quad (16)$$

Example 7. $s=30$.

$$\int_{[0,1]^{30}} \frac{4x_1 x_3^2 e^{2x_1 x_3}}{(1+x_2+x_4)^2} e^{x_5+\dots+x_{20}} x_{21} \dots x_{30} \approx 3.244540. \quad (17)$$

The results are given in the tables below. We have used laptop CPU Core i7 4710HQ at 2.5GHz. The first group of tables contains information about the method used, the relative error obtained, the number of conversions required, and the CPU time required to compute the integral. The second group table contains information about the computational complexity. A comparison is made, which shows what relative error each of the used algorithms gives at a predetermined time. From these results it can be concluded that the lattice method is the most efficient for computing multidimensional integrals of smooth subintegral functions due to the low computational complexity and high accuracy in comparison with the simple Monte Carlo algorithm (crude) and the adaptive approach (adapt). The crude Monte Carlo is the basic and simplest possible Monte Carlo approach. Such kind of applications appear also in some important problems in control theory.

It can be seen that by increasing the dimension, the optimized lattice rule gives the best results (Table I-VII), and the advantage is more clearly pronounced for a preliminary given computational time (Table II, Table IV).

Table III
RELATIVE ERROR FOR 4 DIMENSIONAL INTEGRAL

N	crude	t,s	adapt	t,s	opt. lattice	t,s
10^4	9.31e-3	0.08	1.11e-3	1.97	8.61e-5	0.07
10^5	4.37e-3	0.78	1.44e-4	20.1	3.69e-5	0.99
10^6	7.87e-4	5.86	5.63e-5	210	2.86e-6	5.22
10^7	4.31e-5	50.1	9.11e-6	2035	3.38e-7	58

Table IV
RELATIVE ERROR FOR 4 DIMENSIONAL INTEGRAL FOR PRELIMINARY
GIVEN TIME

time in sec.	crude	adapt	opt. lattice
5	8.61e-4	5.24e-3	8.47e-7
20	2.31e-4	1.44e-4	4.89e-7
100	2.21e-5	8.21e-5	4.53e-8

The lattice method is not applicable to functions with singularities as we will see from the numerical experiments in this section. Let the following model function be given:

$$f(x) = (1 + \sum_{i=1}^d a_i x_i)^{-(s+1)}. \quad (18)$$

The class of test functions in question belongs to a package proposed by Genz [5]. Each individual class of the package is characterized by a peculiarity in computational terms. The selected set of functions has a single local maximum near one of the vertices of the multidimensional single cube, similar to some model functions describing the change in the concentrations of pollutants in the air. The parameters a_i are evaluated, using variables a'_i , uniformly distributed in $[\frac{1}{20}; 1 - \frac{1}{20}]$, and the relation $a = c a'$. The constant c is *parameter of computational complexity* [1], selected so that the "sharpness" of the local maximum is controlled by the following norm $\|a\|_1 = \frac{600}{s^2}$. The adaptive approach is effective for such a class of functions - functions with computational features in a local subdomain of the field of integration.

Table V
RELATIVE ERROR FOR 5 DIMENSIONAL INTEGRAL

N	crude	t,s	adapt	t,s	opt. lattice	t,s
10^3	2.10e-2	0.007	2.15e-3	0.27	1.75e-4	0.007
10^4	4.52e-3	0.07	2.01e-3	2.43	1.28e-5	0.06
10^5	1.19e-3	0.64	8.91e-4	25.2	9.50e-6	0.61
10^6	9.47e-4	6.06	2.92e-4	219.5	5.47e-7	5.98
10^7	6.38e-4	59.9	8.21e-5	2043	7.71e-8	58.4

Table VI
RELATIVE ERROR FOR 7 DIMENSIONAL INTEGRAL

N	crude	t,s	adapt	t,s	opt. lattice	t,s
10^4	1.47e-2	0.11	1.07e-3	2.07	2.19e-4	0.11
10^5	8.26e-3	1.02	7.51e-4	19.3	6.87e-5	0.99
10^6	1.76e-3	10.1	6.30e-5	194	7.39e-6	9.81
10^7	9.85e-4	96.3	2.34e-5	1861	8.89e-7	94.2

Table VII
RELATIVE ERROR FOR 15-DIMENSIONAL INTEGRAL

N	crude	t,s	adapt	t,s	opt. lattice	t,s
10^3	6.31e-2	0.09	3.16e-3	9.24	5.34e-2	0.08
10^4	4.30e-2	0.95	1.49e-3	88	1.22e-3	0.93
10^5	2.77e-2	9.70	5.76e-4	847	3.08e-4	9.65
10^6	2.13e-3	95.8	1.29e-4	8235	1.37e-5	96.9

The results obtained after applying the simple(crude) and adaptive Monte Carlo algorithm for integrals of 5 and 18 are given in Table VIII and Table IX, respectively. The efficiency of the adaptive and lattice algorithms is studied.

In both tables, the value N denotes the total number of conversions in the entire domain for the ordinary algorithm, for the adaptive algorithm, and for the algorithm using a plurality of lattice types. The total number of conversions and approximately the same time to calculate the integrals is actually the basis for comparing the presented results. A number of realizations of the random variable have been chosen so that the times for obtaining an approximate value of the integral are close. The obtained results confirm the reduction of the variance - the adaptive algorithm needs much fewer implementations and gives more accurate results than the ordinary Monte Carlo and the lattice type algorithm, but it is significantly slower (see Table IX).

IV. CONCLUSION

A comprehensive experimental study is done for multidimensional integrals with applications in ecology. The numerical experiments show that the optimized lattice rule is more efficient for multidimensional integrals of smooth functions. The adaptive approach is more efficient for multidimensional integrals with peculiarities and peaks which have applications in air pollution modelling.

REFERENCES

- [1] Berntsen J., Espelid T.O., Genz A. (1991) An adaptive algorithm for the approximate calculation of multiple integrals, ACM Trans. Math. Softw. 17: 437–451.
- [2] Dimov I. (2008) Monte Carlo Methods for Applied Scientists, New Jersey, London, Singapore, World Scientific, 291 p., ISBN-10 981-02-2329-3.
- [3] Dimov I., Karaivanova A., Georgieva R., Ivanovska S. (2003) Parallel Importance Separation and Adaptive Monte Carlo Algorithms for Multiple Integrals, Springer Lecture Notes in Computer Science, 2542, 99–107.
- [4] Dimov I., Georgieva R. (2010) Monte Carlo Algorithms for Evaluating Sobol' Sensitivity Indices. Math. Comput. Simul. 81(3): 506–514.
- [5] A. Genz, Testing multidimensional integration routines. Tools, *Methods and Languages for Scientific and Engineering Computation* (1984) 81–94.
- [6] Hua L.K. and Wang Y. (1981) Applications of Number Theory to Numerical analysis.
- [7] Pencheva, V., I. Georgiev, and A. Asenov. "Evaluation of passenger waiting time in public transport by using the Monte Carlo method." AIP Conference Proceedings. Vol. 2321. No. 1. AIP Publishing LLC, 2021.
- [8] Raeva, E., & Georgiev, I. R. (2018, October). Fourier approximation for modeling limit of insurance liability. In AIP Conference Proceedings (Vol. 2025, No. 1, p. 030006). AIP Publishing LLC.

Table VIII
RELATIVE ERROR FOR $s = 5$, $I[f] = 2.12\text{e-}06$, $a = (5, 5, 5, 5, 4)$.

adapt			crude			opt. lattice		
N	$I_N[f]$	(s)	N	$I_N[f]$	(s)	N	$I_N[f]$	(s)
10^2	3.7735e-03	0.33	10^5	5.4858e-02	0.27	1346269	9.7135e-02	0.38
10^3	1.2877e-03	1.44	10^6	3.8207e-02	1.22	3524578	6.7594e-02	1.32
10^4	4.2452e-04	10.75	10^7	3.3962e-03	12.3	14930352	1.5377e-02	15.07
10^5	4.7169e-05	142.18	10^8	9.4339e-04	124.2	102334155	2.9245e-03	134.58

Table IX
RELATIVE ERROR FOR $s = 18$, $I[f] = 9.919\text{e-}06$, $a = \left(\frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{1}{9}, \frac{1}{9}\right)$.

adapt			crude			opt. lattice		
N	$I_N[f]$	(s)	N	$I_N[f]$	(s)	N	$I_N[f]$	(s)
10	9.2341e-04	15.7	10^7	4.5367e-03	13.6	14930352	7.1579e-02	14.7
10^2	8.0653e-05	142	10^8	2.0163e-03	140	102334155	1.3096e-02	144.1
10^3	1.0081e-05	1408	10^9	5.0480e-04	1353.5	1134903170	7.8883e-03	1344.3

- [9] I.F. Sharygin (1963) A lower estimate for the error of quadrature formulas for certain classes of functions, *Zh. Vychisl. Mat. i Mat. Fiz.* **3**, 370–376.
- [10] I.H. Sloan and P.J. Kachoyan (1987) Lattice methods for multiple integration: Theory, error analysis and examples, *SIAM J. Numer. Anal.* **24**, 116–128.
- [11] I.H. Sloan and S. Joe, *Lattice Methods for Multiple Integration, Lattice methods for multiple Integration*, (Oxford University Press 1994).
- [12] Y. Wang and F. J. Hickernell (2000) *An historical overview of lattice point sets*, in Monte Carlo and Quasi-Monte Carlo Methods 2000, Proceedings of a Conference held at Hong Kong Baptist University, China.