

Abstracts of Dissertations

Institute of Information and
Communication Technologies

BULGARIAN ACADEMY OF
SCIENCES



3 / 2017



MONTE CARLO METHODS
FOR MULTIDIMENSIONAL
INTEGRALS AND
INTEGRAL EQUATIONS
AND APPLICATIONS

Venelin Todorov

МЕТОДИ МОНТЕ КАРЛО ЗА
МНОГОМЕРНИ ИНТЕГРАЛИ
И ИНТЕГРАЛНИ УРАВНЕНИЯ
И ПРИЛОЖЕНИЯ

Венелин Тодоров

Автореферати на дисертации

Институт по информационни и
комуникационни технологии

БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ

ISSN: 1314-6351

Поредицата „Автореферати на дисертации на Института по информационни и комуникационни технологии при Българската академия на науките“ представя в електронен формат автореферати на дисертации за получаване на научната степен „Доктор на науките“ или на образователната и научната степен „Доктор“, защитени в Института по информационни и комуникационни технологии при Българската академия на науките. Представените трудове отразяват нови научни и научно-приложни приноси в редица области на информационните и комуникационните технологии като Компютърни мрежи и архитектури, Паралелни алгоритми, Научни пресмятания, Лингвистично моделиране, Математически методи за обработка на сензорна информация, Информационни технологии в сигурността, Технологии за управление и обработка на знания, Грид-технологии и приложения, Оптимизация и вземане на решения, Обработка на сигнали и разпознаване на образи, Интелигентни системи, Информационни процеси и системи, Вградени интелигентни технологии, Йерархични системи, Комуникационни системи и услуги и др.

Редактори

Генадий Агре

Институт по информационни и комуникационни технологии, Българска академия на науките
E-mail: agre@iinf.bas.bg

Райна Георгиева

Институт по информационни и комуникационни технологии, Българска академия на науките
E-mail: rayna@parallel.bas.bg

Даниела Борисова

Институт по информационни и комуникационни технологии, Българска академия на науките
E-mail: dborissova@iit.bas.bg

Настоящото издание е обект на авторско право. Всички права са запазени при превод, разпечатване, използване на илюстрации, цитирания, разпространение, възпроизвеждане на микрофилми или по други начини, както и съхранение в бази от данни на всички или част от материалите в настоящето издание. Копирането на изданието или на част от съдържанието му е разрешено само със съгласието на авторите и/или редакторите

*The series **Abstracts of Dissertations of the Institute of Information and Communication Technologies at the Bulgarian Academy of Sciences** presents in an electronic format the abstracts of Doctor of Sciences and PhD dissertations defended in the Institute of Information and Communication Technologies at the Bulgarian Academy of Sciences. The studies provide new original results in such areas of Information and Communication Technologies as Computer Networks and Architectures, Parallel Algorithms, Scientific Computations, Linguistic Modelling, Mathematical Methods for Sensor Data Processing, Information Technologies for Security, Technologies for Knowledge management and processing, Grid Technologies and Applications, Optimization and Decision Making, Signal Processing and Pattern Recognition, Information Processing and Systems, Intelligent Systems, Embedded Intelligent Technologies, Hierarchical Systems, Communication Systems and Services, etc.*

Editors

Gennady Agre

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
E-mail: agre@iinf.bas.bg

Rayna Georgieva

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
E-mail: rayna@parallel.bas.bg

Daniela Borissova

Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
E-mail: dborissova@iit.bas.bg

This work is subjected to copyright. All rights are reserved, whether the whole or part of the materials is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in other ways, and storage in data banks. Duplication of this work or part thereof is only permitted under the provisions of the authors and/or editor.



BULGARIAN ACADEMY OF SCIENCES

Abstract of PhD Thesis

MONTE CARLO METHODS FOR MULTIDIMENSIONAL INTEGRALS AND INTEGRAL EQUATIONS AND APPLICATIONS

Venelin Todorov

Supervisor: Prof. Ivan Dimov

Approved by Supervising Committee:

Prof. Stefan Stefanov

Prof. Luben Valkov

Prof. Ivan Dimov

Assoc. Prof. Stanislava Stoilova

Assoc. Prof. Rayna Georgieva



INSTITUTE OF INFORMATION AND
COMMUNICATION TECHNOLOGIES

Department of Parallel Algorithms

The PhD thesis was discussed and allowed to be defended during an extended session of the Department of Parallel Algorithms at IICT-BAS, which had been held on May 23, 2016.

The defense of the PhD thesis had been held on 03.10.2017 at 14:00 am in Room 218, Block 25A, IICT-BAS.

The full volume of the dissertation is 184 pages. It consists of an introduction and three chapters. The list of references contains 237 titles on 18 pages. The text of the dissertation includes 45 tables and 26 figures.

Author: Venelin Lyubomirov Todorov

Title: Monte Carlo methods for multidimensional integrals and integral equations and applications.

Keywords: Monte Carlo and quasi-Monte Carlo methods, multidimensional integrals, integral equations, linear algebraic systems, compact finite difference schemes, Fibonacci lattice rule, Wigner kernel, air pollution modelling.

Introduction

The Monte Carlo (MC) methods are a broad class of computational algorithms based on performing of a big number of simulations of an experiment with randomly chosen input parameters. The year 1949 is generally regarded as the official birthday of the Monte Carlo method when the paper of Metropolis and Ulam [19] was published, although some authors point to earlier dates. Ermakov [8], for example, notes that a solution of a problem by the Monte Carlo method is contained in the Old Testament. The development and intensive applications of the method is connected with the names of J. von Neumann, E. Fermi, G. Kahn and S. M. Ulam, who worked at Los Alamos (USA) for forty years on the Manhattan project¹. A legend says that the method was named in honor of Stanislaw Ulam's uncle, who was a gambler, at the suggestion of Metropolis. Some of the most popular MC methods are the ordinary Crude MC, variance reduction methods such as Stratified sampling, Importance Sampling and Adaptive algorithms, as well as techniques using quasi-random sequences like those of Sobol, Halton and others. An additional and relatively new type of quasi - sequences are those of lattice type, which are appropriate in case of smooth and periodic integrands. Areas, where the Monte Carlo methods are widely used are Physics, Engineering, Financial Mathematics, Gambling, etc. Monte Carlo and/or quasi-Monte Carlo methods can be directly applied to finance problems involving multidimensional integrals [5]. One of the best known physicist Richard Feynman formulated the problem of finding an effective and fast algorithm with linear or polynomial computational complexity for computing multidimensional integrals which represent the Wigner kernel [9].

In general, Monte Carlo numerical algorithms may be divided into two classes - direct and iterative algorithms. The direct algorithms contain only a stochastic error. For example, evaluating integrals can be performed by direct Monte Carlo algorithms. Iterative Monte Carlo algorithms deal with an approximate solution obtaining an improved solution with each iteration of the algorithm. In principle, they require an infinite number of iterations to obtain the exact solution, but usually an approximation with k significant figures is satisfactory. Iterative algorithms are preferred for solving integral equations and large sparse systems of algebraic equations (such as those arising from approximations of partial differential equations). In the latter case there are two errors - stochastic and systematic. The systematic error depends both on the number of iterations performed and the characteristic values of the iteration operator, while the stochastic errors depend on the probabilistic nature of the algorithm. In order to obtain good results the stochastic error r_N must be approximately equal to the systematic error r_k that is $r_N = O(r_k)$. The problem of balancing of the errors is closely connected with the problem of obtaining an

¹Manhattan Project refers to the effort to develop the first nuclear weapons during World War II by the United States with assistance from the United Kingdom and Canada.

optimal ratio between the number of realizations N of the random variable and the mean value k of the number of steps in each random trajectory [6].

In many fields of science and engineering parabolic equations are always used to describe many phenomena, so that the finite-difference method that solves the parabolic equation is frequently a focus of concern, see e.g. [18]. In the context of the finite difference discretization, the standard second-order discretization schemes may need fine gridings to yield approximate solutions of acceptable accuracy. The resulting large size systems have to be solved, which may consume a lot of memory space and CPU cycles even on present generation supercomputers. One approach to reduce computational cost in very large-scale modelings and simulations is to use *higher-order discretization methods*. Other important factor affecting the computational efficiency of a discretized method is to solve the resulting linear and nonlinear systems of algebraic equations in an efficient way. The higher-order methods usually generate algebraic systems of much smaller size, compared to the lower-order methods. Because of this and other advantages of high-order methods, there has been growing interest of developing and using highly accurate numerical schemes for solving partial differential equations, leading to renewed interest in high-order compact difference schemes [20].

Goals and Objectives of the Thesis

The goal of this thesis is the developing, realization and study of various Monte Carlo (quasi-Monte Carlo algorithms) for multidimensional integral, integral equations and algebraic linear systems. The most important part is the application of the methods and algorithms to many areas such as financial computing, physics, biology, environmental sciences. Another important goal is the construction and application of high order accuracy methods, based on finite difference schemes, in air pollution modelling.

The main tasks are:

1. To develop and study of an adaptive Monte Carlo (MC) method, latin hypercube sampling algorithm and a lattice set algorithm, based on generalized Fibonacci numbers of corresponding dimensionality, to make a comparison with Crude Monte Carlo algorithm and Sobol quasi-Monte Carlo algorithm and to analyze the computational complexity of the algorithms.
2. To apply the algorithms for European option pricing, Genz test functions, multidimensional integrals in Bayesian statistics and finding a solution to Richard Feinman's problem for evaluation the Wigner kernel in quantum mechanics.
3. To construct and study an improved Monte Carlo algorithm based on balancing of systematic error and probability error and to obtain the corresponding error balancing conditions.
4. To obtain an improved Monte Carlo method for linear algebraic systems based on „walk on equations“ algorithm [7].

5. To develop and analyze new compact finite difference schemes for semilinear parabolic problems in air pollution modelling. To make a comparison with fourth-order compact difference scheme and the fourth order scheme based on the standard difference scheme with Richardson extrapolation and to obtain a sixth-order approximation based on the fourth-order compact difference scheme combined with Richardson extrapolation.

Methodology

The methodology is based on the fundamental results from the following areas: functional analysis [15] - functional series (convergence, Neuman series, „truncation error“, Taylor series); probabilities [4] - random variables, Markov chains, unbiased and biased estimators, limit theorems; numerical analysis [1] - approximation, quadratures; numerical methods for partial differential equations and theory of difference schemes [3].

The source codes of the algorithms are written on MATLAB [28].

Publications

The main results are published in the following papers:

1. I.T. Dimov, R. Georgieva, V. Todorov, Balancing of Systematic and Stochastic Errors in Monte Carlo Algorithms for Integral Equations, 8th International Conference Numerical Methods and Applications, NMA 2014, Borovets, Bulgaria, August 20-24, 2014, Numerical Methods and Applications (I. Dimov, S. Fidanova, and I. Lirkov - Eds.), LNCS 8962, Springer, 2015, 44–51. DOI: 10.1007/978-3-319-15585-2_5. ISSN: 0302-9743. **SJR (2015): 0.252.**
2. I. Dimov, V. Todorov. Error Analysis of Biased Stochastic Algorithms for the Second Kind Fredholm Integral Equation, DOI:10.1007/978-3-319-32207-0_1, Innovative Approaches and Solutions in Advanced Intelligent Systems, ISSN: 1860949X, pp.3-16, **SJR (2015): 0.24.**
3. V. Todorov, I. Dimov. Monte Carlo methods for multidimensional integration for European option pricing, DOI:10.1063/1.4965003, AIP Conf. Proc. 1773, 100009, ISSN 0094243X, (2016), **SJR (2016): 0.198.**
4. Dimov I.,J. Kandilarov, V. Todorov, L. Vulkov. High-Order Compact Difference Schemes with Richardson Extrapolation for Semilinear Parabolic Systems. IN: Applications of Mathematics in Engineering and Economics, American Institute of Physics, 1789, 030002 (2016), DOI: 10.1063/1.4968448, **SJR (2016): 0.198.**
5. V. Todorov. Computing High Dimensional Integrals with Monte Carlo Methods, Journal Scientific and Applied Research, Journal Scientific and Applied Research, 10, 2016, 11-16, Konstantin Preslavsky Publishing House, ISSN 1314-6289.

The second paper is cited in the following article with IF:

1. Farshid Mirzaee, Nasrin Samadyar. Application of orthonormal Bernstein polynomials to construct a efficient scheme for solving fractional stochastic integro-differential equation, Optik - International Journal for Light and Electron Optics, **IF: 0.742**, Volume 132, March 2017, Pages 262–273, DOI:10.1016/j.ijleo.2016.12.029.

Monte Carlo methods for multidimensional integrals

Latin Hypercube Sampling

One of the main problems of some MC methods is that a sample that is very close to another does not provide much new information about the function being integrated. One powerful variance-reduction technique that addresses this problem is called stratified sampling. Stratified sampling works by splitting up the original integral into a sum of integrals over sub-domains. It has been shown that the stratified sampling can never result in higher variance than pure random sampling. Latin Hypercube Sampling (LHS) is a type of stratified sampling. In the context of statistical 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. Thus following the described algorithm, we obtain a set of points with positions forming a Latin square. Note that this sampling scheme does not require more samples for more dimensions (variables); this independence is one of the main advantages of LHS scheme. Below are given examples of random, stratified and Latin hypercube samplings with 16 points ($s = 2$, $M = 4$ in the LHS case, the figure was taken from [14]).

Adaptive Monte Carlo algorithm

The adaptive method does not use any a priori information about the smoothness of the integrand, but it uses a posteriori information about the variance. The idea of the method consists of the following (see for instance [2, 6, 12]): the domain of integration G is separated into subdomains with identical volume. The interval $[0;1]$ on every dimension coordinate is partitioned into M subintervals. The variance is compared with a preliminary given value ε . If the calculated variance in any region is greater than this constant ε , then this region is divided to new M^s subregions, where s is the dimensionality, again by partitioning the segment of the region on every coordinate to M subintervals. To avoid infinite loops or inefficiency we put a limit to the total number of regions we could have - an integer constant δ which is the maximum number of total regions we could have. In

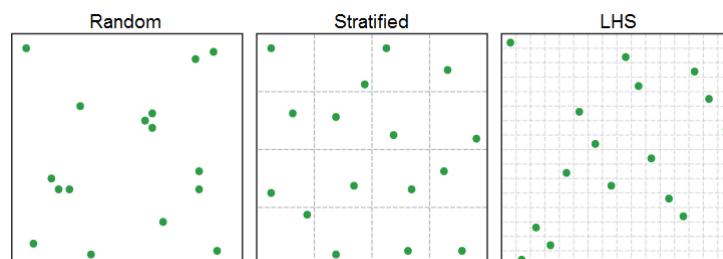


Figure 1: Comparison of the three sampling methods

the end, an approximation for the integral $I = \sum_j I_{G_j}$ is obtained.

Lattices and lattice rules

Quasi-Monte Carlo methods rely on sets of points that are evenly spread over the integration domain, i.e. these sets have small discrepancies. One effective way of obtaining such sets is to use the lattice point method. The most widely used types of lattice sets are the following [22]: $x_n = \{\frac{n}{N}z\}$, $n = 0, \dots, N - 1$. Here $z = (z_1, \dots, z_n)$ is the generating vector of dimensionality s which has no common divisor with N . Every component of the vector is replaced with its fractional part. Using the "lattice rule" then, we shall mean a rule of the form $I_N(f) = \frac{1}{N} \sum_{j=0}^{N-1} f(x_j)$, in which x_0, \dots, x_{N-1} are all the points of a multiple-integration lattice that lie in $[0, 1]^s$.

The most difficult part here is to find a good value of z , such that the points in the set are evenly distributed over the unit cube. Indeed, the choice of a good generating vector, which leads to small errors is not trivial. Complicated methods from theory of numbers are widely used, for example Zaremba's index or error of the worst function. Now, we will focus on one concrete vector with good properties. We will refer to the lattice method using this vector as Fibonacci lattice method or FIBO method. Consider the generating vector below for some natural number n [13]: $z = (1, F_n(2), \dots, F_n(s))$. Here, $F_n(j) = F_{n+j-1} - F_{n+j-2} - \dots - F_n$, where F_i are the corresponding generalized Fibonacci number of dimensionality s , i.e.: $F_{l+s} = F_l + F_{l+1} + \dots + F_{l+s-1}$, $l = 0, 1, \dots$ with initial conditions: $F_0 = F_1 = \dots = F_{s-2} = 0, F_{s-1} = 1$, $l = 0, 1, \dots$. After simplifying, one can see that the generating vector above is: $z = (1, F_{n-1} + F_{n-2} + \dots + F_{n-s+1}, \dots, F_{n-1} + F_{n-2}, F_{n-1})$. It follows easily that the computational complexity of the algorithm is linear, since the number of calculation required to obtain the generating vector is asymptotically $\mathcal{O}(\log F_n)$.

Comparison for multidimensional integrals of smooth functions

We have tested the methods under consideration on 5 and 30 dimensional integrals of smooth functions:

$$\int_{[0,1]^5} \exp(-100x_1x_2x_3)(\sin(x_4) + \cos(x_5))dx \approx 0.1854297367, \tag{1}$$

$$\int_{[0,1]^{30}} \frac{4x_1x_3^2e^{2x_1x_3}}{(1+x_2+x_4)^2} e^{x_5+\dots+x_{20}}x_{21}\dots x_{30}dx \approx 3.244540. \tag{2}$$

We have carried out some numerical experiments to compare the performance of Crude MC method, Adaptive MC algorithm, Sobol quasi-MC algorithm, Latin hypercube sampling (LHS method) and a particular QMC lattice rule based on generalized Fibonacci numbers (FIBO method) for integration of smooth functions with various dimensions. The last two methods have not been compared before and both are generally recommended in case of smooth integrands. The numerical results show (see Table 1 and Table 2) that the FIBO method is better than LHS method for low-dimensional integrals, while LHS outperforms FIBO when the integrand dimension is higher. The Sobol nets (their performance

Table 1: Relative error for 5 dimensional integral

| N | Crude | t,s | Adapt | t,s | FIBO | t,s | Sobol | t,s | LHS | t,s |
|--------|---------|-------|---------|-------|---------|-------|---------|------|---------|-------|
| 10^3 | 2.10e-2 | 0.007 | 2.15e-3 | 0.27 | 1.75e-4 | 0.007 | 5.29e-4 | 0.03 | 9.38e-3 | 0.007 |
| 10^4 | 4.52e-3 | 0.07 | 2.01e-3 | 2.43 | 1.28e-5 | 0.06 | 1.43e-4 | 0.3 | 3.44e-3 | 0.07 |
| 10^5 | 1.19e-3 | 0.64 | 6.91e-4 | 22.2 | 9.50e-6 | 0.61 | 2.36e-5 | 2.77 | 2.01e-3 | 0.69 |
| 10^6 | 8.47e-4 | 6.06 | 2.92e-4 | 219.5 | 5.47e-7 | 5.98 | 6.07e-6 | 24.2 | 1.80e-4 | 6.17 |
| 10^7 | 2.38e-4 | 59.9 | 8.21e-5 | 2043 | 8.71e-9 | 58.4 | 2.30e-6 | 245 | 2.46e-5 | 60.5 |

Table 2: Relative error for 30-dimensional integral

| N | Crude | time(s) | Adapt | time(s) | FIBO | time(s) | Sobol | time(s) | LHS | time(s) |
|--------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 10^3 | 8.56e-1 | 0.02 | 1.56e-1 | 2.27 | 8.73e-1 | 0.02 | 1.29e-1 | 0.27 | 2.31e-2 | 0.02 |
| 10^4 | 7.13e-1 | 0.1 | 6.91e-2 | 20.1 | 1.19e-2 | 0.18 | 8.56e-2 | 2.5 | 6.89e-3 | 0.19 |
| 10^5 | 4.21e-1 | 1.12 | 3.76e-2 | 229 | 2.78e-2 | 1.56 | 1.91e-2 | 20.2 | 1.65e-3 | 1.54 |
| 10^6 | 9.51e-2 | 11.07 | 6.29e-3 | 2271 | 9.56e-3 | 13.61 | 9.47e-3 | 208 | 9.61e-5 | 13.9 |

is given as a reference), are outperformed by at least one of the two discussed methods in any of the considered examples.

Genz test functions

Numerical experiments with this function family

$$f(x) = \left(1 + \sum_{i=1}^s a_i x_i\right)^{-(s+1)}$$

have been performed and the results are described in Table 3. This class of functions belongs to a test package given by Genz (see [10]). The numerical results in the table below confirm the assertion that the adaptive technique is effective for that class of functions that have computational specialty in the local subarea of the integration domain. The adaptive algorithm leads to smaller relative errors (even in 1-2 order) for smaller number of samples in comparison with other algorithms under consideration.

Wigner kernel

There is a new formulation of quantum mechanics in terms of signed classical field-less particles recently developed and it describes quantum objects in terms of classical particles only [21]. The three postulates which completely define the new mathematical formulation of quantum mechanics in terms of signed particles are enough to replicate the results of more conventional quantum theories. Our main interest is the second postulate:

Postulate II. A signed particle, evolving in a potential $V = V(x)$, behaves as a field-less classical point-particle which, during the time interval dt , creates a new pair of signed

Table 3: Relative error and computational time for the case $s = 5$, $I[f] = 2.1200e-06$, $a = (5, 5, 5, 5, 4)$; and for the case $s = 18$, $I[f] = 9.9190e-06$, $a = (\frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{1}{9}, \frac{1}{9})$.

| s | Adaptive alg. | | | LHS | | | FIBO | | |
|----|-----------------|-----------|----------|-----------------|-----------|----------|------------|-----------|----------|
| | N | rel.error | time (s) | N | rel.error | time (s) | N | rel.error | time (s) |
| 5 | 10 ² | 3.773e-03 | 0.33 | 10 ⁵ | 7.227e-02 | 0.27 | 1546352 | 9.714e-02 | 0.38 |
| | 10 ³ | 1.288e-03 | 1.44 | 10 ⁶ | 3.251e-02 | 1.22 | 3040048 | 6.750e-02 | 1.32 |
| | 10 ⁴ | 4.245e-04 | 10.75 | 10 ⁷ | 2.520e-03 | 12.3 | 11749641 | 1.538e-02 | 15.1 |
| | 10 ⁵ | 4.717e-05 | 142.1 | 10 ⁸ | 1.664e-03 | 124.2 | 175514464 | 2.925e-03 | 134.6 |
| 18 | 10 | 9.234e-04 | 15.7 | 10 ⁷ | 8.628e-03 | 13.6 | 16776960 | 7.157e-02 | 14.7 |
| | 10 ² | 8.065e-05 | 142 | 10 ⁸ | 5.119e-03 | 140 | 134214912 | 5.109e-02 | 144.1 |
| | 10 ³ | 1.008e-05 | 1408 | 10 ⁹ | 1.628e-03 | 1353.5 | 1073713152 | 2.888e-02 | 1344.3 |

particles with a probability $\gamma(x(t))dt$ where

$$\gamma(x) = \int_{-\infty}^{+\infty} Dp' V_W^+(x; p') \equiv \lim_{\Delta p' \rightarrow 0^+} \sum_{M=-\infty}^{+\infty} V_W^+(x; M \Delta p'),$$

and $V_W^+(x; p)$ is the positive part of the quantity

$$V_W(x; p) = \frac{i}{\pi^s \hbar^{s+1}} \int_{-\infty}^{+\infty} dx' e^{-\frac{2i}{\hbar} x' p} [V(x+x') - V(x-x')], \quad (3)$$

known as the Wigner kernel in a s -dimensional space [25]. If, at the moment of creation, the parent particle has sign s , position x and momentum p , the new particles are both located in x , have signs $+k$ and $-k$, and momentum $p+p'$ and $p-p'$ respectively, chosen randomly according to the (normalized) probability $\frac{V_W^+(x;p)}{\gamma(x)}$. We want to compute (3) in the 3, 6, 9 dimensional case, where the Wigner potential is $V = V(x) = \{x_1 \dots x_n, x', x, p, x+x', x-x' \in [0, 1]\}$.

The results are given in Table 4. This is the first time Adaptive MC algorithm is applied to this problem. It gives superior results in comparison with the other stochastic algorithms under consideration. To acquire a better intuition about the Wigner kernel we have plotted it as a function of x and p in one dimensional case after integrating on x' using the adaptive and Crude MC algorithm (see Figure 3). The behavior of Wigner kernel explains the advantage and the efficiency of the Adaptive algorithm for this case study. The particle signs of the Wigner kernel in one dimensional case are presented on Figure 2.

Application for European option pricing

Monte Carlo methods can be very useful in such cases if the solution (i.e., the value, V) can be expressed as the mathematical expectation of some random variables. This is made possible by the risk-neutral valuation formula for the European option [5]: $V(S, t) = E(e^{r(T-t)} h(S(T)) \mid S(t) = S, \mu = r)$, where $E(\cdot)$ is the mathematical expectation, $h(S)$ is

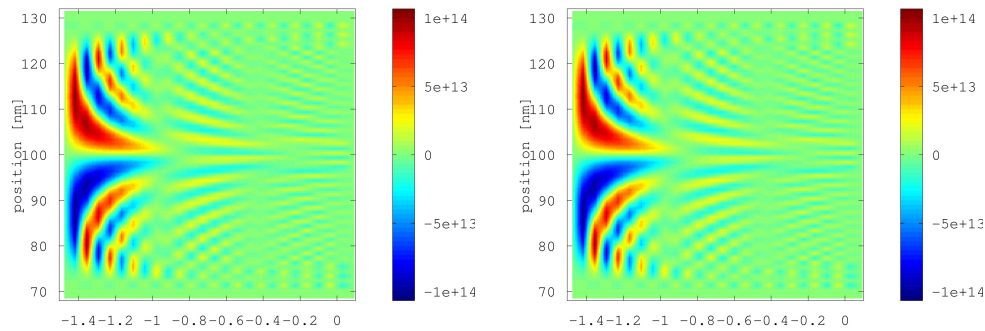


Figure 2: The position of the particle signs in the Wigner kernel: (a) with deterministic method; (b) with stochastic method.

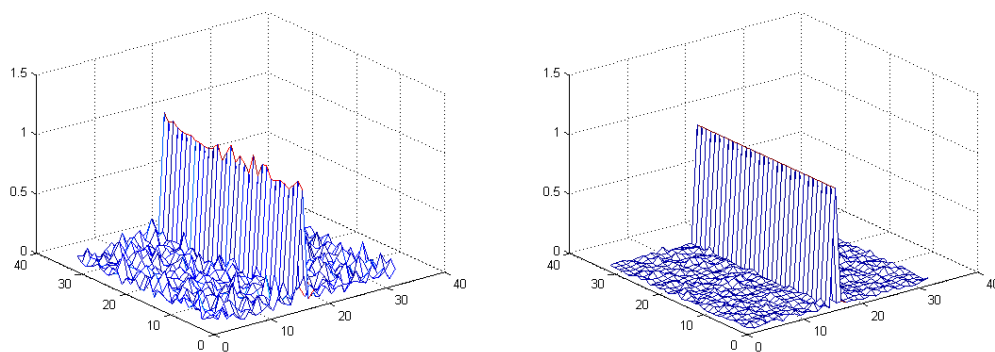


Figure 3: The Wigner kernel, obtained with ordinary MC and adaptive MC approach for 100 points in x' and $p, x, x' \in [0, 1]$

the payoff function, $h(S) = \max(S - E, 0)$ for a call option and $h(S) = \max(E - S, 0)$ for a put option. We follow the idea of Lai and Spanier in [17]. Consider an European option whose payoff depends on $k > 1$ assets with prices $S_i, i = 1, \dots, k$. Each asset follows the random walk $dS_i = \mu_i S_i dt + \sigma_i S_i dX_i$, where σ_i is the annual standard deviation for the i -th asset and dX_i is Brownian motion. Suppose that the payoff is given by $h(S'_1, \dots, S'_k)$ at expiry time T (where S' denotes the value of the i -th asset at expiry). Then the current value, V , of the option (assuming risk neutrality) will be

$$V = e^{-r(T-t)} (2\pi(T-t))^{-k/2} (\det \Sigma)^{-1/2} (\sigma_1, \dots, \sigma_k)^{-1} \times \\ \times \int_0^\infty \dots \int_0^\infty \frac{h(S'_1, \dots, S'_k)}{S'_1, \dots, S'_k} \exp(-0.5\alpha^T \Sigma^{-1} \alpha) dS'_1, \dots, dS'_k,$$

where $\alpha_i = (\sigma_i(T-t)^{1/2})^{-1} (\log \frac{S'_i}{S_i} - (r - \frac{\sigma_i^2}{2})(T-t))$, r is the risk-free interest rate and Σ is the covariance matrix, where the (i, j) -th entry is the covariance of dX_i and dX_j for k assets. The infinite domain of integration can be mapped into the k -dimensional unit hypercube in

Table 4: Relative error for 3, 6 and 9 dimensional integral

| | N | LHS | t,s | Adaptive | t,s | FIBO | t,s | Sobol | t,s |
|-------|-----------------|----------|------|----------|------|----------|------|----------|------|
| s = 3 | 10 ³ | 4.38e-03 | 0.01 | 6.75e-04 | 0.4 | 3.72e-02 | 0.02 | 1.07e-02 | 0.05 |
| | 10 ⁴ | 7.94e-04 | 0.06 | 8.15e-05 | 3.3 | 7.06e-03 | 0.07 | 8.77e-03 | 0.54 |
| | 10 ⁵ | 2.51e-04 | 0.41 | 5.01e-06 | 32.6 | 3.40e-03 | 0.43 | 8.57e-04 | 5.74 |
| | 10 ⁶ | 8.20e-05 | 3.52 | 4.38e-07 | 302 | 1.01e-03 | 4.4 | 6.73e-04 | 45.6 |
| | 10 ⁷ | 1.26e-05 | 35.4 | 9.02e-08 | 2708 | 1.80e-04 | 49.7 | 5.98e-05 | 409 |
| s = 6 | 10 ³ | 1.54e-03 | 0.01 | 2.23e-04 | 0.5 | 7.82e-03 | 0.01 | 2.42e-02 | 0.09 |
| | 10 ⁴ | 6.34e-04 | 0.06 | 4.74e-05 | 4.1 | 5.01e-03 | 0.07 | 5.02e-03 | 0.78 |
| | 10 ⁵ | 4.22e-04 | 0.44 | 5.43e-06 | 37 | 6.88e-03 | 0.43 | 4.60e-04 | 6.19 |
| | 10 ⁶ | 8.57e-05 | 3.7 | 5.04e-07 | 351 | 7.68e-04 | 5.97 | 3.59e-04 | 53 |
| | 10 ⁷ | 3.03e-05 | 38 | 7.12e-08 | 2841 | 4.12e-04 | 48 | 8.11e-05 | 430 |
| s = 9 | 10 ³ | 6.11e-03 | 0.04 | 8.23e-04 | 0.5 | 2.03e-02 | 0.06 | 5.42e-02 | 0.11 |
| | 10 ⁴ | 1.02e-03 | 0.06 | 2.02e-05 | 4.7 | 2.02e-03 | 0.07 | 6.02e-03 | 0.88 |
| | 10 ⁵ | 4.69e-04 | 0.43 | 1.08e-06 | 40 | 9.16e-04 | 0.53 | 3.57e-03 | 6.56 |
| | 10 ⁶ | 8.08e-05 | 3.8 | 4.14e-07 | 381 | 7.13e-04 | 3.7 | 8.02e-04 | 57 |
| | 10 ⁷ | 1.76e-05 | 42 | 8.91e-08 | 2912 | 4.84e-04 | 40 | 5.19e-04 | 451 |

a variety of ways. For example, $\frac{2}{\pi} \arctan(x)$ maps $(0, \infty)$ to $(0, 1)$. Such a mapping transforms the initial problem to a problem, where an integral $\int_0^1 \dots \int_0^1 g(x_1, \dots, x_k) dx_1 \dots dx_k$ over the hypercube is sought. When g is the exponent function with appropriate choices of the constants involved in the equation, we obtain k dimensional integral $\int_{[0,1]^k} \exp(x_1, \dots, x_k) dx_1 \dots dx_k$ for the value of the option according to [17]. Numerical experiments for 5 and 20-dimensional integral are performed.

$$\int_{[0,1]^5} \exp\left(\sum_{i=1}^5 0.5a_i x_i^2 (2 + \sin \sum_{j=1, j \neq i}^5 x_j)\right) \approx 2.923651, a_i = (1, 0.5, 0.2, 0.2, 0.2), \quad (4)$$

$$\int_{[0,1]^{20}} \exp\left(\prod_{i=1}^{20} x_i\right) \approx 1.00000949634. \quad (5)$$

The results are presented on Figure 4 and Figure 5. It can be seen that for lower dimensions FIBO is better, but for higher dimensions the performance of Sobol quasi-random sequence is very similar to FIBO.

Monte Carlo algorithms for integral equations

We study the Fredholm integral equation of the second kind:

$$u(x) = \int_{\Omega} k(x, x')u(x') dx' + f(x) \text{ or } u = \mathcal{K}u + f, \quad (6)$$

where $x, x' \in \Omega \subset \mathbb{R}^d$, $u(x), f(x) \in L_2(\Omega)$, $k(x, x') \in L_2(\Omega \times \Omega)$ and \mathcal{K} is the integral operator. We want to construct Monte Carlo algorithm to evaluate the linear functional of the solution, denoted by the following expression: $J(u) = \int \varphi(x)u(x)dx = (\varphi, u)$. We

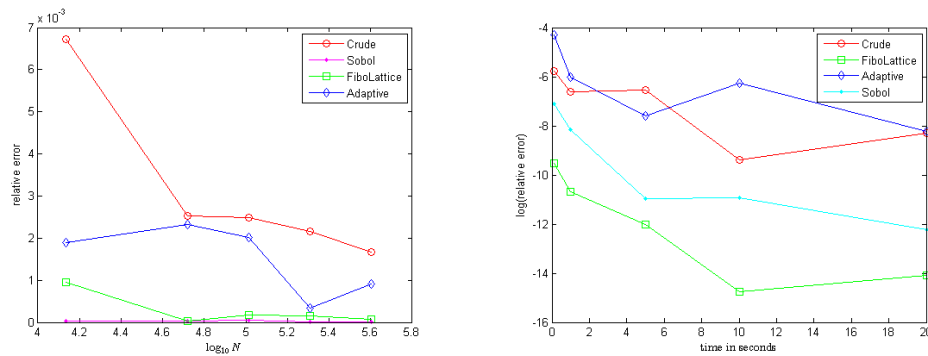


Figure 4: Relative error for 5-dimensional integral.

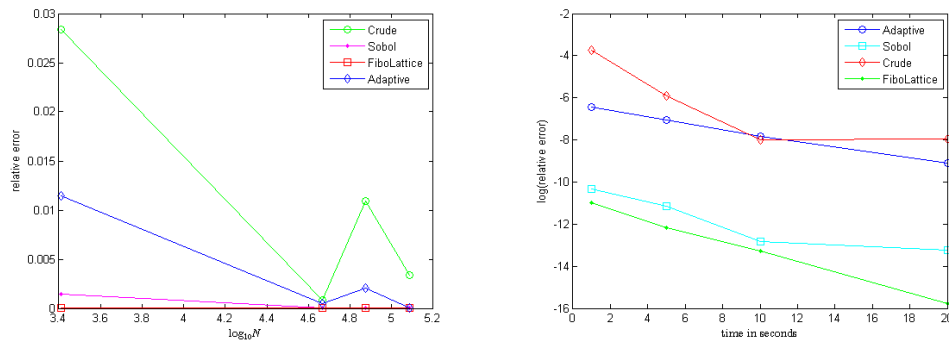


Figure 5: Relative error for 20 dimensional integrals

assume that $\varphi(x) \in L_2(\Omega)$. We construct Monte Carlo method for integral equations based on discrete Markov chains. We define a set of permissible densities: $\pi(x), p(x, x') : \pi(x) \geq 0, p(x, x') \geq 0, \int_{\Omega} \pi(x) dx = 1, \int_{\Omega} p(x, x') dx' = 1, x \in \Omega \subset \mathbb{R}^d$. In correspondence with the initial and transition probabilities we define a discrete Markov chain $T_k : x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_k$ with length k started from the initial state x_0 . Let $u^{(0)} \equiv f$ is the approximate initial solution. It is well known that Monte Carlo algorithm for integral equations is given by the following expressions [23]:

$$E\theta_k[\varphi] = \left(\varphi, u^{(k)}\right), \theta_k[\varphi] = \frac{\varphi(x_0)}{\pi(x_0)} \sum_{j=0}^k W_j f(x_j),$$

$$W_0 = 1, W_j = W_{j-1} \frac{k(x_{j-1}, x_j)}{p(x_{j-1}, x_j)}, j = 1, \dots, k,$$

$$\left(\varphi, u^{(k)}\right) \approx \frac{1}{N} \sum_{n=1}^N \theta_k[\varphi]_n.$$

For the probable error we have that:

$$r_N \leq \frac{0.6745 \|f\|_{L_2} \|\varphi\|_{L_2}}{\sqrt{N} (1 - \|\mathcal{K}\|_{L_2})}.$$

Table 5: Relative error and computational time for the example with balancing of two types of errors (different initial and transition probabilities).

| δ | N | k | expected rel. error | experimental rel. error | time (sec.) | experimental rel. error | time (sec.) |
|----------|--------|---|------------------------|----------------------------|----------------|----------------------------|----------------|
| 0.4 | 865 | 3 | 0.0445 | 0.0052 | 3 | 0.0239 | 5 |
| 0.2 | 3457 | 4 | 0.0223 | 0.0094 | 9 | 0.0121 | 23 |
| 0.1 | 13827 | 4 | 0.0111 | 0.0113 | 28 | 0.0086 | 46 |
| 0.05 | 55306 | 5 | 0.00556 | 0.0177 | 132 | 0.0032 | 222 |
| 0.028 | 176357 | 5 | 0.00312 | 0.0176 | 448 | 0.0031 | 540 |
| 0.02 | 345659 | 6 | 0.00233 | 0.0202 | 901 | 0.0013 | 1090 |

For the systematic error we obtain that:

$$r_k = \left| (\varphi, u) - (\varphi, u^{(k)}) \right| \leq \|\varphi\|_{L_2} \|u - u^{(k)}\|_{L_2} \leq \frac{\|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}^{k+1}}{1 - \|\mathcal{K}\|_{L_2}}.$$

Let δ is the preliminary given error to solve the problem under consideration. We suppose that

$$r_N \leq \frac{0.6745 \|\varphi\|_{L_2} \|f\|_{L_2}}{\sqrt{N} (1 - \|\mathcal{K}\|_{L_2})} \leq \frac{\delta}{2}, \quad r_k \leq \frac{\|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}^{k+1}}{1 - \|\mathcal{K}\|_{L_2}} \leq \frac{\delta}{2}.$$

(Theorem for error balancing conditions.) The lower bounds for N and k for the Monte Carlo algorithm with balancing of the errors are:

$$N \geq \left(\frac{1.349 \|\varphi\|_{L_2} \|f\|_{L_2}}{\delta (1 - \|\mathcal{K}\|_{L_2})} \right)^2, \quad k \geq \frac{\ln \frac{\delta (1 - \|\mathcal{K}\|_{L_2})}{2 \|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}}}{\ln \|\mathcal{K}\|_{L_2}}.$$

The balancing of errors (both, systematic and stochastic) allows to get an approximation of the quantity of interest in the most efficient way by fixing the number of samples N and the number of iterations k if the error is fixed. The error balancing conditions play an important role in reducing the computational complexity of the algorithm.

Numerical experiments

We study the following example taken from neuron networking:

$$u(x) = \int_{\Omega} k(x, x') u(x') dx' + f(x),$$

$\Omega \equiv [-2, 2]$, $k(x, x') = \frac{0.055}{1 + e^{-3x}} + 0.07$, $f(x) = 0.02(3x^2 + e^{-0.35x})$, $\varphi(x) = 0.7((x + 1)^2 \cos(5x) + 20)$. We would like to find (φ, u) , where $\varphi(x) = 0.7((x + 1)^2 \cos(5x) + 20)$. The exact solution is 8.98635750518. This integral equation describes the procedure of teaching of neuron networks. In order to apply the theorem we evaluate the L_2 norms: $\|\varphi\|_{L_2} = 27.7782$, $\|\mathcal{K}\|_{L_2} = 0.2001$, $\|f\|_{L_2} = 0.2510$.

In Table 5 the preliminary given error δ and different estimates of N and k using the theorem are presented. The first two columns with the expected relative error and

computational time in seconds, correspond to the case when the initial and transition probabilities are constant functions. The last two columns correspond to the case when the initial and transition probabilities are chosen to be proportional to the function from the linear functional under consideration and the kernel, respectively, and that leads to an almost optimal Monte Carlo algorithm (MAO). One can see that the MAO gives much better results than the Monte Carlo algorithm with constant transition probabilities for larger values of N and k . For smaller values the Monte Carlo algorithm with constant initial and transition probabilities gives better results, but the results obtained with MAO are closer to the expected theoretical error. It means that experimental relative error of MAO algorithm confirms the expected relative error and this algorithm leads to more reliable results.

Monte Carlo methods for linear systems

Description of the algorithm

We consider a system of LA equations

$$Bx = f, \quad (7)$$

where $B = \{b_{ij}\}_{i,j=1}^n \in \mathbb{R}^{n \times n}$ is a given matrix; $f = (f_1, \dots, f_n)^t \in \mathbb{R}^{n \times 1}$ and $v = (v_1, \dots, v_n) \in \mathbb{R}^{1 \times n}$ are given vectors.

We deal with the matrix $A = \{a_{ij}\}_{i,j=1}^n$, such that $A = I - DB$, where D is a diagonal matrix $D = \text{diag}(d_1, \dots, d_n)$ and $d_i = \frac{\gamma}{b_{ii}}$, $i = 1, \dots, n$, and $\gamma \in (0, 1]$ is a parameter that can be used to accelerate the convergence. The system can be presented in the form of equation

$$x = Ax + b, \quad (8)$$

where $b = Df$. Let us suppose that the matrix B is diagonally dominant. Obviously, if B is a diagonally dominant matrix, then the elements of the matrix A must satisfy the following condition: $\sum_{j=1}^n |a_{ij}| \leq 1$, $i = 1, \dots, n$. A new Monte Carlo algorithm for solving systems of Linear Algebraic (LA) equations is presented and studied. The algorithm is based on the "Walk on Equations" Monte Carlo method recently developed by Ivan Dimov, Sylvain Maire and Jean Michel Sellier [7]. The idea is to compute scores for all the states (seen as new starting states) that are visited during a given trajectory. The initial equation is picked uniformly at random among the n equations. After that for each state i we define the total score $S(i)$ and the total number of visits $V(i)$ that are modified as soon as state i is visited during a walk. For a given trajectory, we store the visited states in a list l to compute the contributions to the score of these visited states. Finally, the solution x_i is approximated by the total score $S(i)$ divided by the total number of visits $V(i)$.

Numerical experiments for linear systems

Example 1. We try to find the solutions x_1 and x_2 defined by the linear systems of algebraic equations $Ax = b$, where the matrix B and the vectors b_1 and b_2 are:

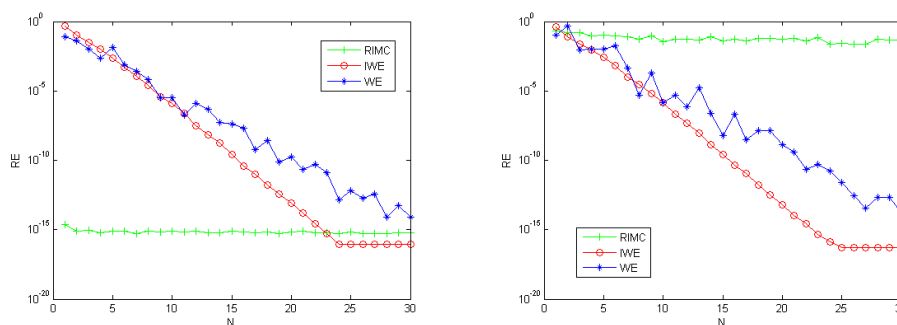
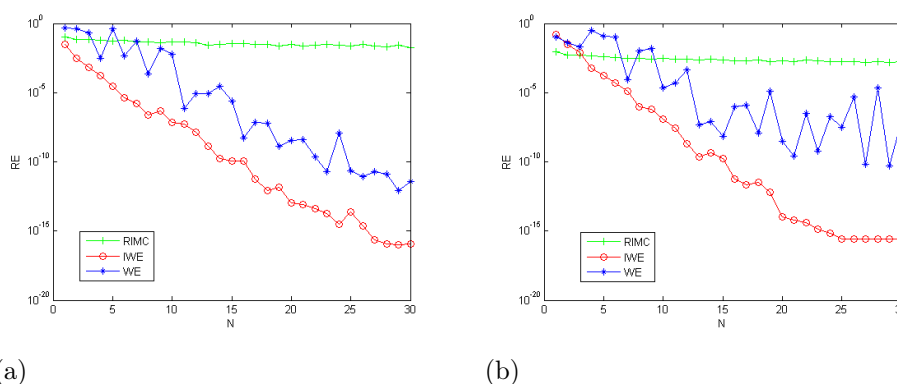


Figure 6: Weighted residual of the solution for matrix 7×7 : (a) for x_1 ; (b) for x_2 .



(a)

(b)

Figure 7: Weighted residual of the solution for the matrix: (a) 100×100 ; (b) 5000×5000 .

$$A = \begin{pmatrix} 5 & -1 & -1 & 0 & 0 & -1 & -1 \\ -1 & 5 & -1 & -1 & 0 & 0 & -1 \\ -1 & -1 & 5 & -1 & -1 & 0 & 0 \\ 0 & -1 & -1 & 5 & -1 & -1 & 0 \\ 0 & 0 & -1 & -1 & 5 & -1 & -1 \\ -1 & 0 & 0 & -1 & -1 & 5 & -1 \\ -1 & -1 & 0 & 0 & -1 & -1 & 5 \end{pmatrix}, \quad b_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 4 \\ -2 \\ -1 \\ 0 \\ -1 \\ -2 \\ 4 \end{pmatrix}. \quad (9)$$

Example 2. Let B is the matrix NOS4 from the Harwell-Boeing Collection [27], and $b \in \mathbb{R}^{100}, b_i = 1, i = 1, \dots, 100$. This particular matrix is taken from an application connected to finite element approximation of a problem describing a beam structure in constructive mechanics [27].

Example 3. Let B is a dense matrix 5000×5000 with elements in $[0,1]$, and $b \in \mathbb{R}^{5000}, b_i = 1, i = 1, \dots, 5000$.

Comparison of relative errors and computational times for a fixed number of iterations between refined iterative MC method (RIMC), walk on equations (WE) and our method (IWE) is shown. The matrix B and the right-hand side b are normalized to accelerate the convergence rate of the stochastic process. Special values of the relaxation parameter γ has been chosen. Numerical experiments show that it leads to balancing of the iteration matrix A . It can be seen that for the 7 dimensional case (see Figure 6) the difference in

the accuracy between the WE and OWE for a given number of iterations is 2-3 order for $N > 15$. It is worth mentioning that the refined iterative Monte Carlo convergence is very slow except for the trivial solution x_1 of Example 1. The prior behavior of the proposed MC algorithm does not depend on the matrix density. The matrix NOS4 has only 5.9 average nonzeros per row and per column. The advantages of the algorithm hold for dense matrices. The choice of relaxation parameters leads to the balancing of iteration matrix. The advantages of the proposed MC algorithm can be observed especially for larger matrix size (see Figure 7).

New high order compact finite difference schemes for models in air pollution

Various processes in chemistry, physics and engineering are described by models consisting from systems of coupled parabolic problems. Here we construct compact high-order finite difference schemes for semilinear parabolic systems. Problems of air pollution transport with coupling in the nonlinear reactions terms are of our main consideration, namely,

$$\frac{\partial u_l}{\partial t} - K \Delta u_l + \mathbf{b}_l \nabla u_l = R_l(x, y, u_1, \dots, u_L), \quad (x, y, t) \in \Omega \times (0, T], \quad (10)$$

$$\mathbf{u} = 0, \quad (x, y, t) \in \partial\Omega \times (0, T], \quad (11)$$

$$\mathbf{u} = \mathbf{u}_0(x, y), \quad (x, y) \in \Omega, \quad (12)$$

where $\mathbf{u} = (u_1, u_2, \dots, u_L)$, $u_l = u_l(x, y, t)$, $l = 1, \dots, L$ are the concentrations of L chemical species (pollutants), $K > 0$ is the diffusion coefficient and $\Omega \in \mathbb{R}^2$ is a bounded domain. The assumption regarding the constant $K := K_x = K_y$ is not a restriction for developing our numerical approach. This just corresponds to the physical model described in [16]. Following [26, 16] the advection part in (10) may be presented in the following form:

$$\mathbf{b}_l \cdot \nabla u_l = \mu(y - y_c) \frac{\partial u_l}{\partial x} + \mu(x_c - x) \frac{\partial u_l}{\partial y},$$

where $x \in (0, X)$, $y \in (0, Y)$, $x_c = X/2$, $y_c = Y/2$. The nonlinear chemical part of the model is (see [16]) given in Table 6:

Table 6: The nonlinear chemical part of the model

| | | | |
|-------|-----------------------------------------------------------|----------|----------------------------------------------------------------|
| R_1 | $k_5 u_2 - (k_6 u_5 + k_4 u_7 + k_3 u_8) u_1$ | R_6 | $k_9 u_2 u_9$ |
| R_2 | $(k_6 u_5 + k_4 u_7 + k_3 u_8) u_1 - (k_5 + k_9 u_9) u_2$ | R_7 | $k_2 u_4 + k_3 u_1 u_8 + k_{10} u_9 - k_4 u_1 u_7$ |
| R_3 | $-k_1 u_3 u_9$ | R_8 | $4k_1 u_3 u_9 - k_3 u_1 u_8$ |
| R_4 | $2k_1 u_3 u_9 + k_3 u_1 u_8 - k_2 u_4$ | R_9 | $k_4 u_1 u_7 + 2k_8 u_{10} - (k_1 u_3 - k_9 u_2 + k_{10}) u_9$ |
| R_5 | $k_2 u_5$ | R_{10} | $k_7 u_5 - k_8 u_{10}$ |

The chemical part of the model is given in Table 7 for the sake of completeness. The rate coefficients can be found in Table 8. Some of the coefficients belong to photochemical reactions (the ones with term $h\nu$), which means that these reactions depend on the light,

Table 7: The chemical reactions of the model

| | | | |
|---|-------------------------------------------|----|---------------------------------------|
| 1 | $HC + OH \rightarrow 4RO_2 + 2ALD$ | 6 | $NO + O_3 \rightarrow NO_2 + O_2$ |
| 2 | $ALD + h\nu \rightarrow 2HO_2 + CO$ | 7 | $O_3 + h\nu \rightarrow O_2 + O(^1D)$ |
| 3 | $RO_2 + NO \rightarrow NO_2 + ALD + HO_2$ | 8 | $O(^1D) + H_2O \rightarrow 2OH$ |
| 4 | $NO + HO_2 \rightarrow NO_2 + OH$ | 9 | $NO_2 + OH \rightarrow HNO_3$ |
| 5 | $NO_2 + h\nu \rightarrow NO + O_3$ | 10 | $CO + OH \rightarrow CO_2 + HO_2$ |

Table 8: The coefficients of the chemical reactions

| | | | |
|-------|----------------------------------------|----------|---------------------------------------|
| k_1 | $6.0e - 12$ | k_6 | $1.6e - 14$ |
| k_2 | $7.8e - 05. \exp(-0.87 / \cos \theta)$ | k_7 | $1.6e - 04. \exp(-1.9 / \cos \theta)$ |
| k_3 | $8.0e - 12$ | k_8 | $2.3e - 10$ |
| k_4 | $8.0e - 12$ | k_9 | $1.0e - 11$ |
| k_5 | $1.0e - 02. \exp(-0.39 / \cos \theta)$ | k_{10} | $2.9e - 13$ |

more precisely on the position of the Sun relative to the horizon: in k_2 , k_5 and k_7 the angle θ denotes the solar zenith angle, which is the angle of the Sun measured from vertical. The chemical species involved in the simplified reactions are written in Table 9. In this study

Table 9: The chemical species in the model

| | | | | | | | | | |
|-------|--------|-------|-------|-------|---------|--------|--------|-------|----------|
| u_1 | u_2 | u_3 | u_4 | u_5 | u_6 | u_7 | u_8 | u_9 | u_{10} |
| NO | NO_2 | HC | ALD | O_3 | HNO_3 | HO_2 | RO_2 | OH | $O(^1D)$ |

we use the following idea for compact difference schemes: we operate on the differential equations as auxiliary relations in order to express high-order derivatives in the truncation error [24]. More details and discussions on construction of compact difference schemes for convection-diffusion problems can be found in [24].

Example 1 (known analytical solution)

Here we consider a problem slightly different from the problem (10)-(12):

$$\frac{\partial u_l}{\partial t} - K \Delta u_l + \mathbf{b}_l \cdot \nabla u_l = R_l(x, y, \mathbf{u}) + \xi_l(x, y, t), (x, y, t) \in \Omega \times (0, T].$$

The functions ξ_l , $l = 1, \dots, 10$, and the initial and boundary conditions are chosen so that the exact solution is

$$u_l = \exp(-t/T) \sin\left(\frac{\pi x}{X}\right) \sin\left(\frac{\pi y}{Y}\right), \quad l = 1, \dots, 10, \quad (x, y, t) \in \bar{\Omega} \times [0, T].$$

The other parameters are as follows: $X = Y = 500$, $T = 1440$, $\mu = 2\pi/(60T)$, $K = 1.8$. For the l^{th} substances with $error_{M,l}$ we denote the error (the difference between the exact and the numerical solution) in maximum norm, obtained on the last time layer $t_N = T$ for the number of space subintervals $M_x = M_y = M$: $error_{M,l} = \max_{i,j \in \bar{\Omega}_h} \|u_l(x_i, y_j, t_N) - u_l^h(i, j, N)\|$. The ratio between the errors obtained on two consecutive mesh refinements (usually doubling) is denoted by $ratio$: $ratio = ratio_{M,l/2M,l} =: error_{M,l}/error_{2M,l}$.

Table 10: Comparison of the maximum absolute errors of the CDS and CFDS for **Example 1**

| CDS, $O(h^2 + \tau^2)$ | | | | | | CFDS, $O(h^4 + \tau^2)$ | | | | | |
|------------------------|-------|-----|------------|---------|-------|-------------------------|-------|------|------------|---------|--------|
| M_x | M_y | N | $error_M$ | $ratio$ | CPU | M_x | M_y | N | $error_M$ | $ratio$ | CPU |
| 4 | 4 | 4 | 5.702 e-03 | - | 0.58 | 4 | 4 | 4 | 5.875 e-03 | - | 0.72 |
| 8 | 8 | 8 | 1.449 e-03 | 3.94 | 1.82 | 8 | 8 | 16 | 3.595 e-04 | 16.34 | 3.04 |
| 16 | 16 | 16 | 3.637 e-04 | 3.99 | 14.42 | 16 | 16 | 64 | 2.232 e-05 | 16.11 | 29.74 |
| 32 | 32 | 32 | 9.102 e-05 | 4.001 | 143.7 | 32 | 32 | 256 | 1.392 e-06 | 16.03 | 1076 |
| 64 | 64 | 64 | 2.276 e-05 | 4.00 | 3959 | 64 | 64 | 1024 | 8.698 e-08 | 16.003 | 60907 |
| 128 | 128 | 128 | 5.691 e-06 | 4.00 | 32709 | 128 | 128 | 4096 | 5.436 e-09 | 16.0001 | 720477 |

In Tables 11 and 12 the mesh refinement analyses using CDS and CFDS, and CDS and CFDS with Richardson extrapolation (RE) in space and time are presented. The results confirm the expected rates of convergence for both numerical methods. The comparison of the results about CPU time in Tables 10 and 11 shows a priority of using RE both in space and time obtaining smaller errors for comparatively same computational time. The advantage of CFDSRE is also clearly seen.

Table 11: Comparison of the errors in maximum norm for the numerical **Example 1** for CDS and CFDS with Richardson extrapolation in space and time

| CDS with RE in space and time, $O(h^4 + \tau^4)$ | | | | | | CFDS with RE in space and time $O(h^6 + \tau^4)$ | | | | | |
|--------------------------------------------------|-------|----|------------|-------|--------|--------------------------------------------------|-------|------|------------|-------|--------|
| M_x | M_y | N | err_N | ratio | CPU | M_x | M_y | N | err_N | ratio | CPU |
| 4 | 4 | 4 | 5.649 e-05 | - | 6.73 | 4 | 4 | 4 | 8.476 e-06 | - | 3.36 |
| 8 | 8 | 8 | 9.722 e-06 | 5.81 | 18.71 | 8 | 8 | 16 | 1.748 e-07 | 48.49 | 30.26 |
| 16 | 16 | 16 | 5.989 e-07 | 16.23 | 194.81 | 16 | 16 | 64 | 2.847 e-09 | 61.39 | 1276 |
| 32 | 32 | 32 | 3.715 e-08 | 16.12 | 4594 | 32 | 32 | 256 | 4.529 e-11 | 62.86 | 66991 |
| 64 | 64 | 64 | 2.171 e-09 | 16.03 | 37101 | 64 | 64 | 1024 | 7.086 e-13 | 63.91 | 790800 |

Example 2 (no exact solution)

In this case we consider more realistic variant of problem (10)-(12) with the following parameters of the domain: the spatial domain is the square $\Omega = [0, 500]^2$ with side length 500 km, the length of the time interval $[0, T]$ is 1440 min and the number of equations is $L = 10$. The initial conditions on the time level $t = 0$ are the constant functions $\mathbf{u}_0(x, y) = (10^3, 10^3, 10^3, 5.10^3, 5.10^3, 10^2, 10^{-2}, 10^{-2}, 10^{-3}, 10^{-11})$, measured in mol/km^3 and the boundary conditions are chosen to be periodic: γ_i has the form $\gamma_l(t) = const_l(\sin(t/C) + 2)$, where $C = 4$ is a constant and the constants $const_l$, $l = 1, \dots, L$ are chosen in such a way that the compatibility of the boundary and initial data is ensured. The diffusion coefficient is set to be $K = 1.8 km^2/min$ and the coefficient μ is $\mu = 2\pi/(60 * T)$.

The log-log plot of the errors versus space mesh size for the *Example 2* is presented on Fig. 8. The increasing of the slope of the lines corresponds with the increasing of the rate of convergence. The lowest line confirms the advantage of the CFDS in combination with Richardson extrapolation.

The numerical solutions obtained by CFDS for $\mu = 2\pi/(60T)$ with mesh parameters $M_x = M_y = 32$, at final time layer $N = 256$ (a) for u_1 ; (b) for u_5 are shown on Fig. 9.

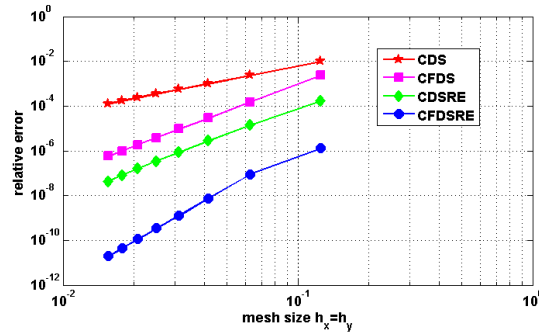


Figure 8: The log-log plot of the errors versus space mesh size for the *Example 2*.

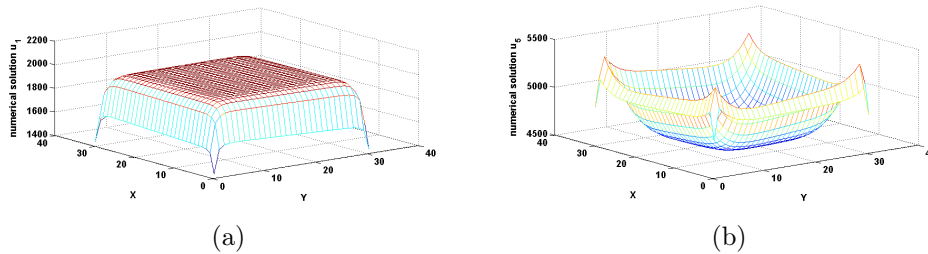


Figure 9: Numerical solution obtained by CFDS for $\mu = 2\pi/(60T)$ with mesh parameters $M_x = M_y = 32$, $N = 256$ *Example 2*: (a) for u_1 ; (b) for u_5

It is interesting to see what is the behaviour of the solutions if the coefficient μ in the convection term is taken to be $\mu = 2\pi/(X)$ as it is in [16] instead $\mu = 2\pi/(60 * T)$ as it is in [11]. The increasing of the convective coefficients leads to significant change of the numerical solution near the corners, see Fig. 10, where $\mu = 2\pi/500$. It can be seen that the constant initial values have been left relatively intact in the middle of the domain, but they have been stretched near the boundary by the sinusoidal boundary conditions.

Comparison with Monte Carlo method for linear systems

We consider the one dimensional problem in $\Omega = (0, 1) \times (0, T)$ with parameters $a = b = c = d = 1$, $f(x, t, u, v) = u(1 - u - v) + \xi_1(x, t)$, $g(x, t, u, v) = v(1 - u - v) + \xi_2(x, t)$, $T = 1$,

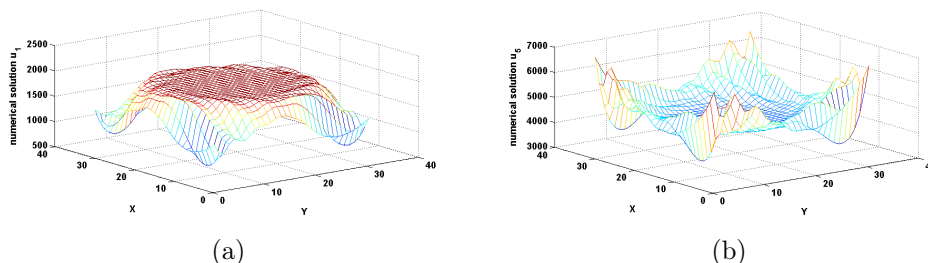


Figure 10: Numerical solution for *Example 2*, obtained by CFDS with $\mu = 2\pi/(500)$ and $M_x = M_y = 32$, $N = 256$: (a) for u_1 ; (b) for u_5

where functions ξ_1 and ξ_2 are chosen, so that the analytical solutions are $u = e^{-t} \sin(\pi x)$ and $v = e^{-t} x(1 - x)$.

The comparison between the computational time for the fastest Matlab linear solver `bicgstabl` (CPU(b)) and the IWE method (CPU(mc)) is given in Table 12. The `bicgstabl` method is at least four times faster than the MC method.

Table 12: Error in maximum norm for the numerical example

| CDS, $O(h^2 + \tau^2)$ | | | | | | CFDS, $O(h^4 + \tau^2)$ | | | | | |
|------------------------|-----|-----------|-------|--------|---------|-------------------------|-------|-----------|-------|--------|---------|
| M | N | err_N | ratio | CPU(b) | CPU(mc) | M | N | err_N | ratio | CPU(b) | CPU(mc) |
| 10 | 10 | 3.82 e-03 | - | 0.36 | 1.12 | 10 | 40 | 1.36 e-05 | - | 0.473 | 1.57 |
| 20 | 20 | 9.11 e-04 | 4.19 | 0.94 | 3.78 | 20 | 160 | 8.34 e-07 | 18.5 | 1.503 | 5.62 |
| 40 | 40 | 2.23 e-04 | 4.08 | 3.49 | 16.02 | 40 | 640 | 5.19 e-08 | 16.1 | 5.354 | 26.78 |
| 80 | 80 | 5.51 e-05 | 4.05 | 8.71 | 33.12 | 80 | 2560 | 3.24 e-09 | 16.02 | 25.98 | 118.32 |
| 160 | 160 | 1.37e-05 | 4.02 | 25 | 114 | 160 | 10240 | 2.02 e-10 | 16.01 | 169 | 733 |
| 320 | 320 | 3.42 e-06 | 4.01 | 166 | 725 | | | | | | |

Contributions

- An adaptive MC method, particular 1-rank lattice Monte Carlo algorithm, based on generalized Fibonacci numbers and a variance reduction method called Latin Hypercube Sampling are constructed and discussed. A comparison with Crude MC method and the Sobol quasi-MC algorithm has been performed. We apply the five methods for multidimensional integrals of smooth functions. The tables with the obtained errors are shown. It can be seen that FIBO method produces better results for lower dimensions, but when the integrand has a dimension higher than 20 the LHS method outperforms the FIBO method. The efficiency of the Adaptive algorithm is clearly shown in the case studies of Genz test functions, where the integrands have computational specialty in the local subarea of the integration domain. The quasi-random Sobol sequence gives good results regardless off the dimension, but it is outperformed by at least one of the discussed methods, in any of the considered examples.
- Five stochastic algorithms have been applied to Wigner kernel case study. The stochastic algorithms under consideration have linear computational complexity that is one solution of Richard Feynman's problem for Wigner kernel evaluation. This is the first time the Adaptive MC algorithm is applied to this problem and it gives superior results in comparison with the other stochastic algorithms under consideration. Four completely different approaches are applied in finance for European option pricing evaluation. FIBO has strong advantage in the case of lower-dimension problems as long as the integrands are sufficiently regular. Some of the multidimensional integrals of smooth functions play an important role in Bayesian statistics.
- Monte Carlo method based on balancing of the systematic error and probability error is presented. Lower bounds for N and k are obtained. Monte Carlo algorithms with various initial and transition probabilities are compared. Experimental relative errors confirm expected theoretical errors. Monte Carlo algorithms with probabilities chosen

to be proportional to the function from the linear functional under consideration and the kernel, respectively, give more reliable results. The algorithm is applied for examples of integral equations in biology, neuron networks and physics.

- A new Monte Carlo algorithm based on WE algorithm for solving linear algebra problems is presented and studied. The algorithm can be used for evaluating all the components of the solution of a linear system. Due to the optimization techniques it gives superior results to the standard “walk on equations” method and it is established as one of the fastest and accurate Monte Carlo algorithm for solving systems of LA equations.
- A fourth-order in space compact difference scheme for semi-linear parabolic systems in air pollution modelling is proposed and studied. Two different ways to derive high-order difference schemes for semilinear parabolic systems of equations are analyzed. First, using central difference approximation with Richardson extrapolation a fourth-order method is defined. Second, the fourth-order in space compact difference scheme is extended by Richardson extrapolation to sixth-order approximation. The time-stepping is carried out using θ -scheme, but in the numerical computations - by the Crank-Nicolson/Newton algorithm. The reported computational results demonstrate that the convergence rate of the CDS is $O(h^2 + \tau^2)$, while for CFDS it is $O(h^4 + \tau^2)$, but in combination with Richardson extrapolation they are respectively $O(h^4 + \tau^2)$ and $O(h^6 + \tau^2)$. The advantages of the CFDS over the CDS both in the accuracy and CPU time have been confirmed numerically. The skilful application of Richardson extrapolation also plays important role in obtaining good results in real time with a small number of grid nodes despite the large intervals of the domain both in space and time in air pollution problems.

References

- [1] Боянов, Б.: *Лекции по числени методи*. Дарба, София, 1998.
- [2] Караиванова, А.: *Стохастични числени методи и симулации*, София, 2012.
- [3] Черногорова, Т.: Теория на диференчните схеми, София, 2005. <https://www.fmi.uni-sofia.bg/econtent/tds.pdf>
- [4] Янев, Н., Димитров, Б.: *Вероятности и статистика*. Наука и изкуство, София, 1990.
- [5] Broadie M., Glasserman P.: Pricing American-style Securities Using Simulation, *J. of Economic Dynamics and Control* 21, 1323-1352, 1997.
- [6] Dimov, I.: *Monte Carlo Methods for Applied Scientists*, New Jersey, London, Singapore, **World Scientific** (2008), 291 p., ISBN-10 981-02-2329-3.
- [7] Dimov, I.T., Maire, S., Sellier, J.M.: A New Walk on Equations Monte Carlo Method for Linear Algebraic Problems, *Applied Mathematical Modelling*, Volume 39, Issue 15, 2015, Pages 4494–4510
- [8] Ermakov, S.M.: *Monte Carlo Methods and Mixed Problems*, **Nauka**, Moscow, (1985).
- [9] Feynman, R.P.: Space-time approach to non-relativistic quantum mechanics, *Rev. Mod. Phys.* 20 (1948).

- [10] Genz, A.: Testing multidimensional integration routines. *Tools, Methods and Languages for Scientific and Engineering Computation* (1984) 81–94.
- [11] Georgiev, K., Zlatev, Z.: Implementation of sparse matrix algorithms in an advection-diffusion-chemistry model. *J. of Comp. Appl. Math.*, **236** (3), 342-353 (2011)
- [12] Georgieva, R.: PhD Thesis: Computational complexity of Monte Carlo algorithms for multidimensional integrals and integral equations, Sofia, 2003
- [13] Hua, L.K., Wang, Y.: *Applications of Number Theory to Numerical analysis*, 1981
- [14] Jarosz, Wojciech: Efficient Monte Carlo Methods for Light Transport in Scattering Media, PhD dissertation, UCSD, 2008
- [15] Kantorovich, L., Akilov, G.: *Functional Analysis*. Nauka, Moscow, 1977.
- [16] Karatson, J., Kurics, T.: A preconditioned iterative solution scheme for nonlinear parabolic systems arising in air pollution modeling. *Math. Modell. Anal.* **18** (5), 641-653 (2013)
- [17] Lai, Y., Spanier, J.: Applications of Monte Carlo/Quasi-Monte Carlo Methods in Finance: Option Pricing, Proceedings of a conference held at the Claremont Graduate Univ, 1998
- [18] Marchuk, G.I., Shaidurov, V.V.: *Difference Methods and Their Extrapolations* (Springer-Verlag, New York Inc. 1983).
- [19] Metropolis, N., Ulam, S.: *The Monte Carlo Method*, **J. of Amer. Statistical Assoc.**, **44**, (1949), No. 247, pp. 335–341.
- [20] Richards, S.: Completed Richardson extrapolation in space and time. *Commun. Numer. Meth. Engineering* **13**, 573–582 (1997).
- [21] Sellier, J.M.: A signed particle formulation of non-relativistic quantum mechanics, *Journal of Computational Physics* 297(2015) 254–265.
- [22] Sloan, I.H., Kachoyan, P.J.: Lattice methods for multiple integration: Theory, error analysis and examples, *SIAM J. Numer. Anal.* 24, (1987) 116-128.
- [23] Sobol, I.M.: *Monte Carlo Numerical Methods*, Nauka, Moscow, 1973, (in Russian).
- [24] Wang, Y.M., Guo, B.Y., Wu, W.J.: Fourth-order compact finite difference methods and monotone iterative algorithms for semilinear elliptic boundary value problems, *Computers and Math. with Appl.*, **68**, 1671-1688 (2014)
- [25] Wigner, E.: On the quantum correction for thermodynamic equilibrium, *Phys. Rev.* 40 (1932) 749.
- [26] Zlatev, Z., Dimov, I.: *Computational and Numerical Challenges in Environmental Modelling* (Elsevier, Amsterdam, 2006).
- [27] Website: Matrix market, NOS4: Lanczos with partial reorthogonalization. Finite element approximation to a beam structure, <http://math.nist.gov/MatrixMarket/data/Harwell-Boeing/lanpro/nos4.html>
- [28] <http://www.mathworks.com/>



АВТОРЕФЕРАТ НА ДИСЕРТАЦИЯ

за присъждане на образователна и научна степен “доктор” по
научна специалност „Математическо моделиране и
приложение на математиката “

*МЕТОДИ МОНТЕ КАРЛО
ЗА МНОГОМЕРНИ ИНТЕГРАЛИ И
ИНТЕГРАЛНИ УРАВНЕНИЯ
И ПРИЛОЖЕНИЯ*

Венелин Любомиров Тодоров

Ръководител: проф. Иван Димов

Научно жури:

Проф. Стефан Стефанов
Проф. Любен Вълков
Проф. Иван Димов
Доц. Станислава Стоилова
Доц. Райна Георгиева



Институт по информационни и
комуникационни технологии

Секция „Паралелни алгоритми“

Дисертационният труд е обсъден и допуснат до защита на разширено заседание на секция „Паралелни алгоритми“ при ИИКТ-БАН, състояло се на 23.05.2017 г.

Дисертационният труд съдържа 184 страници, в които 26 фигури, 45 таблици и 18 страници литература, включваща 237 заглавия.

Защитата на дисертационния труд ще се състои на 03.10.2017 г. от 14:00 часа в зала 218 на ИИКТ-БАН на открито заседание.

Автор: Венелин Любомиров Тодоров

Заглавие: Методи Монте Карло за многомерни интегрални уравнения и приложения

Увод

Методите Монте Карло (МК) са методи за приближено пресмятане на решението на задачи от изчислителната математика чрез използване на случайни процеси, като параметрите на съответния процес съвпадат с решението на задачата. Методът може да гарантира, че грешката при приближеното пресмятане на неизвестната величина е по-малка от зададена стойност с определена вероятност [7]. Методите Монте Карло се използват за оценяване на величини, които могат да бъдат представени като математическо очакване, чрез осредняване на резултатите, получени от голям брой статистически опити. Методите Монте Карло са основно средство за числено решаване на класове задачи в такива важни области като физика на елементарни частици, инженерна химия, молекулярна динамика, финансова математика. Основно научно предизвикателство в развитието на съвременните Монте Карло методи е тяхната относително бавна скорост на сходимост, която в редица случаи има асимптотика $O(N^{-1/2})$, където N е размерът на извадката. Има два подхода за подобряване на сходимостта - намаляване на дисперсията на оценяваната величина и намаляване на дискрепанса на използваните редици.

През Втората световна война, съвместната работа на големи учени като фон Нойман, Ферми, Улам и Метрополис и появата на съвременните дигитални компютри, даде силен тласък на теорията, заложена в методите Монте Карло. Статията на Metropolis и Ulam, публикувана през 1949 г. и озаглавена „The Monte Carlo Method“ [38], се възприема за първата статия, обвързваща наименованието „Монте Карло“ с използването на случайни величини [46]. Но както отбелязва Ермаков в [16] сведения за решение на реален проблем с метод Монте Карло съществува още в Стария Завет, където цар Соломон строи Божия храм. Може би най-ранното документирано използване на случайна извадка за намиране на приближена стойност на интеграл, е това на Буфон за изчисляване на числото π чрез хвърляне на игла. Терминът „Монте Карло“ е въведен от учените, работещи по разработването на ядрени оръжия в Лос Аламос през 40-те, където се е търсел отговор на въпроса дали е възможно предизвикването на ядрена реакция. Известно е, че множество неутрони, движещи се в уран, могат да предизвикат по случаен начин последваща емисия на

други неутрони, но не е можело да се предвиди теоретично дали веригите от реакции, образуващи сложна мрежа, ще предизвикат атомна експлозия. Учените използват компютъра *ENIAC*, за да моделират случайните траектории на неутроните през атомите на урановия заряд. Проектът е бил секретен с кодово наименование „Manhattan“.

Монте Карло методите предлагат простота на конструкциите и често се използват за симулация на процеси, чието поведение може да се интерпретира само в статистически смисъл. Съществуват широк клас задачи, за които методите Монте Карло са единствените възможни числени методи за решаване. При сравнението на обикновения метод Монте Карло за числено интегриране с детерминистични методи се налага изводът, че този подход е един от най-ефективните при числено пресмятане на многомерни интеграли, особено в случая на много високи размерности и без да е отчетена гладкостта на подинтегралната функция (вж. [7]). Независимо от универсалността на Монте Карло методите, техен сериозен недостатък е слабата им сходимост, основана върху грешка с порядък $O(N^{-1/2})$ при статистическа извадка с размер N , когато не се използва допълнителна информация за данните [2]. Дори скромни подобрения в тези методи имат съществен принос по отношение на ефективността и обхвата им на приложимост. Голяма част от усилията в разработването на Монте Карло методите, са насочени към конструиране на методи с намалена дисперсия. За тази цел могат да се използват различни адаптивни процедури, които използват предварителна и/или апостериорна информация, получена в процеса на пресмятанията. В общия случай, при реализациите на равномерно разпределена случайна величина може да се наблюдава струпване в определени подобласти и това да доведе до повишаване на дисперсията, тъй като реализациите не са толкова добре равномерно разпределени в областта, колкото се очаква. Затова изследванията за намаляване на дисперсията са насочени към контролирано разпределяне на реализациите в съответната област. Конструирането на метода на „слоистата“ извадка („stratified sampling“) и квази-Монте Карло методи е резултат от тези усилия. Квази-Монте Карло методите използват квазислучайни редици вместо обичайните псевдослучайни редици. Докато псевдослучайните числа са конструирани, така че да симулират поведението на истинските случайни числа, тези силно равномерно разпределени числа, на-

речени квазислучайни числа, са конструирани да бъдат толкова равномерно разпределени, колкото е математически възможно. Квазислучайните числа са конструирани да минимизират мярката за тяхното отклонение от равномерността, наречена дискрепанс. Така по-високият порядък на сходимост може да бъде получен чрез използване на детерминистични равномерно разпределени редици. Приближеното интегриране с използване на квазислучайни редици има по-бърза сходимост, с порядък $(O)(N^{-1} \log^s N)$, където s е размерността на задачата (за сравнение, обикновеният Монте Карло метод има порядък на сходимост $(O)(N^{-1/2})$). В последните десетилетия квази-Монте Карло методи известни като точкови множества от тип решетка, стават все по-популярни. Много задълбочено изследване на тези алгоритми е направено от Слоан [44]. Този метод е подходящ за гладки и периодични функции в s -мерния хиперкуб.

Съществуват два класа методи Монте Карло - преки и итерационни. Преките методи се характеризират само с един тип грешка, наречена вероятностна. Тъй като с итерационните методи Монте Карло се приближава числено съответното итерационно приближение на точното решение, този клас методи се характеризира с два типа грешки: вероятностна, която зависи от броя на реализациите на веригата на Марков и систематична, която зависи от броя на итерациите в процеса. Задачата за балансиране на систематичната и статистическата грешка е изключително важна, когато се прилагат алгоритми от тип Монте Карло. Тази задача изисква намирането на оптимално съотношение между броя на реализациите и средната дължина на случайните траектории, което спомага за повишаване на ефективността на алгоритъма (вж. [7]).

Частните диференциални уравнения (ЧДУ) се утвърдиха като много успешно средство за математическо моделиране на различни процеси в екологията и други области. Тъй като все по-сложни процеси са обект на моделиране, това се отразява и върху съответните ЧДУ. Численият метод, построен за решаване на дадена математическа задача, която моделира реален процес от практиката, трябва да притежава редица качества. От една страна характеристики, свързани с теоретичните изисквания на числения анализ, като съвместимост, точност, устойчивост и сходимост, а от друга страна, трябва да предава адекватно свойствата на диференциалната задача - например положителност, монотонност, плътност и други. От особена важност е и реализацията на разработения

алгоритъм и неговата ефективност, която се отчита по следните показатели - точност на численото решение, компютърно време и оперативна памет, необходими за пресмятанията. Диференчните схеми са често използвани за решаването на частни диференциални уравнения, поради лесната им реализация и изчислителната им ефективност. При решаването на елиптични и параболични ЧДУ, най-често използваните диференчни схеми са от втори ред. Тези диференчни схеми се наричат стандартни и се прилагат при решаването на много задачи. При използването на стандартните диференчни схеми от втори ред, за да се постигне исканата точност, се получават големи системи, които изискват много голяма изчислителна памет и голям брой процесори върху съвременните суперкомпютри. Напоследък, голямо усилие се съсредоточава в разработването на компактни диференчни схеми с висок ред на точност, които използват само възлите на мрежата, съседни на централния възел. Идеята, която ще използваме в настоящото изследване, е да работим върху диференциалните уравнения, така че да изразим производните от по-висок ред в локалната грешка на апроксимацията. В последните години все повече нараства интересът към компактните диференчните схеми за решаване на параболични ЧДУ в екологията, поради необходимост от точно оценяване на замърсяването в големите европейски градове.

Цели и задачи на дисертационния труд

Целта на настоящата дисертация е разработването, реализирането и изследването на ефективни алгоритми Монте Карло (квази-Монте Карло) за многомерни интегрални уравнения, съответно линейни системи. Важна част е приложението на изследваните алгоритми в много области, от които основните са финанси, физика, биология, екология. Допълнителна цел е конструирането на нови числени методи с висок ред на точност на базата на метода на диференчните схеми за модели, свързани с екологичната безопасност.

Конкретните задачи за постигането на тази цел са:

1. Да се разработи и изследва нов почти оптимален алгоритъм Монте Карло за интегрални уравнения, базиран на балансиране на систематичната и стохастичната грешка. Да се получат оценки за броя на реализациите и

броя на итерациите и да се изследва ефективността на алгоритъма върху интегрални уравнения с приложен характер.

2. Да се реализира квази-МК метод от тип решетки с генериращ вектор обобщената редицата на Фибоначи, МК метод базиран на извадката латински хиперкуб и адаптивен алгоритъм МК. Да се изследва кой от алгоритмите е най-ефективен в зависимост от размерността на интеграла и гладкостта на подинтегралната функция.
3. Да се приложат разгледаните методи върху многомерни интеграли с приложен характер във финансите и в статистиката, както и да се получат оценки за ядрото на Вигнер в квантовата механика и да се изследва кой от методите е най-ефективен за решаване на задачата на Ричард Файнман.
4. Да се конструира нов метод МК за линейни системи на базата на метода МК за линейни системи „случайно блуждаене по уравненията“ и да се направи сравнение между двата алгоритъма и рафинирания алгоритъм МК.
5. Да се разработи нова компактна схема с четвърти ред на точност за системи от параболични ЧДУ. Да се повиши точността с помощта на екстраполация по Ричардсон. Да се приложи компактната схема върху модел на далечен пренос на замърсители във въздуха, както и за други примери.

Методология на изследването

Методологията на настоящите изследвания се основава на фундаментални резултати от следните области:

- функционален анализ [27] - функционални редове (сходимост, ред на Neumann, грешка от „отрязването“ на ред („truncation error“), ред на Тейлър);
- теория на вероятностите [4] - случайни величини, гранични теореми, изместени и неизместени оценки, вериги на Марков;
- числен анализ [1] - апроксимация, квадратурни формули;

- числени методи за диференциални уравнения и теория на диференчните схеми [3].

Програмните кодове на конструираните алгоритми са написани на програмната среда за пресмятания MATLAB [56].

Апробация на резултатите

Резултати, включени в дисертацията, са докладвани на: съвместен семинар на секции „Паралелни алгоритми“ и „Научни пресмятания“ на ИИКТ-БАН;

Част от резултатите са представени на следните конференции:

1. 9th International Conference on „Large-Scale Scientific Computations“ (LSSC'13), Созопол, България, 2013;
2. 8th International Conference on Numerical Methods and Applications: NMA'14, Боровец, 2014;
3. 10th IMACS Seminar on Monte Carlo Methods, (MCM 2015), Linz, Austria, July 6-10, 2015;
4. Advanced Computing for Innovation, AComIn 2015, 10-11 ноември, 2015, София;
5. Международната конференцията Numerical Methods for Scientific Computations and Advanced Applications, NMSCAA'16, 29 май - 2 юни, Хисаря, 2016;
6. APPLICATION OF MATHEMATICS IN TECHNICAL AND NATURAL SCIENCES: 8th International Conference for Promoting the Application of Mathematics in Technical and Natural Sciences - AMiTaNS'16, 22–27 юни 2016, Албена.

Списък на публикациите по дисертацията

Основните резултати по дисертацията са публикувани в:

1. I.T. Dimov, R. Georgieva, V. Todorov, Balancing of Systematic and Stochastic Errors in Monte Carlo Algorithms for Integral Equations, 8th International

Conference Numerical Methods and Applications, NMA 2014, Borovets, Bulgaria, August 20-24, 2014, Numerical Methods and Applications (I. Dimov, S. Fidanova, and I. Lirkov - Eds.), LNCS 8962, Springer, 2015, 44–51. DOI : 10.1007/978–3–319–15585–2_5. ISSN: 0302-9743. **SJR (2015): 0.252.**

2. Dimov, I.T., Todorov, V., Error Analysis of Biased Stochastic Algorithms for the Second Kind Fredholm Integral Equation, International Conference on Advanced Computing for Innovation, AComIn 2015, Sofia; Bulgaria, 10-11 November 2015, 2015 (Margenov S., Angelova G., Agre G. eds.), Studies in Computational Intelligence 648, 2016, 3-16. Springer Verlag. DOI : 10.1007/978–3–319–32207–0_1. ISSN: 1860-949X. **SJR(2015): 0.187.**
3. V. Todorov, I. Dimov. Monte Carlo methods for multidimensional integration for European option pricing, DOI:10.1063/1.4965003, AIP Conf. Proc. 1773, 100009, ISSN 0094243X, (2016), **SJR (2016): 0.198.**
4. Dimov I., J. Kandilarov, V. Todorov, L. Vulkov. High-Order Compact Difference Schemes with Richardson Extrapolation for Semilinear Parabolic Systems. IN: Applications of Mathematics in Engineering and Economics, American Institute of Physics, 1789, 030002 (2016), DOI: 10.1063/1.4968448, **SJR (2016): 0.198.**
5. V. Todorov. Computing High Dimensional Integrals with Monte Carlo Methods, Journal Scientific and Applied Research, Journal Scientific and Applied Research, 10, 2016, 11-16, Konstantin Preslavsky Publishing House, ISSN 1314-6289.

Четири от представените публикации са в издания със SJR фактор.

Втората статия е цитирана в изданието с импакт фактор:

1. Farshid Mirzaee, Nasrin Samadyar. Application of orthonormal Bernstein polynomials to construct a efficient scheme for solving fractional stochastic integro-differential equation, Optik - International Journal for Light and Electron Optics, 132, 2017, pp. 262–273, DOI:10.1016/j.ijleo.2016.12.029, **IF: 0.742.**

Алгоритми Монте Карло за многомерни интеграли

Тази глава е посветена на изследването, реализирането, сравнението и приложението на различни методи Монте Карло и квази-Монте Карло за многомерно интегриране. В дисертацията е направено подробно описание на методът Монте Карло, базиран на извадката латински хиперкуб (LHS). Реализиран е адаптивен метод Монте Карло за многомерни интеграли, който използва само апостериорна информация за порядъка на дисперсията (стандартното отклонение), но не и за гладкостта. Изследвана е изчислителната сложност на алгоритъма, която е съпоставена със сложността на обикновения метод Монте Карло. Направени са числени експерименти с тестови функции на Генц [18], които доказват очаквания ефект на намаляване на дисперсията. Конструиран е и квази-Монте Карло метод от тип решетки, с генериращ вектор обобщената редица на Фибоначи от съответната размерност. Направено е сравнение с квази-случайните редици на Соболев за гладки функции и при оценка на европейски опции във финансите. Най-важното приложение е при оценка на ядрото на Вигнер в квантовата механика. Полученият резултат е оригинален по своя характер и отговаря на въпрос, поставен преди няколко десетилетия от един от най-забележителните физици на 20 век Ричард Файнман за съществуването на алгоритъм с линейна или полиномиална, но не и експоненциална, изчислителна сложност за ядрото на Вигнер. Всички представени методи от тип Монте Карло/квази-Монте Карло дават по-добри резултати от досега използваните методи за оценка на ядрото, като адаптивният алгоритъм се оказва най-ефективен, поради поведението на ядрото.

Резултатите, представени в тази глава, са публикувани в [47, 48].

Описание на алгоритмите

Извадка латински хиперкуб

Подходът на „слоистите“ извадки („stratified sampling“; вж. [46]) се основава само на начално разбиване на областта и подходящ избор на броя на случайните точки в съответната подобласт, което да гарантира намаляване на дисперсията в сравнение с обикновения метод Монте Карло, но не и повишаване на порядъка $\mathcal{O}(\sqrt{1/N})$ на вероятностната грешка. Избират се повече точки в областите,

в които локалната дисперсия е голяма. Подходът не съдържа идеята за адаптивност (рекурсивно разделяне на подобласти) и не използва предварителна информация за гладкостта на подинтегралната функция.

Теорема 1. *Теорема(Соболъ, [46]) За фиксирано разбиване на областта на интегриране и $N_j = N p_j$ е в сила, че*

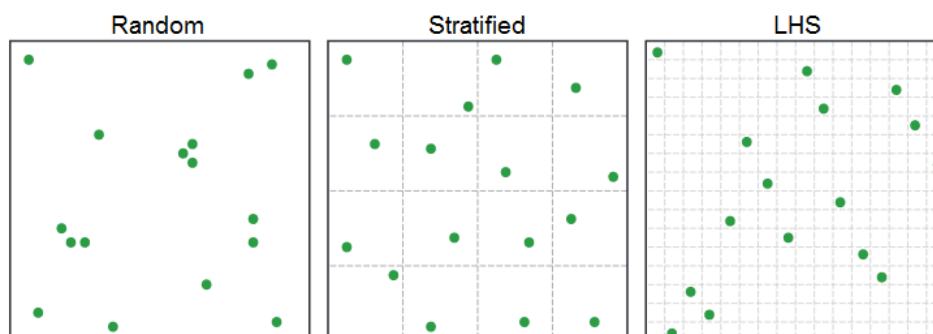
$$D\bar{\theta}_N^* \leq D\bar{\theta}_N,$$

където $D\bar{\theta}_N$ е дисперсията на оценката на интеграла, получена с обикновен метод Монте Карло и N случайни точки в областта на интегриране, а $D\bar{\theta}_N^*$ е дисперсията на оценката на интеграла, получена с метода на „слоистите“ извадки, и p_j е плътността в съответната подобласт.

Съществен недостатък при прилагане на този подход, е разделянето на подобласти. Ако една s -мерна област се разбие на равномерни подобласти чрез разделяне на две по всяко направление, се получават 2^s подобласти. В случая на големи s броят на подобластите рязко нараства, което ограничава силно избора на размерност на случайната извадка. Този проблем се преодолява от извадката "latin hypercube sampling"(LHS) [26]. Извадката LHS е специален вид „слоиста“ извадка, използвана за пръв път през 1979г. в [37] и развита в по скорошни публикации – виж [39]. В двумерния случай квадратна мрежа е латински квадрат, тогава и само тогава, когато във всеки ред и стълб на квадрата попада само една реализация на случайната величина. Латински хиперкуб е обобщение на тази концепция за произволна размерност. Тази техника изисква точно по една случайна точка във всяка подобласт, това свойство е едно от основните предимства на извадката латински хиперкуб. За по-голяма яснота на Фиг. 1 е дадена извадката латински хиперкуб, сравнена със „слоистата“ и случайна извадка с 16 точки ($s = 2$ е размерността, а $M = 4$ са броя подобласти).

Адаптивен алгоритъм Монте Карло

Изследван е адаптивен алгоритъм, базиран на алгоритъма, описан в [8, 20, 28]. Основната идея е концентриране на случайни точки в подобластите, в които дисперсията е голяма (по отношение на предварително зададена точност). Подходът се основава на рекурсивно разделяне на областта, използвайки



Фигура 1: Сравнение на различни типове извадки

апостериорна информация за грешката при текущото разделяне. Така адаптивният алгоритъм дава приближение с грешка $\varepsilon \leq c N^{-1/2}$, където $c \leq 0.6745\sigma(\theta)$ ($\sigma(\theta)$ е стандартното отклонение), но порядъкът е същият, както при обикновения алгоритъм Монте Карло. Реализираният тук адаптивен алгоритъм не използва никаква предварителна информация за гладкостта на подинтегралната функция, но използва апостериорна информация за дисперсията. Адаптивният алгоритъм се оказва най-ефективен за функции с особености в изчислително отношение - при ядрото на Вигнер и тестовите функции на Генц, той дава относителни грешки с няколко порядъка по-добри от останалите алгоритми. Алгоритъмът започва с разделяне на интервалите по всички направления на M подинтервала, като M е зададено като входящ параметър. Във всяка подобласт се пресмята стойността на съответния интеграл $I_{\Omega_j}, j = 1, \dots, M$ и дисперсия. След това получената дисперсия се сравнява с предварително зададена стойност. Получената информация се използва за следващото сгъстяване, тъй като подобластта с най-голямо стандартно отклонение се разделя на 2^s нови подобласти. Изпълнението на алгоритъма спира, когато за стандартното отклонение е достигната предварително зададената точност ε във всички подобласти, получени след делението (или когато е достигнат предварително зададен максимален брой на нивата на разделяне или на подобластите, в които не е изпълнен стоп-критерият). Така се получава едно приближение на интеграла $I = \sum_j I_{\Omega_j}, j = 1, \dots, M^s$ с подход Монте Карло.

Квази-Монте Карло методи с използване на точкови множества от

тип решетка

Да разгледаме квадратурата

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

където $P_N = \{x_0, x_1, \dots, x_{N-1}\}$, $x_i \in [0, 1]^s$ е множество от точки, задаващо интеграционните възли на формулата. Изборът на възлите е много съществен, тъй като от това зависи дискрепансът и точността на полученото приближение. Практическо приложение е намерил теоретико-числовият метод за интегриране, предложен от Корабов [31], при който възлите на квадратурната формула имат вида:

$$x_k = \left(\left\{ \frac{kz_1}{N} \right\}, \left\{ \frac{kz_2}{N} \right\}, \dots, \left\{ \frac{kz_s}{N} \right\} \right), \quad k = 1, 2, \dots, N, \quad (1)$$

където N е зададен брой точки, z е s -мерен целочислен вектор, а чрез $\{a\}$ означаваме дробната част на числото a , т.е. $\{a\} = a - [a]$. Векторът z се нарича генериращ вектор или генератор на множеството.

Да поставим допълнително изискване към гладкостта на функцията $f(x)$, а именно да предположим, че функцията $f(x)$ принадлежи на класа на Корабов $E_s^\alpha(C)$, зададен по следния начин [45]:

Дефиниция 1. Казваме, че $f(x)$ принадлежи на класа $E_s^\alpha(c)$ за $\alpha > 1$ и $c > 0$, ако f е периодична функция с период единица по всяка от променливите си $x_i, i = 1, 2, \dots, s$, дефинирана върху единичния хиперкуб $[0, 1]^s$ и коефициенти й на Фурие удовлетворяват неравенството:

$$|a(m)| \leq \frac{C}{(\bar{m}_1 \dots \bar{m}_s)^\alpha}, \quad (2)$$

където

$$\bar{m} = \begin{cases} |m|, & \text{ако } |m| \neq 0, \\ 0, & \text{ако } m = 0. \end{cases}$$

и константата c не зависи от m_1, \dots, m_s .

Теорема 2. ([5]) Ако $f(x) \in E_s^\alpha(c)$, то съществува оптимален избор на генериращ вектор z , за който за грешката от интегриране е в сила

$$\left| \frac{1}{N} \sum_{k=0}^{N-1} f\left(\left\{\frac{k}{N}z\right\}\right) - \int_{[0,1]^s} f(u) du \right| \leq Cd(s, \alpha) \frac{(\log N)^{\beta(s, \alpha)}}{N^\alpha} \quad (3)$$

за функция $f \in E_s^\alpha(c)$, $\alpha > 1$ и $d(s, \alpha)$, $\beta(s, \alpha)$ не зависят от N . Нещо повече, ако N е просто число, то $\beta(s, \alpha) = \alpha(s - 1)$.

Векторът z , за който неравенството (3) е изпълнено, се нарича добър или оптимален генериращ вектор, а неговите компоненти - оптимални коефициенти в смисъла на Корабов. Точковото множество P_N е множество от добри целочислени точки, а съответният метод за числено интегриране с квадратурата се нарича метод на добрите целочислени точки (Good Lattice Point method - GLP метод). Известни са редица теореми, доказващи съществуването на оптимални генериращи вектори. Трудността е в конструирането на тези оптимални вектори, особено в задачи от голяма размерност. При $s = 2$ оптимална конструкция съществува. Вакхавалов през 1959г. в [5] представя конструкция, основана на числата на Фибоначи. През 1981 г. в [25] е направено обобщение на редицата на Фибоначи за произволна размерност. Ниа и Wang конструират точково множество от тип решетка със следния генериращия вектор [25, 50]:

$$z = (1, F_n(2), \dots, F_n(s)), \quad (4)$$

за някое естествено число n . В сила е $F_n(j) = F_{n+j-1} - F_{n+j-2} - \dots - F_n$, където F_i са съответните обобщени числа на Фибоначи с размерност s , т.е.:

$$F_{l+s} = F_l + F_{l+1} + \dots + F_{l+s-1}, l = 0, 1, \dots \quad (5)$$

с начално условие:

$$F_0 = F_1 = \dots = F_{s-2} = 0, F_{s-1} = 1, \quad (6)$$

за $l = 0, 1, \dots$ След опростяване, може да се види, че генериращият вектор по-горе е:

$$z = (1, F_{n-1} + F_{n-2} + \dots + F_{n-s+1}, \dots, F_{n-1} + F_{n-2}, F_{n-1}) \quad (7)$$

Според [25], имаме следната оценка за дискрепанса на множеството, получено с използването на този вектор и F_n брой точки:

Теорема 3. *Множеството*

$$\left(\left\{ \frac{1}{F_n} k \right\}, \left\{ \frac{F_n(2)}{F_n} k \right\}, \dots, \left\{ \frac{F_n(s)}{F_n} k \right\} \right), \quad 1 \leq k \leq F_n.$$

има дискрепанс

$$D(F_n) = \mathcal{O}\left(F_n^{-\frac{1}{2} - \frac{1}{2^{s+1} \cdot \log 2} - \frac{1}{2^{2s+3}}}\right).$$

Предимството на разглеждания метод е, че броят на операциите, необходими за получаване на генерация вектор, е асимптотично $\mathcal{O}(\log F_n)$. След като имаме този вектор, генерирането на нова точка изисква константен брой операции. Тъй като трябва да генерираме F_n точки, за да получим точково множество от тип решетка от разглеждания вид с F_n точки, ще бъдат необходими $\mathcal{O}(F_n)$ брой операции.

Приложение в Бейсовската статистика

Фундаментален проблем в Бейсовската статистика е точното пресмятане на многомерни интегрални от следните два вида, описани от Shaowei Lin през 2011 в [33, 34]. Първият вид интегрални имат следния вид: $\int_{\Omega} p_1^{u_1}(x) \dots p_k^{u_k}(x) dx$, където $\Omega \in \mathcal{R}^s$, $x = (x_1, \dots, x_d)$, $p_i(x)$ са полиноми и u_i са цели числа. Вторият вид интегрални имат вида $\int_{\Omega} e^{-Nf(x)} \phi(x) dx$, където N е естествено число и $f(x)$, $\phi(x)$ са многомерни полиноми. Такива са 5-мерния и 15-мерния многомерни интегрални, дадени по-долу.

$$\int_{[0,1]^5} \exp(-100x_1x_2x_3)(\sin(x_4) + \cos(x_5)) dx \approx 0.1854297367. \quad (8)$$

$$\int_{[0,1]^{15}} \left(\sum_{i=1}^{10} x_i^2\right)(x_{11} - x_{12}^2 - x_{13}^3 - x_{14}^4 - x_{15}^5)^2 dx \approx 1.96440666. \quad (9)$$

$$\int_{[0,1]^{30}} \frac{4x_1x_3^2e^{2x_1x_3}}{(1+x_2+x_4)^2} e^{x_5+\dots+x_{20}} x_{21} \dots x_{30} dx \approx 3.244540. \quad (10)$$

Направено е детайлно сравнение между обикновен Монте Карло алгоритъм (Crude), адаптивен алгоритъм Монте Карло (Adapt), алгоритъм с използване на квазислучайната редица на Соболев (Sobol), алгоритъм с използване на точково множество от тип решетки базирано на обобщената редица на Фибоначи (FIBO) и извадка от тип латински хиперкуб (LHS). Резултатите са дадени в Таблицы 1, 2 и 3.

За сравнително малка размерност за 5-мерния интеграл методът FIBO постига най-малка относителна грешка при предварително зададен брой реализации - виж Таблица 1. За размерност 15 Sobol постига по-малки относителни грешки от FIBO - виж Таблица 2. Анализите показват, че за висока размерност 30 извадката LHS превъзхожда FIBO и Sobol - виж Таблица 3. От таблиците може да се направи извода, че методът FIBO е ефективен за пресмятане на многомерни интегрални от гладки подинтегрални функции при сравнително малки размерности, заради високата точност и ниската изчислителна сложност. За по-големи размерности за предпочитане е методът LHS. Методът на Соболев е значително по-бавен от FIBO и LHS, затова дори в случаите на сходни относителни грешки FIBO е по-ефективен, защото трудоемкостта му е по-малка. Адаптивният метод е най-бавен, заради рекурсивното делене на подобласти и е най-подходящ за интегриране на функции с особености, което ще бъде показано в следващите параграфи.

Таблица 1: Относителна грешка за 5-мерния интеграл

| N | Crude | време(s) | Adapt | време(s) | FIBO | време(s) | Sobol | време(s) | LHS | време(s) |
|--------|---------|----------|---------|----------|---------|----------|---------|----------|---------|----------|
| 10^3 | 2.10e-2 | 0.007 | 2.15e-3 | 0.27 | 1.75e-4 | 0.007 | 5.29e-4 | 0.03 | 9.38e-3 | 0.007 |
| 10^4 | 4.52e-3 | 0.07 | 2.01e-3 | 2.43 | 1.28e-5 | 0.06 | 1.43e-4 | 0.3 | 3.44e-3 | 0.07 |
| 10^5 | 1.19e-3 | 0.64 | 6.91e-4 | 22.2 | 9.50e-6 | 0.61 | 2.36e-5 | 2.77 | 2.01e-3 | 0.69 |
| 10^6 | 8.47e-4 | 6.06 | 2.92e-4 | 219.5 | 5.47e-7 | 5.98 | 6.07e-6 | 24.2 | 1.80e-4 | 6.17 |
| 10^7 | 2.38e-4 | 59.9 | 8.21e-5 | 2043 | 8.71e-9 | 58.4 | 2.30e-6 | 245 | 2.46e-5 | 60.5 |

Таблица 2: Относителна грешка за 15-мерния интеграл

| N | Crude | време(s) | Adapt | време(s) | FIBO | време(s) | Sobol | време(s) | LHS | време(s) |
|--------|---------|----------|---------|----------|---------|----------|---------|----------|---------|----------|
| 10^3 | 6.31e-2 | 0.09 | 3.16e-3 | 8.24 | 5.34e-2 | 0.08 | 2.04e-3 | 0.98 | 1.06e-2 | 0.12 |
| 10^4 | 4.30e-2 | 0.95 | 1.49e-3 | 68 | 1.22e-3 | 0.93 | 2.89e-4 | 9.3 | 7.33e-3 | 1.07 |
| 10^5 | 2.77e-2 | 9.70 | 5.76e-4 | 547 | 1.08e-4 | 9.65 | 1.13e-5 | 93.8 | 1.54e-4 | 10.11 |
| 10^6 | 7.13e-3 | 95.8 | 1.29e-4 | 5235 | 6.37e-6 | 96.9 | 5.93e-6 | 735 | 1.14e-5 | 99.6 |

Таблица 3: Относителна грешка за 30-мерния интеграл

| N | Crude | време(s) | Adapt | време(s) | FIBO | време(s) | Sobol | време(s) | LHS | време(s) |
|--------|---------|----------|---------|----------|---------|----------|---------|----------|---------|----------|
| 10^3 | 8.56e-1 | 0.02 | 1.56e-1 | 2.27 | 8.73e-1 | 0.02 | 1.29e-1 | 0.27 | 2.31e-2 | 0.02 |
| 10^4 | 7.13e-1 | 0.1 | 6.91e-2 | 20.1 | 1.19e-2 | 0.18 | 8.56e-2 | 2.5 | 6.89e-3 | 0.19 |
| 10^5 | 4.21e-1 | 1.12 | 3.76e-2 | 229 | 2.78e-2 | 1.56 | 1.91e-2 | 20.2 | 1.65e-3 | 1.54 |
| 10^6 | 1.73e-1 | 11.07 | 6.29e-3 | 2271 | 9.56e-3 | 13.61 | 9.47e-3 | 208 | 9.61e-5 | 13.9 |

Тестови функции на Генц

Методът на решетките не е приложим за функции с особености, както се вижда от числените експерименти в тази секция. Нека е дадена следната моделна функция:

$$f(x) = \left(1 + \sum_{i=1}^s a_i x_i\right)^{-(s+1)}. \quad (11)$$

Разглежданият клас от тестови функции принадлежи на пакет, предложен от Genz [18]. Избраното множество от функции имат единствен локален максимум в близост до един от върховете на многомерния единичен куб, подобно на някои моделни функции, описващи изменението в концентрациите на замърсители във въздуха.

Таблица 4: Относителна грешка и изчислително време в секунди за размерност $s = 5$, $I[f] = 2.12e-06$, $a = (5, 5, 5, 5, 4)$.

| Адаптивен алг. МК | | | LHS | | | FIBO | | |
|-------------------|------------|--------|--------|----------------|-------|-----------|------------|--------|
| N | $I_N[f]$ | (s) | N | $I_N[f]$ | (s) | N | $I_N[f]$ | (s) |
| 10^2 | 3.7735e-03 | 0.33 | 10^5 | 7.2274e-02 | 0.27 | 1346269 | 9.7135e-02 | 0.38 |
| 10^3 | 1.2877e-03 | 1.44 | 10^6 | 3.2518e-02e-02 | 1.22 | 3524578 | 6.7594e-02 | 1.32 |
| 10^4 | 4.2452e-04 | 10.75 | 10^7 | 2.5207e-03 | 12.3 | 14930352 | 1.5377e-02 | 15.07 |
| 10^5 | 4.7169e-05 | 142.18 | 10^8 | 1.6646e-03 | 124.2 | 102334155 | 2.9245e-03 | 134.58 |

Резултатите, получени след прилагането на описаните алгоритми в тази глава - латински хиперкуб, точково множество от тип решетки, базирано на обобщената редица на Фибоначи и адаптивен алгоритъм Монте Карло, за интеграли

Таблица 5: Относителна грешка и изчислително време в секунди за размерност $s = 18$, $I[f] = 9.919e-06$, $a = \left(\frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{1}{9}, \frac{1}{9} \right)$.

| Адаптивен алг. МК | | | LHS | | | FIBO | | |
|-------------------|------------|--------|-----------------|------------|--------|------------|------------|--------|
| N | отн. гр. | вр.(s) | N | отн. гр. | вр.(s) | N | отн. гр. | вр(s) |
| 10 | 9.2341e-04 | 15.7 | 10 ⁷ | 8.6285e-03 | 13.6 | 14930352 | 7.1579e-02 | 14.7 |
| 10 ² | 8.0653e-05 | 142 | 10 ⁸ | 5.1195e-03 | 140 | 102334155 | 5.1096e-02 | 144.1 |
| 10 ³ | 1.0081e-05 | 1408 | 10 ⁹ | 1.6283e-03 | 1353.5 | 1134903170 | 2.8883e-02 | 1344.3 |

с размерност 5 и 18, са дадени съответно в Таблица 4 и Таблица 5. Изследвана е ефективността на реализирания адаптивен алгоритъм Монте Карло по отношение на относителна грешка и изчислително време. Резултатите показват, че за почти едно и също време адаптивният алгоритъм дава по-малки относителни грешки, дори в порядъци.

Приложение за ядрото на Вигнер

Един от най-известните физици на нашето време Ричард Файнман поставя преди няколко десетилетия въпроса за съществуване на ефективен алгоритъм с линейна или полиномиална изчислителна сложност за ядрото на Вигнер в многомерния случай [17]. Досега са използвани само детерминистични алгоритми с експоненциална изчислителна сложност, които страдат от „проклятието на размерността“, за което беше вече споменато. Трите постулата, които описват напълно новата математическа формулировка на квантовата механика са дадени в термините на частици със знак и тези три постулата са достатъчни да отговорят на резултатите на по-сложни квантови теории [43]. Тук даваме само втория постулат:

Постулат II. Частица със знак, отделяща се в потенциал $V = V(x)$, се държи като класическа точкова частица без поле, която създава нови частици със знак

с вероятност $\gamma(x(t))dt$ с интервал dt , за времето dt , където

$$\gamma(x) = \int_{-\infty}^{+\infty} Dp' V_W^+(x; p') \equiv \lim_{\Delta p' \rightarrow 0^+} \sum_{M=-\infty}^{+\infty} V_W^+(x; M \Delta p'), \quad (12)$$

и $V_W^+(x; p)$ е положителната част на величината

$$V_W(x; p) = \frac{i}{\pi^s \hbar^{s+1}} \int_{-\infty}^{+\infty} dx' e^{-\frac{2i}{\hbar} x' p} [V(x+x') - V(x-x')], \quad (13)$$

известна като ядрото на Вигнер в s -мерното пространство [51]. Ако по време на създаването, създаващата частица има знак z , позиция x и момент p , новите частици имат същата позиция x , имат знаци $+z$ и $-z$, и моменти $p+p'$ и $p-p'$ съответно, избрани случайно в съответствие с нормализираната вероятност $\frac{V_W^+(x;p)}{\gamma(x)}$.

Използваме следния Вигнеров потенциал:

$$V = V(x) = x_1 \dots x_n, \quad x', x, p, x+x', x-x' \in [0, 1].$$

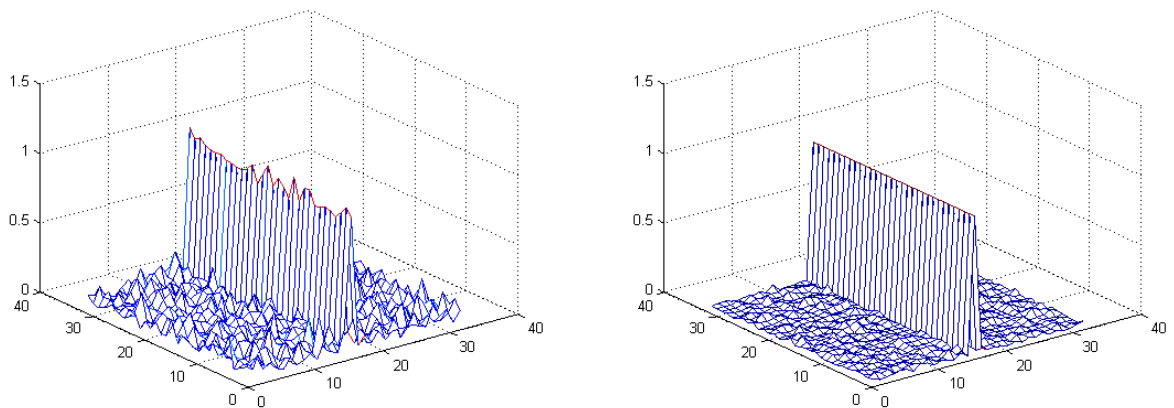
Целта на настоящето изследване е да се пресметне многомерния интеграл (13), представляващ ядрото на Вигнер.

Най-напред прилагаме детерминистичен метод на средните правоъгълници, обикновен метод Монте Карло и квази-Монте Карло с използване на квазислучайни редици на Соболю. Резултатите са дадени в Таблица 6. Детерминистичният метод е най-бавен и започва да губи точност с увеличаване на размерността на интеграла. Методът на Соболю е по-точен от обикновения метод Монте Карло, но е по-бавен - виж Таблица 6. Обикновеният алгоритъм Монте Карло е по-неточен от квази-Монте Карло алгоритъма с редици на Соболю, поради свойствата на добре разпределените редици.

Ядрото на Вигнер, получено с обикновен и адаптивен алгоритъм Монте Карло, е илюстрирано на Фиг. 2. На Фиг. 3 са показани позицията на частиците със знак в ядрото на Вигнер и върхът на ядрото на Вигнер, получени с адаптивния алгоритъм. От поведението на ядрото се вижда, че най-ефективен за пресмятането му би бил адаптивният алгоритъм Монте Карло, което се потвърждава и от проведените експерименти. Числените резултати показват, че методът от

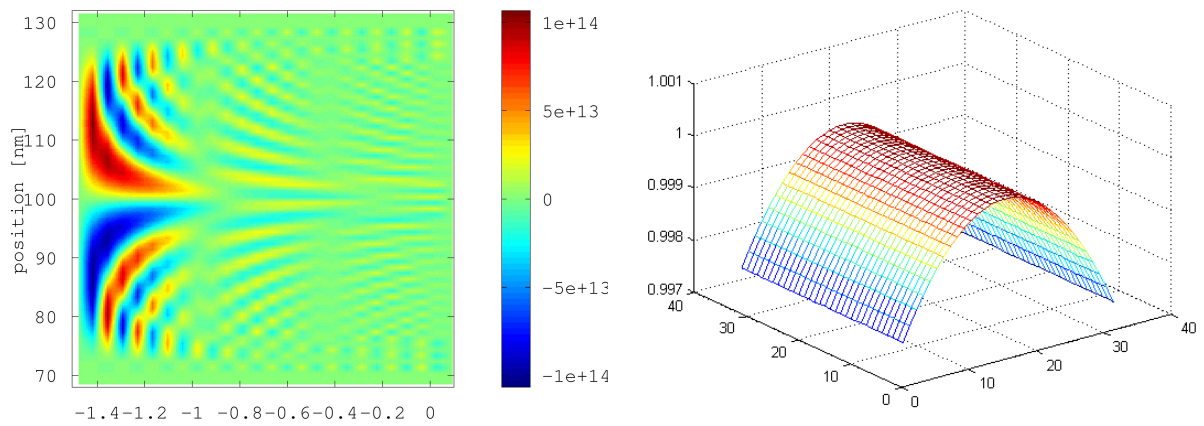
Таблица 6: Относителна грешка за ядрото на Вигнер с детерминистични и стохастични методи

| s | N | Средни правоъг. | време (s) | Собол алг. | време (s) | Обикновен алг. | време (s) |
|---|---------------------|-----------------|-----------|------------|-----------|----------------|-----------|
| 3 | $32^2 \times 50$ | 8.51e-03 | 0.2 | 8.47e-03 | 0.1 | 8.37e-03 | 0.003 |
| | $32^2 \times 100$ | 8.21e-03 | 0.5 | 8.11e-03 | 0.21 | 8.19e-03 | 0.008 |
| | $64^2 \times 50$ | 5.76e-03 | 1 | 5.26e-03 | 0.5 | 5.11e-03 | 0.1 |
| | $64^2 \times 100$ | 4.89e-03 | 1.9 | 4.55e-03 | 1.1 | 4.76e-03 | 0.3 |
| 6 | $8^4 \times 50^2$ | 1.16e-02 | 41.2 | 8.64e-04 | 19.5 | 9.09e-04 | 4.1 |
| | $8^4 \times 100^2$ | 9.75e-03 | 160.6 | 5.73e-04 | 53.4 | 6.44e-04 | 17.9 |
| | $16^4 \times 50^2$ | 7.84e-03 | 635.2 | 1.90e-04 | 149 | 4.37e-04 | 51.5 |
| | $16^4 \times 100^2$ | 2.12e-03 | 2469.1 | 1.29e-04 | 601.6 | 3.80e-04 | 132.1 |
| 9 | $6^6 \times 16^3$ | 1.75e-03 | 835.5 | 5.45e-04 | 188.5 | 9.35e-04 | 46.5 |
| | $6^6 \times 32^3$ | 1.35e-03 | 5544.1 | 2.36e-04 | 1067.6 | 8.29e-04 | 225.1 |
| | $6^6 \times 40^3$ | 1.12e-03 | 10684.4 | 1.07e-04 | 2190.5 | 5.12e-04 | 491.5 |



Фигура 2: Ядрото на Вигнер, получено с обикновен и адаптивен алгоритъм Монте Карло

тип решетки и този на Собол произвеждат подобни относителни грешки за един и същ брой реализации. За 9-мерния интеграл FIBO и Sobol дават сходни ре-



Фигура 3: Позиция на частиците в ядрото на Вигнер и върхът на ядрото на Вигнер с адаптивен алгоритъм

зултати, които са по-неточни от адаптивния алгоритъм и извадката LHS – виж Таблица 7. Адаптивният алгоритъм дава най-добри резултати в сравнение с останалите стохастични алгоритми с до два порядъка по-добри от LHS, но е значително по-бавен – виж Таблица 7. Разработеният адаптивен алгоритъм е с линейна изчислителна сложност, което е един възможен подход за много точно пресмятане на многомерния интеграл, описващ ядрото на Вигнер, и е решение на въпроса, поставен от Ричард Файнман, за който беше споменато в началото. Прави впечатление и добрият резултат на метода LHS, който заради бързината си също е един възможен подход за оценяване на ядрото. Предимството на адаптивния Монте Карло метод следва от вида на ядрото, илюстриран на Фиг. 2. С увеличаване на размерността предимствата на адаптивния алгоритъм пред останалите методи се засилват. Всички използвани методи дават значително по-добри резултати от досега използвания детерминистичен подход, както по отношение на точност, така и по отношение на изчислително време. Изследваният адаптивен алгоритъм позволява да се постигне висока точност със сравнително малка изчислителна сложност, което е един възможен подход за пресмятане на многомерния интеграл, описващ ядрото на Вигнер, и е решение на въпроса, поставен от Ричард Файнман [17].

Таблица 7: Относителна грешка за 3, 6 и 9 мерния интеграл

| | N | Latin | t,s | адаптивен | t,s | FIBO | t,s | Sobol | t,s |
|---------|--------|----------|------|-----------|------|----------|------|----------|------|
| $s = 3$ | 10^3 | 4.38e-03 | 0.01 | 6.75e-04 | 0.4 | 3.72e-02 | 0.02 | 1.07e-02 | 0.05 |
| | 10^4 | 7.94e-04 | 0.06 | 8.15e-05 | 3.3 | 7.06e-03 | 0.07 | 8.77e-03 | 0.54 |
| | 10^5 | 2.51e-04 | 0.41 | 5.01e-06 | 32.6 | 3.40e-03 | 0.43 | 8.57e-04 | 5.74 |
| | 10^6 | 8.20e-05 | 3.52 | 4.38e-07 | 302 | 1.01e-03 | 4.4 | 6.73e-04 | 45.6 |
| $s = 6$ | 10^3 | 1.54e-03 | 0.01 | 2.23e-04 | 0.5 | 7.82e-03 | 0.01 | 2.42e-02 | 0.09 |
| | 10^4 | 6.34e-04 | 0.06 | 4.74e-05 | 4.1 | 5.01e-03 | 0.07 | 5.02e-03 | 0.78 |
| | 10^5 | 4.22e-04 | 0.44 | 5.43e-06 | 37 | 6.88e-03 | 0.43 | 4.60e-04 | 6.19 |
| | 10^6 | 8.57e-05 | 3.7 | 5.04e-07 | 351 | 7.68e-04 | 5.97 | 3.59e-04 | 53 |
| $s = 9$ | 10^3 | 6.11e-03 | 0.04 | 8.23e-04 | 0.5 | 2.03e-02 | 0.06 | 5.42e-02 | 0.11 |
| | 10^4 | 1.02e-03 | 0.06 | 2.02e-05 | 4.7 | 2.02e-03 | 0.07 | 6.02e-03 | 0.88 |
| | 10^5 | 4.69e-04 | 0.43 | 1.08e-06 | 40 | 9.16e-04 | 0.53 | 3.57e-03 | 6.56 |
| | 10^6 | 8.08e-05 | 3.8 | 4.14e-07 | 381 | 7.13e-04 | 3.7 | 8.02e-04 | 57 |

Приложения за европейски опции

Опцията е договор (ценна книга), даваща на собственика правото, но не и задължението да продаде (пут опция) или да купи (кол опция) определено количество ценни книжа или валута на фиксирана в договора цена, в рамките на даден период от време. При опциите притежателя може да упражни или да не упражни своето право в съответния срок. При европейския вид опции собственикът им може да упражни правото си само в края на периода. Монте Карло и квази-Монте Карло методите могат директно да се приложат към финансови проблеми, включващи многомерни интеграли. По-долу ще илюстрираме как изследваните алгоритми могат ефективно да бъдат приложени към пресмятане на Европейски опции. Идеята се състои в следното: стойността на опцията V се формулира в термините на математическо очакване на случайна величина, след това средно аритметичното на независими реализации на случайната величина се използва за оценяване на опцията. Използва се рисково-неутралната

формула за Европейска опция:

$$V(S, t) = E(e^{-r(T-t)}h(S(T)) | S(t) = S, \mu = r), \quad (14)$$

където $E(\cdot)$ е математическото очакване, $h(S)$ е платежната функция, $h(S) = \max(S - E, 0)$ за кол опцията, $h(S) = \max(E - S, 0)$ за пут опцията.

Да разгледаме европейска опция, чиято платежна функция зависи от $k > 1$ актива с цени $S_i, i = 1, \dots, k$. Всеки актив следва случайното движение (така наречената случайна разходка): $dS_i = \mu_i S_i dt + \sigma_i S_i dX_i$, където σ_i е годишното стандартно отклонение на i -тия актив и dX_i е Брауновото движение. Да предположим, че при падежа T платежната функция се дава с: $h(S'_1, \dots, S'_k)$, (където S' означава стойността на i -тия актив при изтичане). Тогава текущата стойност на опцията, V , отхвърляйки възможността за арбитраж (получаване на безрискова печалба) ще бъде

$$V = e^{-r(T-t)}(2\pi(T-t))^{-k/2}(\det\Sigma)^{-1/2}(\sigma_1, \dots, \sigma_k)^{-1} \times \\ \times \int_0^\infty \dots \int_0^\infty \frac{h(S'_1, \dots, S'_k)}{S'_1, \dots, S'_k} \exp(-0.5\alpha^T \Sigma^{-1} \alpha) dS'_1, \dots, dS'_k,$$

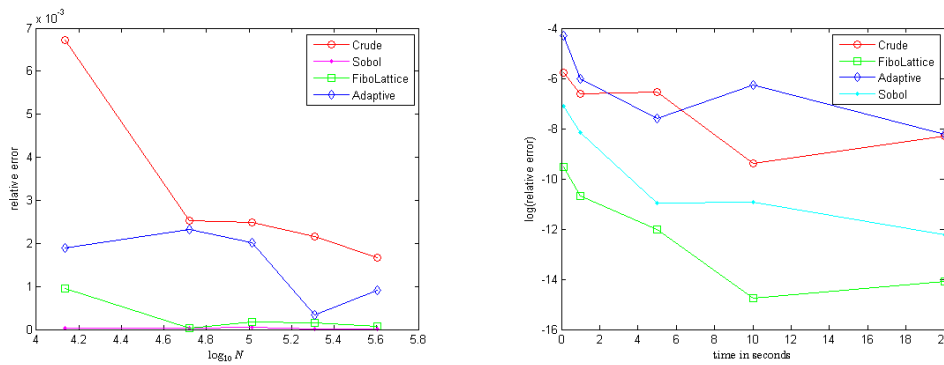
където $\alpha_i = (\sigma_i(T-t)^{1/2})^{-1}(\log \frac{S'_i}{S_i} - (r - \frac{\sigma_i^2}{2})(T-t))$, r е лихвеният процент и Σ е матрицата на ковариация, където елемента (i, j) е ковариацията на dX_i и dX_j за k актива. За задачи като тази, многомерните интеграли могат да се оценят чрез Монте Карло и квази-Монте Карло методи. За да получим търсената формулировка, безкрайната област на интегриране може да се изобрази в s -размерния единичен куб по различни начини. Например, $\frac{2}{\pi} \arctan(x)$ изобразява $(0, \infty)$ в $(0, 1)$. Може да се използва и функция на разпределение на различни случайни величини. Такава трансформация преобразува проблема в такъв, при който търсим стойността на многомерния интеграл $\int_0^1 \dots \int_0^1 g(x_1, \dots, x_s) dx_1 \dots dx_s$ над единичния хиперкуб. Когато g е функцията експонента, с подходящ избор на константите включени в горното уравнение [32] получаваме s -мерния интеграл $\int_{[0,1]^k} \exp(x_1, \dots, x_s) dx_1 \dots dx_s$ Числените експерименти включват пресмятането на 5 и 20-мерни интеграли:

$$\int_{[0,1]^5} \exp\left(\sum_{i=1}^5 0.5a_i x_i^2 (2 + \sin \sum_{j=1, j \neq i}^5 x_j)\right) dx \approx 2.923651, \quad a_i = (1, 0.5, 0.2, 0.2, 0.2), \quad (15)$$

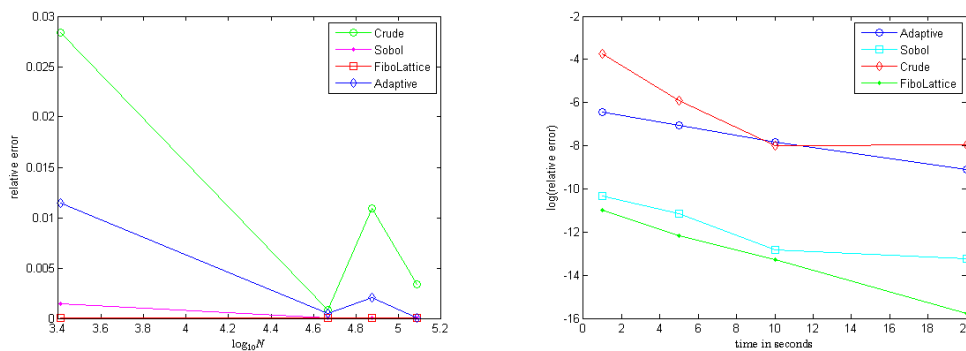
Таблица 8: Относителна грешка за 5-мерния интеграл

| време в сек. | обикновен МК | адаптивен МК | FIBO | Собол |
|--------------|--------------|--------------|---------|---------|
| 0.1 | 3.07e-3 | 1.34e-2 | 7.26e-5 | 8.22e-4 |
| 1 | 1.32e-3 | 2.44e-3 | 2.28e-5 | 2.91e-4 |
| 5 | 1.43e-3 | 4.93e-4 | 5.94e-6 | 1.71e-5 |
| 10 | 8.47e-5 | 1.88e-3 | 3.85e-7 | 1.79e-5 |
| 20 | 2.52e-4 | 2.71e-4 | 7.49e-7 | 4.96e-6 |

$$\int_{[0,1]^{20}} \exp\left(\prod_{i=1}^{20} x_i\right) dx \approx 1.00000949634. \quad (16)$$



Фигура 4: Относителна грешка и изчислително време за 5-мерния интеграл.



Фигура 5: Относителна грешка и изчислително време за 20-мерния интеграл.

Таблица 9: Относителна грешка за 20-мерния интеграл

| време в сек. | обикновен МК | адаптивен МК | FIBO | Sobol |
|--------------|--------------|--------------|---------|---------|
| 1 | 9.14e-3 | 1.58e-3 | 1.48e-5 | 3.25e-5 |
| 2 | 3.68e-3 | 1.028e-3 | 9.17e-6 | 3.97e-5 |
| 5 | 2.67e-3 | 8.58e-4 | 5.19e-6 | 1.45e-5 |
| 10 | 3.34e-4 | 4.02e-4 | 1.73e-6 | 2.71e-6 |
| 20 | 1.53e-4 | 1.13e-4 | 1.38e-7 | 1.76e-6 |

За всеки интеграл е пресметната относителната грешка при отнапред зададен брой реализации и получената относителна грешка при отнапред зададено изчислително време. Прави впечатление, че при малки размерности предимствата на FIBO са безспорни – виж Таблица 8 и Фиг. 4. При голяма размерност за 20-мерния интеграл методът FIBO и този с редици на Соболев дават близки резултати, но пак FIBO има предимство – виж Таблица 9. Адаптивния алгоритъм има предимства пред обикновения метод Монте Карло с увеличаване на размерността, но както видяхме той е най-подходящ за функции с особености в изчислително отношение. Най-бърз алгоритъм е обикновеният метод Монте Карло, като методът FIBO има сходна бързина. Може да се направи извода, че FIBO е най-добър избор за малки размерности, като дори за 20-мерния интеграл, има предимство по отношение на бързина и точност спрямо квази-Монте Карло алгоритъма с редици на Соболев – виж Фиг. 5.

Алгоритми Монте Карло за интегрални уравнения

В първа секция на втора глава е описан модифициран алгоритъм Монте Карло за приближено пресмятане на линеен функционал от решението на интегрално уравнение на Фредхолм от втори род, базиран на балансиране на систематичната и стохастичната грешка.

Резултатите, представени в тази глава, са публикувани в [9, 14] и са цитирани в статията с IF: [40].

Разглежда се задачата за приближено пресмятане на линеен функционал

(φ, u) от решението на интегралното уравнение на Фредхолм от втори род:

$$u(x) = \int_{\Omega} k(x, x')u(x')dx' + f(x), \quad x, x' \in \Omega \subset \mathbb{R}^d, \quad \text{или} \quad u = \mathcal{K}u + f,$$

където \mathcal{K} е интегрален оператор, $k(x, x') \in L_2(\Omega \times \Omega)$, $f(x) \in L_2(\Omega)$ са дадени функции, $u(x) \in L_2(\Omega)$ е неизвестна функция, Ω е ограничена област. В съответствие с началната $\pi(x)$ и преходните $p(x, x')$ вероятности в Ω се конструира случайна траектория (верига на Марков) T_k с дължина k , стартираща със състояние x_0 : $T_k: x_0 \rightarrow x_1 \rightarrow \dots \rightarrow x_k$. В [46] е показано, че

$$E\theta_k[\varphi] = (\varphi, u^{(k)}), \quad \text{където} \quad \theta_k[\varphi] = \frac{\varphi(x_0)}{\pi(x_0)} \sum_{j=0}^k W_j f(x_j),$$

$$W_0 = 1, \quad W_j = W_{j-1} \frac{k(x_{j-1}, x_j)}{p(x_{j-1}, x_j)}, \quad j = 1, \dots, k,$$

и съответната оценка по метод МК за $(\varphi, u^{(k)})$ е: $(\varphi, u^{(k)}) \approx \frac{1}{N} \sum_{n=1}^N \theta_k[\varphi]_n$. Следователно случайната величина $\theta_k[\varphi]$ може да се разгледа като оценка за търсената стойност (φ, u) при достатъчно голямо k с вероятностна грешка с порядък $\mathcal{O}(N^{-1/2})$, където N е броят на реализациите на веригата на Марков, а $\theta_k[\varphi]_n$ е стойността на $\theta_k[\varphi]$, получена върху n -тата траектория. Ако предположим [46]

$$p(x, x') = \frac{|k(x, x')|}{\int_{\Omega} |k(x, x')| dx'}, \quad \pi(x) = \frac{|\varphi(x)|}{\int_{\Omega} |\varphi(x)| dx},$$

тогава алгоритъмът се нарича почти оптимален Монте Карло алгоритъм (МАО) и това гарантира намаляване на дисперсията. За да се повиши изчислителната ефективност на алгоритъма трябва стохастичната грешка r_N да бъде приблизително равна на систематичната грешка r_k или да е изпълнено [7] $r_N = \mathcal{O}(r_k)$. Задачата за балансиране на грешките се трансформира в задача за получаване на оптимално съотношение между броя на реализациите N на случайната величина и средния брой стъпки k във всяка случайна траектория.

За оценка на вероятностната грешка е получено:

$$r_N \leq \frac{0.6745 \|f\|_{L_2} \|\varphi\|_{L_2}}{\sqrt{N} (1 - \|\mathcal{K}\|_{L_2})}.$$

За систематичната грешка получаваме следната оценка чрез неравенството на Коши-Шварц:

$$r_k = |(\varphi, u) - (\varphi, u^{(k)})| \leq \|\varphi\|_{L_2} \|u - u^{(k)}\|_{L_2} \leq \frac{\|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}^{k+1}}{1 - \|\mathcal{K}\|_{L_2}}.$$

Подобреният алгоритъм е базиран на по-долните твърдения, които играят важна роля за изчислителната му сложност. Да допуснем, че

$$r_N \leq \frac{0.6745 \|\varphi\|_{L_2} \|f\|_{L_2}}{\sqrt{N} (1 - \|\mathcal{K}\|_{L_2})} \leq \frac{\delta}{2}, \quad r_k \leq \frac{\|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}^{k+1}}{1 - \|\mathcal{K}\|_{L_2}} \leq \frac{\delta}{2}.$$

Приближаваме $k = k(N)$, като предполагаме, че $0.6745 \|f\|_{L_2} \frac{1}{\sqrt{N}} = \frac{\|\mathcal{K}\|_{L_2}^{k+1} \|f\|_{L_2}}{1 - \|\mathcal{K}\|_{L_2}}$

Теорема 4. (Теорема за балансираност) *В Монте Карло алгоритъма за интегрални уравнения, базиран на балансиране на систематичната и стохастичната грешка, долните граници за N и k са:*

$$N \geq \left(\frac{1.349 \|\varphi\|_{L_2} \|f\|_{L_2}}{\delta (1 - \|\mathcal{K}\|_{L_2})} \right)^2, \quad k \geq \frac{\ln \frac{\delta(1 - \|\mathcal{K}\|_{L_2})}{2 \|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}}}{\ln \|\mathcal{K}\|_{L_2}}.$$

Можем да получим оптимално съотношение между двете грешки по два различни начина:

В Монте Карло алгоритъма за интегрални уравнения базиран на балансиране на систематичната и стохастичната грешки, ако N е близо до своята долна граница, тогава долната граница за k е:

$$k \geq \frac{\ln \frac{0.6745}{\|\mathcal{K}\|_{L_2} \sqrt{N}}}{\ln \|\mathcal{K}\|_{L_2}}.$$

Ако първо изберем k да бъде близо до своята долна граница, по аналогичен начин получаваме долната граница за N :

$$N \geq \frac{0.455}{\|\mathcal{K}\|_{L_2}^{2k+2}}.$$

Горните твърдения са от особено важно значение за изчислителната сложност на алгоритъма.

Числени експерименти

Първият пример има приложно значение в биологията. Разглеждаме уравнението, описващо популационен модел в биологията [15]:

$$u(x) = \int_{\Omega} k(x, x') u(x') dx' + f(x),$$

Таблица 10: Относителна грешка и изчислително време за първия пример с балансиране на грешката (различни преходни плътности).

| δ | N | k | очаквана отн. грешка | експериментална(const) отн. грешка | време сек. | експериментална(MAO) отн. грешка | време сек. |
|----------|--------|---|-------------------------|---------------------------------------|---------------|-------------------------------------|---------------|
| 0.037 | 5101 | 4 | 2.24e-02 | 4.12e-03 | 11 | 4.06e-03 | 7 |
| 0.025 | 11172 | 5 | 1.52e-02 | 1.45e-03 | 16 | 1.21e-03 | 9 |
| 0.014 | 35623 | 6 | 8.49e-03 | 4.572e-04 | 56 | 4.001e-04 | 34 |
| 0.0055 | 230809 | 7 | 3.34e-03 | 1.524e-04 | 424 | 9.881e-05 | 346 |

където $\Omega \equiv [0, 1]$, $k(x, x') = \frac{1}{3}e^x$, $f(x) = \frac{2}{3}e^x$, $\varphi(x) = \delta(x)$.

Точното решение е $u(x) = e^x$. Искаме да намерим решението в средата на интервала. Изчисляваме L_2 нормите: $\|\varphi\|_{L_2} = 1$, $\|\mathcal{K}\|_{L_2} = 0.3917$, $\|f\|_{L_2} = 1.1915$. Монте Карло алгоритъмът започва от $x_0 = 0.5$, така че точното решение е 1.6487, $\pi(x) = \delta(x)$. Направени са 20 алгоритмични стъпки.

В Таблица 10 са представени броя на реализациите N и броя на итерациите k в зависимост от предварително зададената точност δ според получените условия за балансираност на систематичната и стохастичната грешка. Очакваната или теоретична грешка се получава като разделим δ на точната стойност. Седмата и осма колона са за експерименталната относителна грешка с MAO алгоритъма, който сме конструирали и времето за пресмятане на функционала от решението, а пета и шеста колона са съответно за алгоритъма Монте Карло за интегрални уравнения с константни плътности. Веднага се вижда, че очакваната относителна грешка се потвърждава от получената експериментална грешка. Може да се отчете предимство на алгоритъма MAO в сравнение с алгоритъма с константи плътности.

Разглежданият втори пример има важно значение в невронните мрежи и изкуствения интелект. Разглеждам следното интегрално уравнение на Фредхолм от втори род, описващо процеса на обучение на невронни мрежи [20]:

$$u(x) = \int_{\Omega} k(x, x')u(x') dx' + f(x),$$

където $\Omega \equiv [-2, 2]$, $k(x, x') = \frac{0.055}{1+e^{-3x}} + 0.07$, $f(x) = 0.02(3x^2 + e^{-0.35x})$, $\varphi(x) = 0.7((x+1)^2 \cos(5x) + 20)$. Точното решение е 8.98635750518, като $\|\varphi\|_{L_2} = 27.7782$,

Таблица 11: Относителна грешка и изчислително време за втория пример с балансиране на грешката (различни преходни плътности).

| δ | N | k | очаквана отн. грешка | експериментална(const) отн. грешка | време сек. | експериментална(МАО) отн. грешка | време сек. |
|----------|--------|---|-------------------------|---------------------------------------|---------------|-------------------------------------|---------------|
| 0.2 | 3457 | 4 | 2.23e-02 | 9.45e-03 | 9 | 1.21e-02 | 23 |
| 0.1 | 13827 | 4 | 1.11e-02 | 1.13e-02 | 28 | 8.63e-03 | 46 |
| 0.05 | 55306 | 5 | 5.56e-03 | 1.77e-02 | 132 | 3.24e-03 | 222 |
| 0.028 | 176357 | 5 | 3.12e-03 | 1.76e-02 | 448 | 3.11e-03 | 540 |

$\|\mathcal{K}\|_{L_2} = 0.2001$, $\|f\|_{L_2} = 0.2510$. Резултатите са изложени в Таблица 11.

Лесно може да се види, че балансираният МАО алгоритъм дава много по-добри резултати от Монте Карло алгоритма с константни плътности за големи стойности на N и k . При малък брой реализации метода Монте Карло с константни плътности дава по-малка относителна грешка, но резултатите получени с МАО алгоритъма са по-близки до очакваната теоретична относителна грешка. Единствено при балансирания МАО алгоритъм експерименталната грешка потвърждава очакваната относителна грешка. При МАО алгоритъма изчислителното време е по-голямо, защото използваме метода на селекцията за моделиране на началната плътност. В този случай това не е необходимо за преходните плътности, тъй като ядрото на интегралното уравнение е функция само на една променлива и когато поискаме преходните плътности да са пропорционални на ядрото, получаваме, че те са константи. Разработеният балансиран МАО алгоритъм дава много по-добри резултати от балансирания метод Монте Карло с константни плътности при условие, че преходната плътност е различна от δ -функцията. Когато преходната плътност е δ -функцията двата метода дават близки резултати, но отново балансирания МАО е малко по-точен.

Интересно е да се види дали предложения алгоритъм може да се прилага за нелинейни интегрални уравнения с полиномиална нелинейност. Може да се очаква, че ако нелинейността не е много строга вместо да се използват функционали върху разклоняващи се вериги на Марков ([10]), може да се използва конструирания балансиран алгоритъм. Следващия пример е от основно значение при моделирането на процеси във физиката [7]. Разглеждаме следното

Таблица 12: Относителна грешка за третия пример с балансиране на грешката.

| δ | N | k | очаквана отн. грешка | експериментална отн. грешка |
|----------|--------|---|-------------------------|--------------------------------|
| 0.03 | 1487 | 2 | 6.00e-02 | 8.56e-02 |
| 0.01 | 13381 | 3 | 2.00e-02 | 7.10e-02 |
| 0.005 | 53522 | 4 | 1.00e-02 | 6.02e-02 |
| 0.002 | 334512 | 5 | 4.00e-03 | 4.56e-02 |

интегрално уравнение с полиномиална нелинейност, описващо процеса на взаимодействие между две твърди физични тела:

$$u(x) = \int_{\Omega} \int_{\Omega} k(x, y, z) u(y) u(z) dy dz + f(x),$$

където $\Omega = \mathbf{E} \equiv [0, 1]$, $f(x) = c - \frac{x}{a_3}$, $\varphi(x) = \delta(x - x_0)$,

$$p(y, z) = \frac{6}{2a_2^2 - 3a_2 + 2} (a_2 y - z)^2, \quad k(x, y, z) = \frac{x(a_2 y - z)^2}{a_1}.$$

Едно достатъчно условие за сходимост на процеса е:

$$K_2 = \max_{x \in [0, 1]} \int_0^1 \int_0^1 |k(x, y, z)| dy dz = \frac{2a_2^2 - 3a_2 + 2}{6a_1} < \frac{1}{2}.$$

Интегралното уравнение има единствено регулярно решение $u(x) = c$, когато

$c = \pm \left(\frac{6a_1}{a_3(2a_2^2 - 3a_2 + 2)} \right)^{\frac{1}{2}}$. Избираме $a_1 = 11$, $a_2 = 4$, $a_3 = 12$, $c = 0.5$ и точното решение е $u(x) = 0.5$, $\|\mathcal{K}\|_{L_2} = 0.408$, $\|f\|_{L_2} = 0.459$, $\|\varphi\|_{L_2} = 1$. Резултатите са изложени в Таблица 12. Пропускаме метода Монте Карло с константи плътности, тъй като дава много лоши резултати за нелинейни интегрални уравнения.

За проблеми като този учените са доволни да имат грешка от порядъка на 5% или 10%. Оттук може да заключим, че конструираният почти оптимален метод Монте Карло за интегрални уравнения, базиран на балансиране на систематичната и стохастичната грешка, е приложим и за нелинейни интегрални уравнения.

Методи Монте Карло за линейни системи

Във втора секция на втора глава е конструиран и изследван подобрен метод Монте Карло за линейни системи (IWE), конструиран на базата на метода Монте Карло за линейни системи "случайно блуждаене по уравненията" (WE), описан в [13]. Направено е сравнение с рафинирания метод Монте Карло (RIMC), описан в [7]. Проведени са числени експерименти с различни примери на матрици от ниска и висока размерност. Методът може ефективно да се конкурира по бързина с вградените алгоритми в MATLAB и може да се приложи и при решаване на линейните алгебрични системи, получени след дискретизация на частните диференциални уравнения, описани в следващата глава.

Разглеждаме следната система от линейни алгебрични уравнения (СЛАУ): $Bx = f$, където $B = \{b_{ij}\}_{i,j=1}^n \in \mathbb{R}^{n \times n}$ е дадена матрица; $f = (f_1, \dots, f_n)^t \in \mathbb{R}^{n \times 1}$ и $v = (v_1, \dots, v_n) \in \mathbb{R}^{1 \times n}$ са дадени вектори. Използваме матрицата $A = \{a_{ij}\}_{i,j=1}^n$, такава че $A = I - DB$, където D е диагоналната матрица $D = \text{diag}(d_1, \dots, d_n)$ и $d_i = \frac{\gamma}{b_{ii}}$, $i = 1, \dots, n$, и $\gamma \in (0, 1]$ е релаксационен параметър, чийто избор влияе върху скоростта на итерационния процес. Горната система може да се представи като система във вида $x = Ax + b$, където $b = Df$ и a_{ij} са реални числа. Да предположим, че B има диагонално преобладаване. Очевидно, ако B е матрица с преобладаващ главен диагонал, тогава елементите на матрицата A трябва да удовлетворяват следните условия: $\sum_{j=1}^n |a_{ij}| \leq 1$, $i = 1, \dots, n$.

Дефинираме верига на Марков T_k с $n + 1$ състояния $\alpha_1, \dots, \alpha_n, n + 1$, такава че $P(\alpha_{k+1} = j | \alpha_k = i) = |a_{i,j}|$, ако $i \neq n + 1$ и $P(\alpha_{k+1} = n + 1 | \alpha_k = n + 1) = 1$. Дефинираме вектора c , такъв че $c_i = b_i$, ако $1 \leq i \leq n$ и $c(n + 1) = 0$. Означаваме с $\tau = (\alpha_0, \alpha_1, \dots, \alpha_k, n + 1)$ случайната траектория, която започва в началното състояние $\alpha_0 < n + 1$ и минава през $(\alpha_1, \dots, \alpha_k)$, докато стигне до състоянието на поглъщане $\alpha_{k+1} = n + 1$. Вероятността да бъде осъществена траекторията τ е $P(\tau) = p_{\alpha_0} p_{\alpha_0 \alpha_1} \dots p_{\alpha_{k-1}, k} p_{\alpha_k}$. Използваме МАО алгоритъма [7] за вектора на началната вероятност $p = \{p_\alpha\}_{\alpha=1}^n$ и за матрицата на преходните вероятности $P = \{p_{\alpha\beta}\}_{\alpha,\beta=1}^n$, съответно. Теглата Q_α се определят чрез:

$$Q_m = Q_{m-1} \frac{a_{\alpha_{m-1}, \alpha_m}}{p_{\alpha_{m-1}, \alpha_m}}, \quad m = 1, \dots, k, \quad Q_0 = \frac{c_{\alpha_0}}{p_{\alpha_0}}. \quad (17)$$

Величината $X_\alpha(\tau)$ може да се представи като $X_\alpha(\tau) = c_\alpha + Q_k \frac{a_{\alpha k} \alpha}{p_{\alpha k}}$, $\alpha = 1, \dots, n$, взета с вероятност $P(\tau) = p_{\alpha_0} p_{\alpha_0 \alpha_1} \dots p_{\alpha_{k-1}, k} p_{\alpha_k}$. Използваме, че случайната величина $X_\alpha(\tau)$ е неизместена оценка за x_α , т.е. $E\{X_\alpha(\tau)\} = x_\alpha$ [13].

Описание на алгоритъма

Идеята на алгоритъма е да се пресметнат приближено стойностите на всички компоненти на решението (разгледани като начални състояния). Първоначалното уравнение се избира случайно и равномерно сред първите n уравнения. След това, за всяко състояние i дефинираме общ резултат $S(i)$ с общ брой посещения $NV(i)$, които се модифицират, веднага щом състоянието i е посетено по време на случайното блуждаене. За дадена траектория запазваме посетените състояния в списък l , за да пресметнем по-лесно приноса им към резултата на посетените състояния. Накрая решението x_i се приближава с общия резултат $S(i)$, разделен на общия брой посещения $NV(i)$. Ако състоянието никога не е посетено, то $NV(i) = 1$. Ако състоянието е посетено повече от веднъж в рамките на една траектория, обикновено пазим приноса на първото посещение, за да се редуцира дисперсията. Общият брой траектории N се избира близко до размерността n на системата.

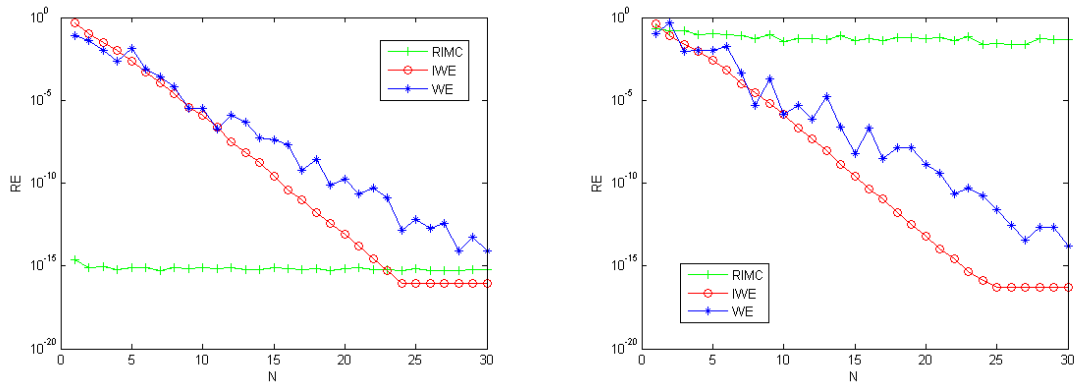
При ускоряване на сходимостта на алгоритмите се използва последователният метод Монте Карло (Sequential Monte Carlo, SMC) за линейни системи, който е реализиран от John Halton през 1960-те [22] и е развит в по-скорошни публикации [24, 23].

Подобреният алгоритъм за намиране на една компонента на решението се състои в избиране на специален избор на релаксационния параметър, което води до балансиране на итерационната матрица и до по-малък брой изчислителни операции спрямо WE, което води до намаляване на изчислителното време.

Числени експерименти

По-долу ще направим сравнения за линейни системи с различна размерност между стандартния рафиниран Монте Карло метод (RIMC) [7], оригиналният Монте Карло метод „случайно блуждаене по уравненията“ (WE) [13] и конструираният подобрен метод Монте Карло за линейни системи, означен с IWE, базиран на метода „случайно блуждаене по уравненията“.

Пример 1. Трябва да намерим решенията x_1 и x_2 дефинирани чрез линейната система $Bx = f$, където матриците B и векторите f_1 и f_2 са:



Фигура 6: Теглови резидуал на решението за матрицата 7×7 : (a) за x_1 ; (b) за x_2 .

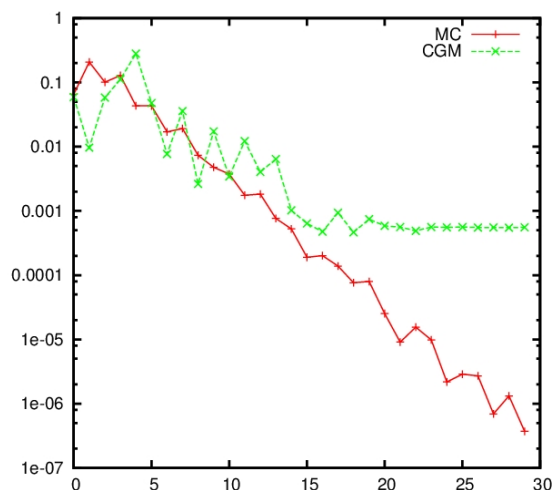
$$A = \begin{pmatrix} 5 & -1 & -1 & 0 & 0 & -1 & -1 \\ -1 & 5 & -1 & -1 & 0 & 0 & -1 \\ -1 & -1 & 5 & -1 & -1 & 0 & 0 \\ 0 & -1 & -1 & 5 & -1 & -1 & 0 \\ 0 & 0 & -1 & -1 & 5 & -1 & -1 \\ -1 & 0 & 0 & -1 & -1 & 5 & -1 \\ -1 & -1 & 0 & 0 & -1 & -1 & 5 \end{pmatrix}, \quad f_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad f_2 = \begin{pmatrix} 4 \\ -2 \\ -1 \\ 0 \\ -1 \\ -2 \\ 4 \end{pmatrix}. \quad (18)$$

Решенията са

$$x_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad x_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \quad (19)$$

Пример 2. Нека B е симетричната положително определена матрицата NOS4 от колекцията на Harwell-Boeing [55], а $f \in \mathbb{R}^{100}$, $b_i = 1$, $i = 1, \dots, 100$. Тази матрица е взета от приложения, свързани с апроксимация по метода на крайните елементи на модел, описващ гредова структура в конструктивната механика. Матрицата NOS4 има точно 597 ненулеви елемента, 100 ненулеви елемента по

главния диагонал и 247 под и над главния диагонал, или средно 5.9 ненулеви елемента във всяка колона и всеки ред. За матрицата NOS4 е показано, че методът Монте Карло WE е по-добър от метода на спрегнатия градиент PCG – виж Фиг. 7. Успехът на PCG се дължи на факта, че той е приложим за оптимални подпространства на Крилов. Методът Монте Карло също е приложим за оптимални подпространства на Крилов. Разликата е, че докато при PCG трябва да се решава оптимизационната задача, и ако има локален минимум до глобалния минимум, сходимостта на процеса може да клони към локалния минимум, като точно такъв е и случая за матрицата NOS4. Методът Монте Карло за линейни системи е независим от такава оптимизационна процедура, и затова резултатите с Монте Карло са по-добри. Този факт не може да се обобщи, понеже успехът на метода зависи от конкретния функционал, който трябва да се минимизира при PCG.



Фигура 7: Сравнение на метода Монте Карло IWE и метода PCG за матрицата NOS4 от колекцията Harwell-Boeing.

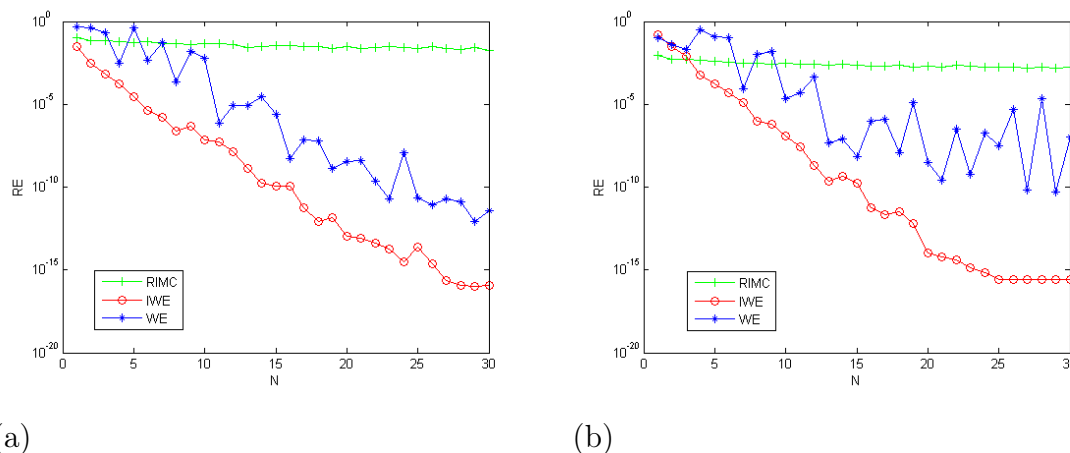
Пример 3. Нека B е плътна положителна матрица 5000×5000 с елементи в $[0,1]$, и $f \in \mathbb{R}^{5000}$, $b_i = 1$, $i = 1, \dots, 5000$.

Матриците B и дясната част f са нормирани, за да се ускори сходимостта на стохастичния процес. Избрани са специални стойности на релаксационния параметър γ . Числените експерименти показват, че това води до балансиране на итерационната матрица A . Може да се види, че в 7-мерния случай, разликата

Таблица 13: Теглови резидуал на решението за матриците $B \in \mathbb{R}^{100 \times 100}$ и $B \in \mathbb{R}^{5000 \times 5000}$

| N | 100 × 100 | | | | | | 5000 × 5000 | | | | | |
|----|-----------|------|----------|------|----------|------|-------------|-------|----------|-------|----------|------|
| | RIMC | t | WE | t | IWE | t | RIMC | t | WE | t | IWE | t |
| 2 | 7.25e-02 | 0.05 | 4.18e-01 | 0.84 | 3.02e-03 | 0.08 | 5.43e-03 | 10.05 | 4.30e-02 | 3.95 | 2.93e-02 | 0.15 |
| 5 | 5.45e-02 | 0.22 | 4.14e-01 | 2.37 | 3.07e-05 | 0.24 | 3.87e-03 | 60.2 | 1.21e-01 | 13.3 | 1.82e-04 | 0.9 |
| 10 | 4.32e-02 | 0.56 | 5.94e-03 | 5.31 | 7.46e-08 | 0.61 | 2.86e-03 | 130.5 | 2.30e-05 | 32.3 | 1.23e-07 | 2.4 |
| 15 | 3.52e-02 | 0.78 | 2.41e-06 | 9.1 | 1.21e-10 | 0.89 | 2.36e-03 | 310.7 | 6.48e-09 | 67.8 | 1.83e-10 | 5.1 |
| 20 | 3.19e-02 | 1.11 | 3.33e-09 | 13.5 | 1.02e-13 | 1.13 | 1.94e-03 | 811 | 3.20e-09 | 171.5 | 1.05e-14 | 11.1 |
| 30 | 1.83e-02 | 2.15 | 3.66e-12 | 24.6 | 1.11e-16 | 1.92 | 1.70e-03 | 2135 | 1.12e-07 | 418.6 | 2.48e-16 | 25.2 |

между WE и IWE при зададен брой итерации е 2-3 порядъка за $N > 15$. Трябва да се спомене, че сходимостта на рафинирания метод Монте Карло е много бавна, освен в случая на тривиалното решение за x_1 от Пример 1 – виж Фиг. 6. За 100-мерния случай на разредената матрица NOS4, подобреният алгоритъм IWE постига много по-добри резултати от WE и е почти 5-6 пъти по-бърз – виж Таблица 4.2. Поведението на Монте Карло алгоритъма не зависи от плътността на матрицата. Предимствата на алгоритъма са в сила и за разредени матрици. Разликата в относителната грешка за фиксиран брой итерации е 3-5 порядъка – виж Фиг. 7. Следователно IWE има предимство по отношение на по-добра точност с увеличаване на размерността. Също предимството на новия метод може ясно да се види по отношение на по-малко изчислително време. За по-голяма размерност, предимствата на IWE в сравнение с WE е още по-добре изразено. След 30 итерации WE постига точност близо до 10^{-10} , докато IWE постига точност 10^{-16} . Изчислителното време за IWE е 15 пъти по-добро в сравнение с WE – виж Таблица 20. Специалният избор на релаксационния параметър води до балансиране на итерационната матрица и това и това обяснява защо IWE е по-бърз. Предимството на подобрения Монте Карло алгоритъм е особено явно за големи размерности. Експериментите показват, че за големи размерности подобрението води до по-малка относителна грешка за по-малък брой SMC итерации за IWE.



Фигура 8: Теглови резидуал на решението за матрицата: (a) 100×100 ; (b) 5000×5000 .

Нови числени методи с висок ред на точност за модели в екологията

В последната глава се конструират нови компактни диференчни схеми (схеми с минимален шаблон) за слабо свързани системи от параболични частни диференциални уравнения (ЧДУ) с нелинейни химични реакции. Осъществено е приложение в едномерния и двумерния случай, както и при модел на далечен пренос на замърсители във въздуха на базата на Unified Danish Eulerian Model, UNI-DEM [52, 53]. Разглежда се подмодел, в който участват 10 химични замърсителя. Разглежда се случая на точно аналитично решение и когато няма точно решение, което съответства на реалната задача. Разглежда се правоъгълна област с размери от 500 км за едно денонощие. Разглежда се също и опростен модел на химични процеси в атмосферата по цикъла на Чапман на базата на три вещества. Направено е сравнение между два различни подхода за получаване на схеми от четвърти ред на точност- компактна диференчна схема и стандартна схема с повишен порядък с екстраполация по Ричардсон. Получена е схема с шести ред на точност по пространството като върху компактната схема е приложена екстраполация по Ричардсон.

Резултатите, представени в тази секция, са публикувани в [12].

Постановка на задачата

Тази глава е посветена на конструирането на нови компактни диференчни схеми (CFDS) с висок ред на точност за полулинейни параболични системи. Задачи от модели на преноси на замърсители във въздуха със свързани нелинейни реакции, са от основен интерес, а именно:

$$\frac{\partial u_l}{\partial t} - K \Delta u_l + \mathbf{b}_l \nabla u_l = R_l(x, y, u_1, \dots, u_L), \quad (x, y, t) \in \Omega \times (0, T], \quad (20)$$

$$\mathbf{u} = 0, \quad (x, y, t) \in \partial\Omega \times (0, T], \quad (21)$$

$$\mathbf{u} = \mathbf{u}_0(x, y), \quad (x, y) \in \Omega, \quad (22)$$

където $\mathbf{u} = (u_1, u_2, \dots, u_L)$, $u_l = u_l(x, y, t)$, $l = 1, \dots, L$ са концентрации на L химични вещества (замърсители) и $K > 0$ са коефициенти на дифузията и $\Omega \in R^2$ е ограничена област. Предположението за константите $K := K_x = K_y$ не е ограничение за развитието на разработения числен подход. Това съответства на физически модел, описан в [19, 29].

Основната цел е приложението и числените експерименти на по-горе споменатите диференчни апроксимации за следната реално физична параболична транспортна система, описана в [19]. Следвайки [19, 29, 54] адвекцията в (20) е

$$\mathbf{b}_l \cdot \nabla u_l = \mu(y - y_c) \frac{\partial u_l}{\partial x} + \mu(x_c - x) \frac{\partial u_l}{\partial y},$$

където $x \in (0, X)$, $y \in (0, Y)$, $x_c = X/2$, $y_c = Y/2$. Нелинейната химична част на модела е следната:

$$\begin{aligned} R_1(u_1, \dots, u_{10}) &= k_5 u_2 - (k_6 u_5 + k_4 u_7 + k_3 u_8) u_1, \\ R_2(u_1, \dots, u_{10}) &= (k_6 u_5 + k_4 u_7 + k_3 u_8) u_1 - (k_5 + k_9 u_9) u_2, \\ R_3(u_1, \dots, u_{10}) &= -k_1 u_3 u_9, \\ R_4(u_1, \dots, u_{10}) &= 2k_1 u_3 u_9 + k_3 u_1 u_8 - k_2 u_4, \\ R_5(u_1, \dots, u_{10}) &= k_2 u_5 \\ R_6(u_1, \dots, u_{10}) &= k_9 u_2 u_9, \\ R_7(u_1, \dots, u_{10}) &= 2k_2 u_4 + k_3 u_1 u_8 + k_{10} u_9 - k_4 u_1 u_7, \\ R_8(u_1, \dots, u_{10}) &= 4k_1 u_3 u_9 - k_3 u_1 u_8, \\ R_9(u_1, \dots, u_{10}) &= k_4 u_1 u_7 + 2k_8 u_{10} - (k_1 u_3 - k_9 u_2 + k_{10}) u_9, \\ R_{10}(u_1, \dots, u_{10}) &= k_7 u_5 - k_8 u_{10}. \end{aligned} \quad (23)$$

Таблица 14: Химични реакции на модела

| | | | |
|---|-------------------------------------------|----|---------------------------------------|
| 1 | $HC + OH \rightarrow 4RO_2 + 2ALD$ | 6 | $NO + O_3 \rightarrow NO_2 + O_2$ |
| 2 | $ALD + h\nu \rightarrow 2HO_2 + CO$ | 7 | $O_3 + h\nu \rightarrow O_2 + O(^1D)$ |
| 3 | $RO_2 + NO \rightarrow NO_2 + ALD + HO_2$ | 8 | $O(^1D) + H_2O \rightarrow 2OH$ |
| 4 | $NO + HO_2 \rightarrow NO_2 + OH$ | 9 | $NO_2 + OH \rightarrow HNO_3$ |
| 5 | $NO_2 + h\nu \rightarrow NO + O_3$ | 10 | $CO + OH \rightarrow CO_2 + HO_2$ |

Таблица 15: Скоростни константи на химичните реакции

| | | | |
|-------|--------------------------------------|----------|-------------------------------------|
| k_1 | $6.0e - 12$ | k_6 | $1.6e - 14$ |
| k_2 | $7.8e - 05. \exp(-0.87/\cos \theta)$ | k_7 | $1.6e - 04. \exp(-1.9/\cos \theta)$ |
| k_3 | $8.0e - 12$ | k_8 | $2.3e - 10$ |
| k_4 | $8.0e - 12$ | k_9 | $1.0e - 11$ |
| k_5 | $1.0e - 02. \exp(-0.39/\cos \theta)$ | k_{10} | $2.9e - 13$ |

Химичната част на модела е дадена в Таблица 14 за пълнота. Скоростните константи на химичните реакции са дадени в Таблица 15. Някои от коефициентите принадлежат на фотохимични реакции (тези с член $h\nu$), което значи, че тези реакции зависят от светлината, по точно от позицията на Слънцето спрямо хоризонта: в k_2 , k_5 и k_7 ъгълът θ означава слънчевия зенитен ъгъл, който е ъгъла на Слънцето, измерен по вертикала. Химичните вещества, включени в опростените реакции, са дадени в Таблица 16.

Таблица 16: Химичните вещества на модела

| | | | | | | | | | |
|-------|--------|-------|-------|-------|---------|--------|--------|-------|----------|
| u_1 | u_2 | u_3 | u_4 | u_5 | u_6 | u_7 | u_8 | u_9 | u_{10} |
| NO | NO_2 | HC | ALD | O_3 | HNO_3 | HO_2 | RO_2 | OH | $O(^1D)$ |

От практическа и математическа гледна точка, интерес представлява съществуването и качественото поведение (неотрицателността) на решението на задачата (20)-(22). Коректността на начално-граничната задача за по-обща система

от (20) е получена в [41]. Ще допускаме до края съществуване и единственост на класическо решение на задачата (20)-(22), което означава, че функцията принадлежи на $C([0, T] \times \bar{\Omega}) \cap C^1((0, T); C(\bar{\Omega})) \cap (C(0, T); C^2(\bar{\Omega}))$ и удовлетворява (20)-(22) поточно.

Разглеждаме система, описваща химични концентрации, неотрицателността на решението трябва да се запази. Доказано е [41], че ако

1. $\mathbf{u}_0(x, y) \geq 0$;
2. $R_l(x, y, \mathbf{u})$, $l = 1, \dots, L$ е непрекъсната по Липшиц по отношение на концентрациите u_1, u_2, \dots, u_L и удовлетворява неравенството $R_l(x, y, \mathbf{u}) \geq 0$, където $u_l = 0$, и $\mathbf{u} \in R_+^L \equiv \{u_k \geq 0, k = 1, \dots, L\}$,

тогава $\mathbf{u} \geq 0$ за всяко $(x, y) \in \Omega$ и $t \in [0, T]$.

Лесно е да се провери, че химичните реакции $R_l(u_1, u_2, \dots, u_{10})$, $l = 1, \dots, 10$ зададени от (23) удовлетворяват точка 2. и решението на задачата (20)-(22) с (23) е неотрицателно по времето $t > 0$, ако началните данни $\mathbf{u}_0(x, y) \geq 0$.

В последните години се наблюдава нов и все по-нарастващ интерес към компактните диференчни схеми с висок ред на точност за решаване на ЧДУ [42]. Напоследък, голямо усилие се съсредоточава в развитието на компактни диференчни схеми с висок ред на точност, които използват само възлите на мрежата, съседни на централния възел. Компактните схеми, предложени от Крайс и Олигер [21] използват подобен шаблон, но изискват тридиагонално или петдиагонално обръщане. Друга идея, която се използва, е да се работи върху диференциалните уравнения, така че да се изразят производните от по-висок ред (трети и четвърти) в локалната грешка на апроксимация (LTE) [49].

Компактни диференчни схеми (CFDS) в двумерния случай

В тази секция се описва построяването на компактни диференчни схеми (CFDS) в двумерния случай за системата от две уравнения:

$$\frac{\partial u}{\partial t} - a(x, y) \frac{\partial^2 u}{\partial x^2} - b(x, y) \frac{\partial^2 u}{\partial y^2} + c(x, y) \frac{\partial u}{\partial x} + d(x, y) \frac{\partial u}{\partial y} = r(x, y, t, u, v), \quad (24a)$$

$$\frac{\partial v}{\partial t} - e(x, y) \frac{\partial^2 v}{\partial x^2} - f(x, y) \frac{\partial^2 v}{\partial y^2} + g(x, y) \frac{\partial v}{\partial x} + h(x, y) \frac{\partial v}{\partial y} = s(x, y, t, u, v), \quad (24b)$$

дефинирана в областта $Q_T = \Omega \times (0, T)$, където $\Omega \subset R^2$ е ограничена област с Липшицова граница. Нелинейните функции r и s са достатъчно гладки по тех-

ните аргументи. Коефициентите $a(x, y)$, $b(x, y)$, $e(x, y)$ и $f(x, y)$ са положителни в Ω . Предполагаме гранични условия на Дирихле

$$u(x, y, t) = \bar{\phi}(x, y, t), \quad v(x, y, t) = \bar{\bar{\phi}}(x, y, t), \quad (x, y, t) \in \partial\Omega \times (0, T] \quad (25)$$

и начални условия

$$u(x, y, 0) = \bar{\psi}(x, y), \quad v(x, y, 0) = \bar{\bar{\psi}}(x, y), \quad (x, y) \in \Omega, \quad (26)$$

където $\bar{\phi}$, $\bar{\bar{\phi}}$, $\bar{\psi}$ и $\bar{\bar{\psi}}$ са дадени и гладки и съвместимост с началните и гранични условия е осигурена. Нека за простота областта Ω е правоъгълник $\Omega = [0, X] \times [0, Y]$. Да въведем равномерните мрежи $\bar{\omega}_{h,x} = \{x_i = ih_x, i = 0, 1, \dots, N_x, h_x = X/N_x\}$, $\bar{\omega}_{h,y} = \{y_j = jh_y, j = 0, 1, \dots, N_y, h_y = Y/N_y\}$ и тогава $\bar{\Omega}_h = \omega_{h,x} \times \omega_{h,y}$, $\bar{\bar{\Omega}}_h = \Omega_h \cup \partial\Omega_h$, където Ω_h се състои от всички вътрешни мрежови точки, а $\partial\Omega_h$ - от всички гранични мрежови точки.

Дискретизация по пространството. Идеята е, че в локалната грешка на апроксимация членовете от втори порядък съдържат трета и четвърта производна на решението, и за да останем на същия шаблон, трябва да ги изразим чрез производни от по-нисък ред, като диференцираме двукратно диференциалните уравнения. Следователно, за да елиминираме членовете от ред $\mathcal{O}(h_x^2 + h_y^2)$, диференцираме уравнението (24a) два пъти по x и получаваме израз за $\frac{\partial^3 u}{\partial x^3}$, $\frac{\partial^4 u}{\partial x^4}$, и два пъти по y за $\frac{\partial^3 u}{\partial y^3}$, $\frac{\partial^4 u}{\partial y^4}$. В подробности тази процедура е описана в [12, 49].

Получаваме следната система ОДУ

$$\bar{Q} \frac{d}{dt} U^h + \bar{P} U^h = \bar{Q} R + \bar{\Phi}, \quad t \in (0, T], \quad (27)$$

$$\bar{\bar{Q}} \frac{d}{dt} V^h + \bar{\bar{P}} V^h = \bar{\bar{Q}} S + \bar{\bar{\Phi}}, \quad (28)$$

с начални условия $U^h(0)$ и $V^h(0)$, получени от $\bar{\psi}$ и $\bar{\bar{\psi}}$ за $(i, j) \in \Omega_h$ след пренареждането. В системата (27), (28) матрицата \bar{P} (аналогично $\bar{\bar{P}}$) е $(N_y - 1) \times (N_y - 1)$ блочно тридиагонална матрица $\bar{P} = \text{tridiag}(\bar{P}_{k,k-1}, \bar{P}_{k,k}, \bar{P}_{k,k+1})$ и $\bar{P}_{k,l}$, $l = k - 1, k, k + 1$ са също тридиагонални матрици от ред $(N_x - 1) \times (N_x - 1)$. Тогава елементите на $\bar{P}_{k,l}$ са

$$\bar{P}_{k,l} = \text{tridiag}(p_{k,2:N_x-1}^{(-1,\varepsilon)}, p_{k,2:N_x}^{(0,\varepsilon)}, p_{k,1:N_x-2}^{(1,\varepsilon)}) \quad l = k + \varepsilon, \quad \varepsilon = 0, \pm 1. \quad (29)$$

Елементите на $\bar{\bar{Q}}_{k,l}$ (аналогично \bar{Q}) са

$$\bar{\bar{Q}}_{k,l} = \text{tridiag}(q_{k,2:N_x-1}^{(-1,\varepsilon)}, q_{k,2:N_x}^{(0,\varepsilon)}, q_{k,1:N_x-2}^{(1,\varepsilon)}) \quad l = k + \varepsilon, \quad \varepsilon = 0, \pm 1 \quad (30)$$

със забележката, че за $\varepsilon = \pm 1$ матриците $\overline{Q}_{k,l}$ са диагонални (вместо тридиагонални) матрици. Векторите $\overline{\Phi}$ и $\overline{\overline{\Phi}}$ са свързани с граничните функции и също зависят от времето t .

Дискретизация по времето. За дискретизация по времето е използвана схема с тегла с $\theta = 1/2$. Тогава пълната дискретизация по Кранк-Никълсън за (27), (28), е както следва:

$$\begin{aligned} \overline{Q} \frac{U^{n+1} - U^n}{\tau} + \overline{P} U^{n,\theta} &= \overline{Q} R^{n,\theta} + \overline{\Phi}^{n,\theta}, \quad n = 1, \dots, N - 1, \\ \overline{\overline{Q}} \frac{V^{n+1} - V^n}{\tau} + \overline{\overline{P}} V^{n,\theta} &= \overline{\overline{Q}} S^{n,\theta} + \overline{\overline{\Phi}}^{n,\theta}, \quad n = 1, \dots, N - 1. \end{aligned} \quad (31)$$

Системата (31) е пренаписана във формата $\Upsilon(W) = 0$, където $W = [U^T, V^T]^T$ е вектор с дължина $2(N_x - 1)(N_y - 1)$. Поставяме W^{n+1} като начално приближение на новия слой по времето $t = t_{n+1}$ да бъде решението на стария слой по времето $t = t_n$. Тогава за да намерим решението на слоя $t = t_{n+1}$ итерационен процес с подходящо стоп правило е използван:

$$\begin{cases} \Upsilon'(W^{n+1}) \Delta^k = -\Upsilon(W^{n+1}), \\ W^{n+1} = W^{n+1} + \Delta^k. \end{cases} \quad (32)$$

Тук Δ^k е вектора на нарастванията и матрицата на Якобиана $\Upsilon'(W^{n+1})$ за $\theta = 1/2$ е

$$\Upsilon'(W^{n+1}) = \begin{pmatrix} \frac{1}{\tau} \overline{Q} I + \frac{1}{2} \overline{P} - \frac{1}{2} \overline{Q} \frac{\partial R}{\partial U} & \frac{1}{2} \overline{Q} \frac{\partial R}{\partial V} \\ \frac{1}{2} \overline{\overline{Q}} \frac{\partial S}{\partial U} & \frac{1}{\tau} \overline{\overline{Q}} I + \frac{1}{2} \overline{\overline{P}} - \frac{1}{2} \overline{\overline{Q}} \frac{\partial S}{\partial V} \end{pmatrix} \Big|_{(U,V)=(U^k, V^k)}. \quad (33)$$

В числените експерименти, за да решим първото уравнение в (32), който е линейна система от $2(N_x - 1)(N_y - 1)$ уравнения, използваме така наречения „неточен метод“ на Нютон (inexact Newton) [6], т.е. решаваме системата приближено с вградената MatLab функция bicgstab(1) (biconjugate gradients stabilized (1) method), който дава най-добрите резултати за числените експерименти по отношение на брой вътрешни операции и изчислително време. Може да се използва и методът Монте Карло за линейни системи, но е значително по-бавен за разглежданите задачи.

Екстраполация на Ричардсон

Друг начин за получаване на диференчни схеми с висок ред на точност, е да се използва метода на екстраполация на Ричардсон (RE). Главната идея [36] е да се реши диференчната схема на две или повече вложени мрежи и след това да се комбинират получените числени решения с подходящи тегла. Да допуснем, че $h_x = h_y = h$ и за численото решение на n -тия слой по времето, следният израз е верен:

$$U_h^\tau = U_{(i,j)}^n = u(x_i, y_j, t^n) + C_1 h^\sigma + \chi(h, \tau), \quad (x_i, y_j, t_n) \in \Omega_{h,\tau}, \quad (34)$$

където функцията $\chi(h, \tau)$ е остатъчен член и константата C_1 не зависи от h_x , h_y и τ . Ако искаме да елиминираме члена $C_1 h^\sigma$, се правят следните стъпки:

- решаваме диференчната схема на две последователни мрежи: груба $\Omega_{h,\tau}$ и фина $\Omega_{h/2,\tau}$ и нека съответстващите числени решения да бъдат U_h^τ и $U_{h/2}^\tau$;
- намираме теглата γ_1 и γ_2 от системата

$$\begin{aligned} \gamma_1 + \gamma_2 &= 1 \\ \gamma_1 + \frac{\gamma_2}{2^\sigma} &= 0 \end{aligned} \quad (35)$$

- получаваме ново числено решение на грубата мрежа

$$U_{extr} = \gamma_1 U_h^\tau + \gamma_2 U_{h/2}^\tau \quad (x_i, y_j, t_n) \in \Omega_{h,\tau}.$$

От (35) имаме за случая на стандартна схема (CDS) на Кранк-Никълсън ($\sigma = 2$), че коефициентите на екстраполацията по Ричардсон са

$$\gamma_1 = -1/3 \quad \gamma_2 = 4/3. \quad (36)$$

За случая на компактна схема (CFDS) и екстраполация по Ричардсон ($\sigma = 4$) съответните теглови коефициенти са

$$\gamma_1 = -1/15 \quad \gamma_2 = 16/15. \quad (37)$$

Числени експерименти за модел в екологията - системата ЧДУ от 10 уравнения

В тази секция се разглеждат два числени експеримента за потвърждение на теоретичните резултати. Първата задача (Пример1) е с точно решение, а втората (Пример 2) е опростена двумерна моделна задача с пренос на замърсители, описан в [19].

Пример 1 (известно аналитично решение)

Тук разглеждаме проблем, малко по различен от проблема (20):

$$\frac{\partial u_l}{\partial t} - K \Delta u_l + \mathbf{b}_l \cdot \nabla u_l = R_l(x, y, \mathbf{u}) + \xi_l(x, y, t), (x, y, t) \in \Omega \times (0, T]. \quad (38)$$

Функциите ξ_l , $l = 1, \dots, 10$, и началните и гранични условия са избрани, така че точното решение да бъде

$$u_l = \exp(-t/T) \sin\left(\frac{\pi x}{X}\right) \sin\left(\frac{\pi y}{Y}\right), \quad l = 1, \dots, 10, \quad (x, y, t) \in \partial\Omega \times [0, T]. \quad (39)$$

Другите параметри са, както следва: $X = Y = 500$, $T = 1440$, $\mu = 2\pi/(60T)$, $K = 1.8$.

За l -тата субстанция с $error_{M,l}$ означаваме грешката (разликата между точното и приближеното решение) в равномерна норма, получена на последния слой по времето $t_N = T$ за брой подинтервали по времето $M_x = M_y = M$:

$$error_{M,l} = \max_{i,j \in \overline{\Omega}_h} \|u_l(x_i, y_j, t_N) - u_l^h(i, j, N)\|.$$

Отношението между грешките, получени на две последователни мрежи (обикновено удвоени) е означено с $ratio$:

$$ratio = ratio_{M,l/2M,l} =: error_{M,l}/error_{2M,l}.$$

В таблиците са представени анализи със съгъстяване на мрежата с използването на CDS и CFDS, CDS и CFDS с (RE) по пространството и CDS и CFDS с (RE) по пространството и времето. Резултатите потвърждават теоретичния ред на сходимост за двата числени метода. От сравняването на CPU времето за Таблицы 17, 18 и 19 следва предимство на използването на RE едновременно по времето и пространството за получаването на по-малки грешки за по-кратко време. Предимството на CFDSRE е също очевидно. Фиг. 9 показва грешката в

Таблица 17: Сравнение на грешката за CDS и CFDS за **Пример 1**

| CDS, $O(h^2 + \tau^2)$ | | | | | | CFDS, $O(h^4 + \tau^2)$ | | | | | |
|------------------------|-------|-----|------------|---------|-------|-------------------------|-------|------|------------|---------|--------|
| M_x | M_y | N | $error_M$ | $ratio$ | CPU | M_x | M_y | N | $error_M$ | $ratio$ | CPU |
| 4 | 4 | 4 | 5.702 e-03 | - | 0.58 | 4 | 4 | 4 | 5.875 e-03 | - | 0.72 |
| 8 | 8 | 8 | 1.449 e-03 | 3.94 | 1.82 | 8 | 8 | 16 | 3.595 e-04 | 16.34 | 3.04 |
| 16 | 16 | 16 | 3.637 e-04 | 3.99 | 14.42 | 16 | 16 | 64 | 2.232 e-05 | 16.11 | 29.74 |
| 32 | 32 | 32 | 9.102 e-05 | 4.001 | 143.7 | 32 | 32 | 256 | 1.392 e-06 | 16.03 | 1076 |
| 64 | 64 | 64 | 2.276 e-05 | 4.00 | 3959 | 64 | 64 | 1024 | 8.698 e-08 | 16.003 | 60907 |
| 128 | 128 | 128 | 5.691 e-06 | 4.00 | 32709 | 128 | 128 | 4096 | 5.436 e-09 | 16.0001 | 720477 |

Таблица 18: Грешка в равномерна норма за **Пример 1** за CDS и CFDS в RE по пространството

| CDS с RE по пространството, $O(h^4 + \tau^2)$ | | | | | | CFDS с RE по пространството, $O(h^6 + \tau^2)$ | | | | | |
|-----------------------------------------------|-------|-----|------------|---------|-------|------------------------------------------------|-------|------|-------------|---------|-------|
| M_x | M_y | N | err_N | $ratio$ | CPU | M_x | M_y | N | err_N | $ratio$ | CPU |
| 4 | 4 | 4 | 5.677 e-03 | - | 1.34 | 4 | 4 | 4 | 5.711 e-03 | - | 1.38 |
| 8 | 8 | 16 | 3.545 e-04 | 16.014 | 16.17 | 8 | 8 | 32 | 8.912 e-05 | 64.087 | 17.45 |
| 16 | 16 | 64 | 2.216 e-05 | 15.997 | 544 | 16 | 16 | 256 | 1.392 e-06 | 64.022 | 1497 |
| 32 | 32 | 256 | 1.385 e-06 | 16.001 | 3055 | 32 | 32 | 2048 | 2.1757 e-08 | 63.989 | 23390 |

максимална норма за *Пример 1* (а) със CDS и RE по пространството и времето за $M_x = M_y = 16$, $N = 16$; (б) с CFDS и RE по пространството и времето за $M_x = M_y = 16$, $N = 64$ и е в съответствие с резултатите в Таблица 19.

Пример 2 (без точно решение)

В този случай е разгледан по-реалистичен вариант на задачата (20)-(22) със следните параметри на областта: областта е квадрат $\Omega = [0, 500]^2$ с дължина 500 km, дължината на интервала по времето $[0, T]$ е 1440 min и броят на уравненията е $L = 10$. Началните условия на първия слой по времето $t = 0$ са константни функции

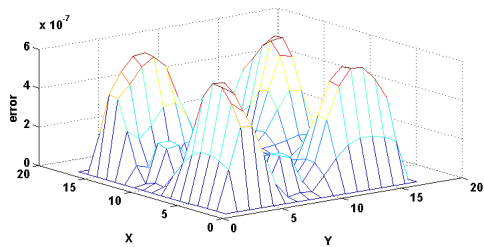
$$\mathbf{u}_0(x, y) = (10^3, 10^3, 10^3, 5.10^3, 5.10^3, 10^2, 10^{-2}, 10^{-2}, 10^{-3}, 10^{-11}),$$

измерени в mol/km^3 и граничните условия са избрани периодични функции: γ_i които имат вида

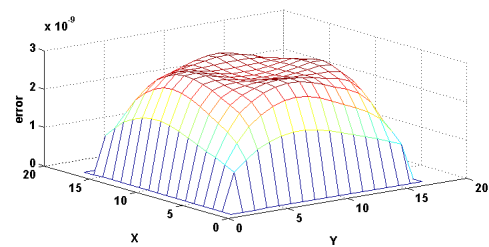
$$\gamma_i(t) = const_i(\sin(t/C) + 2),$$

Таблица 19: Грешка в равномерна норма за **Пример 1** за CDS и CFDS в RE по пространството и времето

| CDS с RE, $O(h^4 + \tau^4)$ | | | | | | CFDS с RE, $O(h^6 + \tau^4)$ | | | | | |
|-----------------------------|-------|----|------------|-------|--------|------------------------------|-------|------|------------|-------|--------|
| M_x | M_y | N | err_N | ratio | CPU | M_1 | M_y | N | err_N | ratio | CPU |
| 4 | 4 | 4 | 5.649 e-05 | - | 6.73 | 4 | 4 | 4 | 8.476 e-06 | - | 3.36 |
| 8 | 8 | 8 | 9.722 e-06 | 5.81 | 18.71 | 8 | 8 | 16 | 1.748 e-07 | 48.49 | 30.26 |
| 16 | 16 | 16 | 5.989 e-07 | 16.23 | 194.81 | 16 | 16 | 64 | 2.847 e-09 | 61.39 | 1276 |
| 32 | 32 | 32 | 3.715 e-08 | 16.12 | 4594 | 32 | 32 | 256 | 4.529 e-11 | 62.86 | 66991 |
| 64 | 64 | 64 | 2.171 e-09 | 16.03 | 37101 | 64 | 64 | 1024 | 7.086 e-13 | 63.91 | 790800 |



(a)



(b)

Фигура 9: Грешката в максимална норма за *Пример 1*: (a) CDS с RE по пространството и времето за $M_x = M_y = 16, N = 16$; (b) CFDS с RE по пространството и времето за $M_x = M_y = 16, N = 64$

където C е константа и константите $const_l, l = 1, \dots, L$ са избрани по такъв начин, че да е осигурена съгласуваност на началните и граничните условия. Коефициентът на дифузия е избран да бъде $K = 1.8 km^2/min$ и коефициентът $\mu = 2\pi/(60 * T)$. В този случай няма аналитично решение. Един възможен начин за пресмятане на реда на сходимост е методът на Рунге на три вложени мрежи. Тук използваме друга идея. Като „точно” решение вземаме решението, получено с най-малките размери на мрежата по пространството. В таблицата това решение е означено с удебелен шрифт. Следим реда на сходимост означен с *order* и пресметнат с помощта по формулата: $order = \log_2(ratio)$ когато удвояваме броя на мрежовите точки и в другия случай $order = \log(error_{M',l}/error_{M'',l})/\log(M''/M')$, където M' и M'' са два последователни броя мрежови точки по пространството в анализите на съгъстванията на мре-

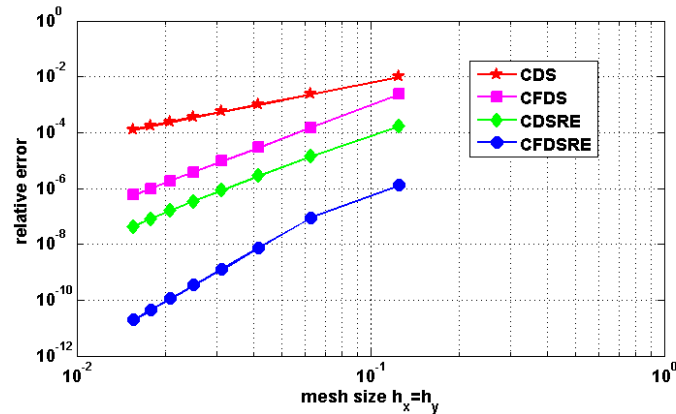
Таблица 20: Числените стойности, относителната грешка и реда на сходимост за **Пример 2** за CFDS за централния възел $(x, y) = (X/2, Y/2)$ със слоеве по времето $N = 256$ за u_1 и u_5

| U_1 | | | | | U_5 | | | | |
|-------|-------|---------------------|------------|-------|-------|-------|---------------------|------------|-------|
| M_x | M_y | числ. ст. | отн. гр. | ред | M_x | M_y | числ. ст. | отн. гр. | ред |
| 8 | 8 | 2000.6332968 | 2.273 e-03 | - | 8 | 8 | 4580.1540334 | 2.417 e-03 | - |
| 16 | 16 | 1996.1958272 | 1.495 e-04 | 3.988 | 16 | 16 | 4569.7951222 | 1.495 e-04 | 4.014 |
| 24 | 24 | 1995.9567369 | 2.972 e-05 | 3.984 | 24 | 24 | 4569.2477794 | 2.972 e-05 | 3.984 |
| 32 | 32 | 1995.9162197 | 9.419 e-06 | 3.994 | 32 | 32 | 4569.1550256 | 9.420 e-06 | 3.994 |
| 40 | 40 | 1995.9051188 | 3.858 e-06 | 4.000 | 40 | 40 | 4569.1296128 | 3.858 e-06 | 4.000 |
| 48 | 48 | 1995.9011265 | 1.858 e-06 | 4.008 | 48 | 48 | 4569.1204736 | 1.858 e-06 | 4.008 |
| 56 | 56 | 1995.8994140 | 9.997 e-07 | 4.019 | 56 | 56 | 4569.1165532 | 9.997 e-07 | 4.019 |
| 64 | 64 | 1995.8985824 | 5.831 e-07 | 4.037 | 64 | 64 | 4569.1146497 | 5.831 e-07 | 4.037 |
| 192 | 192 | 1995.8974186 | | | 192 | 192 | 4569.1121206 | | |

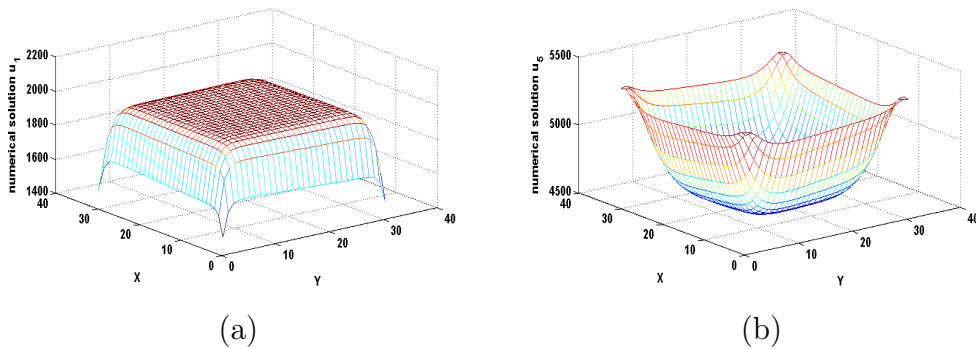
жата.

В Таблица 20 са представени резултатите от CDFS брой слоеве по времето $N = 256$ за първото и петото вещество u_1 и u_5 в централния възел с координати $(x_{M/2}, y_{M/2}) = (X/2, Y/2) = (250, 250)$. Със същите параметри експериментите са повторени с използването на CDS с RE. Четвъртият ред на сходимост в двата случая (централен възел $(x,y)=(X/2,Y/2)$ и възел $(x,y)=(X/6,Y/6)$) за двете вещества u_1 и u_5 е потвърден. Отново в централния възел относителните грешки са почти еднакви. Потвърден е и шестия ред на сходимост пространството на CFDS с RE.

На Фиг. 10 Логаритмична графика на грешките и пространствените мрежови възли за *Пример 2* е представена, получена с: CDS - червена линия; CFDS - виненочервена линия; CDSRE - зелена линия; CFDSRE - синя линия. Нарастването на наклона на линиите съответства на нарастването на реда на сходимост. Най-наклонената линия съответства на предимството на CFDS в комбинация с екстраполация по Ричардсон. Проведени са и други експерименти. Интересно е да се види поведението на решението, ако коефициентите μ на конвекцията



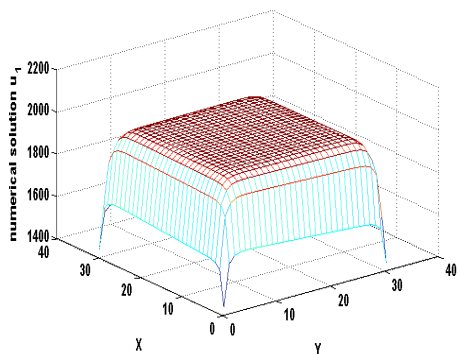
Фигура 10: Логаритмична графика на грешките и пространствените мрежови възли за *Пример 2*, получен с: CDS - червена линия, —★—; CFDS - виненочервена линия, —■—; CDSRE - зелена линия, —◆—; CFDSRE - синя линия, —●—.



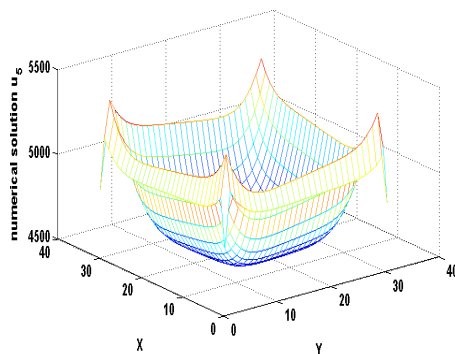
Фигура 11: Численото решение получено с CDS за $\mu = 2\pi/(60T)$ с мрежови параметри $M_x = M_y = 32$, $N = 256$ за *Пример 2*: (a) за u_1 ; (b) за u_5

са $\mu = 2\pi/(X)$, както е в [29], вместо $\mu = 2\pi/(60T)$, както е в [19]. На Фиг. 11 е показано численото решение получено с CDS за $\mu = 2\pi/(60T)$ с мрежови параметри $M_x = M_y = 32$, на последния слой по времето $N = 256$ (a) за u_1 ; (b) за u_5 . Аналогично, на Фиг. 12 е показано численото решение, получено с CFDS. Нарастването на конвективните членове води до съществено изменение на численото решение близо до границата – виж Фиг. 13.

Въпреки всички предимства на CFDS по отношение на точност и CPU време, има някои недостатъци. Шаблонът на CFDS е 9 точков и условието на знаците на дискретния принцип на максимума не е изпълнено. В резултат на

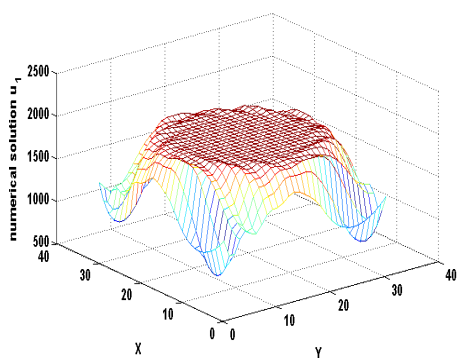


(a)

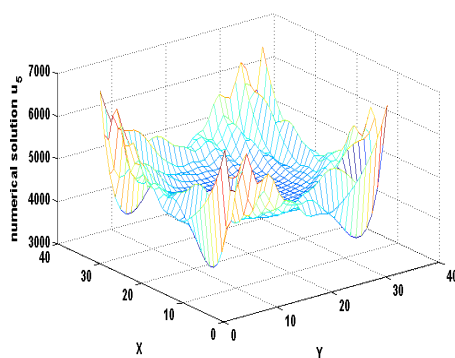


(b)

Фигура 12: Численото решение получено с CFDS за $\mu = 2\pi/(60T)$ с мрежови параметри $M_x = M_y = 32$, $N = 256$ за *Пример 2*: (a) за u_1 ; (b) за u_5

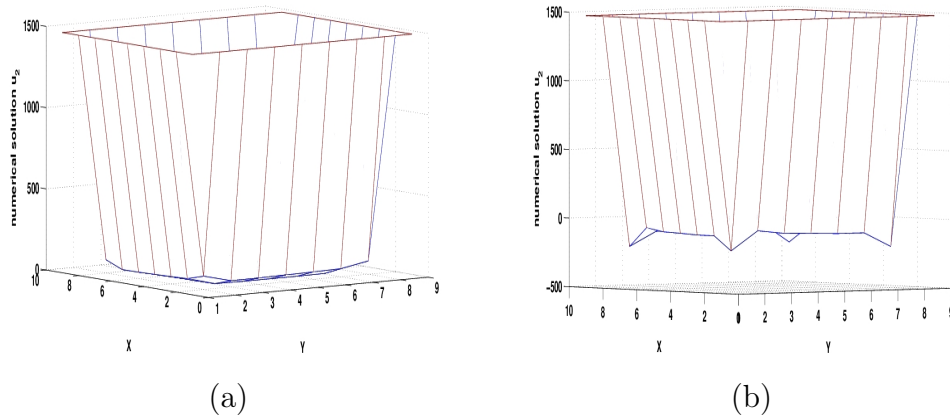


(a)



(b)

Фигура 13: Численото решение за *Пример 2*, получено с CFDS за $\mu = 2\pi/(500)$ и $M_x = M_y = 32$, $N = 256$: (a) за u_1 ; (b) за u_5



Фигура 14: Численото решение за замърсителя NO_2 - u_2 за $\mu = 2\pi/(60T)$ с параметри на мрежата $M_x = M_y = 8$, $N = 256$ Пример 2, получено с: (a) CDS; (b) CFDS

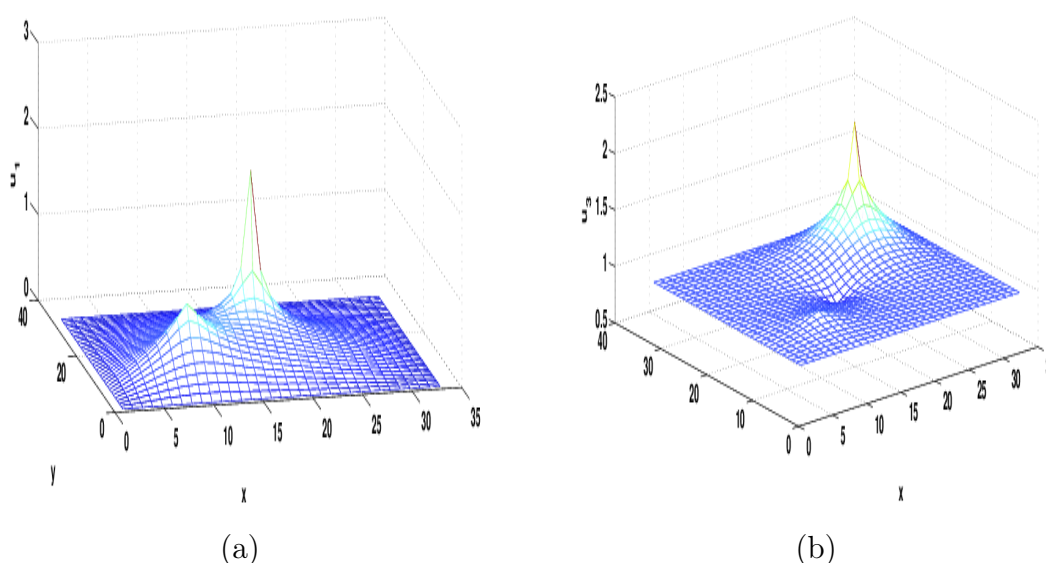
това положителността на численото решение е нарушена за някои стойности на параметрите на мрежата по времето и пространството. На Фиг. 14 численото решение за замърсителя NO_2 (u_2) за Пример 2 когато $\mu = 2\pi/(60T)$ и $M_x = M_y = 8$, $N = 256$, получено със (a) CDS и с (b) CFDS е представено. Схемата CDS запазва положителността на численото решение, докато CFDS не го запазва - близо до ъглите численото решение е отрицателно и няма теоретично обяснение.

Числени експерименти за атмосферен модел на базата на цикъла на Чапман

Използваме опростен, но реалистичен, модел на три компонентна химична система, която моделира химични процеси, протичащи в атмосферата. В тази секция се концентрираме на системата (20)-(22) за $L = 3$ с коефициенти, реакции и източници, които отговарят на модел в атмосферата, базиран на цикъла на Чапман [30, 35]. Докато в един реалистичен атмосферен модел в екологията могат да участват концентрации на много реагиращи вещества [53, 19], опростеният модел тук притежава основните свойства на големите практически модели. Разглеждаме системата:

$$\frac{\partial u_l}{\partial t} - K\Delta u_l + \mathbf{b}_l \cdot \nabla u_l = R_l(x, y, \mathbf{u}) + \xi_l(x, y, t), (x, y, t) \in \Omega \times (0, T]. \quad (40)$$

Компонентите на системата са азотен оксид (NO), азотен диоксид (NO_2) и озон



Фигура 15: Численото решение с параметри на мрежата $M_x = M_y = 32$, $N = 256$ за Пример 2: (а) за $NO - u_1$; (б) за $O_3 - u_3$

(O_3) означени с u_1, u_2, u_3 съответно:

$$R_l(\mathbf{u}) = -r(\mathbf{u}), \quad l = 1, 3, \quad R_2(\mathbf{u}) = r(\mathbf{u}), \quad r(\mathbf{u}) = k_1 u_1 u_3 - k_2 u_2,$$

където k_1, k_2 са скоростите на реакциите.

В изследвания пример се фокусираме върху локални източници, които се описват от Делта - функции. Разглеждаме задачата (40), където функциите ξ_l са точкови източници от вида

$$\xi_l(x, y, t) = f_l(t)\delta(x - \bar{x}_l, y - \bar{y}_l), \quad l = 1, 2, 3.$$

Параметрите са, както следва: $\bar{\Omega} = [0, 500] \times [0, 500]$, $T = 1440$, $b_l = (-0.1, 0)$, за $l = 1, 2, 3$, $K_1 = 1$, $K_2 = K_3 = 5$, $k_1 = 1000$, $k_2 = 2000$, $(\bar{x}_1, \bar{y}_1) = (0.5, 0.5)$, $(\bar{x}_2, \bar{y}_2) = (0.25, 0.25)$, $(\bar{x}_3, \bar{y}_3) = (0.75, 0.75)$, $f_1(t) = 7$, $f_2(t) = 11$, $f_3(t) = 13$. На Фиг. 15 е представено численото решение получено с CFDS с параметри $M_x = M_y = 32$, $N = 256$: (а) за $NO - u_1$; (б) за $O_3 - u_3$. Влиянието на точковите източници се вижда ясно.

Авторска справка

Основните научни приноси на настоящата дисертация са:

1. Изследвани са различни методи Монте Карло и квази-Монте Карло и подходи за генериране на случайни извадки - адаптивен метод Монте Карло, специална решетка, базирана на обобщената редица на Фибоначи, извадка латински хиперкуб. Адаптивният алгоритъм има предимство пред разглежданите алгоритми за функции с изчислителни особености. За гладки функции с по-ниска размерност най-ефективна е решетката с числа на Фибоначи, а при високи размерности - извадката латински хиперкуб.
2. Конструиран е нов почти оптимален алгоритъм Монте Карло за интегрални уравнения, базиран на балансиране на систематичната и стохастичната грешка. Изведени са оценки за избор на броя реализации на случайната величина и броя итерации във веригата на Марков, които са от съществено значение за неговата ефективност и точност.
3. Разработен е нов метод Монте Карло за линейни системи, който е подобрение на метода „случайно блуждаене по уравненията на СЛАУ“. Методът успешно се прилага при решаване на СЛАУ, получени след дискретизация на ЧДУ.
4. Разработени са и са изследвани нови компактни диференчни схеми с четвърти ред на точност по пространството за системи от параболични ЧДУ със свързани нелинейни реакции.

Основните научно-приложни приноси на настоящата дисертация са:

1. Приложени са различни ефективни Монте Карло и квази-Монте Карло алгоритми за ядрото на Вигнер, които са по-ефективни от досега използваните детерминистични алгоритми по отношение на точността и изчислителната сложност.
2. Приложени са различни ефективни Монте Карло и квази-Монте Карло алгоритми за гладки подинтегрални функции при оценки на Европейски опции с важно значение във финансите и за многомерни интегрални в Бейсовската статистика с приложение при машинното обучение.

3. Приложена е схема с шести ред на точност по пространствената променлива за реална физична параболична транспортна система, описваща далечен пренос на замърсители във въздуха, от Датския Ойлеров модел и за атмосферен модел, базиран на цикъла на Чапман.

Благодарности

Издавам своята искрена признателност и благодарност на научния си ръководител проф. д-н Иван Димов за неговите ценни напътствия, професионална компетентност и съдействие при провеждане на настоящите изследвания и при подготовката на дисертацията. Изключително благодаря и за неговата неопценяема морална подкрепа и за проявеното търпение.

Благодаря изключително много на доц. д-р Райна Георгиева за нейните ценни напътствия, готовността ѝ за съдействие във всеки момент и неопценяемата помощ при подготовката на дисертацията.

Благодаря на колегите от Русенския Университет проф. д-р Любен Вълков и доц. д-р Юрий Кандиларов за полезните дискусии и предоставянето на интересни задачи.

Авторът изказва и благодарности на екипа от секция "Паралелни алгоритми" проф. Стефка Фиданова, доц. Пенчо Маринов, доц. Цветан Остромски и доц. Михаил Недялков за приобщаването на автора към колектива и за ценните напътствия и оказаното съдействие. Благодарности дължа и за целия екип на Института по информационни и комуникационни технологии и към неговия Директор за осигурените качествени условия за работа и обучение.

Благодаря на моите колеги Стоян Димитров от ФМИ-СУ и д-р Николай Икономов от ИМИ-БАН за съвместната работа.

Благодаря специално на баща ми и брат ми за тяхната морална и финансова подкрепа, търпение и стимулиращи напътствия.

Благодаря за финансовата подкрепа по проектите: #ФНИ И-02-20, #ДФНП 91-А1, #ДМУ 03/61.

Библиография

- [1] Боянов, Б.: *Лекции по числени методи*. Дарба, София, 1998.
- [2] Караиванова, А.: *Стохастични числени методи и симулации*, София, 2012.
- [3] Черногорова, Т.: Теория на диференчните схеми, София, 2005.
<https://www.fmi.uni-sofia.bg/econtent/tds.pdf>
- [4] Янев, Н., Димитров, Б.: *Вероятности и статистика*. Наука и изкуство, София, 1990.
- [5] Bahvalov, N.: On the Approximate Computation of Multiple Integrals. In *Vestnik Moscow State University, Ser. Mat., Mech.*, volume 4, pages 3–18, 1959.
- [6] Dembo R.S., Eisenstat, S.C., Steihaug, T.: Inexact Newton methods. *SIAM Journal on Numerical Analysis*, **19**(2), 400-408 (1982)
- [7] Dimov, I.: *Monte Carlo Methods for Applied Scientists*, New Jersey, London, Singapore, **World Scientific** (2008), 291 p., ISBN-10 981-02-2329-3.
- [8] Dimov, I., R. Georgieva: Monte Carlo Algorithms for Evaluating Sobol' Sensitivity Indices. *Math. Comput. Simul.* **81** (2010) (3), 506-514.
- [9] Dimov, I.T., Georgieva, R., Todorov, V.: Balancing of Systematic and Stochastic Errors in Monte Carlo Algorithms for Integral Equations, 8th NMA 2014, Borovets, Bulgaria, August 20-24, 2014, LNCS 8962, Springer, 2015, 44–51.
- [10] Dimov, I., Gurov, T.: Monte Carlo Algorithm for Solving Integral Equations with Polynomial Non-linearity. Parallel Implementation, Pliska (Studia Mathematica Bulgarica), Vol. 13 2000, Proceedings of the 9th International Summer School on Probability Theory and Mathematical Statistics, Sozopol, 1997, pp. 117-132.
- [11] Dimov, I.T., Gurov, T.: *A New Iterative Monte Carlo Approach for Inverse Matrix Problem*, **Journal of Computational and Applied Mathematics**, Vol. **92** (1998), pp. 15-35.
- [12] Dimov I., Kandilarov, J., Todorov, V., Vulkov, L.: High-Order Compact Difference Schemes with Richardson Extrapolation for Semilinear Parabolic Systems. IN:

- Applications of Mathematics in Engineering and Economics, American Institute of Physics, 1789, 030002 (2016).
- [13] Dimov, I.T., Maire, S., Sellier, J.M.: A New Walk on Equations Monte Carlo Method for Linear Algebraic Problems, Applied Mathematical Modelling, Volume 39, Issue 15, 2015, Pages 4494–4510.
- [14] Dimov, I., Todorov, V.: Error Analysis of Biased Stochastic Algorithms for the Second Kind Fredholm Integral Equation, Studies in Computational Intelligence 648, 2016, pp. 3-16.
- [15] Doucet, A., Johansen, A.M., Tadic, V.B.: On solving integral equations using Markov chain Monte Carlo methods. *Applied Mathematics and Computations*, **216** 2869–2880, (2010).
- [16] Ermakov, S.M.: *Monte Carlo Methods and Mixed Problems*, **Nauka**, Moscow, (1985).
- [17] Feynman, R.P.: Space-time approach to non-relativistic quantum mechanics, Rev. Mod. Phys. 20 (1948).
- [18] Genz, A.: Testing multidimensional integration routines. Tools, *Methods and Languages for Scientific and Engineering Computation* (1984) 81–94.
- [19] Georgiev, K., Zlatev, Z.: Implementation of sparse matrix algorithms in an advection-diffusion-chemistry model. J. of Comp. Appl. Math., **236** (3), 342-353 (2011)
- [20] Georgieva, R.: PhD Thesis: Computational complexity of Monte Carlo algorithms for multidimensional integrals and integral equations, Sofia, 2003
- [21] Gustafsson, B., Kreiss, H., Oliger, J.: Time Dependent Problems and Difference Methods, Wiley, New York (1995)
- [22] Halton, J.: *Sequential Monte Carlo, Proceedings of the Cambridge Philosophical Society*, **Vol. 58** (1962) pp. 57-78.
- [23] Halton, J.: *Sequential Monte Carlo for linear systems - a practical summary*, **Monte Carlo Methods & Applications**, **14** (2008) pp. 1–27.
- [24] Halton, J.: *Sequential Monte Carlo. University of Wisconsin*, Madison, Mathematics Research Center Technical Summary Report No. 816 (1967) 38 pp.
- [25] Hua, L.K., Wang, Y.: *Applications of Number Theory to Numerical analysis*, 1981
- [26] Jarosz, Wojciech: Efficient Monte Carlo Methods for Light Transport in Scattering Media, PhD dissertation, UCSD, 2008
- [27] Kantorovich, L., Akilov, G.: *Functional Analysis*. Nauka, Moscow, 1977.

- [28] Karaivanova, A., Dimov, I.: Error analysis of an adaptive Monte Carlo method for numerical integration, *Mathematics and Computers in Simulation* **47** (1998) 201–213.
- [29] Karatson, J., Kurics. T.: A preconditioned iterative solution scheme for nonlinear parabolic systems arising in air pollution modeling. *Math. Modell. Anal.* **18** (5), 641-653 (2013)
- [30] Kim, J., Cho, S.: Computation accuracy and efficiency of the time splitting method. *J. Atmosph. Envir.* **31**(15), pp. 2215-2224 (1997).
- [31] Korobov, N.M.: *Number-Theoretical Methods in Approximate Analysis*, Fizmatgiz, Moscow, 1963.
- [32] Lai, Y., Spanier, J.: *Applications of Monte Carlo/Quasi-Monte Carlo Methods in Finance: Option Pricing*, Proceedings of a conference held at the Claremont Graduate Univ, 1998.
- [33] Lin, S.: “Algebraic Methods for Evaluating Integrals in Bayesian Statistics,” Ph.D. dissertation, UC Berkeley, May 2011.
- [34] Lin, S., Sturmfels B., Xu Z.: Marginal Likelihood Integrals for Mixtures of Independence Models, *Journal of Machine Learning Research*, Vol. 10 (2009), pp. 1611-1631.
- [35] Mamonov, A. V., Tsai, Y.-H. R.: Point source identification in non-linear advection-diffusion-reaction systems. *Inverse Problems* **29**(3), pp. 035009 (2012)
- [36] Marchuk, G.I., Shaidurov, V.V.: *Difference Methods and Their Extrapolations* (Springer-Verlag, New York Inc. 1983).
- [37] McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 21(2), 239–245 (1979), doi:10.2307/1268522
- [38] Metropolis, N., Ulam, S.: *The Monte Carlo Method*, **J. of Amer. Statistical Assoc.**, **44**, (1949), No. 247, pp. 335–341.
- [39] Minasny B., McBratney B.: A conditioned Latin hypercube method for sampling in the presence of ancillary information *Journal Computers and Geosciences archive*, Volume 32 Issue 9, November, 2006, Pages 1378-1388.
- [40] Mirzaee F., Samadyar N.: Application of orthonormal Bernstein polynomials to construct a efficient scheme for solving fractional stochastic integro-differential equation, *Optik - International Journal for Light and Electron Optics*, Volume 132, March 2017, Pages 262–273.
- [41] Pao C. V.: *Nonlinear Parabolic and Elliptic Equations*, Springer, US, 1992.

- [42] Richards, S.: Completed Richardson extrapolation in space and time. *Commun. Numer. Meth. Engineering* **13**, 573–582 (1997).
- [43] Sellier, J.M.: A signed particle formulation of non-relativistic quantum mechanics, *Journal of Computational Physics* 297(2015) 254–265.
- [44] Sloan, I.H., Joe, S.: *Lattice Methods for Multiple Integration*, Oxford University Press, Oxford (1994).
- [45] Sloan, I.H., Kachoyan, P.J.: Lattice methods for multiple integration: Theory, error analysis and examples, *SIAM J. Numer. Anal.* 24, (1987) 116-128.
- [46] Sobol, I.M.: *Monte Carlo Numerical Methods*, Nauka, Moscow, 1973, (in Russian).
- [47] Todorov, V., Dimov, I.: Monte Carlo methods for multidimensional integration for European option pricing, *AIP Conf. Proc.* 1773, 100009, (2016).
- [48] Todorov, V.: Computing high dimensional integrals with Monte Carlo methods, *Journal Scientific and Applied Research*, Vol.10, 2016, 11-16, ISSN 1314-6289.
- [49] Wang, Y.M., Guo, B.Y., Wu, W.J.: Fourth-order compact finite difference methods and monotone iterative algorithms for semilinear elliptic boundary value problems, *Computers and Math. with Appl.*, **68**, 1671-1688 (2014)
- [50] Wang, Y., Hickernell, F.J.: *An historical overview of lattice point sets*, (2002).
- [51] Wigner, E.: On the quantum correction for thermodynamic equilibrium, *Phys. Rev.* 40 (1932) 749.
- [52] Zlatev, Z.: *Computer Treatment of Large Air Pollution Models* (Kluwer Academic Publishers, 1995).
- [53] Zlatev, Z., Dimov, I.: *Computational and Numerical Challenges in Environmental Modelling* (Elsevier, Amsterdam, 2006).
- [54] Zlatev, Z., Dimov, I., Farago, I., Georgiev, K., Havasi. A.: Application of Richardson extrapolation for multi-dimensional advection equations, *Comp. Math. Appl.*, **67**, 2279-2293 (2014)
- [55] Website: Matrix market, NOS4: Lanczos with partial reorthogonalization. Finite element approximation to a beam structure, <http://math.nist.gov/MatrixMarket/data/Harwell-Boeing/lanpro/nos4.html>
- [56] MATLAB - MathWorks: <http://www.mathworks.com/>

Abstracts of Dissertations

Number 3, 2017

INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES
BULGARIAN ACADEMY OF SCIENCES

БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО ИНФОРМАЦИОННИ И КОМУНИКАЦИОННИ ТЕХНОЛОГИИ

Брой 3, 2017

Автореферати на дисертации