БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО ИНФОРМАЦИОННИ И
КОМУНИКАЦИОННИ ТЕХНОЛОГИИ

Венелин Любомиров Тодоров

МЕТОДИ МОНТЕ КАРЛО ЗА МНОГОМЕРНИ ИНТЕГРАЛИ И ИНТЕГРАЛНИ УРАВНЕНИЯ И ПРИЛОЖЕНИЯ

ДИСЕРТАЦИЯ

за присъждане на
образователна и научна степен „Доктор“
Докторска програма „Математическо моделиране и
приложение на математиката“
Професионално направление 4.5 „Математика“

Научен ръководител: проф. д.т.н. Иван Димов

София, 2017

В памет на майка ми

Съдържание

Увод

Актуалност на темата

*Методите **М**онте **К**арло (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* [50].

Заслугата за откриването на концепцията, която е в същността на Монте Карло методите, се приписва на полския математик от еврейски произход Станислав Улам, макар че подобни идеи са били наблюдавани и много по-рано, например при класическото решение на задачата за иглата на Буфон. Улам прави тези открития през 1940г., а чак след това, през 1983, разказва как, за да стигне до тях се е опитвал да реши задачата за вероятността за победа при игра на Solitaire: *„След дълго време, прекарано в чисто комбинаторни изчисления, се замислих, дали един по-практичен метод не би могъл да бъде да се изиграят 100 игри и просто да се преброи колко от тях са били успешни..“* [73]. Монте Карло подходът е добил известност и заради фон Нойман и неговото добро познанство с Улам. Фон Нойман доразвива идеята на своя приятел. По негова идея е конструиран и методът на селекцията. Ето какво споменава за техните взаимоотношения Джан-Карло Рота – техен общ приятел и известен учен (бащата на съвременната комбинаторика) в своята книга „Indiscrete thoughts“ [177]: „Стан беше единственият близък приятел, който фон Нойман е имал. Фон Нойман се въртеше постоянно около Стан и успяваше да бъде близко до него за колкото се може повече време... Стан беше по-оригиналният математик от

двамата, въпреки че е постигнал много по-малко в математиката в сравнение с Фон Нойман... Като всеки занимаващ се с абстрактна наука, фон Нойман имал нужда от постоянно потвърждение на идеите си в отговор на своите вътрешни колебания. Мисленето на Стан Улам коригирало много точно несъвършенствата в работата на фон Нойман. От тяхната свободна обмяна на идеи са дошли едни от най-значимите идеи в приложната математика – Монте Карло методите, симулацията на математически модели на компютър, клетъчните автомати и др.“

Може би най-ранното документирано използване на случайна извадка за намиране на приближена стойност на интеграл е това на граф дьо Буфон. През 1777 г. той описва следния експеримент: *Игла с дължина L се хвърля случайно върху хоризонтална равнина с разграфени прави линии на разстояние d, $d > L$. Каква е вероятността P иглата да пресече една от тези линии?* За да определи P, граф дьо Буфон извършва многократно експеримента „хвърляне на игла“. Математическия анализ, който той извършва на задачата, показва, че $P = \frac{2L}{\pi d}$. Няколко години по-късно Лаплас предлага тази формула за пресмятане на числото π, като се използва експерименталната оценка за вероятността. Това е Монте Карло метод за определяне на числото π, скоростта на сходимост, обаче, е бавна. Експериментът може да се намери в [234]. Метод Монте Карло е предложен по-късно и от Hol през 1873 в [91] за изчисляване на числото π чрез хвърляне на игла и придобива особена популярност като забавление. През 20-ти век са направени много открития в теория на вероятностите и теория на случайните процеси, които са използвани в теоретичната основа на методите Монте Карло. Например, Курант, Фридрихс и Леви доказват еквивалентността на поведението на определени случайни процеси и решението на частни диференциални уравнения (ЧДУ). През 1930 г. Енрико Ферми прави числени експерименти, които по-късно се наричат Монте Карло пресмятания, в изследванията си на новооткрития неутрон.

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 Метрополис и Улам, публикувана през 1949 г. и озаглавена „The Monte Carlo Method“ [148], се възприема за първата статия, обвързваща

наименованието „Монте Карло“ с използването на случайни величини [194]. Но както отбелязва Ермаков в [77, 76] сведения за решение на реална задача с метод Монте Карло съществува още в Стария Завет, където цар Соломон строи Божия храм. Терминът „Монте Карло“ е измислен 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“. В края на 40-те и началото на 50-те на миналия век се наблюдава повишен интерес към тематиката. Появяват се много статии, описващи новия метод и как той може да бъде използван за решаване на задачи в областта на статистическата механика, радиационния транспорт, икономическите модели и други. По това време компютрите все още не са особено добре развити. По-късното им развитие позволява извършването на все по-интензивни пресмятания. Последните успехи отговарят на оптимистичните очаквания на основателите на Монте Карло методите от средата на 20-ти век.

Методите Монте Карло се прилагат в широк спектър от направления – физически базирани рендъринговi алгоритми (стохастично проследяване на лъчи; проследяване на частици, които се излъчват от някакъв източник, например светлинен, и изчисляване на енергийния им трансфер; двупосочно трасиране на маршрут: проследяване и съединяване на пътища, които стартират от две посоки – от екрана и от източник), задачи за пренос на неутрони, теория на опашките, трансфер на излъчена топлина, компютърна графика [203], дифузия на неутрони в материал, пренос на частици [79], радиационно екраниране, приложения в полимерните кристали [97] и съвременни приложения в невронни мрежи и автоматично разпознаване на обекти [70], прогнозиране на индустриални индекси, решаване на **ч**астни **д**иференциални **у**равнения (ЧДУ), обработка на образи от сателити, моделиране на популациите в определени региони. Монте Карло методите намират широко приложение и за математическо моде-

лиране и числени симулации в динамика на разредени газове, микрофлуидика, механика на флуидите – виж [20, 154, 199, 200].

Методът Монте Карло е подходящ за задачи, чието решаване е свързано с отчитане на множество случайни фактори, като оценяване и прогнозиране на рискове за безопасност. Стохастичните методи, към които се отнасят методите Монте Карло, са изключително подходящи за описанието на стохастични процеси, но също така може да се окаже полезно и трансформирането на даден детерминистичен проблем в еквивалентен стохастичен проблем [38, 135]. Названието „Монте Карло“ идва от града Монте Карло (Монако), известен в цял свят с многото си казина, а един от най-простите механични прибори за получаване на случайни числа е всъщност рулетката.

При решаването на дадена математическа задача с предварително зададена точност, използвайки различни **а**лгоритми **М**онте **К**арло (АМК), е важно да съществуват числени показатели, въз основа на които да се направи сравнение между алгоритмите.

Прилагането на даден числен алгоритъм е съпроводено и с определяне на мярка за неговата прецизност и времето за реализацията му. Такава мярка за оценяване на числените алгоритми се дефинира с понятията трудоемкост и ефективност. При сравнението на два алгоритъма, по-ефективен е този, чиято трудоемкост е по-малка.

Дефиниция 1. Трудоемкост на метод Монте Карло (вж. [194]) се нарича произведението на вероятната грешка и времето, необходимо за пресмятане на една реализация на случайната величина.

Основната компонента на грешката, която се допуска при прилагането на методи Монте Карло, произтича от вероятностният им характер, т.е. може да се твърди, че грешката е $\varepsilon(p),\ \varepsilon > 0$, с вероятност $p \in (0;1)$.

Ефективността на един алгоритъм е индикатор за реалното време, необходимо за пресмятането на една приближена оценка на неизвестната величина с предварително зададена точност. Ефективността е характеристика, която зависи от дисперсията на оценката D и времето T, необходимо за пресмятане на оценката, [203]:

$$eff = \frac{1}{T\ D}.$$

Двамата учени Stearns и Hartmanis поставят началото на съвременното изучаване на изчислителната сложност през 1965 г. в статията [98].

Дефиниция 2. Изчислителната сложност на алгоритъм от тип Монте Карло се дефинира като

$$C_N = t\ N,$$

където t е осредненото време (или брой операции), необходимо за пресмятане на една реализация на случайната величина, и N е броят на реализациите на случайната величина [50].

Обзор на основните резултати в областта

Стохастичните числени методи, известни под името методи Монте Карло, са числени методи за решаване на математически задачи с помощта на моделиране на случайни величини и/или случайни функции и статистическа оценка на техните характеристики [8]. 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. Порядъкът на сходимост на обикновеното Монте Карло интегриране е пропорционален на $\sqrt{1/N}$, при статистическа извадка с размер N, с което се демонстрира едно съществено предимство на описания подход, тъй като порядъкът не зависи от размерността на подинтегралната функция. Съществуват широк клас задачи, за които методите Монте Карло са единствените възможни числени методи за решаване. Независимо от универсалността на Монте Карло методите, техен сериозен недостатък е слабата им сходимост, основана върху грешка с порядък $O(N^{-1/2})$, когато не се използва допълнителна информация за данните[8]. В резултат от комбинацията на простота на конструкцията, широк обхват на приложимост и бавна сходимост, за Монте Карло пресмятанията се използва твърде много компютърно време. Например според данни от мониторирането на Европейската грид инициатива (www.egi.eu) около 80% от общото процесорно време се използва от грид приложенията с Монте Карло пресмятания. Според данни от Департамента по енергетика на САЩ около 70% от компютърното време на изчислителните машини в департамента се използва за Монте Карло пресмята-

ния [8]. Това е предизвикателство в областта на Монте Карло методите. 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, които водят до ускоряване на сходимостта чрез намаляване на константата пред $O(N^{-1/2})$ в оценката за грешката. Един друг подход за подобряване на сходимостта на Монте Карло методите е да се модифицира използваната случайна редица, т.е., да се замени чрез алтернативна редица, която подобрява степенния показател $-1/2$ във втория множител на израза за грешката. Квази-Монте Карло методите използват квазислучайни редици вместо обичайните псевдослучайни редици.

Предимствата на Монте Карло методите в сравнение с детерминистичните методи са следните [5]:

- Връзката между размерността на задачата и необходимата компютърна памет е линейна, което позволява да се решават задачи с големи размерности.

- Едно важно предимство на метода Монте Карло е възможността за директно пресмятане на неизвестен функционал от решението на интегрални уравнения със същия брой операции, необходими за пресмятане на стойността на решението само в една точка от дефиниционната област [194].

- Алгоритмите Монте Карло са сравнително прости за реализация и дават възможност да се решават задачи със сложни граници.

- Вероятностната природа на тези методи позволява да се намира в известен смисъл обобщено решение на различни задачи, свързани с решаване на уравнения-частни диференциални и интегрални уравнения, а алгоритмите практически не се усложняват при решаване на задачи с особености на граничните условия, негладки граници и сложни десни части.

- Алгоритмите Монте Карло притежават вътрешно присъщ паралелизъм: изчисленията по различни траектории може да се осъществят независимо, но поради наличието на различни типове архитектури, въпросът за избор на конкретна паралелна реализация не е тривиален.

- В хода на изчисленията оценяването на грешката на метода е без съществени допълнителни затруднения.

В резултат на посочените предимства методите Монте Карло се разглеждат като едни от най-важните методи за моделиране във различни области като екологията при определяне на източници на замърсявания, тъй като не е необходимо да се намира решение в цялата област, а се пресмята само интересуващия ни функционал от решението.

Идеята на методите Монте Карло се състои в построяването на случайна величина, чието математическо очакване е равно или близко до решението или функционал от решението на изходната задача. За да се реши приближено изходната задача, достатъчно е да се моделира съответната случайна величина. Когато е построена случайна величина с очакване, равно или близко до решението на дадената задача, казваме че е даден метод Монте Карло за решаването ѝ. Когато е указано как точно може да се моделира случайната величина на компютър с използването на генератор на псевдослучайни числа, се смята че е зададен Монте Карло алгоритъм за решаването ѝ. При такъв подход от известна гледна точка всеки метод Монте Карло може да се разглежда като задача за приближено пресмятане на определен интеграл в подходящо многомерно или безкрайно пространство. В такъв случай размерността на съответното пространство се нарича конструктивна размерност (дефиницията е дадена в [194], стр. 254) на съответния Монте Карло алгоритъм. Идеята на методите квази-Монте Карло се състои в това да се подобри сходимостта на стандартните методи Монте Карло чрез използването на равномерно разпределени редици вместо генераторите на псевдослучайни числа. В такъв случай обаче са налице някои трудности като по-трудната емпирична оценка за грешката и по-трудната теоретична обосновка на тяхното използване. Смята се, че квази-Монте Карло методите и алгоритмите са подходящи, когато размерността на съответния Монте Карло алгоритъм е ниска, и функциите, с които се работи, са сравнително гладки. Съществуват области, в които квази-Монте Карло алгоритмите са доказали своята ефективност, въпреки високата размерност. Така например във финансовата математика често се налага изчисляването на интеграли в 360-мерни пространства (най-често използваните ипотеки са за 30 годишен период

и лихвата може да се изменя всеки месец, така че симулациите изискват да се използват $30 \times 12 = 360$ точки) [1].

Съществуват два класа алгоритми, основаващи се на числените методи Монте Карло – преки и итерационни.

Преки методи Монте Карло

Дадена е случайна величина θ, чието математическо очакване съществува и нека $\mathrm{E}\theta = I$ (по определение $\mathrm{E}\theta$ съществува тогава и само тогава, когато съществува $\mathrm{E}|\theta|$). При оценяването на неизвестната величина I, се използва, че за N независими еднакво разпределени случайни величини с крайно математическо очакване е в сила законът на Хинчин (слаб закон за големите числа), т.е. за всяко $\varepsilon > 0$

$$P\{|\overline{\theta}_N - I| > \varepsilon\} \xrightarrow[N\longrightarrow\infty]{} 0.$$

В горната формула с $\overline{\theta}_N$ е означено средното аритметично на N независими реализации $\theta_1, \ldots, \theta_N$ на случайната величина θ:

$$\overline{\theta}_N = \frac{1}{N}\sum_{i=1}^{N}\theta_i. \tag{1}$$

Преките методи се характеризират само с един тип грешка, наречена вероятностна. Използвайки централната гранична теорема за независими еднакво разпределени случайни величини с математическо очакване I и с крайна дисперсия, за грешката $\overline{\theta}_N - I$ е показано [194], че съществува множество от оценки, зависещи от параметъра c_β:

$$\lim_{N\to\infty} P\left\{\left|\overline{\theta}_N - I\right| < c_\beta\sqrt{\mathrm{D}\theta/N}\right\} = \beta.$$

Тук с β е означен коефициентът на доверие. Следователно задавайки $\beta \in (0;1)$, може да се определи доверителния интервал $(-c_\beta\sqrt{\mathrm{D}\theta/N}; c_\beta\sqrt{\mathrm{D}\theta/N})$ за грешката $\overline{\theta}_N - I$.

Дефиниция 3. Величината $R_N := c_\beta\sqrt{\mathrm{D}\theta/N}$ се нарича вероятностна грешка. При $\beta = 0.5$ (и $c_\beta \approx 0.6745$) съответната грешка r_N се нарича вероятна, т.е.

$$P\left\{\left|\overline{\theta}_N - I\right| < r_N\right\} = \frac{1}{2} = P\left\{\left|\overline{\theta}_N - I\right| \geq r_N\right\}.$$

В хода на пресмятанията може да се конструира и емпирична оценка за дисперсията [194]:

$$\mathrm{D}\theta \approx \frac{1}{N}\sum_{i=1}^{N}(\theta_i)^2 - \left(\frac{1}{N}\sum_{i=1}^{N}\theta_i\right)^2. \tag{2}$$

Забележка 1. Оценката (1) е неизместена и състоятелна оценка за математическото очакване I. Оценката (2) е изместена оценка за дисперсията на θ, т.е.

$$\mathrm{E}\left(\frac{1}{N}\sum_{i=1}^{N}(\theta_i)^2 - (\overline{\theta}_N)^2\right) = \left(1 - \frac{1}{N}\right)\mathrm{D}\theta.$$

Неизместена оценка за дисперсията на θ дава формулата

$$\mathrm{D}\theta \approx \frac{1}{N-1}\sum_{i=1}^{N}(\theta_i)^2 - \frac{1}{N(N-1)}\left(\sum_{i=1}^{N}\theta_i\right)^2.$$

Итерационни методи Монте Карло

За разлика от преките, итерационните алгоритми Монте Карло се характеризират и със систематична грешка. Систематичната грешка зависи от броя на итерациите в използвания итерационен метод, докато вероятностната грешка зависи от стохастичната природа на методите Монте Карло.

Нека X е Банахово пространство от реални функции, функциите $f = f(\mathrm{x}) \in \mathrm{X}$ и $u = u(\mathrm{x}) \in \mathrm{X}$ са дефинирани в $\mathbb{R}^s$ и $\mathcal{K} = \mathcal{K}u$ е линеен оператор, дефиниран върху X. Предполага се, че се търси решението на уравнението

$$u = \mathcal{K}u + f \tag{3}$$

или, по-общо, линеен функционал от решението $J(u) = (\varphi, u),\ \varphi(\mathrm{x}) \in \mathrm{X}$.

Съществуват две възможности за линейния оператор $\mathcal{K}$:

- $\mathcal{K}$ е матрица, а u и f са вектори;
- $\mathcal{K}$ е интегрален оператор, а $u(\mathrm{x})$ и $f(\mathrm{x})$ са функции.

Дефиниция 4. Итерационният процес за уравнението (3):

$$u^{(k)} = \mathcal{K}u^{(k-1)} + f,\ k = 1, 2, \ldots .$$

дефинира следния „отрязан“ ред на Neumann („truncated Neamann series“)

$$u^{(k)} = \sum_{j=0}^{k} \mathcal{K}^{(j)} f = f + \mathcal{K}f + \ldots + \mathcal{K}^{(k-1)} f + \mathcal{K}^{(k)} u^{(0)}, \quad k = 1, 2, \ldots, \tag{4}$$

където $u^{(0)}(\mathrm{x}) \equiv f(\mathrm{x})$ и $\mathcal{K}^{(k)}$ е k-тата итерация на оператора $\mathcal{K}$.

Ако безкрайната сума (4) е сходяща, то нейната граница е решението на уравнението (3). От (4) се вижда, че всяко k-то итерационно приближение на решението u (или съответно на функционал от решението) зависи само от непосредствено предхождащото го $(k-1)$-во приближение. Затова произволна реализация на случайната величина, която се конструира при приближеното пресмятане на неизвестната величина по метод Монте Карло, се получава в следствие на дискретен процес на Марков (дискретна верига на Марков; верига на Марков с дискретен параметър; [6]).

Дефиниция 5. Редицата от случайни величини

$$\{\theta_n, n \geq 0\}$$

се нарича верига на Марков с дискретен параметър, ако за произволни числа $n, i_0, \ldots, i_{n-1}, i, j \geq 0$ е изпълнено равенството

$$P\{\theta_{n+1} = j \,|\, \theta_0 = i_0, \theta_1 = i_1, \ldots, \theta_n = i_n\} = P\{\theta_{n+1} = j \,|\, \theta_n = i\} \tag{5}$$

и съответната условна вероятност съществува. Вероятността в (5) се нарича вероятност за прехода или преходна вероятност.

Веригата на Марков с дискретен параметър $\{\theta_n, n \geq 0\}$ се счита за зададена, ако освен преходната вероятност, са известни и началните вероятности $\pi = \{\pi_i\}_i$, където π_i е вероятността случайната величина θ_0 да приема стойност i, т.е.

$$\pi_i = P(\theta_0 = i).$$

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:

- *систематична* грешка r_k, която зависи от броя на итерациите в процеса:

$$r_k = u^{(k)} - u = \mathcal{K}^{(k)}(u^{(0)} - u),\ k = 1, 2, \ldots;$$

- *вероятностна* грешка r_N, която зависи от броя на реализациите на веригата на Марков:

$$r_N = c_\beta \sqrt{\frac{\mathrm{D}\,\theta}{N}}, \qquad \text{където } \beta \text{ е доверителната вероятност.}$$

С вероятностна грешка се характеризират само изместените оценки, но не и неизместените. Това е грешката r_k на оценката на неизвестната величина след извършване само на k прехода във веригата на Марков. Задачата за балансиране на вероятностната r_k и систематичната грешка r_N е изключително важна, когато се прилагат алгоритми от тип Монте Карло, т.е.

$$r_N = \mathcal{O}\,(r_k).$$

Всъщност тази задача изисква намирането на оптимално съотношение между броя на реализациите N и средната дължина на случайните траектории, на която е посветена втората глава от настоящото изследване. В нея е формулирана теорема, която задава условия за балансираност и е конструиран нов алгоритъм Монте Карло, базиран на тези условия.

С оценките от тип Монте Карло са асоциират следните понятия дадени подолу, които определят съответна характеристика [113]:

- *Грешка.* $Error(I_N) = I - I_N$.

- *Неизместеност.* Оценката I_N на величината I се нарича неизместена, ако

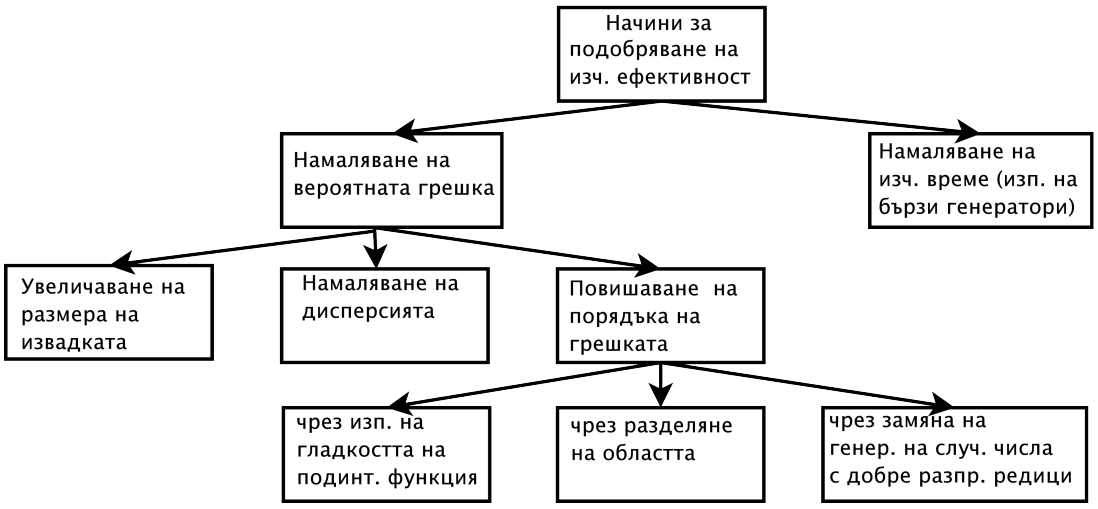
$$\mathrm{E}\,[I_N] = I \qquad \text{за всяко } N.$$

- *Изместване.* Изместването („the bias“) на оценката се дефинира така

$$\mathrm{b}\,[I_N] := I - \mathrm{E}\,[I_N].$$

- *Състоятелност.* Една оценка е състоятелна, ако

$$\lim_{N\to\infty} \mathrm{b}\,[I_N] = 0.$$



Фигура 1: Подходи за конструиране на ефективни Монте Карло алгоритми

Числено интегриране с методи Монте Карло

Монте Карло интегрирането е математическа техника за числено пресмятане на интеграли, която се основава на статистическите свойства на случайните величини и случайните извадки. От дефиницията на вероятната грешка – виж Дефиниция 3, следва че изчислителното време, което методът изисква, е пропорционално на броя на реализациите N и следователно нараства много бързо, ако се изисква по-добра точност.

Следната диаграма на Фиг. 1 показва начините за конструиране на ефективни Монте Карло алгоритми. Целта на настоящото изследване е подобряване на изчислителната ефективност чрез намаляването на вероятната грешка, което може да се постигне по различни начини, както е показано. В настоящата дисертация при изследването на различни Монте Карло и квази-Монте Карло алгоритми, вниманието е насочено към намаляване на дисперсията и подобряване на скоростта на сходимост чрез разделяне на областта и чрез замяна на генератора на случайни числа с добре разпределени редици.

Най-естественият възможен подход за пресмятане на многомерни интеграли, който използва дефиницията на математическото очакване, е обикновеният метод Монте Карло (Plain/Crude Monte Carlo) [50, 54, 194].

Нека е дадена задачата за приближено пресмятане на интеграла

$$I[g] = \int_{\Omega} g(\mathrm{x})p(\mathrm{x})\mathrm{dx}, \quad \Omega \in \mathbb{R}^s. \tag{6}$$

Нека ξ е случайна точка с вероятностна плътност $p(\mathrm{x})$, а $\theta = g(\xi)$ е случайна величина, за която $\mathrm{E}\theta = \int_{\Omega} g(\mathrm{x})p(\mathrm{x})\mathrm{dx}$. Нека случайните точки $\xi_1, \xi_2, \ldots, \xi_N$ са независими реализации на случайната точка ξ с вероятностна плътност $p(\mathrm{x})$ и $\theta_1 = g(\xi_1), \ldots, \theta_N = g(\xi_N)$. Тогава една приближена стойност на $I[g]$ е

$$\overline{\theta}_N = \frac{1}{N}\sum_{i=1}^{N} \theta_i. \tag{7}$$

Изразът (7) дефинира обикновен алгоритъм Монте Карло за приближеното пресмятане на I. Стойността на величината θ_i се пресмята за всяка случайна точка.

Оценките от тип Монте Карло за неизвестни величини се характеризират с някои специфични особености, които маркират предимствата и недостатъците на метода [203] и ще бъдат разгледани по-долу.

- Получените резултати са статистически по своята същност, което означава, че оценката може да бъде (произволно) лоша, но могат да бъдат пресметнати интервалите на доверие, които да посочат колко приближената стойност се различава от истинската стойност. Тези интервали могат да се конструират достатъчно малки, например чрез генериране на повече реализации. Следователно статистическата природа на резултатите се компенсира до голяма степен от предимствата на Монте Карло интегрирането.

- При пресмятането на оценката е необходимо генериране на случайни точки и пресмятане на подинтегралната функция в тези точки. Това са минимални изисквания, определящи метода като лесно приложим към задачи за интегриране, характеризиращи се с изчислителни трудности.

- Методът Монте Карло е робастен. Под робастност на алгоритъм се разбира *устойчивост по отношение на грешки в резултатите, дължащи се на различни отклонения в предположенията.* Основанията за определянето

на метода МК за робастен са, че се изисква генериране на случайни точки само в съответната област на интегриране, а сходимостта се гарантира от Централната гранична теорема [19].

- За да бъде приложен метод Монте Карло за приближено пресмятане на интеграли, не се налага изискване подинтегралната функция да е гладка. При интегрирането на функция с прекъсвания може да се подходи чрез разделяне на областта на интегриране според точките на прекъсванията.

Следователно обикновеният метод Монте Карло за числено интегриране 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. [47, 50, 113]). Методите Монте Карло не се влияят от размерността, т.нар. „проклятие на размерността“ („curse of dimensionality“ – понятие, въведено от Bellman [27]). Монте Карло интегрирането е единственият възможен подход за големи размерности и успешно се прилгага при оценката на опции и пресмятане на ядрото на Вигнер, на което е посветена част от първа глава на дисертацията.

Детерминистичните методи за интегриране от ред (алгебрична степен на точност) r се характеризират с порядък на сходимост N^{-r} в едномерния случай и $N^{-r/s}$ в s-мерния случай, където N е броят на възлите. Следователно за произволен ред r методът Монте Карло ще бъде по-бързо сходящ за достатъчно висока размерност на интеграла. Освен това, квадратурните (кубатурните) формули от по-висок ред изискват „твърде“ гладки подинтегрални функции (например при прилагането на квадратурната формула на Гаус се предполага съществуване на $2N$ непрекъснати производни). Така, дори в случаите на по-малка размерност, за които методът Монте Карло не е измежду най-ефективните, е важно да се получи сравнително груба оценка, но сравнително бързо и лесно. От друга страна, съществуват редица техники за намаляване на дисперсията (отделяне на главна част, симетризация на подинтегралната функция, съществена извадка, разделяне на областта на интегриране на подобласти, „control“ и „antithetic variates“, вж. [22, 50, 97, 113, 194]), чрез които да се конструират алгоритми Монте Карло с повишена скорост на сходимост.

С цел намаляване на изчислителната сложност при решаването на даден математически проблем с предварително зададена точност са разработени редица техники за намаляване на дисперсията (без увеличаване на размера на случайната извадка), което довежда до значителни подобрения на вероятностната грешка, но не повлиява порядъка $(1/\sqrt{N})$ на сходимостта. От друга страна, известни са и методи, които постигат подобрения и в порядъка на сходимост.

Дефиниция 6. ([22]). Нека s и k са цели числа и $s, k \geq 1$. Разглежда се клас $W^k(||f||; U^s)$ от реални функции f, дефинирани в единичния куб $U^s = [0,1]^s$, притежаващи всички частни производни от ред k

$$\frac{\partial^r f(\mathrm{x})}{\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d}}, \quad \alpha_1 + \ldots + \alpha_d = r \leq k,$$

които са непрекъснати за $r < k$ и ограничени (по отношение на sup norm) за $r = k$.

Дефиниция 7. Полунормата $||\cdot||$ върху W^k се дефинира като

$$||f|| = sup\left\{\left|\frac{\partial f(\mathrm{x})}{\partial x_1^{\alpha_1} \ldots \partial x_s^{\alpha_s}}\right|, \alpha_1 + \ldots + \alpha_s = k, x \equiv (x_1, \ldots, x_s) \in U^s\right\}.$$

Съществуват два класа методи за числено интегриране на такива функции върху $[0,1]^s$ – детерминистични и стохастични или методи Монте Карло.

Дефиниция 8. Дадена е следната квадратурна формула:

$$I(f) = \sum_{i=1}^{N} c_i f(x^{(i)}), \tag{8}$$

където $x^{(i)} \equiv (x_1^{(i)}, \ldots, x_s^{(i)}) \in U^s$, $i = 1, \ldots, N$ са възлите и c_i, $i = 1, \ldots, N$ са теглата. Ако $x^{(i)}$ и c_i са реални стойности, формулата (8) дефинира детерминистична квадратурна формула. Ако $x^{(i)}, i = 1, \ldots, N$ са случайни точки, дефинирани в U^s, и c_i са случайни величини, дефинирани в $\mathbb{R}$, формулата (8) дефинира Монте Карло квадратурна формула.

Следните резултати на Бахвалов [23, 24] указват долните граници за грешката при интегриране и в двата случая:

Теорема 1. *(Бахвалов [23, 24]). Съществува константа $c(s,k)$, така че за всяка квадратурна формула $I(f)$, която е напълно детерминистична и използва стойностите на функцията в N точки, съществува функция $f \in W^k$, такава че*

$$\left| \int_{U^s} f(x)\mathrm{dx} - I(f) \right| \geq c(s,k)||f||N^{-\frac{k}{s}}.$$

Теорема 2. *(Бахвалов [23, 24]). Съществува константа $c(s,k)$, такава че за всяка квадратурна формула $I(f)$, в която участват случайни величини и стойностите на функцията в N точки, съществува функция $f \in W^k$, такава че*

$$\left\{ \mathrm{E} \left[\int_{U^s} f(\mathrm{x})\mathrm{dx} - I(f) \right]^2 \right\}^{1/2} \geq c(s,k)||f||N^{-\frac{1}{2}-\frac{k}{s}}.$$

Ясно е, че когато s е достатъчно голямо, методите, използващи случайни величини, имат предимство пред детерминистичните методи. Това дава възможност със стохастични методи да се решават задачи, за които детерминистичните алгоритми досега дават незадоволителни резултати, като оценка на многомерни интеграли, които възникват в Бейсовската статистика и оценка на ядрото на Вигнер в квантовата механика, които обстойно се разглеждат в първа глава.

Дефиниция 9. Алгоритми Монте Карло с порядък на вероятната грешка $\mathcal{O}(N^{-\frac{1}{2}-\psi(s)})$, където $\psi(s) > 0$, а s е размерността на задачата, се наричат суперсходящи алгоритми.

Методите Монте Карло, за които порядъкът на сходимост е $\mathcal{O}(N^{-\frac{1}{2}-\frac{k}{s}})$, са оптимални. В действителност, методи от този тип са суперсходящи (следвайки определението, дадено от Соболь в [194]) с неподобряем порядък на сходимост. Задачата за конструиране на универсален метод с такъв порядък на сходимост за произволна размерност s и стойност на k не е тривиална. В случая на $k = 1$ и $k = 2$ са известни различни методи Монте Карло за числено интегриране, които се характеризират с порядък $\mathcal{O}(N^{-\frac{1}{2}-\frac{k}{s}})$ [167, 185, 196, 204, 206].

В случая, когато освен идеята за разделяне на областта (но без рекурсивен елемент), се използва и информацията за гладкостта на подинтегралната функция, се постига повишаване на порядъка на сходимост. Първите ни известни резултати в тази насока за област $\Omega = [0,1]^s$, вероятностна плътност $p(\mathrm{x}) = 1$

и разбиване на областта на равни части по всички направления са получени от Dupach (вж. [72]).

Теорема 3. *(Dupach, [72]). Нека* $g(\mathrm{x})$ *и всички нейни частни производни от първи ред* $\dfrac{\partial g}{\partial x_k}$ *са непрекъснати в* Ω *и ограничени, т.е. за всяко* $1 \leq k \leq s$: $\left|\dfrac{\partial g}{\partial x_k}\right| \leq L$, *и съществуват константи* $c_1, c_2 > 0$, *за които са изпълнени условията*

$$p_j \leq \frac{c_1}{N}, \qquad за\ j = 1, \ldots, N, \qquad d_j \leq \frac{c_2}{N^{1/s}},$$

където d_j *е диаметърът на подобластта* Ω_j, *т.е.* $d_j = \sup\limits_{\mathrm{x}_1, \mathrm{x}_2 \in \Omega_j} |\mathrm{x}_1 - \mathrm{x}_2|$. *Тогава за дисперсията на оценката* $\overline{\theta}_N^*$ *(4) в случая на* $M = N$ *и* $N_j = 1, j = 1, \ldots, M$ *е в сила:*

$$\mathrm{D}\overline{\theta}_N^* \leq c^2 L^2 N^{-1-2/s}, \quad \text{където} \quad c = s\ c_1\ c_2.$$

Използвайки неравенството на Чебишев и оценката от Теорема 3 (вж. [194]), за грешката $\overline{\theta}_N^* - I[g]$ се получава

$$P\left\{|\overline{\theta}_N^* - I[g]| < \frac{c\ L}{\varepsilon}\ N^{-1/2-1/s}\right\} \geq 1 - \varepsilon^2, \tag{9}$$

където ε е достатъчно малко положително число. Тази идея е използвана [185, 204] за конструирането на методи Монте Карло с повишен порядък на сходимост.

Същият резултат за порядъка на сходимост може да се постигне и при по-слаби условия, а именно - само съответната функция да е непрекъсната, като доказателството е направено от Димов и Тонев в [67]. Съществуват различни подходи от тип Монте Карло за числено интегриране, чийто порядък на сходимост е $\mathcal{O}\left(N^{-\frac{1}{2}-\frac{k}{s}}\right)$. За $k = 1$ и $k = 2$ тези методи сравнително лесно могат да бъдат конструирани, следвайки идеите на Dupach за $k = 1$, описани по-горе (вж. Теорема 3). Но при $k \geq 3$ не е така. Използвайки метода на контролиране на дисперсията върху начасти (във всяка подобласт) интерполационни полиноми, Атанасов и Димов [22] формулират условия за конструиране на метод с оптимален порядък на сходимост за s-мерни функции от класа W^k.

За функции с особености в изчислително отношение като ядрото на Вигнер [220] са подходящи друг тип методи Монте Карло. Това са така наречените адаптивни методи Монте Карло, предложени от Lautrup (вж. стр. 391-392 от книгата

на Davis и Rabinowitz, [47]), които използват предварителна и/или апостериорна информация, получена в процеса на пресмятанията. Повечето то адаптивните алгоритми използват редици от подобласти с намаляващ обем на дадената област, избрани така, че да концентрират пресмятанията на подинтегралната функция в подобластите, в които има особености. Най-общо използват се два вида стратегии за разделяне: локално и глобално разделяне [69]. Основния недостатък на локалното разделяне е, че се изисква локална абсолютна точност, която може да се определи след удовлетворяването на глобалната абсолютна точност. Основното предимство на стратегията за локално разделяне е лесната процедура за обработка на подобластите (не се съхранява информация за неактивните подобласти). Обикновено глобалните адаптивни алгоритми използват повече работна памет, отколкото локалните, и определянето на извадката става по-бавно. Тези алгоритми целят минимизиране на глобалната грешка, колкото е възможно по-бързо, независимо от определеното изискване за точност [28]. Разработени са и подходи за паралелно адаптивно интегриране [36, 37, 82].

При конструирането на основната Монте Карло оценка (7) съответната случайна величина се моделира със зададена вероятностна плътност $p(\mathrm{x})$. В [194] е показано, че най-добрият избор (в смисъл на минимална дисперсия на оценката) за вероятностна плътност е да бъде избрана, пропорционална на подинтегралната функция (метод на съществената извадка). Методът на съществената извадка („importance sampling“) е вероятно най-широко използваният метод Монте Карло с намалена дисперсия (вж. [39]). Идеята на метода се състои в „избирането“ на редки, но важни събития, т.е. малки подобласти (региони от областта на интегриране), в които стойностите на подинтегралната функция „по модул“ са големи. В [52, 117, 118, 86] е представен и изследван метод, наречен метод на разделяне по важност („importance separation“), който обединява идеите на метода на разделяне на областта на подобласти и метода на съществената извадка. Този метод има възможно най-добър (оптимален) порядък на сходимост за даден клас функции. Нека е дадена задачата за приближено пресмятане на интеграла

$$I[g] = \int_{\Omega} g(\mathrm{x})p(\mathrm{x})\mathrm{dx}, \quad \Omega \in \mathbb{R}^s. \tag{10}$$

Да разгледаме задачата 1 и да използваме следните означения:

$$\Omega_0 = \{\mathrm{x} : g(\mathrm{x}) = 0\} \quad \text{и} \quad \Omega_+ = \Omega - \Omega_0.$$

Дефиниция 10. Вероятностната плътност $p(\mathrm{x})$ е допустима за $g(\mathrm{x})$, ако

$$p(\mathrm{x}) \begin{cases} > 0, & \text{за } \mathrm{x} \in \Omega_+, \\ \geq 0, & \text{за } \mathrm{x} \in \Omega_0. \end{cases}$$

Интерес представлява задачата за намирането на допустима плътност $p(\mathrm{x})$, която минимизира дисперсията на случайната величина θ_0, дефинирана така:

$$\theta_0 = \begin{cases} \dfrac{g(\mathrm{x})}{p(\mathrm{x})}, & \text{за } \mathrm{x} \in \Omega_+, \\ 0, & \text{за } \mathrm{x} \in \Omega_0. \end{cases}$$

Определянето на такава плътност означава, че съществува оптимален алгоритъм Монте Карло за описания клас задачи. Следната фундаментална теорема дава отговор на поставения въпрос:

Теорема 4. *(Kahn, [112]). Вероятностната плътност* $c|g(\mathrm{x})|$ *минимизира* $\mathrm{D}\theta_0$ *и минималната дисперсия е:*

$$\mathrm{D}\widehat{\theta_0} = \left[\int_\Omega |g(\mathrm{x})|\mathrm{dx}\right]^2 - I_0^2. \tag{11}$$

Непосредствено следствие от (11) е, че $\mathrm{D}\widehat{\theta_0} = 0$, ако подинтегралната функция $g(\mathrm{x})$ не си сменя знака в Ω.

Практическата реализация на метода е съпроводена от някои трудности. Недостатък на метода е моделирането на плътността, което може да се преодолее чрез прилагане на метода на селекцията (вж. [194]). Owen [170] дава основни насоки за избора на вероятностна плътност, който да осигури необходимата ефективност на метода. Методът на съществената извадка може да доведе до съществено повишаване на дисперсията в някои случаи, например ако функцията на вероятностна плътност намалява към нула по-бързо от квадрата на подинтегралната функция (вж. [170]). Това отношение участва във водещото събираемо в израза за дисперсията. В [99] е представен метод на „defensive importance sampling“. Друг метод, подобен на „defensive importance sampling“, е представен в [213, 212].

Един общ подход за подобряване на сходимостта е използването на силно равномерно разпределени числа, вместо обичайните псевдослучайни числа. Случайните числа, генерирани с компютър, са само псевдослучайни и в този смисъл детерминистични, както и квазислучайните. Разликата е, че докато псевдослучайните числа имат свойства, подобни на реалните случайни числа, квазислучайните нямат такива свойства. Докато псевдослучайните числа са конструирани така, че да симулират поведението на истинските случайни числа, тези силно равномерно разпределени числа, наречени 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, известни още като редици с малък дискрепанс. Методите, използващи такива редици, са известни като методи квази-Монте Карло. Съответно, приближеното интегриране с използване на квазислучайни редици има по-бърза сходимост, с порядък $\mathcal{O}(N^{-1}\log^s N)$. (за сравнение, обикновеният Монте Карло метод има порядък на сходимост $\mathcal{O}(N^{-1/2})$.

Методите квази-Монте Карло „изоставят“ случайността при генерирането на извадка. При теоретично оценяване на грешката на методите квази-Монте Карло обикновено се използва неравенството на Коксма-Хлавка, което свързва грешката на интегриране с дискрепанса на равномерно разпределената редица. Поради това, дискрепансът е най-важната мярка за равномерността на разпределението на числови редици. Известно е, че дискрепанса на псевдослучайните редици е $N^{\frac{1}{2}}\log\log N$ [8]. Въпросът за най-добрия възможен порядък на намаляване на дискрепанса на безкрайни редици е все още открит. Оценяването на дискрепанса на известни редици, както и построяването на редици с малък дискрепанс, има важно теоретично и практическо значение. Освен дискрепанса, като мерки за неравномерност на разпределението се използват още L_2 дискрепанса, диафонията, двоичната диафония, b-ичната диафония [14, 65, 138]. Теорията на равномерно разпределените редици води началото си от работата на Херман и Вайл, където е дадено определение и критерий за равномерност на разпределението на дадена безкрайна редица.

Дефиниция 11. Нека $\sigma = \{x_n\}_{n=1}^{\infty}$ е редица от числа в s-мерния хиперкуб $E^s = [0,1)^s$. Нека за всеки подинтервал $J \subset E^s$ с $A_N(J)$ означаваме броя на точките на σ измежду първите N, които попадат в J. Редицата се нарича равномерно разпределена, ако е изпълнено

$$\lim_{N\to\infty} \frac{A_N(J)}{N} = \mu(J)$$

за всеки подинтервал $J \subset E^s$, където с $\mu(J)$ сме означили Лебеговата мярка на J.

Като характеристика на равномерността на разпределението най-често се използва неговия дискрепанс [1].

Дефиниция 12. За всеки s-мерен интервал $J = \Pi_{i=1}^{s}[c_i, d_i) \subseteq E^s$, означаваме с $A_N(J)$ броя на членовете на редицата $\sigma = x_j$ измежду първите N, такива че $x_j \in J$, нека $\mu(J)$ е Лебеговата мярка или обемът на J. Дискрепансът $D_N(\sigma)$ на редицата σ се дефинира като следния супремум:

$$\sup_{J\subset E^s} \left| \frac{A_N(J)}{N} - \mu(J) \right|.$$

Да означим

$$\lambda_N(x) = \frac{A_N(\Pi_{i=1}^{s}[0, x_i))}{N} - \Pi_{i=1}^{s} x_i.$$

Стар-дискрепансът [1] или още дискрепансът-звезда $D_N^*(\sigma)$ на редицата се получава като супремум на $\lambda_N(x)$ в E^s, а L_2 дискрепансът $D_N^{(2)}(\sigma)$ се дефинира като L_2 нормата на функцията λ_N:

$$D_N^{(2)}(\sigma) = \left(\int_{E^s} |\lambda_N(x)|^2 dx \right)^{\frac{1}{2}}.$$

Различните видове дискрепанс са свързани с неравенството [92]:

$$D_N^{(2)}(\sigma) \leq D_N^*(\sigma) \leq D_N(\sigma) \leq 2^s D_N^*(\sigma).$$

То показва, че порядъкът на мерките е един и същ, затова изборът дали да се използва дискрепанса или дискрепанса-звезда е по-скоро въпрос на удобство [1]. В Първа глава ще използваме дискрепанса за оценка на грешката на редиците от тип решетка с използване на обобщени числа на Фибоначи от съответната размерност. Ще използваме следната опростена дефиниция на дикрепанса-звезда [**?**, 8]:

Дефиниция 13. Нека е зададено точковото множество $X = \{x_i \mid i = 1, 2, \dots N\}$ в $[0,1)^s$ и $N > 1$. Да означим $x_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(s)})$ и $J(v) = [0, v_1) \times [0, v_2) \times \dots \times [0, v_s)$. Тогава дискрепанса-звезда се дефинира като

$$D_N^*(X) := \sup_{0 \le v_j \le 1} \left| \frac{\#\{x_i \in J(v)\}}{N} - \prod_{j=1}^{s} v_j \right|. \tag{12}$$

Познаването на дискрепанса на една редица дава теоретична оценка на грешката при интегриране в класа от функциите с ограничена вариация в смисъла на Харди и Краузе.

Дефиниция 14. Нека функцията f е дефинирана в E^s. Разбиване на куба E^s наричаме набор от s редици

$$\eta_0^j, \eta_1^j, \dots, \eta_{m_j}^j, j = 1, 2, \dots, s,$$

такива че

$$0 = \eta_0^j \le \eta_1^j \le \dots \le \eta_{m_j}^j = 1.$$

Въвеждаме оператора Δ_j с равенството [1]:

$$\Delta_j f(x^{(1)}, \dots, x^{(j-1)}, \eta_i^{(j)}, x^{(j+1)}, \dots, x^{(k)} =$$

$$f(x^{(1)}, \dots, x^{(j-1)}, \eta_{i+1}^j, x^{(j+1)}, \dots, x^{(k)} - f(x^{(1)}, \dots, x^{(j-1)}, \eta_i^j, x^{(j+1)}, \dots, x^{(k)},\ 0 \le m_j.$$

Операторите с различни индекси очевидно комутират. Означаваме $\Delta_{j_1,\dots,j_p} = \Delta_{j_1} \dots \Delta_{j_p}$. Означаваме още

$$V^{(s)}(f) = \sup_P \sum_{i_1=0}^{m_1} \cdots \sum_{i_s=0}^{m_s} |\Delta_{1,\dots,s(\eta_{i_1}^{(1)},\dots,\eta_{i_s}^{(s)})}|,$$

където супремумът се взима по всички разбивания на единичния куб. Ако той е краен, се казва, че функцията има ограничена вариация в смисъла на Витали. Очевидно при $s \le 2$ съществуват функции с нулева вариация, които не са константи. Затова се казва, че функцията f е ограничена в смисъла на Харди и Краузе, ако рестрикцията на f върху всяка от стените на куба с размерност $1, 2, \dots, s-1$ е функция с ограничена вариация в смисъла на Витали.

Известното неравенство на Коксма-Хлавка [101, 102, 123] установява връзката между дискрепанса и грешката при интегриране на функции с ограничена вариация и мотивира изследванията на дискрепанса с цел подобряване на сходимостта на квази-Монте Карло методите.

Теорема 5. *[101] Нека функцията f е дефинирана в E^s и е с ограничена вариация в смисъла на Харди и Краузе, а $\sigma = \{x_n\}_{n=1}^{\infty}$ е равномерно разпределена редица. Изпълнено е*

$$\left| \frac{1}{N} \sum_{i=1}^{N} f(x_i) - \int_{E^s} f(x)dx \right| \leq \sum_{p=1}^{s} \sum_{i_1,\ldots,i_p \subset 1,\ldots,s} V^{(p)}(f_{i_1,\ldots,i_p}) D_N^*(\sigma_{i_1,\ldots,i_p}),$$

където с $V^{(p)}(f_{i_1,\ldots,i_p})$ е означена p-мерната вариация на рестрикцията на f върху съответната стена на единичния куб, в смисъл на Витали, а $\sigma_{i_1,\ldots,i_p}$ е съответната проекция на σ.

За този фундаментален резултат в теорията на квази-Монте Карло методите, Кафлиш [39] прави много просто и елегантно доказателство на неравенството за случая на функции с ограничени s производни, което илюстрира изключителната му важност в теорията на квази-Монте Карло методите [8]. Оттук се вижда важността на конструирането на редици с малък дискрепанс. Предимството на тези редици е малкият дискрепанс, за който има както теоретични, така и практически резултати, и лекотата, с която тези редици се генерират с помощта на компютър.

Оттук следва, че подобрени оценки за сходимостта на квази-Монте Карло методите за интегриране на ограничени функции в смисъл на Харди и Краузе може да се получат като се подобряват оценките за дискрепанса на известните редици или като се конструират редици с по-малка оценка на дискрепанса. На практика се използват хиляди и милиони членове на равномерно разпределените редици. Затова активно се изследва асимптотичното поведение на дискрепанса при N, клонящо към безкрайност, което се обосновава от следната теорема [157].

Теорема 6. *Една редица σ е равномерно разпределена тогава и само тогава, когато дискрепансът ѝ клони към нула при N, клонящо към безкрайност.*

Когато е фиксиран отнапред броя на членовете на редицата, в терминологията на теорията на равномерното разпределение е възприето да се говори за мрежи. Известни са неравенства, свързващи дискрепанса на безкрайната редица и съответните ѝ мрежи [1]. Въпросът за най-добрия възможен порядък за

намаляване на дискрепанса е решен окончателно само при едномерните редици. Смята се, че за всяка размерност той е $\mathcal{O}(N^{-1}\log^s N)$, но при $s \geq 2$ това не е доказано, като дълго време най-добър беше резултата на К.Ф. Рот [178], според който той е $\mathcal{O}(N^{-1}\log^{\frac{s}{2}} N)$. Този резултат е подобрен от Бейкър в [25] през 1999 г. Всички популярни редици, изброени по-долу имат оценки за дискрепанса с порядък $\mathcal{O}(N^{-1}\log^s N)$. Редици с такава оценка се наричат "редици с малък дискрепанс". Много често оценката се получава във вида:

$$D_N(\sigma) \leq c_s \frac{\log^s N}{N} + O\left(\frac{\log^{s-1} N}{N}\right).$$

Затова представлява голям интерес въпросът за получаване на оценки с най-малка възможна стойност за константата c_s, който специално е отбелязан от Niederreiter в [161] като критерий за равномерност на разпределението.

Първоначалното конструиране на квазислучайните редици е свързано с редицата на Ван дер Корпут [211], която е едномерна квазислучайна редица, основана на цифрова инверсия. Редиците на Холтън, въведени в [92], се явяват многомерно обобщение на редиците на Ван дер Корпут. Дълго време се е смятало, че редиците на Ван дер Корпут и на Холтън имат най-малък възможен дискрепанс при $N \to \infty$, което се оказва невярно при $s = 1$. В своята работа Faure [78] предлага обобщение на редиците на Холтън и Ван дер Корпут и показва как могат да се построят редици с по-малък дискрепанс при $s = 1$.

Съществено обобщение на метода на инверсията прави Faure [78] и през 1982 г. конструира редица, която днес носи неговото име. По късно, Niederreiter обобщава известните конструкции на Собол и Faure и създава редици с произволни основи, известни като редиците на Niederreiter (1992) [161]. По-късно Tezuka [207] обобщава редиците на Niederreiter чрез използване на полиномиален аналог на редиците на Halton. Нови редици се свързват с имената на Xing [164, 165], Niedereiter (1993) [163, 159, 160], Owen (1998) [169], Hickernell (1996) [100]. От неравенството на Коксма-Хлавка следва, че тези редици имат грешка, ограничена от $O(N^{-1}(\log N)^{s-1})$ и $O(N^{-1}(\log N)^s)$, съответно, и са по-ефективни от $O(N^{-1/2})$ за големи N.

Редиците на Собол (1967) [193] са може би най-популярните редици в квази-Монте Карло методите. В редица сравнения на качествата на равномерно разпределените редици те се оказват най-добри или измежду най-добрите [1]. Затова

в първа глава те са широко използвани и е направено сравнение с множествата от тип решетки с обобщената редица на Фибоначи за многомерни интеграли с различни размерности. Поради широкото им използване в дисертацията ще се спрем по-подробно на тях и ще дадем няколко дефиниции.

Дефиниция 15. [193] Нека са дадени безкрайните матрици $A_1, A_2, \ldots, A_s$ с нули и единици, като по главния диагонал има само единици, а над него само нули. Съответната редица $\sigma(A_1, A_2, \ldots, A_s) = (x_j)_{j=0}^{\infty}$ се получава като се разложи j в двоична бройна система:

$$j = \sum c_k 2^k$$

и се положи

$$x_j^{(i)} \sum_{r=1}^{s} 2^{-r} \bigoplus k = 1^{s+1} a_{kr} c_{k-1},$$

където с $\bigoplus$ е означена побитовата операция сумиране по модул 2.

Дефиниция 16. [193] Редицата σ се нарича LP_τ редица, ако за всеки каноничен интервал $J \subset E^s$ с обем 2^{-k} и всяко естествено число M има точно 2^τ членове на редицата в J с индекси j в интервала $2^{k+\tau} \leq j < (M+1)2^{k+\tau}$.

Собол е показал как може да се построят LP_τ редици за всяко s. Този параметър τ се явява мярка за качеството на разпределението на съответната LP_τ редица. Нека двоичното представяне на цялото неотрицателно число n е $n = n_1 2^0 + n_2 2^1 + \cdots + n_w 2^{w-1}$. Тогава n-тият елемент от j-тата размерност на редицата на Собол може да се дефинира като

$$x_n^{(j)} = n_1 \nu_1^{(j)} \bigoplus n_2 \nu_2^{(j)} \bigoplus \cdots \bigoplus n_w \nu_w^{(j)},$$

където $\nu_i^{(j)}, i = 1, \ldots, w$ са i-тите направляващи числа за j-тата размерност. Тези направляващи числа се генерират чрез следната рекурентна зависимост

$$\nu_i^{(j)} = a_1 \nu_{i-1}^{(j)} \bigoplus a_2 \nu_{i-2}^{(j)} \bigoplus \ldots a_q \nu_{i-q+1}^{(j)} \bigoplus \nu_{i-q}^{(j)} \bigoplus \nu_{i-q}^{(s)} / 2^q.$$

Тук $i > q$ и a_i са коефициентите на примитивен полином от степен q над крайно поле $F_2 = \{0, 1\}$. Трябва да се отбележи, че за различни размерности в редицата на Собол трябва да се използват различни примитивни полиноми.

В практическата реализация на редицата на Собол широко се използват кодовете на Грей [?]. Ако $G(n)$ е двоичното представяне на n с код на Грей, то двоичното представяне на n и $n+1$ ($G(n)$ и $G(n+1)$) е различават точно с един бит. С използването на кодовете на Грей елементите на една редица на Собол могат да се генерират рекурсивно. Антонов и Салеев [21] първи използват това представяне и доказват, че редицата на Собол асимптотично не се променя [195].

В първа глава се използва специфична реализация на редицата на Собол, която следва идеята на Антонов и Салеев в [21]. Прави се адаптация на алгоритмите INSOBL и GOSOBL, използвани в ACM TOMS Algorithm 647 [81] и ACM TOMS Algorithm 659 [33]. За генериране на редицата на Собол се използва готова matlab функция [231], която генерира вектор, съдържащ числа от квазислучайната редица на Собол от зададена размерност и има максимална размерност 40, тъй като програмната среда не поддържа 64 битови числа.

Ако подинтегралната функция е гладка и периодична, могат да се получат по-добри оценки за грешката, ако се използват точкови множества от тип решетка [32, 161, 189, 190]. В последните десетилетия, Монте Карло методи известни като множества от тип решетка, стават все по-популярни. Много задълбочено изследване на тези алгоритми е направено от Слоан. Този метод е подходящ за гладки и периодични функции в s-мерния хиперкуб. Този метод подробно е разгледан в първа глава и са дадени теореми за анализ на грешката.

Правилата от тип решетка са обобщение на квадратурните правила на Korobov (1959) [125] за интегриране върху s мерния хиперкуб. За първи път са представени от Sloan и Kachoyan (1987) [190].

Дефиниция 17. Казваме, че L е решетка, ако L е безкрайно множество от точки със следните три свойства:

1. Ако x и x' принадлежат на L, тогава принадлежат и $x + x'$ и $x - x'$.
2. L съдържа s линейно независими точки.
3. Съществува сфера с център 0, която не съдържа други точки на решетката, освен 0.

Така многомерна интеграционна решетка е дискретно подмножество на $\mathbb{R}^s$, затворено относно операциите събиране и изваждане, и съдържащо $\mathbb{Z}^s$ като

свое подмножество. Търсеното множество от точки се формира от възлите на интеграционната решетка, попадащи в областта на интегриране [9]. За s-мерния единичен хиперкуб редиците на Коробов се дефинират като:

$$x_k = \left(\left\{ \frac{kz_1}{N} \right\}, \left\{ \frac{kz_2}{N} \right\}, \ldots, \left\{ \frac{kz_s}{N} \right\} \right), \ k = 1, 2, \ldots, N, \tag{13}$$

където N е предварително 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. Доказано е [127], че съществува оптимален избор на генериращия вектор, при който за дискрепанса на редицата на Коробов е в сила:

$$D_N = \mathcal{O}\left(\frac{(\log N)^{\beta(s,\alpha)}}{N^{\alpha}} \right),$$

където β е реално число, независещо от N, s размерността на задачата и $\alpha > 1$. При такъв оптимален избор на z числата x_k играят роля на s-мерни квазислучайни числа. Основната трудност при използването на тези квази-случайни числа е свързана с намирането на оптималния вектор, особено в задачи от голяма размерност. В първа глава е построен квази-Монте Карло метод с генериращ вектор обобщената редица на Фибоначи от съответната размерност.

Квази-Монте Карло методите могат да постигнат асимптотично порядък на сходимост $\mathcal{O}(1/N)$ (вж. [162]), което означава, че при достатъчно голямо N методите квази-Монте Карло трябва да имат предимство пред методите Монте Карло, като се има предвид порядъка на обикновения метод Монте Карло $\mathcal{O}(\sqrt{N})$. Предимствата на методите квази-Монте Карло могат и да се загубят за многомерни математически задачи – например при решаването на (пространственото) хомогенно уравнение на Болцман (вж. [132]) и уравнението на топлопроводността (вж. [155]). Загубата на теоретично очакваната точност при интегриране с квази-Монте Карло метод се дължи главно на две причини. Първата е прекъснатост или липса на гладкост на подинтегралната функция. В дисертацията е реализиран адаптивен Монте Карло алгоритъм, който ефективно отчита такива особености. Оценката на грешката при квази-Монте Карло $\mathcal{O}((\log N)^s N^{-1})$ се определя от неравенството на Коксма-Хлавка, което е в сила за функции с ограничена вариация $V(f)$. Но за да бъде $V(f)$ крайна, е необхо-

димо f да е гладка функция. Монте Карло методите често се използват за пресмятане на интеграли от прекъснати функции. Много методи включват процес на вземане на решение, при което компонент на подинтегралната функция е 0 или 1 при „не“ или „да“. За такива прекъснати подинтегрални функции неравенството на Коксма-Хлавка не е в сила и много числени експерименти показват, че грешката е с порядък $\mathcal{O}(N^{-1/2})$. Друга причина е високата размерност. За големи s дискрепансът на квазислучайната редица се доближава до $O(N^{-1/2})$ при средно големи N, но за достатъчно големи N остава $\mathcal{O}((\log N)^s N^{-1})$. Преходът от $O(N^{-1/2})$ към $\mathcal{O}((\log N)^s N^{-1})$ се осъществява приблизително при $N = e^s$. В първа глава числените експерименти показват, че за високи размерности за гладки подинтегрални функции извадката латински хиперкуб дава по-добри резултати от квази-Монте Карло методите.

В дисертацията квази-Монте Карло методите се използват за оценка на Европейски опции и дават значително по-добри резултати от Монте Карло методите. Пасков установява в емпирични изследвания, че методът квази-Монте Карло превъзхожда в някои отношения Монте Карло за реални задачи от финансова математика [173]. В опитите си да разберат защо Sloan и Wozniakowski въвеждат идеята за теглови пространства („weighted spaces“, [192]). В [219] е показано, че с увеличаване на размерността събираемите от по-нисък ред в ANOVA (**AN**alysis **O**f **VA**riance, анализ на дисперсията) разлагането продължават да имат съществен принос, докато ролята на събираемите от по-висок ред може да бъде пренебрегната. Това внася яснота в многомерните задачи от финансова математика и обяснява защо алгоритмите квази-Монте Карло са ефективни за такъв клас от задачи [8]. За фиксирано s и достатъчно голямо N квази-Монте Карло квадратурата превъзхожда Монте Карло квадратурата. Факторът $N^{-1}(\log N)^s$ е съществено по-голям от $N^{-1/2}$, когато s е голямо, освен ако N не е достатъчно голямо число. Функцията $kN^{-1}(\log N)^s$, където k е константа, има максимум за $N = e^s$. Асимптотичният режим не е достигнат преди тази стойност на N. Това е причината дълго време да се смята, че квази-Монте Карло методите не бива да се използват за интегриране при висока размерност. Това следва и от концепцията "ефективна размерност която дава обяснение защо квази-Монте Карло методите продължават да са ефективни за високи размерности.

В резултат на описаните изследвания може да се направи извод, че предимството на методите Монте Карло или квази-Монте Карло при числото интегриране се определя съществено от подинтегралната функция: гладкост, чувствителност по отношение на колебания в стойностите на променливите, ефективна размерност, изотропия.

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. Числениият метод, построен за решаване на дадена математическа задача, която моделира реален процес от практиката, трябва да притежава редица качества. От една страна, свързаните 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 числото решение, компютърно време и оперативна памет, необходими за пресмятанията. Разглеждаме нелинейни системи ЧДУ от тип адвекция-реакция-дифузия. Реакцията обикновено моделира взаимодействието между различни компоненти, дифузията моделира свободното движение на всеки един от компонентите, а адвекцията моделира насоченото движение на един или повече компоненти в отговор на повишената концентрация на друг компонент. Числените апроксимации на разглежданите в последната глава на настоящата дисертация задачи са получени на базата на метода на диференчните схеми. Получена е компактна диференчна схема за едно, две, три и система от 10 нелинейни ЧДУ, описваща преноса на различни замърсители във въздуха.

Диференчните схеми са често използвани за решаването на частни диференциални уравнения, поради лесната им реализация и изчислителната им ефективност. При решаването на елиптични ЧДУ, най-често използваните диференчни схеми са от втори ред, въпреки че в повечето случаи диференчни схеми от първи ред трябва да се използват за избягване на осцилации. 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 използваме диференциалните уравнения, така че да изразим производните от по-висок ред, които участват в локалната грешка на апроксимацията, чрез производните от по-нисък ред.

За да се намали изчислителното време за големите математически модели, се използва дискретизация с висок ред на точност. Друг важен фактор, който влияе на изчислителната ефективност, е да се решат ефективно нелинейните и линейните системи от алгебрични уравнения. Методи с висок ред на точност обикновено генерират алгебрични системи с много по-малък размер в сравнение с с методите с по-нисък ред на точност. Това е основната причина да има много голям интерес в развитието на методи с висок ред на точност за решаването на ЧДУ, което поражда все по-нарастващ интерес в конструирането на компактни диференчни схеми (схеми с минимален шаблон) с висок ред на точност [43, 130, 175, 197, 202, 210, 216].

Друг начин за повишаване на реда на точност на диференчните схеми е да се използва екстраполация по Ричардсон [141]. Това е ефективна изчислителна техника, която изисква минимално усилие за повишаване на точността. Въпреки, че тази техника е добре позната, тя не се счита за достатъчно ефективна в сравнение с директното използване на компактни диференчни схеми от четвърти ред, виж [228]. По-долу ще бъдат представени някои основни понятия от теорията на диференчните схеми.

За да се построи една диференчна схема, апроксимираща дадена диференциална задача, е необходимо [2, 11, 12, 13, 15]:

1. Да се направи дискретизация на областта, т.е. да се замени областта на непрекъснато изменение на аргумента с област на дискретно изменение.

2. Да се направи дискретизация на задачата, т.е. да се заменят основните диференциални уравнения и допълнителните условия с диференчни ана-

лози.

Към основните понятия в теорията на диференчните схеми се отнасят:

- ***Мрежа*** - това е множеството от точки, в което се търси приближено решение на задачата. Мрежата може да бъде както равномерна, така и неравномерна.
- ***Възли*** на мрежата се наричат точките на мрежата.
- ***Мрежова функция*** се нарича всяка функция, дефинирана във възлите на мрежата.
- ***Шаблон*** на даден възел x от мрежата, който ще означаваме с Ш, се нарича множеството от възела x и съседните му възли, в които се използват стойностите на мрежовата функция за апроксимиране на производните във възела x.
- ***Околност*** на възела x: $\text{Ш}'(x) = \text{Ш}(x) \backslash \{x\}$ - това е множеството от точките, включени в шаблона на възела x без самия възел x.

Други важни понятия от теорията на диференчните схеми (ДС) са: ***апроксимация, устойчивост, сходимост*** и ***точност***.

Нека е дадена линейната диференциална задача:

$$Lu = f(x), \qquad x \in G, \tag{14}$$

$$lu = \mu(x), \qquad x \in \Gamma, \qquad \overline{G} = G \cup \Gamma, \tag{15}$$

където $u(x)$, $f(x)$ и $\mu(x)$ са елементи на нормираното пространство H с норма $\|.\|$, L и l са линейни диференциални оператори, действащи от H в H. Уравнение (14) обобщава основните диференциални уравнения, а (15) - допълнителните условия (начални и/или гранични) на разглежданата задача. Ще предполагаме, че решението на задача (14), (15) съществува и е единствено.

На задача (14), (15) съпоставяме следната диференчна задача (схема):

$$L_h y_h = \varphi_h, \qquad x \in \omega_h, \tag{16}$$

$$l_h y_h = \widetilde{\mu}_h, \qquad x \in \gamma_h, \qquad \overline{\omega} = \omega_h \cup \gamma_h, \tag{17}$$

където $\overline{\omega}_h$ е мрежата от точки, въведена в областта $\overline{G}$, функциите y_h, φ_h, $\widetilde{\mu}_h$ са мрежови функции, принадлежащи на нормираното пространство H_h с норма $\|.\|_h$, а L_h и l_h са линейни диференчни оператори $H_h \longrightarrow H_h$. Решението на задача (16), (17) зависи от стъпката на мрежата h. Ще отбележим, че ако мрежата е неравномерна, то под h трябва да се разбира векторът $h = (h_1, h_2, ..., h_p)$ с компоненти - разстоянията между всеки две последователни точки от мрежата. Ако областта G е многомерна и $x = (x_1, x_2, ..., x_p)$, то $h = (h_1, h_2, ..., h_p)$, ако мрежата е равномерна по всеки от аргументите $x_1, x_2, ..., x_p$.

Основен интерес в теорията на диференчните схеми представлява разликата между функциите u и y_h. Но те са функции от различни пространства. За тяхното сравняване има две възможности:

а) мрежовата функция y_h, дефинирана върху мрежата $\overline{\omega}_h$, да се додефинира (например чрез линейна интерполация) в цялата област $\overline{G}$. Тогава разликата между непрекъснатата функция $\widetilde{y}$, съответстваща на y_h, и u е елемент на H, т.е. $\widetilde{y} - u \in H$ и близостта им се характеризира с $\|\widetilde{y} - u\|$;

б) пространството H се изобразява в пространството H_h с помощта на някакъв проектор P, т.е. на функцията $u \in H$ се съпоставя $u_h \in H_h$ така, че $u_h = Pu$ (ако $u(x)$ е непрекъсната, полагаме $u_h(x) = u(x), x \in \overline{\omega}_h$; ако $u(x)$ е интегруема, полагаме $u_h(x) = \frac{1}{2h}\int_{x-h}^{x+h} u(x)dx$). Тогава $y_h - u_h \in H_h$ и близостта им се оценява с $\|y_h - u_h\|_h$.

Считаме, че H и H_h са нормирани пространства. Ако в H е въведена някаква норма (най-често ще използваме C и L_2 норми), то в H_h се въвежда мрежов аналог на нормата в H, като при това е естествено да се иска $\|\cdot\|_h$ да апроксимира $\|\cdot\|$ в следния смисъл:

$$\lim_{\|h\|\to 0} \|u_h\|_h = \|u\|, \qquad u \in H.$$

Това условие се нарича условие за съгласуваност на нормите в H и H_h. Тук под $\|h\|$ разбираме аналог на C-нормата, т.е. $\|h\| = \max_i h_i$, или $\|h\| = (\sum h_i^2)^{1/2}$, което е аналог на L_2-нормата. Ако в H е въведена C-норма, т.е.

$$\|u\|_C = \max_{x\in\overline{G}} |u(x)|,$$

аналог на C-нормата в H_h е

$$\|y_h\|_C = \max_{x\in\overline{\omega}_h} |y_h(x)|.$$

Аналогично, мрежов аналог на L_2-нормата

$$\|u\|_{L_2} = \left(\int_{\overline{G}} u^2(x)dx \right)^{1/2}$$

е например

$$\|y_h\|_{L_2} = \left(\sum_{\overline{\omega}_h} h y_h^2(x) \right)^{1/2}.$$

Дефиниция 18. Под грешка на диференчната схема (16), (17) се разбира мрежовата функция

$$z_h = y_h - u_h. \tag{18}$$

Навсякъде при сравняване на точното и приближеното решение ще използваме втория подход б), т.е. ще оценяваме нормата на $y_h - u_h$, като най-често ще пропускаме индекса h.

Замествайки y_h от (18) в задача (16), (17), получаваме:

$$\begin{aligned} L_h z_h &= \varphi_h - L_h u_h, \qquad x \in \omega_h, \\ l_h z_h &= \widetilde{\mu}_h - l_h u_h, \qquad x \in \gamma_h, \qquad \overline{\overline{\omega}} = \omega_h \cup \gamma_h. \end{aligned}$$

Дефиниция 19. Под грешка на апроксимация на диференциалното уравнение (14) с диференчното уравнение (16), пресметната за точното решение на диференциалното уравнение, се разбира мрежовата функция

$$\psi_h = \varphi_h - L_h u_h.$$

Аналогично се дефинира и грешка на апроксимация на граничното условие (15) с диференчното условие (17), а именно

$$\nu_h = \widetilde{\mu}_h - l_h u_h.$$

Ако се разглеждат тези грешки в една конкретна точка от мрежата, се говори за *локална грешка на апроксимацията.*

Да представим грешката ψ_h по друг начин. Имаме

$$\begin{aligned} \psi_h = \varphi_h - L_h u_h &= \varphi_h - L_h u_h + (Lu - f)_h \\ &= \varphi_h - f_h + (Lu)_h - L_h u_h = \psi_h^{(1)} + \psi_h^{(2)}, \end{aligned}$$

където изразът $\psi_h^{(1)} = \varphi_h - f_h$ е грешката на апроксимация на дясната част на уравнение (14), а изразът $\psi_h^{(2)} = (Lu)_h - L_h u_h$ е грешката на апроксимация на диференциалния L оператор с диференчния оператор L_h.

Ще отбележим, че при дефинирането на понятието грешка на апроксимацията навсякъде използваме точното решение на диференциалната задача (14), (15). Грешката на апроксимация понякога се дефинира за произволна функция v, $v \in V$, където V е класът на достатъчно гладките функции.

Дефиниция 20. Ако $\|\psi_h^{(2)}\|_h \to 0$ при $\|h\| \to 0$, се казва, че диференчният оператор L_h апроксимира диференциалния оператор L. Ако $\|\psi_h^{(2)}\|_h = O(\|h\|^m)$, се казва, че апроксимацията на L с L_h е от m-ти ред ($m > 0$). Казва се, че диференчната схема (16), (17) има *n-ти ред на апроксимация*, ако $\|\psi_h^{(2)}\|_h = O(\|h\|^n)$ и $\|\nu_h^{(2)}\|_h = O(\|h\|^n)$.

Дефиниция 21. Казва се, че диференчната схема (16), (17) е *сходяща* (т.е. решението на диференчната схема (16), (17) е сходящо към решението на диференциалната задача (14), (15)), ако $\|z_h\|_h = \|y_h - u_h\|_h \to 0$ при $\|h\| \to 0$. Ако $\|z_h\| = O(\|h\|^n)$, казва се, че диференчната схема (16), (17) е сходяща със скорост $O(\|h\|^n)$, или че има *n-ти ред на точност*.

Дефиниция 22. Казва се, че диференчната схема (16), (17) е коректно поставена, ако за всички достатъчно малки h, $\|h\| \le h_0$:

- решението y_h съществува и е единствено за всички входни данни φ_h и $\widetilde{\mu}_h$ от някакъв допустим клас;

- решението y_h непрекъснато зависи от φ_h и $\widetilde{\mu}_h$, като тази зависимост е равномерна относно h. Последното означава, че съществуват константите $M > 0$ и $N > 0$, независещи от h, такива че за всички достатъчно малки h, $\|h\| \le h_0$, е в сила неравенството

$$\|\overline{y}_h - y_h\|_h \le M\|\overline{\varphi}_h - \varphi_h\|_h + N\|\overline{\mu}_h - \mu_h\|_h,$$

  където y_h е решение на задача (16), (17), а $\overline{y}_h$ - на задача (16), (17), но с входни данни $\overline{\varphi}$ и $\overline{\mu}$ вместо φ_h и $\widetilde{\mu}$. Това свойство се нарича *устойчивост* на диференчната схема по входни данни.

Ще отбележим, че ако y_h е решение на задача (16), (17), то y_h, φ_h и $\widetilde{\mu}_h$ зависят от h като параметър. Изменяйки h, получаваме редица от входни данни $\{\varphi_h, \widetilde{\mu}_h\}$ и редица от решения $\{y_h\}$, и следователно не една диференчна задача, а семейство задачи, зависещи от параметъра h. Понятието коректност се въвежда за семейство диференчни схеми при $\|h\| \to 0$.

Връзка между понятията апроксимация, устойчивост и сходимост дава следната

Теорема 7. *(Лакс) Ако задачата (16), (17) е коректно поставена и апроксимира задача (14), (15), то тя е сходяща, като при това скоростта на сходимост (редът на точност) се определя от реда на апроксимация.*

Важна задача на дисертацията е да се разработят нови диференчни схеми с висок ред на точност и които имат шаблон с минимален брой възли за приложни задачи в екологията и други области. Напоследък нараства интересът към компактните диференчни схеми с висок ред на точност 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 интеграли и 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-Монте Карло метод от тип решетки с генериращ вектор обобщената редицата на Фибоначи, Монте Карло метод базиран на извадката латински хиперкуб и адаптивен алгоритъм Монте Карло. Да се изследва кой от алгоритмите е най-ефективен в зависимост от размерността на интеграла и гладкостта на подинтегралната функция.

3. Да се приложат разработените алгоритми за многомерни интеграли с приложен характер във финансите за оценка на европейски опции и в Бейсовската статистика. Да се получат оценки за ядрото на Вигнер в квантовата механика и да се изследва кой от алгоритмите е най-ефективен за решаване на проблема на Ричард Файнман.

4. Да се конструира нов метод Монте Карло за линейни системи на базата на метода Монте Карло за линейни системи „случайно блуждаене по уравненията“ и да се направи сравнение между двата алгоритъма и рафинирания алгоритъм Монте Карло.

5. Да се разработи нова компактна схема с четвърти ред на точност за системи от слабо свързани частни диференциални параболични системи с нелинейни химични реакции с приложение при опазване на околната среда. 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 в едномерния и двумерния случай.

Методология на изследването

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

- функционален анализ [114] – функционални редове (сходимост, ред на Neumann, грешка от „отрязването“ на ред („truncation error“), ред на Taylor, оценка на остатъчния член) [180].

- теория на вероятностите [19] – случайни величини, гранични теореми, изместени и неизместени оценки, вериги на Марков;

- числен анализ [3, 10] – апроксимация, квадратурни формули;
- числени методи за диференциални уравнения [11, 12, 13] и теория на диференчните схеми [15].

Програмните кодове и числените експерименти са написани на MATLAB [237].

Структура на съдържанието

Настоящата дисертация се състои от увод, три глави, заключение и списък на цитираната литература.

Основните дефиниции и твърдения от теорията на методите Монте Карло, свързани с численото пресмятане на интеграли и решаване на интегрални уравнения, са дадени в увода. Описана е същността на интегриране от тип Монте Карло, както и специфичните характеристики на приближената оценка на неизвестна величина, получена с техника Монте Карло. Представен е най-естественият възможен подход за числено интегриране от тип Монте Карло, основаващ се на стохастичната същност на подхода – обикновен метод Монте Карло. Направен е кратък коментар относно основните предимства и недостатъци на методите от тип Монте Карло, както и сравнение по отношение на ефективността (имайки предвид скоростта на сходимост и изчислително време) с детерминистични подходи за числено интегриране. Дадени са резултатите на Бахвалов, които указват долните граници за грешката при интегриране и в случая на детерминистични подходи, и на стохастични. Описана е същността на подходи за конструиране на оценки с намалена вероятностна грешка – съответно с намалена дисперсия (метод на съществената извадка и негови модификации, адаптивни процедури) или с повишен порядък на сходимост (използване на квазислучайни числа). Дадени са дефиниции на квазислучайна редица и нейния дискрепанс. Разгледан е въпроса за различните мерки на дискрепанс и тяхната еквивалентност. Описана е квазислучайната редица на Собол, която е широко използвана в първа глава. Дефинирани са и точковите множества от тип решетка и редиците на Коробов. Разгледани са и основни понятия от теорията на диференчните схеми като апроксимация, устойчивост, сходимост и

точност. Дадено е описание и на различни подходи за повишаване на реда на сходимост на диференчните схеми и е въведено понятието компактна диференчна схема.

Основните научни и научно-приложни приноси са представени в следващите три глави.

Първа глава е посветена на методите Монте Карло и квази-Монте Карло за многомерно интегриране. Направено е описание на метода латински хиперкуб, който е използван широко в числените експерименти. 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 Genz [84], които доказват очакваният ефект на намаляване на дисперсията. Конструиран е и квази-Монте Карло метод от тип решетки с генериращ вектор обобщената редица на Фибоначи. Направено е сравнение с квазислучайните редици на Собол за гладки функции при оценка на Европейски опции във финансите. Получено е представяне на стойността на опцията с многомерен интеграл с помощта на математическото очакване на случайни величини, които се използват за оценка на опцията. Най-важното приложение е при оценка на ядрото на Вигнер в квантовата механика. Получения 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 при по-високи размерности. Всички представени методи от тип Монте Карло/квази-Монте Карло дават по добри резултати от досега използваните методи за оценка на ядрото като адаптивния алгоритъм се оказва най-ефективен, поради вида на ядрото.

Резултатите, представени в тази глава, са публикувани в [208, 209].

Втора глава е посветена на разработването на нови алгоритми Монте Карло за интегрални уравнения и линейни системи. Описан е нов 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. Разгледан е проблема за намиране на оптимално съотношение между броя на реализациите на случайната величина и средния брой стъпки в една случайна траектория във веригата на Марков с цел решаване на задачата с предварително зададена точност. Изведена е теорема за балансираност и следствия за оптималното съотношение между двата параметъра в алгоритъма. Най-напред са получени неравенства за систематичната и вероятната грешка. С помощта на тези оценки се извеждат условията за балансираност. Получена е теорема и следствия за долни оценки за броя на реализациите и броя на итерациите в разглеждания алгоритъм, които са от важно значение за качеството на алгоритъма. Разработеният алгоритъм е приложен върху няколко примера, които имат приложен характер-популационен модел в биологията и при обучение на невронните мрежи. Разгледан е и въпроса дали алгоритъмът е приложим за нелинейно интегрално уравнение с приложение във физиката.

Изследван е нов метод Монте Карло за линейни системи. Подробно е описан един от най-бързите и точни методи за линейни системи "случайно блуждаене по уравненията" ("walk on equations", WE) и на базата на този метод се конструира нов метод Монте Карло за линейни системи, който е по-точен и по-бърз в сравнение с първоначалния алгоритъм. Направено е сравнение и с рафинираният метод Монте Карло за линейни системи. Разгледани са различни примери на матрици от висока размерност. Методът може ефективно да се конкурира по бързина с вградените градиентни алгоритми в MATLAB и може да се приложи и при решаване на линейните системи, получени след дискретизация на частните диференциални уравнения, описани в следващата глава.

Резултатите, представени в тази глава, са публикувани в [57, 66] и са цитирани в статията с IF [153].

В трета глава се конструират нови компактни диференчни схеми за системи от параболични частни диференциални уравнения. Направено е приложение в едномерния и двумерния случай за системи от нелинейни параболични частни диференциални уравнения, както и при модел на далечен пренос на замърсители във въздуха на базата на Unified Danish Eulerian Model, UNI-DEM [227]. Разглежда се модел, в който участват 10 химични замърсителя. Разглежда се случая на точно аналитично решение и когато няма точно решение, което съ-

ответства на реалната задача. Разглежда се правоъгълна област с размери от 500 км за едно денонощие. Разглежда се 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, получен с екстраполация по Ричардсон. Получена е и схема с шести ред на точност по пространството като върху компактната схема е приложена екстраполация по Ричардсон. За решаване на част от системите от линейни алгебрични уравнения, получени след дискретизация е използван метода Монте Карло за линейни системи и е направено сравнение с вградения алгоритъм в Матлаб bicgstabl [236]. За първи път за моделните задачи се прилага диференчна схема с шести ред на точност.

Резултатите, представени в тази глава, са публикувани в [60].

Глава 1

Алгоритми Монте Карло за многомерни интеграли

Многомерните числени квадратури са от изключителна важност в много приложни области – от атомната физика до финансите. Подходът Монте Карло е единственият метод, който може да се използва на практика за редица задачи с висока (голяма) размерност. Основна част от усилията за подобряване на методите Монте Карло се съсредоточава в конструирането на методи с намалена дисперсия, които ускоряват пресмятанията или в конструиране на квазислучайни редици с нисък дискрепанс.

Понякога методите Монте Карло са единственият възможен метод за многомерни интеграли, тъй като неговата сходимост не зависи от размерността на задачата. Това е така, тъй като обикновеният метод Монте Карло се характеризира с порядък на сходимост $O(N^{-1/2})$, който не зависи от размерността на интеграла. За конструирането на методи за числено интегриране съществуват два основни подхода. Първият подход използва априорна информация за подинтегралната функция (например непрекъснатост на функцията и на производните от съответен ред, ограничени производни), а вторият - апостериорна информация (например оценка за дисперсията на съответната случайна величина.)

Нека е дадена задачата за приближено пресмятане на интеграла

$$I[g] = \int_{\Omega} g(\mathrm{x})p(\mathrm{x})\mathrm{dx}, \quad \Omega \in \mathbb{R}^s. \tag{1}$$

За малки стойности на размерността на задачата s, числените методи за интегриране, като правилото на Симпсън и правило на трапеците (виж Davis и Rabinowitz [48]), може да се използват за апроксимиране на интеграла (1). Тези методи обаче страдат от така нареченото „проклятие на размерността“ и стават непрактични, когато s надхвърли 3.

Методите Монте Карло имат съществено предимство пред детерминистичните методи когато размерността на задачата е голяма, което е показано подолу. Да разгледаме многомерен интеграл [50], за който $s = 30$. За да приложим детерминистичен метод, генерираме мрежа в s-мерния хиперкуб и вземаме сумата (със съответните коефициенти според избраната формула) на функционалните стойности в точките на мрежата. Нека мрежата е избрана с 10 възела по всяка една от координатните оси в s мерния хиперкуб $G = [0, 1]^s$.

В този случай трябва да пресметнем 10^{30} стойности на функцията $f(x)$. Да предположим, че време от $10^{-7}s$ е необходимо за пресмятане на една стойност на функцията. Следователно време от порядъка на 10^{23}s ще е необходимо за пресмятане на интеграла (отбелязваме, че 1 година $= 31536 \times 10^3$s, и е изминало повече от 9×10^{10}s от раждането на Питагор). Така, че да пресметнем 30 мерния интеграл с детерминистична формула ще ни трябват $10^{23}/31556926 = 3 \times 10^{15}$ години.

Да разгледаме обикновен Монте Карло алгоритъм за този проблем с вероятна грешка от същия ред. Алгоритъмът се състои в генерирането на N псевдо случайни числа (точки) (PRV) в G; в пресмятането на $f(x)$ в тези точки; и осредняване на получените стойности на функцията. За всяка равномерно разпределена точка в G трябва да генерираме 30 случайни числа, равномерно разпределени в $[0, 1]$.

Вероятната грешка е:

$$\epsilon \leq 0.6745||f||_{L_2}\frac{1}{\sqrt{N}}. \tag{2}$$

От горното неравенство и като използваме, че $\epsilon \leq cMh^3$ получаваме

$$N \approx (\frac{0.6745||f||_{L_2}}{cM})^2 \times h^{-6}. \tag{3}$$

Да предположим, че изразът пред h^{-6} е от ред 1.

В нашия пример $h = 0, 1$, следователно $N \approx 10^6$; и значи трябва да генерираме $30 \times 10^6 = 3 \times 10^7$ PRV. Обикновено две операции са нужни да се генерира

една PRV. Да предположим, че времето, необходимо да се генерира една PRV, е същото като времето за изчисление на стойността на функцията в една точка от областта. Следователно, за да решим проблема със същата точност, време от

$$3 \times 10^7 \times 2 \times 10^7 \approx 6s$$

ще бъде необходимо. Предимството на Монте Карло метода пред детерминистичния за разрешаване на този проблем е очевидно.

За генерирането на случайните числа при обикновения метод Монте Карло се използва функцията rand() в MATLAB, която използва генератора Mersenne Twister (MT) [143, 232], разработен от Makoto Matsumoto и Takuji Nishimura през 1996-1997 г. Този тип генератори се отличава със следните важни характеристики:

- при конструирането му са отчетени недостатъците на съществуващи генератори;
- голям период $(2^{19937}-1)$ и висока размерност, за която са в сила свойствата на равномерно разпределение при фиксирана битова точност (623-орки са равномерно разпределени в 623-мерния единичен куб при 32-битова точност);
- бързо генериране;
- ефективно използване на паметта.

Максималната дължина на редица от псевдослучайни числа преди появата на повторения се определя от „размера“ в битове на инициализиращото число („seed“). Генераторът Mersenne Twister има период $2^{19937} - 1 \sim 10^{6001}$, което е много повече, сравнено с период $\sim 10^8$ за най-добрите варианти на линейни конгруентни генератори на случайни числа. От друга страна, проведени числени експерименти показват (вж. [142]), че за псевдослучайните числа е характерно струпване в отделни подобласти, а в други подобласти не попада нито една точка. Именно това е един от недостатъците на генераторите на псевдослучайни числа, който генераторът Mersenne Twister преодолява.

1.1 Извадка латински хиперкуб

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.

Идеята на метода на „слоистата“ извадка е разделяне на областта на различни, неприпокриващи се подобласти, като във всяка подобласт (наречена „stratum“) попада предварително зададен набор от реализации, например една реализация за подобласт.

По метода на „слоистите“ извадки („stratified sampling“; вж. [97, 179, 194]) областта на интегриране се разбива на M непресичащи се подобласти Ω_j, $j = 1, \ldots, M$, като

$$p_j = \int_{\Omega_j} p(\mathrm{x})\mathrm{dx}, \quad I_j[g] = \int_{\Omega_j} g(\mathrm{x})p(\mathrm{x})\mathrm{dx}.$$

Следователно $\sum_{j=1}^{M} p_j = 1$ и $\sum_{j=1}^{M} I_j[g] = I[g]$.

Нека $\xi^j \in \Omega_j$ е случайна точка с вероятностна плътност $p(\mathrm{x})/p_j$. Така, прилагайки обикновения метод Монте Карло, е в сила следната зависимост: $I_j[g] = p_j\ \mathrm{E}(\theta^j)$, където $\theta^j = g(\xi^j)$. Следователно за неизвестната стойност на интеграла $I[g]$, се получава следната неизместена оценка:

$$\overline{\theta}^*_N = \sum_{j=1}^{M} \frac{p_j}{N_j} \sum_{i=1}^{N_j} g(\xi_i^j). \tag{4}$$

Във формула (4) с N_j е означен броят на независимите реализации на случайната точка ξ^j с вероятностна плътност $p(\mathrm{x})/p_j$, където $\sum_{j=1}^{M} N_j = N$. Дисперсията на оценката (4) се изразява така:

$$\mathrm{D}\overline{\theta}^*_N = \sum_{j=1}^{M} p_j^2\ \mathrm{D}\overline{\theta}^{*j}_N, \quad \text{където} \quad \mathrm{D}\overline{\theta}^{*j}_N = \frac{1}{N_j}\sum_{i=1}^{N_j} g^2(\xi_i^j) - \frac{1}{N_j^2}\left(\sum_{i=1}^{N_j} g(\xi_i^j)\right)^2$$

е дисперсията на оценката за интеграла в подобласт Ω_j. Валидна е следната теорема:

Теорема 1.1.1. *(Соболь, [194]) За фиксирано разбиване на областта на интегриране и $N_j = N\ p_j$ е в сила, че*

$$\mathrm{D}\overline{\theta}^*_N \leq \mathrm{D}\overline{\theta}_N,$$

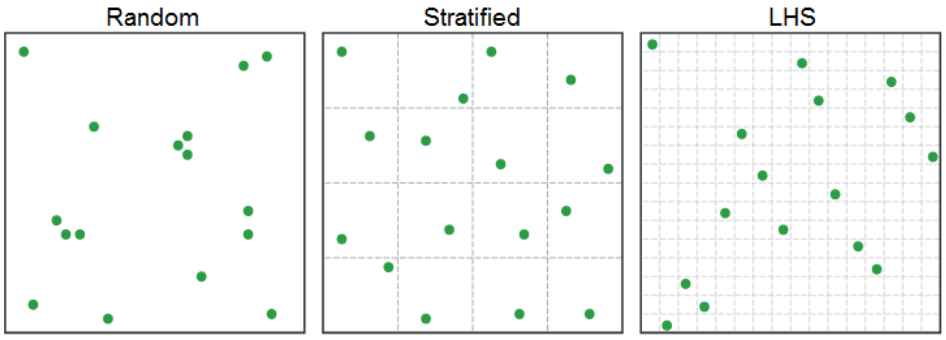
където $\mathrm{D}\overline{\theta}_N$ е дисперсията на оценката на интеграла $I[g]$, получена с обикновен метод Монте Карло и N случайни точки в Ω.

При метода на „слоистите“ извадки се избират повече точки в областите, в които локалната дисперсия е голяма, докато при метода на „съществената извадка“ - в областите, в които подинтегралната функция има по-големи стойности по модул. В заключение трябва да се подчертае, че подходът на „слоистите“ извадки не съдържа идеята за адаптивност (рекурсивно разделяне на подобласти) и не използва предварителна информация за гладкостта на подинтегралната функция. Подходът 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.

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 „orthogonal array sampling“ [168]. Основен недостатък при нея е трудната реализация, тъй като всички случайни точки трябва да се генерират едновременно. Предимството на извадката латински хиперкуб (LHS) [129, 214] е, че точките могат да се вземат една след друга, като се запомня разположението на предишните.

Извадката латински хиперкуб (LHS) е специален вид „слоистата“ извадка, описана за първи път от McKay през 1979г. в [145]. Независимо подобна идея е описана по-рано от Eglajs през 1977г. в [74]. По-късно методът LHS е обстойно проучен от Ronald L. Iman и колектива му в [106, 107]. Наскоро са разработени

Фигура 1.1: Сравнение на различни типове извадки



ефективни реализации на извадката латински хиперкуб с подобрена точност от Budiman Minasny в [150, 151].

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. Тази конфигурация съответства на шахматна дъска, в която топовете са разположени, така че нито един не застрашава друг. 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. 1.1 е дадена извадката латински хиперкуб, сравнена с метода на „слоистата“ извадка и на случайната извадка с 16 точки ($s = 2$ е размерността, а $M = 4$ са броя подобласти, фигурата е взета от [108]).

1.2 Адаптивен алгоритъм Монте Карло

Когато подинтегралната функция не е гладка адаптивните алгоритми са особено подходящи за използване. Съществуват различни подходи при конструирането на адаптивни алгоритми Монте Карло [50, 118]. Реализираният тук адаптивен алгоритъм не използва никаква предварителна информация за гладкостта на подинтегралната функция, но използва апостериорна информация за дисперсията.

Основната идея е концентриране на случайни точки в подобластите, в ко-

ито дисперсията е голяма (по отношение на предварително зададена точност), т.е. подходът се основава на рекурсивно разделяне на областта, използвайки апостериорна информация за грешката при текущото разделяне. Адаптивният алгоритъм дава приближение с грешка $\varepsilon \leq c\, N^{-1/2}$, където $c \leq 0.6745\sigma(\theta)$ ($\sigma(\theta)$ е стандартното отклонение). От оценката за грешката може да се направи извод, че адаптивния алгоритъм Монте Карло дава грешка, по-малка от грешката на обикновения алгоритъм Монте Карло, но порядъкът е същият.

Реализиран е адаптивен алгоритъм, базиран на на алгоритъма, описан в [4, 56, 61]. Алгоритъм започва с разделяне на интервалите по всички направления на M подинтервала, като M е зададено като входящ параметър, т.е.

$$\Omega = \sum_j \Omega_j, \quad j = 1,\, M^s.$$

С p_j и I_{Ω_j} са означени следните величини:

$$p_j = \int_{\Omega_j} p(\mathrm{x})\,\mathrm{dx} \quad \text{и} \quad I_{\Omega_j} = \int_{\Omega_j} f(\mathrm{x})p(\mathrm{x})\,\mathrm{dx}.$$

Разглеждаме случайна точка $\xi^{(j)} \in \Omega_j$ с вероятностна плътност $p(\mathrm{x})/p_j$. В този случай

$$I_{\Omega_j} = \mathrm{E}\left[\frac{p_j}{N}\sum_{i=1}^{N} f(\xi_i^{(j)})\right].$$

Във всяка подобласт се пресмята стойността на съответния интеграл I_{Ω_j} 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. Алгоритъмът спира, когато стандартното отклонение във всички, получени след деленето подобласти, е достигната предварително зададената точност ε (или когато е достигнат предварително зададен максимален брой на подобластите, в които не е изпълнен стоп-критерият, или на нивата на разделяне). Така се получава едно приближение на интеграла $I = \sum_j I_{\Omega_j}, j = 1, \ldots, M^s$ с подход Монте Карло. Алгоритъмът е описан по-долу.

Описание на адаптивния алгоритъм

1. **Входни данни**: *брой реализации N, константа M (първоначален брой подобласти), константа ε (максимална стойност на дисперсията във всяка подобласт), s-размерност на задачата, f - функцията, която разглеждаме.*

2. **Начално разделяне** на областта на интегриране Ω.

3. **За** $j = 1,\ M^s$ *($M \geq 2$ за началното разделяне и $M = 2$ за останалите)*:

   3.1. Пресмятане *на приближение на I_{Ω_j} и дисперсията D_{Ω_j} в подобластта Ω_j на основата на N независими реализации на случайната величина θ_N*;

   3.2. Ако ($\sqrt{\mathrm{D}_{\Omega_j}} \geq \varepsilon$), **тогава**

      3.2.1. прибавяне *на подобластта Ω_j в базата от данни на подобласти, където стандартното отклонение е по-голямо от ε*,

      3.2.2. избор *на подобласт от базата с максимално стандартно отклонение от всички нива на разделяне*,

      3.2.3. разделяне *на избраната подобласт на 2^d подобласти на две по всяко направление* и **преминаване** *на стъпка 3.1*;

   3.3. Иначе ако ($\mathrm{D}_{\Omega_j} < \varepsilon$), **тогава**

      3.3.1. изваждане *на настоящата подобласт от базата с данни*,

      3.3.2. ако *базата с данни не е пълна*, **тогава преминаване** *на стъпка 3.2.2*;

   3.4. Сумиране *на приближенията I_N за получаване на I.*

4. **Край на алгоритъма**.

Изчислителна сложност

Разглеждаме случайната величина

$$\theta = f(\xi),$$

чието математическо очакване съвпада със стойността на интеграла I_{G_j}, т.е.

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

Нека $\xi_1, \xi_2, \ldots, \xi_N$ са независими реализации на случайната точка ξ с плътност $p(x)$ и $\theta_1 = f(\xi_1), \ldots, \theta_N = f(\xi_N)$. Тогава приближената стойност на I_{G_j} е

$$\hat{\theta}_N = \frac{1}{N} \sum_{i=1}^{N} \theta_i.$$

Сега лесно следва, че изчислителната сложност на обикновения Монте Карло алгоритъм е линейна, тъй като в този случай трябва да изберем N случайни точки в областта и всеки такъв избор е на цената на $\mathcal{O}(1)$ операции. Едно изчисление на функцията в тези точки изисква също константен брой операции. т.е. $\mathcal{O}(1)$ операции.

Сега да забележим, че при адаптивния алгоритъм правим същия брой операции. Да разгледаме простия случай при $N = 2$ и на първа стъпка имаме 4 подобласти и

$$\hat{\theta}_N = \frac{1}{N_1} \sum_{i=1}^{N_1} \theta_i + \frac{1}{N_2} \sum_{i=1}^{N_2} \theta_i + \frac{1}{N_3} \sum_{i=1}^{N_3} \theta_i + \frac{1}{N_4} \sum_{i=1}^{N_4} \theta_i,$$

където $N_1 + N_2 + N_3 + N_4 = N$, така че имаме същия брой операции като обикновения Монте Карло алгоритъм за изчислението на I_{G_j}.

На практика, когато разделяме областта, трябва да изберем $\mathcal{O}(1)$ подобласти, където дисперсията е по-голяма от параметъра ε и този избор е независим от размерността на задачата. Може лесно да се провери, че на всяка стъпка адаптивността не е във всички подобласти, а само в $\mathcal{O}(1)$ подобласти. В началото избираме $\frac{N}{k_0}$ случайни точки, където k_0 е делител на N. На следващата стъпка, след като сме разделили областта на 2^N подобласти, избираме $\mathcal{O}(1)$ подобласти и там избираме $\frac{N}{k_1}$ точки. Следователно на j^{th} стъпка на адаптивния алгоритъм избираме $\mathcal{O}(1)$ подобласти и в тях избираме $\frac{N}{k_j}$ точки. Използваме, че $\sum_{j=0}^{i} \frac{1}{k_j} = 1$. Следователно за изчислителната сложност получаваме

$$\frac{N}{k_0} + \mathcal{O}(1)\frac{N}{k_1} + \cdots + \mathcal{O}(1)\frac{N}{k_i} = N\mathcal{O}(1)\left(\sum_{j=0}^{i} \frac{1}{k_j}\right) = N\mathcal{O}(1) = \mathcal{O}(N).$$

Така получихме, че изчислителната сложност на адаптивния алгоритъм е линейна като използвахме, че максималният брой подобласти, на които делим, е фиксирана константа и ще имаме само константен брой операции в повече

от обикновения метод Монте Карло. Трябва да се отбележи, че тази константа понякога е съществена, защото може да стигнем до няколко нива на разделяне, а се включват и сравненията дали дисперсията е по малка от ε.

1.3 Квази-Монте Карло методи с използване на точкови множества от тип решетка

Разглеждаме задачата за приближение на интеграла

$$I(f) = \int_{[0,1)^s} f(x)dx, \tag{1.3.1}$$

където f приема реални стойности и има интегруем квадрат в $[0,1)^s$.

За апроксимирането на горния интеграл може да се използва квадратурата:

$$I_N(f) = \frac{1}{N}\sum_{j=0}^{N-1} f(x_j), \tag{1.3.2}$$

където $P_N = \{x_0, x_1, \ldots, x_{N-1}\}$, $x_i \in [0,1)^s$ е множество от точки, задаващо възлите на квадратурната формула.

Съществуват различни начини за определяне на множеството P_N. Когато възлите на квадратурната формула са независими и еднакво разпределени случайни точки, получаваме обикновения метод Монте Карло за оценка на интеграла (1.3.1). Неравенството на Коксма-Хлавка (виж Увода) определя двата основни фактора, формиращи грешката от числено интегриране. От една страна, това е гладкостта на подинтегралната функция, а, от друга, дискрепансът на редицата от точки, чиито елементи се явяват възли на квадратурната формула. Следователно при фиксирана функция f, точковите множества с малък дискрепанс би трябвало да водят до оценки с по-голяма точност.

Когато точките от множеството P_N са възлите на квадратурната формула, попадащи в областта на интегриране Ω, получаваме метод за апроксимиране на интеграла (1.3.1), наречен теоретико-числов метод за интегриране на възлите на квадратурната формула (lattice rule):

$$Q(f) = \frac{1}{N}\sum_{k=0}^{N-1} f(x_k), \ \{x_k : k = 0, 1, \ldots, N-1\}. \tag{1.3.3}$$

Пример за такова интегриране в едномерния случай е апроксимирането по метода на правоъгълниците:

$$R_n(f) = \frac{1}{n}\sum_{j=0}^{N-1} f\left(\frac{j}{n}\right).$$

Неговата точност е $(O)(n^{-2})$. Най-очевидното обобщение на метода на правоъгълниците за по-голяма размерност е прилагането му по всяка координата:

$$I(f) \approx \frac{1}{n^s}\sum_{j_1=0}^{n-1}\cdots\sum_{j_s=0}^{n-1} f\left(\frac{j_1}{n},\ldots,\frac{j_s}{n}\right). \tag{1.3.4}$$

Общият брой възли в този случай е $N = n^s$. Грешката в (1.3.4) е $\mathcal{O}(N^{-2/s})$, което при големи s прави приближението нецелесъобразно. Например, за постигане на точност 10^{-2} трябва да се използват повече от 10^s възли [9]. В този случай изчислителната сложност нараства експоненциално с размерността на задачата и този феномен е известен като "проклятието на размерността".

Практическо приложение е намерил теоретико-числовият метод за интегриране, предложен от Корабов [127], при който възлите на квадратурната формула (1.3.3) имат вида:

$$x_k = \left(\left\{\frac{kz_1}{N}\right\},\left\{\frac{kz_2}{N}\right\},\ldots,\left\{\frac{kz_s}{N}\right\}\right),\ k = 1,2,\ldots,N, \tag{1.3.5}$$

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

Съответният квази-Монте Карло метод за апроксимиране на интеграла има вида:

$$Q_{N,s}(f) = \frac{1}{N}\sum_{k=1}^{N} f\left(\left\{\frac{k}{N}z\right\}\right). \tag{1.3.6}$$

В статията на Sloan и Lyness от 1989 [191] се въвежда понятието ранг на решетка. За всяка решетка L с размерност s съществува уникално число r, $1 \le r \le s$, и положителни числа $n_1,\ldots,n_r$, за които $n_{i+1}|n_i, i = 1,\ldots,r-1, n_r > 1$ такива, че възловото множество P се поражда от

$$\left\{\frac{k_1}{n_1}z_1 + \cdots + \frac{k_r}{n_r}z_r\right\}, \tag{1.3.7}$$

където $1 \leq k_i \leq n_i, 1 \leq i \leq r$. Най-малкото естествено число r, което удовлетворява (1.3.7) се нарича ранг на решетката, а $n_1, \ldots, n_r$ са инварианти. Ще разглеждаме решетки с ранг 1.

Нека $f(x) \in [0,1]^s$ е периодична функция с период единица по всяка от променливите си $x_i, i = 1, 2 \ldots, s$ и нека $f(x)$ да може да се представи в абсолютно сходящ ред на Фурие:

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

където

$$a(m) = \int\limits_{[0,1)^s} e^{-2\pi i m.x} f(x) dx,$$

и скаларното произведение $m.x = m_1 x_1 + m_2 x_2 + \ldots m_s x_s$. Записано по-подробно, ако $\hat{f}$ е периодичното продължение на $f(x)$ в цялото $\mathbb{R}^s$, то

$$a(m) = \hat{f}(m_1, \ldots, m_s) = \int_0^1 \ldots \int_0^1 f(x_1, \ldots, x_s) e^{-2\pi i (m_1 x_1 + \ldots m_s x_s)} dx_1 \ldots dx_s.$$

Да дефинираме дуалната решетка на L като

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

В случая, когато решетката е с ранг 1, имаме

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

Дуалната (реципрочна, полярна) решетка е концепция на геометричната теория на числата и е полезна в теория на кодирането, Х-лъчевата дифракция и физиката на твърдото тяло и се появява преди няколко десетилетия в контекста на многомерното интегриране.

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 [190]:

Дефиниция 1.3.1. 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 $x_i, i = 1, 2 \ldots, s$, дефинирана върху единичния хиперкуб $[0,1]^s$ и коефициенти й на Фурие удовлетворяват неравенството:

$$|a(m)| \leqq \frac{C}{(\overline{m}_1 \ldots \overline{m}_s)^{\alpha}}, \tag{1.3.8}$$

където

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

и константата c не зависи от $m_1, \ldots, m_s$.

Дефинираме индекс на Заремба [223, 225] като

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

Ключът към анализа на грешката на множествата от тип решетки е експоненциалната сума в следващата теорема, доказана от Sloan и Kachoyan(1987) [190]. Доказателството използва понятия от теория на групите.

Теорема 1.3.1. *Нека L е решетка с точки $x_0, x_1, \ldots, x_{N-1}$ в $[0,1)^s$ и нека $m \in \mathbb{Z}^s$. Тогава*

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

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

Теорема 1.3.2. *[190] Нека L е решетка с точки $x_0, x_1, \ldots, x_{N-1}$ в $[0,1)^s$. Тогава за грешката на приближение е изпълнено*

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

Теорема 1.3.3. *[190] Нека L е решетка с точки $x_0, x_1, \ldots, x_{N-1}$ в $[0,1)^s$ и нека $f \in E_s^{\alpha}(c), \alpha > 1$. Тогава*

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

Доказателство. От предишната теорема имаме, че

$$|I_N(f) - I(f)| = \sum_{m \in L^{\perp}, m \neq 0} |a(m)| \leqq c \sum_{m \in L^{\perp}, m \neq 0} \frac{1}{(\overline{m}_1 \ldots \overline{m}_s)^{\alpha}}$$

и това доказва исканото. □

В случая на решетка с ранг 1 имаме:

$$\left|\frac{1}{N}\sum_{k=0}^{N-1} f\left(\left\{\frac{k}{N}z\right\}\right) - \int\limits_{[0,1)^s} f(u)du\right| \leq c \sum_{m.x\equiv 0(modN), m\neq 0} \frac{1}{(\overline{m}_1 \dots \overline{m}_s)^\alpha}. \qquad (1.3.9)$$

Теорема 1.3.4. *[190] Нека L е решетка с точки $x_0, x_1, \dots, x_{N-1}$ в $[0,1)^s$ и нека $f \in E_s^\alpha(c), \alpha > 1$ и $\rho \geqq 2$ е индексът на Заремба. Тогава*

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

Да дефинираме числото $R_l, l = 1, 2, \dots,$ което да показва броя точки $m \in L^\perp$, такива че

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

Доказателството използва следната лема, доказана от Hua и Wang (1981) [104]:

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

където $e(s)$ зависи само от s.

Доказателство. Съгласно Теорема 1.3.3

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

Сумата по m може да се разбие на сума от събираеми по области $E_1, E_2, \dots,$ където E_l е дефинира от неравенствата

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

Съгласно дефиницията на R_l имаме следните неравенства:

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

Имаме, че

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

и сега използвайки лемата

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

където $d_1(s,\alpha)$ зависи само от s и α. С това теоремата е доказана. □

Доказан е следният резултат [23]:

Теорема 1.3.5. *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*

$$\left| \frac{1}{N} \sum_{k=0}^{N-1} f\left(\left\{ \frac{k}{N} z \right\} \right) - \int_{[0,1)^s} f(u)du \right| \leq Cd(s,\alpha) \frac{(\log N)^{\beta(s,\alpha)}}{N^\alpha} \tag{1.3.10}$$

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 (1.3.10) 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 квадратура (1.3.6) се нарича метод на добрите целочислени точки (Good Lattice Point method- GLP метод). Корабов е доказал [126] съществуването на добри целочислени точки в случая, когато N е просто число или произведение на две прости числа. Известни са редица теореми, доказващи съществуването на оптимални генериращи вектори. Трудността е в конструирането на тези оптимални вектори, особено в задачи от голяма размерност.

Основната идея за получаването на оптимален генериращ вектор z е свързана с търсенето на минимума на най-лошата грешка, получена при интегрирането на тестови функции, принадлежащи на предварително дефиниран клас. За всички функции от даден клас се прави оценка на грешката $|Q_{N,s}(f) - I(f)|$ от численото интегриране. Намира се „най-лошата“ функция за класа, т.е. функцията, за която грешката от численото интегриране е максимална. Тогава под

добър генериращ вектор се разбира векторът, минимизиращ грешката получена при интегрирането на най-лошата функция в класа. Следователно при GLP метода, минимизирането на максимума на грешката от численото интегриране води до намиране на оптимален генериращ вектор z.

В теорията на решетките важна роля играят функциите $f_\alpha, \alpha = 2, 4, \ldots$. Всяка функция f_α е „най- лошата функция“ [218] за подходящ клас $E_s^\alpha(1)$. Тези функции се дефинират чрез

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

Освен това $f_\alpha \in E_s^\alpha(1), I(f_\alpha) = 1$. Нека $P_\alpha(z, N) = P_\alpha$ означава грешката, която имаме в $I(f_\alpha)$. Тогава от Теорема 1.3.2

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

Сега за $f \in E_s^\alpha(c)$ по Теорема 1.3.3 грешката се записва като

$$|I_N(f) - I(f)| \leq c P_\alpha(N, z), \tag{1.3.11}$$

където $\alpha = 2, 4, \ldots$ и грешката се достига ако $f = f_\alpha$. Стойностите на $P_\alpha(z, N)$ при фиксирано α се използват като индикация за относително качество на отделните решетки. В случая на решетка с ранг 1 и $f \in E_\alpha^s(c)$ имаме

$$P_\alpha(z, N) = \sum_{z.a \equiv 0 \ (mod N), a \neq 0} \frac{c}{(\overline{m}_1 \ldots \overline{m}_s)^\alpha}.$$

Korobov и Bakhvalov (1959) [23, 125] доказват следната теорема:

Теорема 1.3.6. *Ако N е просто число, то съществува избор на генериращ вектор z, така че*

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

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

Niederreiter показва [158], че ако N е съставно число, то съществуват точкови множества от тип решетка, за които:

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

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

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

където $\phi(N)$ е функцията тотиента на Ойлер и $\tau(N)$ е броят на положителните делители на N. Аналогични оценки са получени от Sloan и Disney [68]. За прости числа тези формули показват съществуването на z, така че

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

В сила е следната теорема на Sharygin (1963) [186]:

Теорема 1.3.7. *За всяко точково множество от тип решетка е в сила следната оценка:*

$$P_\alpha(z,N) \geq O(N^{-\alpha}\log^{s-1} N). \tag{1.3.12}$$

Броят на различните квадратури от вида (1.3.2) с N интеграционни възела е краен, тъй като без ограничение на общността можем да считаме, че векторът z принадлежи на крайно множество $\mathbb{Z}_N^s$, където

$$Z_N = 0, 1, 2, \ldots, N-1.$$

Но дори за средно големи стойности на N и s, пълното търсене сред елементите на $\mathbb{Z}_N^s$ за определяне на минимума на най-лошата грешка на практика е невъзможно, поради прекалено големия брой възможни стойности на генериращия вектор. За да редуцира този брой Корабов е предложил следната форма за z:

$$z = \{1, a, a^2, \ldots, a^{s-1}\},$$

където N е просто число и $a \in \{1, 2, 3, \ldots, N-1\}$.

Търсенето на оптимални генериращи вектори при висока размерност s при фиксирано N има висока изчислителна трудност. Изборът на добър генериращ вектор, който води до малки грешки при интегриране, не е тривиален. Използват се сложни методи от теория на числата, основани на различни критерии като индекс на Заремба и грешка на най-лошата функция, описани по-горе. Установено е, че за добър генериращ вектор е необходимо индекса на Заремба да бъде максимален. При $s = 2$ оптимална конструкция съществува. Bakhavalov

(1959) [23], Hua и Wang (1960) [103] представят конструкция, основана на числа на Фибоначи, които се дефинират рекурсивно чрез

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

Избираме $N = F_l$ и $z = (1, F_{l-1})$. За полученото точково множество от тип решетка Bakhavalov и Hua и Wang показват, че

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

което е оптимално съгласно теоремата на Sharygin. През 1966 Заремба [223] показва, че

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

което има оптимален порядък съгласно Schmidt (1972) [183]. Важно е да се отбележи, че за намиране на F_l са нужни само $O(\log F_l)$ елементарни операции. Различни техники се изследват за построяване на оптимални конструкции при $s \geq 2$. Нека $s = \frac{p-1}{2}$, където $p \geq 5$ е просто число. Разглеждаме циклотомичното поле $Q(2\cos\frac{2\pi}{p})$, което е алгебрически числово поле от степен s. То има базис $2\cos(2\pi j/p) \mid j = 1, \ldots, s$, така че конструираме редицата $\eta_l, l = 1, 2, \ldots$, която удовлетворява:

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

където c_s е константа и $\eta^{(j)}$ е спрегнатото на η. Дефинираме генериращия вектор чрез:

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

където η_l е броя на точките и [.] е функцията цяла част. При такъв избор на z Hua и Wang показват, че

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

където ε е предварително зададено положително число.

През 1981г. Hua и Wang в [104] обобщават редицата на Фибоначи и разглеждат точково множество от тип решетка с генериращ вектор, който е описан

по-долу.Разглеждаме следния генериращ вектор за някое естествено число n [104, 218]:

$$z = (1, F_n(2), \ldots, F_n(s)) \tag{1.3.13}$$

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

$$F_{l+s} = F_l + F_{l+1} + ... + F_{l+s-1}, l = 0, 1, \ldots \tag{1.3.14}$$

с начално условие:

$$F_0 = F_1 = \ldots = F_{s-2} = 0, F_{s-1} = 1, \tag{1.3.15}$$

за $l = 0, 1, \ldots$.

След опростяване, може да се види, че генериращият вектор по-горе е:

$$z = (1, F_{n-1} + F_{n-2} + \ldots + F_{n-s+1}, \ldots, F_{n-1} + F_{n-2}, F_{n-1}) \tag{1.3.16}$$

Според [104], имаме следната оценка за дискрепанса на множеството, получено с използването на този вектор и F_n на брой точки:

Теорема 1.3.8. *Множеството*

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

има дискрепанс

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

и за най-лошата грешка е в сила:

$$P_\alpha(z, N_l) = \mathcal{O}\left(N_l^{-\frac{\alpha}{2} - \frac{\alpha}{2^{s+1} \cdot \log 2} - \frac{\alpha}{2^{2s+4}}} \right).$$

Броят на операциите, необходими за получаване на генериращия 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. Следователно алгоритъмът има линейна изчислителна сложност, при това експериментите показват, че скоростта е близка до тази на най-бързия обикновен Монте Карло метод.

1.4 Приложение в Бейсовската статистика

В тази секция разглеждаме примери на няколко многомерни интеграла съответно 5, 15 и 30-мерен, като ще се използват техните референтни стойности. Това са примери на многомерни интеграли с гладки подинтегрални функции. Методът на решетките с обобщени числа на Фибоначи е сравнен с широко използваната квазислучайна редица на Собол. Както отбелязват Sloan и Joe [189] решетките са конструирани, така че да се възползват от гладкостта на подинтегралната функция. Оказва се, че за високи размерности $s > 20$ предимство има извадката латински хиперкуб, която се оказва дори по-точна от адаптивния алгоритъм Монте Карло за фиксиран брой реализации.

В съвременната статистика има две парадигми [18]. Едната, която е най-позната и най-използвана, се нарича честотна ("frequentist"). Другата, която е почти непозната (особено в България) и е използвана само от тесен кръг учени и изследователи (главно по света), се нарича Бейсовска (Bayesian) [26]. Но дори и там, където Бейсовската статистика е позната, тя не се осъзнава като отделна парадигма. Това е валидно в еднаква степен както за представителите на честотната парадигма, така и за представителите на Бейсовската статистика. В Бейсовската статистика имаме следната опростена постановка [109] - при всяко изследване разполагаме само с данните от изследването и искаме да проверим някакви хипотези. В такъв случай, вероятността конкретна хипотеза да е вярна и едновременно с това да разполагаме точно с тези данни, с които разполагаме, може да се представи по два начина. Първо, това е вероятността хипотезата да е вярна умножена по вероятността да разполагаме точно с тези данни, при условие че хипотезата е вярна, или второ, вероятността да разполагаме точно с тези данни умножена по вероятността хипотезата да е вярна, при условие че това са данните. На практика, освен данните от изследването и проверяваните хипотези, при всяко изследване разполагаме още и с априорна информация. Това може да е теоретично знание за изследвания обект, а може да е информация от някакви минали изследвания на същия обект. Априорната вероятност е вероятността хипотезата да е вярна само в светлината на априорната информация за изследвания обект. Извадковото разпределение е вероятността да се получат точно тези данни, които са получени, ако хипотезата е вярна. Пълната

вероятност е вероятността да се получат точно тези данни, които са получени, независимо дали хипотезата е вярна или не [18].

Едно от фундаменталните различия между честотната и Бейсовската статистика е в разбирането на това какво е вероятност. Според честотната статистика вероятността е обективна характеристика на изследвания обект, която се проявява при безкраен брой опити [139]. Според Бейсовската статистика вероятността е измерител на знанието(state of knowledge) за изследвания обект. В този смисъл, от гледна точка на честотната статистика вероятността е вътрешно присъща характеристика на изследвания обект, а от гледна точка на Бейсовската статистика вероятността не характеризира обекта, а знанието за него [34].

Фундаментален проблем в Бейсовската статистика е точното пресмятане на многомерни интеграли от следните два вида, описани по долу и разгледани от Shaowei Lin в [136]. Такива интеграли имат важно приложение в машинното обучение [137] и изчислителната биология [44]. Първият вид интеграли имат следния вид:

$$\int_{\Omega} p_1^{u_1}(x)\dots p_k^{u_k}(x)dx, \tag{1.4.1}$$

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 интеграли имат вида

$$\int_{\Omega} e^{-Nf(x)}\phi(x)dx, \tag{1.4.2}$$

където N е естествено число и $f(x), \phi(x)$ са многомерни полиноми. Такива са 5-мерния и 15-мерния интеграли от числените експерименти.

Пример 1. $s = 5$.

$$\int\limits_{[0,1]^5} \exp(-100x_1x_2x_3)(\sin(x_4) + \cos(x_5))dx \approx 0.1854297367. \tag{1.4.3}$$

Пример 2. $s = 15$.

$$\int\limits_{[0,1]^{15}} (\sum_{i=1}^{10} x_i^2)(x_{11} - x_{12}^2 - x_{13}^3 - x_{14}^4 - x_{15}^5)^2 dx \approx 1.96440666. \tag{1.4.4}$$

Пример 3. $s = 30$.

$$\int\limits_{[0,1]^{30}} \frac{4x_1x_3^2e^{2x_1x_3}}{(1 + x_2 + x_4)^2} e^{x_5+\dots+x_{20}} x_{21}\dots x_{30}dx \approx 3.244540. \tag{1.4.5}$$

Направено детайлно сравнение между обикновен Монте Карло алгоритъм (Crude), адаптивен алгоритъм Монте Карло (Adapt), алгоритъм с използване на квазислучайната редица на Собол (Sobol), алгоритъм с използване на точково множество от тип решетки базирано на обобщената редица на Фибоначи (FIBO) и извадка латински хиперкуб (LHS). Резултатите са дадени в таблиците по-долу. Първата група таблици съдържат информация за метода, който се използва, получената относителна грешка, необходимия брой реализации и необходимото процесорно време, измерено в секунди, за пресмятане на интеграла. Втората група таблици съдържат информация за относителната грешка при отнапред зададено време в секунди, което е мярка за изчислителната сложност. Очевидно е, че Crude, FIBO и LHS имат сходна бързина при отнапред зададен брой реализации. Методът на Собол е по-бавен и в случаите на ниска и висока размерност винаги е превъзхождан от поне един от останалите методи. 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.

От таблиците е очевидно, че за малка размерност методът FIBO постига най-добра точност - виж Таблици 1.1 и 1.2. За размерност 15 при отнапред зададен брой точки FIBO и Sobol постигат сходни резултати- виж Таблица 1.3, като FIBO постига по-добра точност за много по-малко време - виж Таблица 1.4. Анализите показват, че за висока размерност 30 извадката LHS превъзхожда FIBO и квази-Монте Карло агоритъма на Собол - виж Таблици 1.5 и 1.6, където методът FIBO започва да губи точност. Адаптивния алгоритъм е подходящ за по-високи размерности поради малкия брой реализации за достигане на отнапред зададената точност. Като бъдеща работа ще бъде сравнението на алгоритъмът FIBO с оптималният метод, разработен от Атанасов и Димов в [22].

От таблиците може да се направи извода, че методът FIBO е най-ефективен за пресмятане на многомерни интеграли от гладки подинтегрални функции заради ниската изчислителна сложност при ниски размерности. За високи размерност $s > 20$ за предпочитане е методът Монте Карло LHS. Методът Sobol дава добри резултати, независимо от размерността на интеграла.

Получените числени резултати за относителната грешка са в съгласие с тео-

ретичните оценки за дискрепанса на използваните редици, които се разглеждат в редица публикации на проф. Караиванова в [8, 118, 119, 120].

Таблица 1.1: Относителна грешка и изчислително време 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 1.2: Относителна грешка за 5-мерния интеграл при фиксирано изчислително време

време в сек.	Crude	Adapt	FIBO	Sobol	LHS
0.1	3.16e-3	3.48e-3	1.09e-5	1.34e-4	3.21e-3
1	1.08e-3	2.08e-3	5.58e-6	7.21e-5	8.54e-4
5	8.79e-4	8.20e-4	8.71e-7	1.54e-5	3.25e-4
10	5.85e-4	7.51e-4	4.15e-7	9.32e-6	8.65e-5
20	3.99e-4	6.95e-4	8.37e-8	7.39e-6	5.02e-5

Таблица 1.3: Относителна грешка и изчислително време 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 1.4: Относителна грешка за 15-мерния интеграл при фиксирано изчислително време

време в сек.	Crude	Adapt	FIBO	Sobol	LHS
1	4.16e-2	6.30e-2	1.10e-3	2.04e-3	7.89e-3
5	3.72e-2	1.68e-2	2.45e-4	7.32e-4	6.78e-4
10	2.33e-2	2.89e-3	9.48e-5	1.94e-4	1.64e-4
20	1.03e-2	1.66e-3	9.87e-6	4.05e-5	5.67e-5

Таблица 1.5: Относителна грешка и изчислително време 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	1.06e-2	13.61	9.47e-3	208	9.61e-5	13.9

Таблица 1.6: Относителна грешка за 30-мерния интеграл при фиксирано изчислително време

време,s	Crude	Adapt	FIBO	Sobol	LHS
1	5.01e-1	2.32e-1	2.38e-2	1.01e-1	4.38e-3
5	3.21e-1	9.21e-2	1.81e-2	7.76e-2	8.16e-4
10	1.13e-1	7.05e-2	9.48e-3	5.71e-2	3.11e-4
20	9.06e-2	6.99e-2	7.87e-3	1.28e-2	8.63e-5

1.5 Тестови функции на Генц

Методът на решетките, базиран на обобщената редица на Фибоначи и извадката латински хиперкуб, не са приложими за функции с особености, както се вижда от числените експерименти в тази подсекция. Няма универсален метод и

за функции с особености в локална подобласт на областта на интегриране, адаптивният алгоритъм позволява да се постигне висока точност със сравнително малка изчислителна сложност.

Нека е дадена следната моделна функция:

$$f(x) = (1 + \sum_{i=1}^{s} a_i \; x_i)^{-(s+1)}. \tag{1.5.1}$$

Разглежданият клас от тестови функции принадлежи на пакет, предложен от Генц [84]. Всеки отделен клас от пакета се характеризира с особеност в изчислително отношение. 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. Параметрите a_i са пресметнати, използвайки временни стойности a'_i, равномерно разпределени в $[\frac{1}{20}; 1 - \frac{1}{20}]$, и зависимостта $a = c \; a'$. Константата c представлява *параметър на изчислителна трудност* [28], избран, така че „заостреността“ на локалния максимум да се контролира от следната норма $||a||_1 = \frac{600}{s^2}$. Адаптивният подход е ефективен за такъв клас от функции – функции с изчислителни особености в локална подобласт на областта на интегриране.

Резултатите, получени след прилагането описаните алгоритми в тази глава- латински хиперкуб, точково множество от тип решетки, базирано на обобщени числа на Фибоначи и адаптивен алгоритъм Монте Карло за интеграли с размерност 5 и 18, са дадени съответно в Таблица 1.7 и Таблица 1.8. Изследвана е ефективността на разработения адаптивен алгоритъм Монте Карло.

Таблица 1.7: Относителна грешка и изчислително време в секунди за размерност $s = 5$, $I[f] = 2.12\mathbf{e}\text{-}06$, $a = (5, 5, 5, 5, 4)$.

Адаптивен алг. МК			латински хиперкуб			Решетки		
N	отн. гр.	вр.(s)	N	отн. гр.	вр.(s)	N	отн. гр.	вр.(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

Таблица 1.8: Относителна грешка и изчислително време в секунди за размерност $s = 18$, $I[f] = 9.919\mathbf{e}\text{-}06$, $a = \left(\frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{2}{27}, \frac{2}{27}, \frac{1}{9}, \frac{1}{9}, \frac{4}{27}, \frac{1}{9}, \frac{1}{9}\right)$.

Адаптивен алг. МК			латински хиперкуб			Решетки		
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

И в двете таблици с величината N се означава общият брой на реализациите в цялата област за извадката латински хиперкуб и за алгоритъма с използване на множество от тип решетки, както и броя реализации във всяка подобласт за адаптивния алгоритъм. Избран е брой реализации на случайната величина, така че времената за получаване на приближена стойност на интеграла да са близки. Получените резултати потвърждават намаляването на дисперсията – адаптивният алгоритъм има нужда от много по-малко реализации и дава по-точни резултати, отколкото Монте Карло метода, базиран на извадката латински хиперкуб и алгоритъма от тип решетки, но е по-бавен (вж. Таблици 1.7 и 1.8).

1.6 Приложение за ядрото на Вигнер

Един от най-известните физици на нашето време Ричард Файнман поставя през миналия век въпроса за съществуване на ефективен алгоритъм с линейна или полиномиална изчислителна сложност за пресмятане на многомерните интеграли при високи размерности, които описват ядрото на Вигнер в многомерния случай [80]. Досега са използвани само детерминистични алгоритми с експоненциална изчислителна сложност, които страдат от „проклтието на размерността“, за което беше вече споменато. Решението на този проблем е от фундаментално значение за формализма на Вигнер в квантовата механика, който е описан накратко по-долу. Вигнеровият формализъм в квантовата механика е интуитивен подход, който позволява разбирането и предсказването на феноме-

ни в квантовата механика в термините на квази-разпределени функции.

Трябва да се отбележи, че тук за пръв път за изчисляване на ядрото на Вигнер се използват стохастични методи, които не страдат от „проклятието а размерността“, което е присъщо за детерминистичните методи.

В днешно време съществуват различни формулировки на квантовата механика, сред които предложените от E. Schrodinger, E. Wigner, R. Feynman, L.V. Keldysh, K. Husimi, D. Bohm са сред най-популярните. Докато на пръв поглед те изглеждат драстично различни, може да се покаже, че всички те са математически еквивалентни. Ситуацията е същата като в класическата механика, където съществуват различни формулировки като тези на Нютон, Лагранж и Хамилтон и може да се докаже тяхната еквивалентност.

Наскоро е направена нова формулировка на квантовата механика в термините на квантови частици със знак. Тази теория е базирана на интерпретация на новия Вигнер Монте Карло метод, който може да симулира във времето едно и повече уравнения на Вигнер, с което специалистите в областта са запознати. Той наистина описва квантовите обекти само в термините на класическите квантови частици [220]. Новият Вигнеров Монте Карло алгоритъм за първи път позволява зависещо от времето, многоразмерно моделиране на системи с голям брой частици (дори за силно свързани системи), което е уникално свойство на метода [63]. Методът позволява за първи път да бъдат моделирани системи в квантовата химия, за които по-рано е било невъзможно да бъдат симулирани. Точната оценка на ядрото на Вигнер е от съществено значение за ефективността на алгоритъма и по този начин се отварят нови възможности при изследване на възбудени състояния на атоми, молекули, кристали и други системи в квантовата химия [156].

Може да се забележи, че новата формулировката на квантови частици със знак е еквивалента на обикновената формулировка [63]. Новата формулировка е обоснована на класически частици, които имат позиция и момент едновременно, въпреки, че принципът на Хайзенберг е спазен в новата формулировка. В частност, знакът на частицата не може да се пресметне експериментално и няма физическо измерение, което да представи разликите с останалите формализми. Въпреки всичко, този формализъм предлага няколко предимства. От една страна, той представя много интуитивен подход, който осигурява нов начин за

описване на природата на квантово ниво. От друга страна, това е изчислително атрактивна формулировка, основана на независимо отделящи се квантови частици, което позволява дълбоки нива на паралелизация и симулация във времето на една или много квантови системи. И не на последно място новата формулировка позволява включването на физични ефекти, които е трудно да бъдат третирани в другите формулировки на квантовата механика.

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 предизвикателствата на по-сложни квантови теории. Показано е в [184], че трите постулата са достатъчни да се възстанови квази функцията на разпределение на системата, следователно и нейната вълнова функция. С други думи време зависимите процеси в квантова система могат да се изразят напълно по отношение на създаване и унищожение само на частици със знак.

Постулат I. Физичните системи могат да се опишат чрез (виртуални) Нютонови частици, имащи позиция x и момент p едновременно, или с положителен или с отрицателен знак.

Постулат II. Частица със знак, която има 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

$$\gamma(x) = \int_{-\infty}^{+\infty} Dp' V_W^+(x;p') \equiv \lim_{\triangle p' \to 0^+} \sum_{M=-\infty}^{+\infty} V_W^+(x; M \triangle p'), \tag{1.6.1}$$

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

$$V_W(x;p) = \frac{i}{\pi^s \hbar^{s+1}} \int_{-\infty}^{+\infty} dx' e^{-\frac{2i}{\hbar}x'p}[V(x+x') - V(x-x')], \tag{1.6.2}$$

известна като ядрото на Вигнер в s-мерното пространство [220]. Ако по време на създаването, създаваната 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 нормираната вероятност $\frac{V_W^+(x;p)}{\gamma(x)}$.

Постулат III. Две частици с противоположни знаци и едни и същи координати (x,p) във фазовото пространство анихилират.

Таблица 1.9: 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

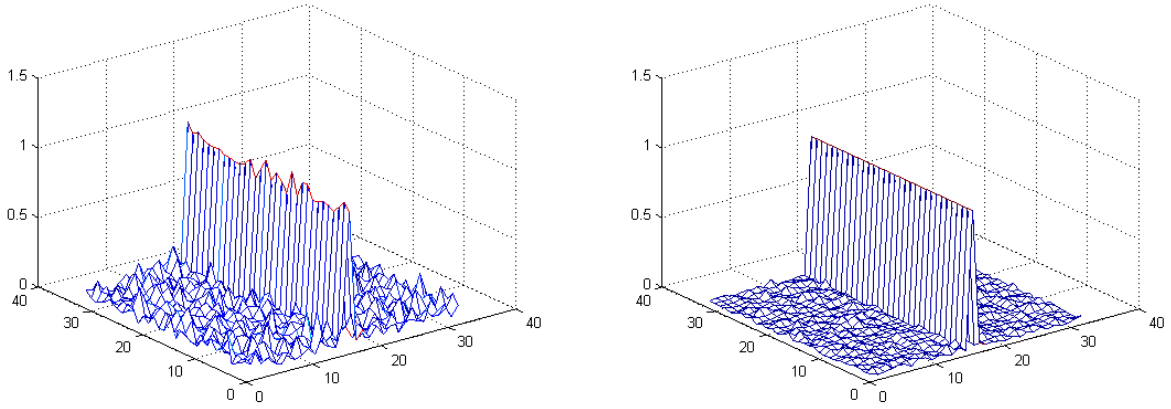
Използваме следния Вигнеров потенциал:

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

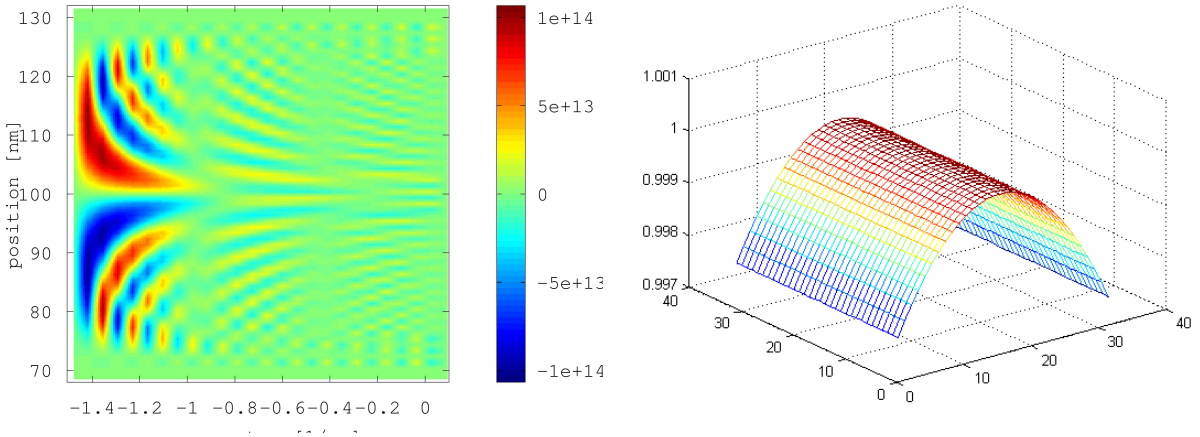
Целта на настоящето изследване е да се пресметне многомерния интеграл (1.6.2), представляващ ядрото на Вигнер с интегриране по трите променливи. Най-напред прилагаме детерминистичен метод на средните правоъгълници, обикновен метод Монте Карло и квази-Монте Карло с използване на квазислучайни редици на Собол. Резултатите са дадени в Таблица 1.9 по-долу. Детерминистичният метод е най-бавен и започва да губи точност с увеличаване на размерността на интеграла. Методът на Собол е по-точен от обикновения метод Монте Карло, но е по-бавен- виж Таблица 1.9. Обикновения алгоритъм Монте Карло е по-неточен от квази-Монте Карло алгоритъма с редици на Собол, поради свойствата на добре разпределените редици.

На следните графики е изобразено ядрото на Вигнер с помощта на адаптивен и обикновен Монте Карло метод – виж Фиг. 1.2 и позицията на частиците със

знак в ядрото и неговия връх на Фиг 1.3, получено с адаптивен алгоритъм Монте Карло.



Фигура 1.2: Ядрото на Вигнер, получено с обикновен и адаптивен алгоритъм Монте Карло



Фигура 1.3: Позиция на частиците в ядрото на Вигнер и върхът на ядрото на Вигнер с адаптивен алгоритъм

В таблиците по долу са представени получените резултати за относителната грешка за ядрото на Вигнер с четирите метода. Трябва да се отбележи, че четирите алгоритъма, които използваме са по-добри от досега използваните детерминистични методи за пресмятане на многомерните интеграл, свързан с

ядрото на Вигнер. За пръв път е използван адаптивен метод Монте Карло за пресмятането му и от таблиците се вижда, че той дава най-добри резултати при отнапред зададено време и брой точки. Предимството на адаптивния Монте Карло метод следва от вида на ядрото, илюстриран на Фиг. 1.2. На Фиг. 1.3 са показани позицията на частиците със знак в ядрото на Вигнер и върхът на ядрото на Вигнер, получени с адаптивния алгоритъм. 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.

Числените резултати показват, че алгоритъма от тип решетки, базиран на обобщената редица на Фибоначи, е по-добър и по-бърз от Собол за по-малко изчислително време за 3 и 6-мерния интеграл, но двата алгоритъма произвеждат подобни относителни грешки за един и същ брой реализации. За 9-мерни интеграл двата метода дават сходни резултати, които са по-неточни от адаптивния алгоритъм и извадкатата латински хиперкуб – виж Таблици 1.10 и 1.11. За първи път за пресмятане на ядрото на Вигнер се използва Монте Карло метода с LHS. Предимството е, че дава резултати с един порядък по-добри от метода на Собол и има сходно изчислително време като обикновения Монте Карло алгоритъм и линейна изчислителна сложност. Както вече отбелязахме, адаптивният алгоритъм дава най-добри резултати в сравнение с останалите стохастични алгоритми с до два порядъка по-добри от латинския хиперкуб, но е значително по-бавен – виж Таблици 1.10 и 1.11. Пиковете на ядрото на Вигнер, което се вижда от приложените графики, обяснява предимството на адаптивния алгоритъм за този важен многомерен интеграл с фундаментално значение в квантовата механика.

Получените резултати потвърждават очакваното, че адаптивния алгоритъм е най-точен за ядрото на Вигнер с увеличаване на размерността. LHS метода заради бързината си също е един възможен подход за оценяване на ядрото, макар и по-неточно от адаптивния алгоритъм. 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, за който беше споменато в началото.

Таблица 1.10: Относителна грешка за 3, 6 и 9-мерния интеграл

	N	LHS	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

Таблица 1.11: Относителна грешка при фиксирано изчислително време

	време (s)	LHS	Адаптивен	FIBO	Sobol
$s = 3$	1	1.14e-04	9.62e-05	5.42e-03	7.27e-03
	10	7.21e-05	9.12e-06	2.11e-03	7.83e-04
	100	9.11e-06	8.18e-07	9.50e-04	2.18e-04
$s = 6$	1	1.14e-04	9.09e-05	9.25e-04	3.31e-03
	10	6.15e-05	8.13e-06	5.11e-04	9.34e-04
	100	1.65e-05	9.08e-07	1.05e-04	1.27e-04
$s = 9$	1	6.54e-04	8.05e-05	8.72e-04	5.59e-03
	10	5.41e-05	6.32e-06	6.51e-04	5.84e-03
	100	9.22e-06	7.58e-07	3.70e-04	6.39e-04

1.7 Приложение за Европейски опции

В последните десетилетия се разработват нови методи, които превъзхождат стандартните приближения във финансите. Финансовите проблеми стават все по-трудни, но по-използвани в последните две десетилетия. Опциите са много обстойно изследвани от създаването на организираната обмяна на пари през 1973 г. Все повече усилия се влагат в разработването на ефективни Монте Карло/квази-Монте Карло алгоритми за многомерни интеграли с много висока размерност. Интерес представляват и проблеми с малка $s \leq 5$ и средна размерност $s \leq 15$, тъй като има много контракти, които зависят само от малък и среден брой променливи величини. Такива са например опциите в енергийната индустрия и баскет опциите в стоковата борса [32].

По долу ще илюстрираме как изследваните стохастични алгоритми в тази глава могат ефективно да бъдат приложени към пресмятане на Европейски опции. Идеята се състои в следното: стойността на опцията 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. Освен за оценяване на опции Монте Карло и квази-Монте Карло методите се използват и за решаване на други задачи от финансовата математика [105], [30], [131]. Приложения на редици с малък дискрепанс за решаване на други задачи от финансовата математика могат да се намерят в статиите на Boyle, Broadie и Glasserman (1997) [31], Caffisch, Morokoff и Owen (1997) [40], Joy, Boyle и Tan [111], Ninomiya и Tezuka (1996) [166], Tan и Boyle (2000) [205] и Paskov и Traub (1995) [174].

Първо ще бъдат представени някои финансови термини, по-детайлна информация може да се намери в [131], [41], [45], [83], [105] or [221].

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, а в последните години и фючърси, на фиксирана в договора цена, в рамките на даден период от време. За тази възможност той плаща съответна сума, наречена премия. За разлика от фючърса, който е задължаващ договор, при опциите притежателя може да упражни или да не упражни своето право в съответния срок. В зависимост от това, кога притежателя

може да упражни правото си по тях, те се делят на европейски, американски или бермудски тип. При американския тип ценните книжа, предмет на сделката, могат да се купят или продадат през целия договорен период, докато при европейския вид опции собственикът им може да упражни правото си само в края на периода, а при бермудските само на предварително фиксирани дати. Има и други видове по-нестандартни опции като азиатска, бариерна и други. Обект на настоящото разглеждане са европейски кол и пут опции [30, 31].

Огромен тласък в развитието на финансовата математика дава моделът за оценяване на цената на европейската кол опция (теоретичната стойност на премията), представен през 1973 г. от Фишер Блек (Fisher Black) и Майрън Шолс (Myron Scholes) в [29] и независимо от тях Робърт Мертън (Robert Merton) [147]

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0. \tag{1.7.1}$$

Тук $V(S,t')$, $(S,t') \in [0,\infty] \times (0,T]$ е цената на европейската кол опция. Тази цена зависи от следните два фактора: текущата пазарна цена на актива (акцията) - S, за която се отнася времето t', което остава до матуритета (финалната дата за падежа на опцията) T. Освен от тези два фактора, цената на европейската кол опция се определя и от основния лихвен процент r, волатилност на цената на акцията σ, т.е. степента на случайност на акцията. Величината σ е мярка за несигурността на възвръщаемостта на акцията, т.е. относно бъдещото движение на цената на акцията. Волатилитета се дефинира по такъв начин, че величината $\sigma\sqrt{\triangle t'}$ да бъде стандартното отклонение на възвръщаемостта на акцията за един малък период от време $\triangle t'$.

Да означим с E договорената в опционния договор цена (наричана още цена на упражняване), по която ще се закупи даден актив. При падеж T, притежателят на европейска кол опция взема решение, дали да упражни опцията в зависимост от пазарната цена на акцията $S := S(T)$. Ако тя е по-голяма от договорената, т.е. $S \geq E$, тогава той упражнява опцията, т.е. правото си да купи опции на цена E и след това веднага да ги продаде на цена S, с печалба $V = S - E$. В този случай се казва, че опцията е „в пари“ ("in-the-money option"). Ако $S = E$, тогава печалбата е нулева и опцията е „при пари“ ("at-the-money"). Когато $S \leq E$ има загуба и опцията е „извън пари“ ("out-of-the-money option"). В последните два случая собственикът на опцията няма да упражни

кол опцията, тъй като тези акции могат да се закупят на пазарна цена E или по-малко от E. Затова на падежна дата, цената на кол опцията, известна още като „pay-off“ функция, се задава с условието

$$V(S,T) = \max\left(S - E, 0\right), \ 0 \leq S \leq \infty. \tag{1.7.2}$$

Уравнението на Блек Шолс ([29] или [221]) е параболично с израждане на елиптичния оператор. При нулева текуща цена на базовия актив ($S = 0$) имаме

$$\frac{\partial V}{\partial t'} - rV = 0, \ 0 \leq t' \leq T. \tag{1.7.3}$$

Затова при $S = 0$ не е необходимо гранично условие. Ако цената на акцията е нула, то естествено и цената на опцията да е нула и затова вместо (1.7.3) може да бъде поставено условието

$$V(0,t') = 0, \ 0 \leq t' \leq T. \tag{1.7.4}$$

Когато цената на акцията расте неограничено, за цената на опцията имаме

$$V(S,t') \simeq S - Ee^{-r(T-t')}, \ 0 \leq t' \leq T, \ S \to \infty. \tag{1.7.5}$$

Точното решение на задачата в случая на постоянни коефициенти е известно и се дава с формулата:

$$V_{call}(S,t') = SN(d_1) - Ee^{-r(T-t')}N(d_2), \tag{1.7.6}$$

където

$$N(x) = \frac{1}{\sqrt{2\Pi}} \int\limits_{-\infty}^{x} e^{\frac{-x^2}{2}} dx, d_1 = \frac{ln(\frac{S}{E}) + (r + \frac{\sigma^2}{2})t'}{\sigma\sqrt{T-t'}}, d_2 = d_1 - \sigma\sqrt{T-t'}.$$

Цената на европейската пут опция удовлетворява същото уравнение на Блек-Шолс (1.7.1), но при други терминални и гранични условия. Тя е "в пари ако $E > S$, "при пари ако $E = S$ и "извън пари при $E < S$. Затова платежната фумкция (pay-off) се дефинира така:

$$V(S,T) = \max\left(S - E, 0\right), \ 0 \leq S \leq \infty. \tag{1.7.7}$$

Граничното условие при $S = 0$ се задава или с (1.7.3) или с

$$V(0,T) = Ee^{-r(T-t')}, 0 \leq t' \leq T, \tag{1.7.8}$$

а при $S \to \infty$ имаме

$$V(S,t') = 0,\ 0 \leq t'T. \tag{1.7.9}$$

Точното решение за европейската пут опция се задава с формулата:

$$V_{put}(S,t') = Ee^{-r(T-t')}N(-d_2) - SN(-d_1) = V_{call} + Ee^{-r(T-t')} - S, \tag{1.7.10}$$

Един от основните проблеми в оценката на опции е следният: при дадена текуща цена на актива S, цена на упражняване (или фиксираната цена, по която могат да се закупят ценните книжа) E, време на падеж T, лихвен процент r, и зададено уравнение, което описва поведението на S като функция на t:

$$dS = \mu S dt' + \sigma S dX, \tag{1.7.11}$$

където dX е Винеров процес, μ (мярка за средния ръст на растеж на цена на актива) е тенденцията (генералната посока на изменение на актива) и σ е изменението на актива (характеризира флуктуациите в цената S), как може да се определи справедлива цена на $V(S,t')$ на опцията?

Методите Монте Карло могат да бъдат много полезни [131, 172] в случаи, когато решението (т.е. стойността на опцията V) може да се представи като математическо очакване на случайна величина. Това е възможно с помощта на рисково-неутралната формула за Европейска опция [35]:

$$V(S,t) = E(e^{-r(T-t)}h(S(T)) \mid S(t) = S, \mu = r), \tag{1.7.12}$$

където $E(.)$ 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. От тази формула, добре известната формула на Блек-Шолс може да се възстанови.

Монте Карло и квази-Монте Карло методите могат директно да се приложат към задачи от финансовата математика, включващи многомерни интеграли. Например Пасков използва квазислучайната редица на Собол за да намери текущите стойности на ценни книжа за 360 мерни интеграли [172]. Целта на настоящото изследване е да сравни квазислучайната редица на Собол с алгоритъма, базиран на обобщената редица на Фибоначи при размерности $s \leq 20$.

Да разгледаме европейска опция, чиято платежна функция зависи от $k > 1$ актива с цени $S_i, i = 1, ..., k$. Всеки актив следва случайното движение (така нареченото случайно блуждаене):

$$dS_i = \mu_i S_i dt + \sigma_i S_i dX_i,$$

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: $h(S_1', \ldots, S_k')$, (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

$$V = e^{-r(T-t)}(2\pi(T-t))^{-k/2}(det\Sigma)^{-1/2}(\sigma_1, \ldots, \sigma_k)^{-1} \times$$
$$\times \int_0^\infty \ldots \int_0^\infty \frac{h(S_1', \ldots, S_k')}{S_1', \ldots, S_k'} exp(-0.5\alpha^T \Sigma^{-1} \alpha) dS_1', \ldots, dS_k',$$

където

$$\alpha_i = (\sigma_i (T-t)^{1/2})^{-1}(log\frac{S_i'}{S_i} - (r - \frac{\sigma^2}{2})(T-t)),$$

r е основният лихвен процент и Σ е матрицата на ковариация. 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-мерния 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. Когато платежната функция h е функцията експонента, с подходящ избор на константите, включени в горното уравнение [131] получаваме s-мерния интеграл $\int\limits_{[0,1]^k} exp(x_1, \ldots, x_s) dx_1 \ldots dx_s$

Числените експерименти включват пресмятането на следните 5 и 20-мерни интеграли, за които използваме референтни стойности пресметнати с помощта на програмата Математика [235]:

$$\int\limits_{[0,1]^5} \exp(\sum_{i=1}^{5} 0.5 a_i x_i^2 (2 + \sin \sum_{j=1, j\neq i}^{5} x_j)) dx \approx 2.923651, a_i = (1, 0.5, 0.2, 0.2, 0.2), \tag{1.7.13}$$

$$\int\limits_{[0,1]^{20}} \exp(\prod_{i=1}^{20} x_i) dx \approx 1.00000949634. \tag{1.7.14}$$

Резултатите са представени по долу. Отново 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 Таблици 1.12 и 1.13. При 20-мерния интеграл методът FIBO и този с редицата на Собол дават много близки резултати – виж Таблица 1.14. Адаптивния алгоритъм има предимство 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, близка до тази на квази-Монте Карло алгоритъма с редици на Собол – виж Таблица 1.15. Поведението на грешката е дадено и на Фиг. 1.4 и 1.5.

1.8 Заключение

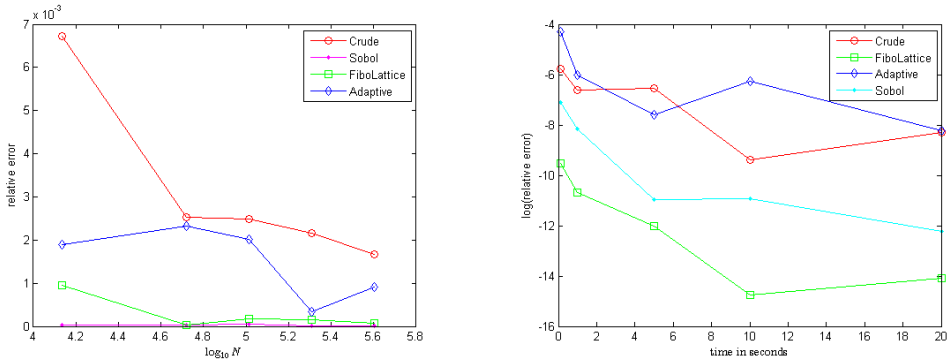
Изследван е адаптивен алгоритъм Монте Карло за числено интегриране и е приложен за клас от подинтегрални функции, като е направено сравнение с обикновения метод Монте Карло. Изследван е квази-Монте Карло метод за числено интегриране, базиран на множества от тип решетка с генериращ вектор обобщената редица на Фибоначи и е сравнен с метода на Собол, който се

Таблица 1.12: Относителна грешка за 5-мерния интеграл

брой точки	обикновен	T	адаптивен	T	FIBO	T	Собол	T
13624	6.72e-3	0.02	1.89e-3	2.33	9.59e-4	0.03	1.76e-4	0.56
52656	2.53e-3	0.06	2.31e-3	6.18	6.96e-4	0.06	5.05e-5	1.45
103519	2.48e-3	0.09	2.01e-3	9.94	8.72e-5	0.13	2.70e-5	2.52
203513	2.15e-3	0.15	3.42e-4	16.2	8.04e-5	0.25	7.57e-6	6.07
400096	1.66e-3	0.40	9.12e-4	45.6	7.26e-5	0.50	2.52e-6	10.63

Таблица 1.13: Относителна грешка за 5-мерния интеграл при фиксирано изчислително време

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



Фигура 1.4: Относителна грешка и изчислително време за 5-мерния интеграл.

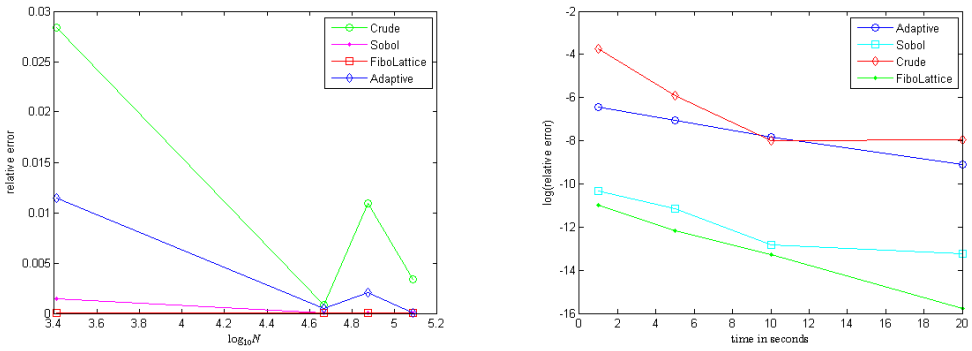
сочи за един от най-добрите квази-Монте Карло алгоритми. Реализиран е Монте Карло метод, базиран на извадката латински хиперкуб, който има същото бързодействие като обикновения метод Монте Карло, но дава много по-точни

Таблица 1.14: Относителна грешка за 20-мерния интеграл

брой точки	обикновен	T	адаптивен	T	FIBO	T	Собол	T
2048	2.84e-2	0.02	1.14e-2	8.6	8.22e-5	0.03	8.44e-4	0.13
16384	8.23e-4	0.12	4.96e-4	60.3	3.12e-5	0.13	6.82e-5	1.68
65536	8.61e-3	0.91	9.75e-4	474.2	1.36e-5	1.17	8.34e-6	8.69
131072	4.13e-4	2.13	1.25e-5	888.3	8.85e-6	2.34	3.77e-6	14.36
524288	1.22e-4	8.13	1.96e-6	2356	2.15e-6	8.34	1.91e-7	57

Таблица 1.15: Относителна грешка за 20-мерния интеграл при фиксирано изчислително време

време в сек.	обикновен МК	адаптивен МК	FIBO	Собол
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



Фигура 1.5: Относителна грешка и изчислително време за 20-мерния интеграл.

резултати. Направено е детайлно сравнение на различните методи за гладки подинтегрални функции с различни размерности. Направени са изводите кой от методите е за препоръчване при малки и големи размерности. При средни

размерности разглежданите квази-Монте Карло методи показват сходни резултати. Приложени са таблици с относителната грешка при отнапред зададен брой реализации. Направено е и по-задълбочено изследване за относителната грешка при отнапред зададено изчислително време, което е мярка за изчислителната сложност. Направено е приложение на разглежданите методи за оценка на опции във финансите и за многомерни интеграли, които възникват в Бейсовската статистика. Най-важното приложение е за ядрото на Вигнер, където са посочени възможни решения на задачата на Ричард Файнман за съществуване на алгоритъм с линейна или полиномиална изчислителна сложност за многомерния интеграл, описващ ядрото на Вигнер. Резултатите, получени с адаптивния алгоритъм са по-добри от останалите разглеждани стохастични методи, които пък от своя страна са по-добри от досега използваните детерминистични методи за пресмятане на ядрото. Резултатите от проведените числени експерименти могат да се обобщят така:

- И петте подхода са приложими за конкретния клас от функции.
- Получените числени резултати потвърждават теоретичните оценки и са очаквани.
- Квази-Монте Карло методът, базиран на точково множество от тип решетки с обобщената редица на Фибоначи има предимство при гладки подинтегрални функции заради бързината си при малки и средни размерности. За големи размерности $s > 20$ започва да губи точност.
- Адаптивният алгоритъм има предимство при подинтегрални функции с особености като ядрото на Вигнер или тестовите функции на Генц, като с увеличаване на размерността предимствата му пред обикновения метод Монте Карло стават по големи, дори за гладки подинтегрални функции.
- Методът, базиран на извадката латински хиперкуб дава най-добри резултати за големи размерност $s = 30$, като експериментите показват, че е по-точен при отнапред задени реализации от квазислучайната редица на Собол и от адаптивния алгоритъм Монте Карло.

- По отношение на изчислителна сложност най-бърз е квази-Монте Карло алгоритъма, който използва точкови множества от тип решетки заедно с извадката латински хиперкуб, а адаптивният алгоритъм е най-бавен, защото се основава на рекурсивно разделяне на областта, използвайки апостериорна информация за грешката при текущото разделяне.

- Представени са различни методи Монте Карло и квази-Монте Карло, които дават по-добри резултати от досега използваните детерминистични методи за пресмятане на ядрото на Вигнер. За пръв път е използван адаптивен подход за ядрото и се оказва, че той има най-добра точност в сравнение с останалите алгоритми и е едно възможно решение на проблема поставен от Ричард Файнман за намиране на ефективен алгоритъм за пресмятане на ядрото на Вигнер с големи размерности. Трябва да се отбележи, че метода, базиран на извадката латински хиперкуб също е приложим за проблема и дава много добри резултати.

- Реализираните методи са използвани за оценка на Европейски опции във финансите и за пресмятане на многомерни интеграли в Бейсовската статистика с приложение в изчислителната биоинформатика и машинното обучение.

Глава 2

Алгоритми Монте Карло за интегрални уравнения и линейни системи

2.1 Алгоритми Монте Карло за интегрални уравнения

2.1.1 Постановка на задачата

Нека е дадено е интегрално уравнение на Фредхолм от втори род:

$$u(x) = \int_\Omega k(x,x')u(x')\mathrm{d}x' + f(x) \tag{2.1.1}$$

или

$$u = \mathcal{K}u + f \ \ (\mathcal{K} \text{ е интегрален оператор}),$$

където $k(x,x') \in L_2(\Omega \times \Omega), f(x) \in L_2(\Omega)$ са дадени функции и $u(x) \in L_2(\Omega)$ е неизвестна функция, $x, x' \in \Omega \subset \mathbb{R}^d$ (Ω е ограничена област).

Целта е конструиране и изследване на нов алгоритъм Монте Карло за пресмятане на линейни функционали от решението, зададени по следния начин:

$$J(u) = \int \varphi(x)u(x)\mathrm{d}x = (\varphi, u). \tag{2.1.2}$$

Предполага се, че $\varphi(x) \in L_2(\Omega)$. В много задачи от статистическата физика се налага пресмятането на линеен функционал от типа (2.1.2) от решението

на уравнение на Boltzmann, Wigner или Schrödinger, например определяне на вероятността за попадането на частица във фиксирана точка от пространството или в даден момент от времето (интеграл от решението), определяне на средна скорост на частиците (първи интегрален момент на скоростта) или енергията (втори интегрален момент на скоростта).

Приложен е метод на последователните приближения:

$$u^{(k)} = \sum_{j=0}^{k} \mathcal{K}^{(j)} f = f + \mathcal{K}f + \ldots + \mathcal{K}^{(k-1)} f + \mathcal{K}^{(k)} u^{(0)}, \quad k = 1, 2, \ldots, \tag{2.1.3}$$

където $u^{(0)}(x) \equiv f(x)$. Известно е [114], че следното неравенство $\|\mathcal{K}\|_{L_2} < 1$ представлява достатъчно условие за сходимостта на реда на Neumann.

Следователно, когато това условие е удовлетворено, е в сила следното свойство:

$$u^{(k)} \longrightarrow u, \quad \text{когато} \quad k \to \infty.$$

Следователно

$$J(u) = (\varphi, u) = \lim_{k\to\infty} (\varphi, u^{(k)}) = \lim_{k\to\infty} \left(\varphi, \sum_{j=0}^{k} \mathcal{K}^{(k)} f \right) = \lim_{k\to\infty} \sum_{j=0}^{k} \left(\varphi, \mathcal{K}^{(j)} f \right).$$

Приближение на неизвестната стойност (φ, u) може да се получи като се използва „отрязан“ ред на Neumann (2.1.3) при достатъчно голямо k (в зависимост от нормата на интегралния оператор):

$$(\varphi, u^{(k)}) = (\varphi, f) + (\varphi, \mathcal{K}f) + \ldots + (\varphi, \mathcal{K}^{(k-1)} f) + (\varphi, \mathcal{K}^{(k)} f).$$

Най-напред, 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:

$$T_k : x_0 \longrightarrow x_1 \longrightarrow \ldots \longrightarrow x_k.$$

Функциите $\pi(x)$ и $p(x, x')$ удовлетворяват условията за неотрицателност и за нормираност:

$$\int_\Omega \pi(x) \mathrm{d}x = 1, \quad \int_\Omega p(x, x') \mathrm{d}x' = 1 \ \text{ за всяко } x \in \Omega \subset \mathbb{R}^d.$$

Освен това са допустими съответно за функцията $\varphi(x)$ и ядрото $k(x,x')$. От направените предположения за ядрото $k(x,x')$ и свойствата на математическото очакване

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

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

следва, че съответната оценка по метод Монте Карло 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.

Важно е да се отбележи, че траекториите от същия тип T_k могат да се използват за приближеното пресмятане на функционала $(\varphi, u^{(k)})$ за различни функции $\varphi(x)$, както и за различни интегрални уравнения със същото ядро $k(x,x')$, но с различна дясна част $f(x)$.

Ако предположим [194]

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

тогава алгоритъма се нарича почти оптимален Монте Карло алгоритъм (МАО) и това гарантира намаляване на дисперсията.

2.1.2 Балансиране на грешката

Има два класа алгоритми. Първите са изместените алгоритми, когато се търси случайна величина, чието математическо очакване е равно на приближеното решение на проблема с помощта на „отрязания“ ред на Liouville-Neumann (2.1.3) за достатъчно голямо k. Неизместените оценки допускат, че формулираната случайна величина е такава, че нейното математическо очакване приближава точното решение на проблема. Очевидно първият клас от методи Монте

Карло се характеризират с два типа грешки – систематична (грешка от „отрязване“, „truncation error“) r_k и вероятностна (стохастична, „probability error“) r_N, която зависи от броя на реализациите на случайната величина или от броя вериги, който използваме за намиране на приближението. В случая на изместените стохастични методи трябва да се извърши допълнителен анализ на грешката: балансиране на систематичната и стохастичната грешка, за да се минимизира изчислителната сложност на алгоритъма. Разглеждаме задачата за конструиране на нов балансиран МАО алгоритъм за интегрални уравнения. Балансирането на грешките спомага за повишаването на точността на алгоритъма, ако е фиксирана изчислителната сложност, или за намаляването на изчислителната сложност, ако е фиксирана грешката на алгоритъма (вж. [50]).

Систематичната грешка зависи от броя на итерациите и собствените стойности на оператора, докато стохастичната грешка зависи от вероятностната природа на алгоритъма. За да получим добри резултати трябва стохастичната грешка r_N да бъде приблизително равна на систематичната грешка r_k или да е изпълнено

$$r_N = O(r_k).$$

Проблема за балансиране на грешките е тясно свързан със задачата за получаване на оптимално съотношение между броя на реализациите N на случайната величина и средния брой стъпки k във всяка случайна траектория.

Оценка на вероятната грешка

За вероятната грешка r_N е в сила, че [50] $r_N \leq 0.6745\sigma(\theta)\frac{1}{\sqrt{N}}$, където $\sigma(\theta) = (D\theta)^{1/2}$ е стандартното отклонение на случайната величина θ, за която

$$E\theta_k[\varphi] = \left(\varphi, u^{(k)}\right) = \sum_{j=0}^{k} (\varphi, \mathcal{K}^{(j)} f).$$

За $x = (x_0, ..., x_k) \in G \equiv \Omega^{k+1} \subset \mathbb{R}^{d(k+1)}$ имаме:

$$\begin{aligned}(\varphi, \mathcal{K}^{(j)} f) &= \int_\Omega \varphi(x_0)\mathcal{K}^{(j)} f(x_0)dx_0 = \\ &= \int_G \varphi(x_0)k(x_0, x_1)\dots k(x_{k-1}, x_j)f(x_j)dx_0dx_1\dots dx_j = \int_G F(x)dx,\end{aligned}$$

като

$$F(x) = \varphi(x_0)k(x_0, x_1)\dots k(x_{k-1}, x_j)f(x_j),\ x \in G \subset \mathbb{R}^{d(j+1)}$$

и N е броят на реализациите на случайната величина. Като вземем предвид, че

$$D\sum_{j=0}^{k}\theta_k^{(j)} \leq \left(\sum_{j=0}^{k}\sqrt{D\theta_k^{(j)}}\right)^2,$$

и използвайки неравенствата:

$$r_N \leq \frac{0.6745}{\sqrt{N}}\sum_{j=0}^{k}\left(\int_G \left(\mathcal{K}^{(j)}\varphi f\right)^2 p dx - \left(\int_G \mathcal{K}^{(j)}\varphi f p dx\right)^2\right)^{1/2} \leq$$
$$\leq \frac{0.6745}{\sqrt{N}}\sum_{j=0}^{k}\left(\int_G \left(\mathcal{K}^{(j)}\varphi f\right)^2 p dx\right)^{1/2} = \frac{0.6745}{\sqrt{N}}\|\varphi\|_{L_2}\|f\|_{L_2}\sum_{j=0}^{k}\left\|\mathcal{K}^{(j)}\right\|_{L_2},$$

получаваме, че

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

В този случай оценката в неравенството включва L_2 нормата на подинтегралната функция.

Оценка на систематичната грешка

Разглеждаме редицата $u^{(1)},\ u^{(2)}, \dots,$ дефинирана рекурсивно чрез формулата

$$u^{(k)} = \mathcal{K}u^{(k-1)} + f, k = 1, 2, \dots.$$

Формалното решение на (2.1.1) е „отрязания“ ред на Нойман

$$u^{(k)} = f + \mathcal{K}f + \dots + \mathcal{K}^{(k-1)}f + \mathcal{K}^{(k)}u^{(0)},\ k > 0,$$

където $\mathcal{K}^{(k)}$ означава k-та итерация на $\mathcal{K}$,

$$u^{(k)} = \sum_{i=0}^{k-1}\mathcal{K}^{(i)}f + \mathcal{K}^{(k)}u^{(0)}.$$

Дефинираме k – резидуал $r^{(k)}$. По дефиниция: $r^{(k)} = u^{(k+1)} - u^{(k)},\ r^{(k+1)} = \mathcal{K}r^{(k)}$. След преобразувания:

$$r^{(k)} = f - (I - \mathcal{K})\,u^{(k)} = (I - \mathcal{K})\left(u - u^{(k)}\right),$$

и

$$f - u^{(k)} + Ku^{(k)} = u - u^{(k)} + Ku^{(k)} - Ku = u - u^{(k)} - K\left(u - u^{(k)}\right) = (I - K)\, r_k.$$

От дефиницията на $r^{(k)}$:

$$r^{(k)} = f - u^{(k)} + \mathcal{K}u^{(k)} = u^{(k+1)} - u^{(k)}$$

и

$$r^{(k+1)} = u^{(k+2)} - u^{(k+1)} = \mathcal{K}u^{(k+1)} + f - \mathcal{K}u^{(k)} - f = \mathcal{K}\left(u^{(k+1)} - u^{(k)}\right) = \mathcal{K}r^{(k)}.$$

За $u^{(0)} := f$ получаваме, че:

$$r^{(0)} = u^{(1)} - u^{(0)} = \mathcal{K}u^{(0)} + f - u^{(0)} = \mathcal{K}f,$$

като

$$r^{(k+1)} = \mathcal{K}r^{(k)} = \mathcal{K}^{(2)}r^{(k-1)} = \cdots = \mathcal{K}^{(k+1)}r^{(0)}.$$

От горното получаваме [46], че:

$$u^{(k+1)} = u^{(k)} + r^{(k)} = u^{(k-1)} + r^{(k-1)} + r^{(k)} = \cdots = u^{(0)} + r^{(0)} + \cdots + r^{(k)} =$$
$$= u^{(0)} + r^{(0)} + \mathcal{K}r^{(0)} + \mathcal{K}^{(2)}r^{(0)} + \cdots + \mathcal{K}^{(k)}r^{(0)} = u^{(0)} + \left(I + \mathcal{K} + \cdots + \mathcal{K}^{(k)}\right) r^{(0)}.$$

Ако е изпълнено $\|\mathcal{K}\|_{L_2} < 1$, тогава редът на Нойман $u = \sum\limits_{i=0}^{\infty} \mathcal{K}^{(i)} f$ е сходящ и $u^{(k+1)} \xrightarrow{k\to\infty} u$, следователно от

$$u^{(k+1)} = u^{(0)} + \left(I + \mathcal{K} + \cdots + \mathcal{K}^{(k)}\right) r^{(0)} \quad \text{при} \quad k \to \infty$$

получаваме $u = u^{(0)} + (I - \mathcal{K})^{-1} r^{(0)}$. След тривиални преобразувания:

$$u = \mathcal{K}u + f = \mathcal{K}u^{(0)} + \mathcal{K}(I - \mathcal{K})^{-1} r^{(0)} + f = u^{(1)} + \mathcal{K}(I - \mathcal{K})^{-1} r^{(0)}.$$

Повтаряме k пъти тази процедура: $u = u^{(k)} + \mathcal{K}^{(k)}(I - \mathcal{K})^{-1} r^{(0)}$. Използваме неравенството на Кощи-Буняковски-Шварц:

$$\left\|u - u^{(k)}\right\|_{L_2} \le \frac{\|\mathcal{K}\|_{L_2}^{k} \left\|r^{(0)}\right\|_{L_2}}{1 - \|\mathcal{K}\|_{L_2}} \le \frac{\|\mathcal{K}\|_{L_2}^{k} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}}{1 - \|\mathcal{K}\|_{L_2}} = \frac{\|\mathcal{K}\|_{L_2}^{k+1} \|f\|_{L_2}}{1 - \|\mathcal{K}\|_{L_2}}.$$

Накрая получаваме следната оценка за систематичната грешка при приближено пресмтане на линеен функционал от решението на интегралното уравнение (2.1.1):

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

2.1.3 Теорема за балансираност

В тази секция се изведен основният резултат, който решава задачата за намиране на оптимално съотношение между броя на реализациите на случайната величина и броя на случайните траектории (скокове) във веригата на Марков. Получени са теореми и следствия за връзката между двата основни параметъра в алгоритъма - броя на реализациите N и броя на итерациите k. Новият алгоритъм е базиран на по-долните твърдения, които играят важна роля за изчислителната сложност на алгоритъма.

Да допуснем, че

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

Поставяме условие за приблизително равенство на трите израза:

$$r_N \approx r_k \approx \frac{\delta}{2}$$

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

Теорема 2.1.1 (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:*

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

Ако δ се изрази от неравенството за N по-горе и се замести в оценката за k се получава следното съотношение между двете грешки:

Теорема 2.1.2 (Следствие за оптималното съотношение)**.** *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:*

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

Може да се докаже, че двете долни граници за k са еквивалентни.

Може да се формулират и следните допълнителни твърдения, които са от особено важно значение за изчислителната сложност на разработения алгоритъм Монте Карло за интегрални уравнения, базиран на балансиране на систематичната и стохастичната грешка.

Теорема 2.1.3 (Следствие за оптималното съотношение)**.** *Ако първо изберем k да бъде близко до своята долна граница, по аналогичен начин получаваме долната граница за N:*

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

Аналогично се вижда, че двете долни граници за N са еквивалентни.

От следствията за оптималното съотношение между двете грешки може да получим по друг начин точна оценка за броя на реализациите и броя на итерациите. Получаваме, че може да се пресметне точно N и k с помощта на следните изрази:

$$N = \left\lceil \left(\frac{1.349 \|\varphi\|_{L_2} \|f\|_{L_2}}{\delta \left(1 - \|\mathcal{K}\|_{L_2}\right)} \right)^2 \right\rceil, \; k = \left\lceil \frac{\ln \dfrac{0.6745}{\|\mathcal{K}\|_{L_2} \sqrt{N}}}{\ln \|\mathcal{K}\|_{L_2}} \right\rceil. \tag{2.1.4}$$

$$k = \left\lceil \frac{\ln \dfrac{\delta\left(1 - \|\mathcal{K}\|_{L_2}\right)}{2\|\varphi\|_{L_2} \|f\|_{L_2} \|\mathcal{K}\|_{L_2}}}{\ln \|\mathcal{K}\|_{L_2}} \right\rceil, \; N = \left\lceil \frac{0.455}{\|\mathcal{K}\|_{L_2}^{2k+2}} \right\rceil. \tag{2.1.5}$$

2.1.4 Приложения и числени експерименти

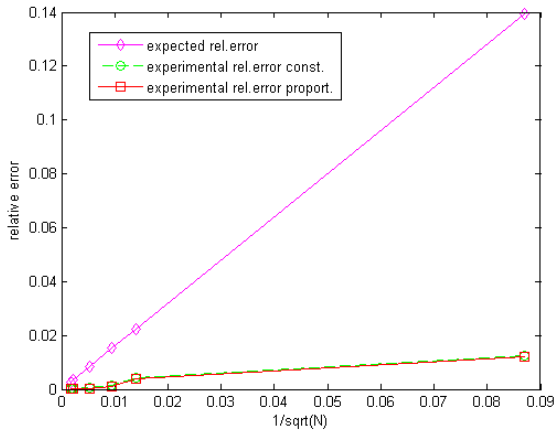
Приложение в биологията

Първият пример има важно значение в биологията. Разглеждаме уравнението, описващо популационен модел в биологията [71]:

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

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: $\|\varphi\|_{L_2} = 1$, $\|\mathcal{K}\|_{L_2} = 0.3917$, $\|f\|_{L_2} = 1.1915$.



Фигура 2.1: Експериментална и очаквана относителна грешка.

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.

Резултатите са дадени в Таблица 2.1.

В Таблица 2.1 оценяваме 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. Сравняваме почти оптималния алгоритъм Монте Карло за интегрални уравнения, при който са избрани съответно плътности, пропорционални на ядрото и на функцията от линейния функционал и алгоритъма за интегрални уравнения

Таблица 2.1: 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(МАО) 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

Таблица 2.2: 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

с константни плътности. Седмата и осма колона са за експерименталната относителна грешка с МАО 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 потвърждава получената експериментална грешка. Може да се отчете предимство на алгоритъма МАО в сравнение с метода с константи плътности.

Приложение в невронните мрежи

Разглеждания пример има важно значение в невронните мрежи и изкуствения интелект. Разглеждаме следния пример описващ процеса на обучение на невронни мрежи [4]:

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

където $\Omega \equiv [-2, 2]$, $k\left(x, x'\right) = \frac{0.055}{1+e^{-3x}} + 0.07$, $f\left(x\right) = 0.02\left(3x^2 + e^{-0.35x}\right)$, $\varphi(x) = 0.7((x+1)^2\cos(5x) + 20)$.

Точното решение е 8.98635750518, като $\|\varphi\|_{L_2} = 27.7782$, $\|\mathcal{K}\|_{L_2} = 0.2001$, $\|f\|_{L_2} = 0.2510$.

Резултатите са представени в Таблица 2.2.

Лесно може да се види, че баланисрания МАО алгоритъм дава много подобри резултати от Монте Карло алгоритъма с константни плътности за големи стойности на N и k. При малък брой реализации методът 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, което определя преходните плътности да са константи.

В обобщение може да се каже, че разработеният МАО алгоритъм с балансирани систематична и стохастична грешка дава по-надеждни резултати от балансирания метод Монте Карло с константни плътности при условие, че началната плътност е различна от δ-функцията. Когато началната плътност е δ-функцията двата алгоритъма дават близки резултати, но отново МАО има предимство (дори в изчислителното време).

Приложение във физиката

Интересно е да се види дали предложеният 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 ([58]), може да се използва конструирания балансиран алгоритъм.

Следващият пример е от основно значение при моделирането на процеси във физиката [50]. Разглеждаме следното интегрално уравнение с полиномиална нелинейност, описващо процеса на взаимодействие между две твърди физични тела:

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

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

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

Едно достатъчно условие за сходимост на процеса е:

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

Таблица 2.3: 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

Съгласно Теоремата на Миранда [152, 215] интегралното уравнение има единствено регулярно решение $u(x) = c$, когато е изпълнено следното условие:

$$c = \pm \left(\frac{6a_1}{a_3 \left(2a_2^2 - 3a_2 + 2\right)} \right)^{\frac{1}{2}}.$$

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 дадени в Таблица 2.3. 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 се направи заключение, че конструираният почти оптимален алгоритъм Монте Карло за интегрални уравнения, базиран на балансиране на систематичната и стохастичната грешка, е приложим и за нелинейни интегрални уравнения.

2.1.5 Заключение

В настоящата секция са постигнати следните резултати:

- Разработен е нов Монте Карло метод, базиран на балансиране на систематичната и стохастичната грешка, като са получени долни граници за N и k.

- Сравнени са Монте Карло алгоритми с различни подходи за избор на начални и преходни плътности, като експерименталната относителна грешка потвърждава очакваната относителна грешка.

- Показано е, че почти оптималният Монте Карло алгоритъм, базиран на балансиране на систематичната и стохастичната грешка, дава по добри резултати от Монте Карло метода с константни плътности, като резултатите са най-добри, когато $\varphi(x) \neq \delta(x)$.

- Алгоритъмът се оказва ефективен дори за такъв клас задачи, които имат фундаментално значение за популационни модели в биологията и за описване на процеса на обучение на невронни мрежи.

- Посредством конструираният почти оптимален алгоритъм Монте Карло, базиран на балансиране на систематичната и стохастичната грешка, е пресметнат линеен функционал от решението на нелинейното интегрално уравнение с втори ред на нелинейност, което описва взаимодействието между две тела и намира широко приложение във физиката и химията.

2.2 Алгоритми Монте Карло за линейни системи

2.2.1 Постановка на задачата: решаване на система от линейни алгебрични уравнения

Разглеждаме следната система от линейни уравнения (СЛАУ):

$$Bx = f, \quad B \in \mathbb{R}^{n\times n}; \;\; f, x \in \mathbb{R}^{n\times 1}, \tag{2.2.1}$$

където $B = \{b_{ij}\}_{i,j=1}^{n} \in \mathbb{R}^{n\times n}$ е дадена матрица; $f = (f_1, \ldots, f_n)^t \in \mathbb{R}^{n\times 1}$ и $v = (v_1, \ldots, v_n) \in \mathbb{R}^{1\times n}$ са дадени вектори. Използваме матрицата $A = \{a_{ij}\}_{ij=1}^{n}$, такава че $A = I - DB$, където D е диагоналната матрица $D = \mathrm{diag}(d_1, \ldots, d_n)$ и $d_i = \frac{\gamma}{b_{ii}}$, $i = 1, \ldots, n$, и $\gamma \in (0, 1]$ е параметър, който се избира за ускоряване на сходимостта. Да предположим, че матрицата B има преобладаващ главен диагонал. Очевидно, ако B е такава матрица, тогава елементите на матрицата A трябва да удовлетворяват следните условия: $\sum_{j=1}^{n} |a_{ij}| \leq 1, \qquad i = 1, \ldots, n.$ Системата (2.2.1) може да се представи като уравнение във вида:

$$x = Ax + b, \tag{2.2.2}$$

където $b = Df$. Предполага се, че

$$(i) \qquad \begin{cases} 1. \;\; \text{Матриците } D \;\; \text{и} \;\; A \;\; \text{са несингулярни;} \\ 2. \;\; |\lambda(A)| < 1 \;\; \text{за всички собствени стойности} \;\; \lambda(A) \;\; \text{на} \;\; A, \end{cases}$$

т.е. всички стойности $\lambda(A)$ на матрицата A, за които $Ay = \lambda(A)y$. Ако условието (i) е изпълнено, тогава се задава *стационарен линеен итерационен процес от първи ред за системата 2.2.2* [58, 50]:

$$x_k = Ax_{k-1} + b, \quad k = 1, 2, \ldots, \tag{2.2.3}$$

който дефинира следния ред на Нойман, чрез който се представя решението x:

$$x = \sum_{k=0}^{\infty} A^k b = b + Ab + A^2 b + A^3 b + \ldots \tag{2.2.4}$$

В резултат, сходимостта на Монте Карло алгоритъма зависи от грешката от прекъсване на (2.2.4) (вж, [50]).

Анализ на условията за сходимост на Монте Карло алгоритмите за линейни системи, основани на стационарни линейни итерационни методи, е представен в две публикации [110, 201]. Да допуснем, че се интересуваме от пресмятането на линейната форма $V(x)$ на решението x на системата (2.2.2), т.е., $V(x) \equiv (v,x) = \sum_{i=1}^{n} v_i x_i$, където $v \in \mathbb{R}^{n\times 1}$. Ще дефинираме случайна величина $X[v]$, чието математическо очакване е равно на горе дефинираната линейна форма, т.е., $EX[v] = V(x)$, използвайки дискретна верига на Марков с краен брой състояния. В такъв случай задачата се свежда до пресмятане на повтарящи се реализации на $X[v]$ и комбинирането им в подходяща статистическа оценка за $V(x)$.

Разглеждаме вектор на началните вероятности $p = \{p_i\}_{i=1}^{n} \in \mathbb{R}^n$, такъв че $p_i \geq 0, i = 1,\ldots,n$ и $\sum_{i=1}^{n} p_i = 1$. Разглеждаме също матрица на преходните вероятности $P = \{p_{ij}\}_{i,j=1}^{n} \in \mathbb{R}^{n\times n}$, такава че $p_{ij} \geq 0,\ i,j = 1,\ldots,n$ и $\sum_{j=1}^{n} p_{ij} = 1$, за $i = 1,\ldots,n$. Дефинираме множество от *допустими* вероятности $\mathcal{P}_b$ и $\mathcal{P}_A$ за системата 2.2.2.

Векторът на началните вероятности $p = \{p_i\}_{i=1}^{n}$ се нарича допустим за вектора $v = \{v_i\}_{i=1}^{n} \in \mathbb{R}^n$, i.e. $p \in \mathcal{P}_b$, ако

$$\begin{cases} p_{\alpha_s} > 0, & \text{когато } v_{\alpha_s} \neq 0, \\ p_{\alpha_s} = 0, & \text{когато } v_{\alpha_s} = 0. \end{cases} \tag{2.2.5}$$

Аналогично, матрицата на преходните вероятности $P = \{p_{ij}\}_{i,j=1}^{n}$ се нарича *допустима* за матрицата $A = \{a_{ij}\}_{i,j=1}^{n}$, т.е. $P \in \mathcal{P}_A$, ако

$$\begin{cases} p_{\alpha_{s-1},\alpha_s} > 0, & \text{когато } a_{\alpha_{s-1},\alpha_s} \neq 0, \\ p_{\alpha_{s-1},\alpha_s} = 0, & \text{когато } a_{\alpha_{s-1},\alpha_s} = 0. \end{cases} \tag{2.2.6}$$

Ще използваме само *допустими* вероятности.

Очевидно е, че по такъв начин случайните траектории, конструирани за решаването на задачата, никога не посещават *нулевите елементи* на матрицата. По такъв начин се намалява изчислителната сложност на алгоритъма. Този подход е много подходящ и при решаване на задачи с големи разредени матрици. Да предположим, че имаме верига на Марков:

$$T = \alpha_0 \to \alpha_1 \to \ldots \to \alpha_k \to \ldots, \tag{2.2.7}$$

със състояния $\alpha_0, \alpha_1, \ldots, \alpha_k, \ldots$. Една случайна траектория (верига) T_k с дължина k, стартираща със състоянието α_0 се дефинира като:

$$T_k = \alpha_0 \to \alpha_1 \to \ldots \alpha_j \to \ldots \alpha_k, \tag{2.2.8}$$

където α_j означава номера на избраното състояние за $j = 1, \ldots, n$. Да допуснем, че

$$P(\alpha_0 = \alpha) = p_\alpha \quad \text{и} \quad P(\alpha_j = \beta | \alpha_{j-1} = \alpha) = p_{\alpha\beta}, \tag{2.2.9}$$

където p_α е вероятността веригата да започва в състояние α и $p_{\alpha\beta}$ е преходната вероятност за преход от състояние α в състояние β. Вероятностите $p_{\alpha\beta}$ дефинират преходната матрица P, за която са в сила следните условия:

$$\sum_{\alpha=1}^{n} p_\alpha = 1 \quad \sum_{\beta=1}^{n} p_{\alpha\beta} = 1 \quad \alpha = 1, \ldots, n. \tag{2.2.10}$$

Горното позволява конструирането на следната случайна величина, която е неизместена оценка за $V(x)$ ([59]):

$$X[v] = \frac{v_{\alpha_0}}{p_0} \sum_{m=0}^{\infty} Q_m v_{\alpha_m} \ , \tag{2.2.11}$$

където

$$Q_0 = 1; \qquad Q_m = Q_{m-1} \frac{a_{\alpha_{m-1},\alpha_m}}{p_{\alpha_{m-1},\alpha_m}}, \qquad m = 1, 2, \ldots \tag{2.2.12}$$

са тегла и $\alpha_0, \alpha_1, \ldots$ е верига на Марков с елементи на матрицата A, конструирана с използването на началните вероятности p_α и преходните вероятности $p_{\alpha_{m-1},\alpha_m}$ за избиране на елемента $a_{\alpha_{m-1},\alpha_m}$ от матрицата A.

Един възможен избор на вероятности е следният:

$$p = \{p_\alpha\}_{\alpha=1}^{n} \in \mathcal{P}_b, \quad p_\alpha = \frac{|v_\alpha|}{\| \ v \ \|};$$

и

$$P = \{p_{\alpha\beta}\}_{\alpha,\beta=1}^{n} \in \mathcal{P}_A, \quad p_{\alpha\beta} = \frac{|a_{\alpha\beta}|}{\| \ a_\alpha \ \|}, \alpha = 1, \ldots, n. \tag{2.2.13}$$

Такъв избор на вектора на началните вероятности и на матрицата на преходните вероятности води до почти оптимален Монте Карло алгоритъм (МАО) ([50, 51, 58, 64]).

Такъв избор на вектора на началните вероятности и на матрицата на преходните вероятности води до почти оптимални Монте Карло алгоритми в класа на матриците A и векторите h (МАО) ([50, 51, 58, 64]), които съвпадат с оптималните теглови алгоритми, дефинирани в [75] и описани в [149] (повече подробности има в [51]). В частност, МАО алгоритъмът става оптимален (с нулева дисперсия), ако всички елементи на вектора в дясната част на линейната система са равни (виж, [51]). От друга страна, оптималните алгоритми изискват много време, тъй като за да се дефинират оптималните вероятности са необходими допълнителни пресмятания, еквивалентни с оригиналната задача. Това прави процедурата много скъпа. Известни са и други подходи, но те изискват по-голяма изчислителна сложност за линейни системи [124, 146]. Това е причината да се използва МАО алгоритъма.

Забележка 2.2.2. Ясно е, че ако се използва специалния избор на линейната форма $V(x)$, съответстваща на вектора $v = e_i = (0, 0, \ldots, \underbrace{1}_{i}, 0, \ldots, 0)$, където на i-тото място се взема 1, ще получим i-тата компонента на решението x_i. За алгоритъма, описан по-долу, се прави нещо допълнително: брои се номера на посещенията на всяко уравнение и така се изчисляват всички компоненти на решението x.

Малка модификации на $X[v]$ позволява да се пресметне обратната матрица. Наистина, за да се пресметне обратната матрица $G = B^{-1}$ на $B \in \mathbb{R}^{n\times n}$, се допуска, че B е несингулярна, и $||\lambda(B)| - 1| < 1$ за всички собствени стойности $\lambda(B)$ на B и *итерационната* матрица е $A = I - B$. Тогава обратната матрица може да се представи като $G = \sum_{i=0}^{\infty} A^i$. Очевидно,

$$E\left\{\sum_{i|k_i=r'} Q_i\right\} = g_{rr'},$$

където $(i|k_i = r')$ означава сумиране само за теглата Q_i, за които $k_i = r'$ и $G = \{g_{rr'}\}_{r,r'=1}^{n}$.

Друга малка модификация на $X[v]$ позволява да се пресметнат първите $k \leq n$ собствени стойности на реална симетрична матрица, понеже в този случай собствените стойности са реални числа ([64]). Това показва, че използваният подход Монте Карло за линейни системи може да се приложи за по-широк клас от алгебрични задачи.

Да разгледаме Монте Карло алгоритъм със *състояния на поглъщане*: вместо крайна случайна траектория T_i в алгоритъма ние разглеждаме безкрайна траектория с координати на състоянието δ_i $(i = 1, 2, \dots)$. Да допуснем, че $\delta_i = 0$, ако траекторията се прекъсва в състояние α_i и $\delta_i = 1$ в останалите случаи. Нека

$$\Delta_i = \delta_0 \times \delta_1 \times \cdots \times \delta_i.$$

Така $\Delta_i = 1$ до първото състояние на прекъсване и $\Delta_i = 0$ след това. Вероятностите за попадане в абсорбиращо (поглъщащо) състояние се избират по специален начин, различни са за различните редове на матрицата и зависят от $\sum_{j=1}^{n} |a_{ij}|$. В този случай веригата на Марков има $n + 1$ състояния, $\{1, \dots, n, n + 1\}$. Матрицата на преходните вероятности (2.2.10) е такава, че $\sum_{\beta=1}^{n} p_{\alpha\beta} \leq 1$ и вероятността за поглъщане на всяко състояние, с изключение на първото, е $p_{\alpha,n+1} = p_\alpha = 1 - \sum_{\beta=1}^{n} p_{\alpha\beta}, \alpha = 1, \dots, n$.

2.2.2 Описание на алгоритъма

Разглеждаме реална система $x = Ax + b$, където матрицата A с размер $n \times n$ е такава, че нейният радиус на сходимост е $\varrho(A) < 1$ и нейните коефициенти $a_{i,j}$ са реални числа и

$$\sum_{j=1}^{n} |a_{i,j}| \leq 1, \forall 1 \leq i \leq n.$$

Дефинираме верига на Марков T_k с $n+1$ състояния $\alpha_1, \dots, \alpha_n, n+1$, такива че

$$P(\alpha_{k+1} = j | \alpha_k = i) = \frac{|a_{i,j}|}{\sum_{j=1}^{n} |a_{ij}|},$$

ако $i \neq n + 1$ и

$$P(\alpha_{k+1} = n + 1 | \alpha_k = n + 1) = 1.$$

Дефинираме вектор 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 с начално състояние $\alpha_0 < n + 1$ и минава през $(\alpha_1, \dots, \alpha_k)$ до състоянието на поглъщане $\alpha_{k+1} = n + 1$. Вероятността за осъществяване на траекторията τ е

$$P(\tau) = p_{\alpha_0} p_{\alpha_0\alpha_1}, \dots p_{\alpha_{k-1,k}\alpha_k} p_{\alpha_k}.$$

Използваме МАО алгоритъм, дефиниран чрез (2.2.13), с вектор на началните вероятности $p = \{p_\alpha\}_{\alpha=1}^n$ и с матрица на преходните вероятности $P = \{p_{\alpha\beta}\}_{\alpha,\beta=1}^n$. Теглата Q_α са дефинирани по същия начин като в (2.2.12), т.е.

$$Q_m = Q_{m-1}\frac{a_{\alpha_{m-1},\alpha_m}}{p_{\alpha_{m-1},\alpha_m}}, \quad m = 1,\ldots,k, \quad Q_0 = \frac{c_{\alpha_0}}{p_{\alpha_0}}. \tag{2.2.14}$$

Оценката $X_\alpha(\tau)$ може да се представи като $X_\alpha(\tau) = c_\alpha + Q_k\frac{a_{\alpha_k\alpha}}{p_{\alpha_k}}$, $\alpha = 1,\ldots n$, взета с вероятност $P(\tau) = p_{\alpha_0}p_{\alpha_0\alpha_1},\ldots p_{\alpha_{k-1,k}\alpha_k}p_{\alpha_k}$. Димов, Мер и Селие [62] доказват следната теорема:

Теорема 2.2.1. *Случайната величина $X_\alpha(\tau)$ е неизместена оценка за x_α, т.е.*

$$E\{X_\alpha(\tau)\} = x_\alpha.$$

Да отбележим, че нито един от елементите $c_{\alpha_0}, a_{\alpha_0\alpha_1}\ldots a_{\alpha_{k-1}\alpha_k}$ не е нула, поради специалния избор на *допустимите* вероятности p_α и $p_{\alpha\beta}$, дефинирани чрез МАО плътностите (2.2.13). Условията (2.2.13) осигуряват, че веригата на Марков ще *посещава* само ненулевите елементи. Да допуснем, че можем да пресметнем N стойности на случайната величина X_α, именно $X_{\alpha,i}$, $i = 1,\ldots,N$. Нека разгледаме стойността $\overline{X}_{\alpha,N} = \frac{1}{N}\sum_{i=1}^N X_{\alpha,i}$ като Монте Карло приближение на компонентата x_α на решението. Тогава, използвайки свойствата на математическото очакване следва, че:

$$E\{\overline{X}_{\alpha,N}\} = x_\alpha, \tag{2.2.15}$$

т.е. случайната величина $\overline{X}_{\alpha,N}$ е неизместена оценка за x_α, и имаме, че дисперсията $Var\{\overline{X}_{\alpha,N}\}$ клони към нула, когато N клони към безкрайност, т.е.

$$\lim_{N\to\infty} Var\{\overline{X}_{\alpha,N}\} = 0. \tag{2.2.16}$$

Ясно е, че качеството на алгоритъма, или колко е голяма дисперсията и колко малка вероятностната грешка, зависи от свойствата на матрицата A и дясната страна b. Балансирането на матрицата, заедно със спектралния радиус, играе важна роля за ефективността на Монте Карло алгоритъма.

Да разгледаме дисперсията на случайната величина $X_\alpha(\tau)$ за пресмятане на линейната форма $V(x)$. Използваме следните означения: $\overline{A} = \{|a_{ij}|\}_{i,j=1}^n$,

$\widehat{c} = \{c_i^2\}_{i=1}^{n+1}$. Изборът на МАО вероятностите води до крайна верига: $c_{\alpha_0} \to a_{\alpha_0\alpha_1} \to \ldots \to a_{\alpha_{k-1}\alpha_k}$. Дисперсията на случайната величина $X_\alpha^k(\tau)$ се дефинира като [62]

$$X_\alpha^k(\tau) = \frac{c_{\alpha_0}}{p_{\alpha_0}} \frac{a_{\alpha_0\alpha_1}}{p_{\alpha_0\alpha_1}} \frac{a_{\alpha_1\alpha_2}}{p_{\alpha_1\alpha_2}} \cdots \frac{a_{\alpha_{k-1}\alpha_k}}{p_{\alpha_{k-1}\alpha_k}} \frac{c_{\alpha_k}}{p_{\alpha_k}} = \frac{A_c^k c_{\alpha_k}}{P^k(\tau)}$$

и играе важна роля за качеството на алгоритъма. По-малка стойност на дисперсията $Var\{X_\alpha^k(\tau)\}$ води до по-добра сходимост на алгоритъма. Доказано е [62], че $Var\{X_\alpha^k(\tau)\} = \frac{c_{\alpha_0}}{p_{\alpha_0} p_\alpha}(\overline{A}_c^k \hat{c})_\alpha - (A_c^k c)_\alpha^2$.

2.2.3 Подобрен метод Монте Карло за изчисляване на решението

Подобреният алгоритъм за намиране на една и на всички компоненти на решението на СЛАУ се състои в специален избор на релаксационния параметър, което води до балансиране на итерационната матрица и до до повишаване на точността на алгоритъма, както и на по-малък брой изчислителни операции спрямо методът „случайното блуждаене по уравненията“ на СЛАУ, което води до намаляване на изчислителното време.

Първо ще бъде представен метод Монте Карло за намиране на една компонента x_{i_0} на решението на линейна система с реални коефициенти. Траекторията започва от реда i_0 на системата, b_{i_0} се добавя към резултата от „случайното блуждаене по уравненията“ на СЛАУ. След това траекторията се прекъсва или се посещава друго уравнение според вероятността $p_{i_0 j}$, описана в алгоритъма по-долу. Знакът на приноса на резултата се сменя в зависимост от знака на коефициентите на системата. Това е описано детайлно в алгоритъма по долу:

Алгоритъм 1. Пресмятане на една компонента x_{i_0} на решението x_i, $i = 1, \ldots n$

1. *Инициализация.* **Въведи** *начални данни: матрицата B, вектора f, константата $\gamma_i = b_{ii}$, $i = 1, \ldots, n$ и номера на случайните траектории N.*

2. *Предварителни пресмятания:*

2.1. **Изчисли** *матрицата A с използването на параметър $\gamma = (\gamma_1, \ldots, \gamma_n,\ i = 1, \ldots, n)$:*

$$\{a_{ij}\}_{i,j=1}^{n} = \begin{cases} 1 - b_{ii}, & \textit{когато} \quad i = j \\ -b_{ij}, & \textit{когато} \quad i \neq j\,. \end{cases}$$

3. **Положи** $S := 0$.

4. **За** $k = 1$ **до** N **повтаряй**

4.1 **положи** $m := i_0$.

4.2 **положи** $S := S + b_m$.

4.3 $test = 0;\ sign = 1$

4.4 генериране на случайно равномерно разпределено число $r \in (0, 1)$

4.5 **обнови** $S := S + sign * b_m$;

5. **край на цикъла**

6. $x_{i_0} = \frac{S}{N}$

Тук се описва подобреният алгоритъм за намиране на всички компоненти на решението. Идеята е да се пресметне резултата за всички състояния (разгледани като начални състояния), които са посетени по време на една траектория. 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 $W(i)$, които се модифицират, веднага щом състоянието i е посетено по време на траекторията. За дадена траектория запазваме посетените състояния в списък l, за да пресметнем приноса им към резултата на посетените състояния лесно. Накрая решението x_i се приближава с общия резултат $S(i)$, разделен на общия брой посещения $W(i)$. Ако състоянието никога не е посетено, то $W(i) = 1$. Ако състоянието е посетено повече от веднъж по време на същата траектория, обикновено пазим приноса на първото посещение, за да се редуцира дисперсията. Общият брой траектории N се избира кратно на размерността n на системата.

Алгоритъм 2. Пресмятане на всички компоненти $x_i,\ i = 1, \ldots n$ на решението с подобрения алгоритъм

1. Инициализация. **Въведи** *начални данни: матрицата B, вектора f, константата $\gamma_i = b_{ii},\ i = 1, \ldots, n$ и броя на случайните траектории N.*

2. Предварителни пресмятания:

2.1. **Изчисли** *матрицата A с използването на параметър $\gamma = (\gamma_1, \ldots, \gamma_n,\ i = 1, \ldots, n)$:*

$$\{a_{ij}\}_{i,j=1}^{n} = \begin{cases} 1 - b_{ii}, & \textit{когато} \quad i = j \\ -b_{ij}, & \textit{когато} \quad i \neq j\,. \end{cases}$$

3. **за** $i = 1$ **до** n **повтаряй**

3.1. $S(i) := 0$; $W(i) := 0$

3.2. **за** $k = 1$ **до** N **повтаряй**

3.2.1 **положи** $m := rand(1 : n)$.

3.2.2 **положи** $test := 0$; $m_1 := 0$;

3.2.3 $W(m) := W(m) + 1$; $m = m_1 + 1$; $l(m_1) = m$;

3.2.4 **за** $q = 1, m_1$ **повтаряй**:

$S(l(q)) := S(l(q)) + b_{l(q)}$;

3.3. **край на цикъла**

3.4. **за** $j = 1, n$ **повтаряй**:

$W(j) := \max\{1, W(j)\}$;

$x_j = \frac{S(j)}{W(j)}$.

край на цикъла

За 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 [93], и е развит в по-скорошни публикации [94, 96, 95]. Техниката се използва също да се намери приближение на ортогонален базис [140] или за решаване на частни диференциални уравнения с висока точност [87].

Основната идея на SMC метода е много ефективна. В случая на линейни системи трябва да пресметнем вектора x, който е решение на системата $x = Ax+b$. Очевидно, ако b е малко, дисперсията на алгоритъма е малка и ако $b = 0$, то дисперсията е също нула. Понеже задачата е линейна, можем да конструираме нова система, чието решение е резидуал на първоначалната система, като имаме малко b. Нека $x^{(1)}$ е приближеното решение на системата с използването на горния алгоритъм. Тогава резидуалът $y = x - x^{(1)}$ е решение на новото уравнение $y = Ay + b - x^{(1)} + Ax^{(1)}$, където $b - x^{(1)} + Ax^{(1)}$ трябва да бъде близко до нула. Пресмятаме приближението $y^{(1)}$ на това уравнение с използването на нашия алгоритъм отново и това води до приближение $x^{(2)} = x^{(1)} + y^{(1)}$ за оригиналното уравнение. Същата идея се използва итерационно, за да се получи приближението $x^{(k)}$ след k стъпки на алгоритъма. Може да се докаже (виж [87]), че ако броят реализации N, използвани да се пресметне $x^{(k)}$, е достатъчно голям, тогава съответната дисперсия клони към нула с геометрична скорост на сходимост.

Алгоритмите, характеризиращи се с такава скорост на сходимост, се означават като алгоритми с нулева дисперсия [50, 51, 64, 124, 133, 134, 140]. Алгоритмите от този тип могат успешно да се конкурират с оптималните алгоритми, основани на подпространства на Крилов, и с градиентните методи.

2.2.4 Числени експерименти

За да пресметнем точността на полученото числено решение на системата $\hat{x}$, пресмятаме резидуала $r := B\hat{x} - f$ и тегловия резидуал („weighted residual“) [229, 230] по формулата:

$$\rho := \frac{||r||}{||B||\ ||\hat{x}||}.$$

В таблиците и фигурите по-долу е дадено сравнение на относителната греш-

ка и изчислителното време за фиксиран брой SMC итерации между стандартния рафиниран Монте Карло метод (RIMC) [50], оригиналния Монте Карло метод „случайно блуждаене по уравненията“ (WE) [62] и конструирания подобрен метод Монте Карло за линейни системи (IWE). Броят на итерациите е означен с N, тегловият резидуал с RE, а изчислителното време с t и е измерено в секунди. 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 итерационата матрица A. Търсим решението на линейната система $Bx = f$, като са разгледани примери с плътна и разредена матрица. Числените експерименти са проведени с различни СЛАУ с размерност $n = 7, 100, 1000, 5000$, където броят на уравненията в линейната система е n и $B \in \mathbb{R}^{n\times n}$.

Пример 1. Първо тестваме методите върху матрици с малка размерност. Трябва да намерим решенията x_1 и x_2 на линейната система $Bx = f$:

$$\mathbf{B} = \begin{pmatrix} 5 & -1 & -1 & 0 & 0 & -1 & -1 \\ -1 & 5 & -1 & -1 & 0 & 0 & -1 \\ -1 & -1 & 5 & -1 & -1 & 0 & 0 \\ 0 & -1 & -1 & 5 & -1 & -1 & 0 \\ 0 & 0 & -1 & -1 & 5 & -1 & -1 \\ -1 & 0 & 0 & -1 & -1 & 5 & -1 \\ -1 & -1 & 0 & 0 & -1 & -1 & 5 \end{pmatrix}, \quad \mathbf{f}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad \mathbf{f}_2 = \begin{pmatrix} 4 \\ -2 \\ -1 \\ 0 \\ -1 \\ -2 \\ 4 \end{pmatrix}. \tag{2.2.17}$$

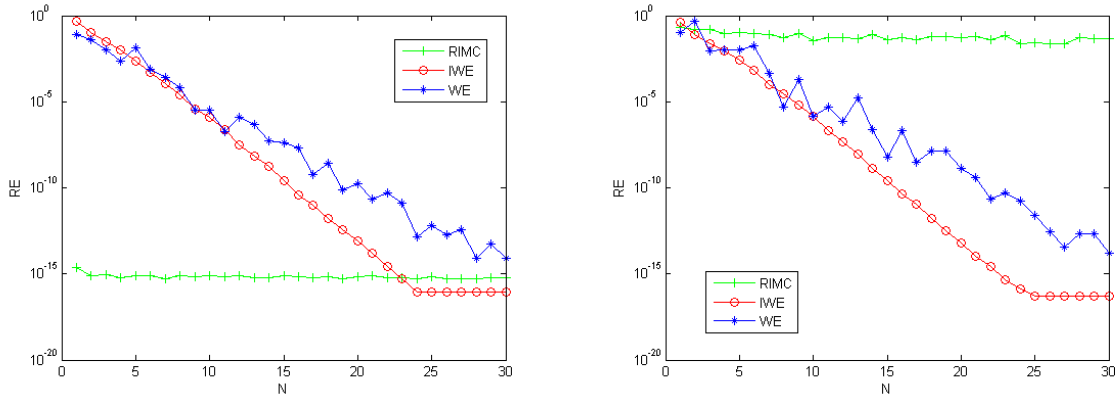
Решенията са

$$\mathbf{x}_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad \mathbf{x}_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \tag{2.2.18}$$

За малки размерности на матрицата B за разгледаните примери с размерност 7, могат да се направят изводи, че методът WE дава близки резултати

Таблица 2.4: Теглови резидуал на решението за матрицата $B \in \mathbb{R}^{7\times 7}$

N	x_1						x_2					
	RIMC	t	WE	t	IWE	t	RIMC	t	WE	t	IWE	t
2	2.28e-15	0.02	4.15e-02	0.11	1.00e-01	0.007	1.54e-01	0.003	4.63e-01	0.11	8.21e-02	0.003
5	7.89e-16	0.07	1.39e-02	0.23	2.29e-03	0.026	1.01e-01	0.01	9.93e-03	0.23	2.48e-03	0.008
10	8.07e-16	0.21	3.18e-06	0.68	1.24e-06	0.04	3.55e-02	0.04	1.43e-06	0.68	1.42e-06	0.05
15	7.45e-16	0.35	3.94e-08	1.11	2.55e-10	0.1	4.04e-02	0.08	6.17e-09	1.11	2.66e-10	0.09
20	6.66e-16	0.69	1.71e-10	2.53	7.78e-14	0.23	5.00e-02	0.14	1.40e-09	2.53	6.56e-14	0.16
30	5.79e-16	1.14	8.12e-15	3.69	8.32e-17	0.49	4.72e-02	0.24	1.53e-14	3.69	5.03e-17	0.29



Фигура 2.2: Теглови резидуал на решението за *пример 1*: (a) за x_1; (b) за x_2.

Таблица 2.5: Теглови резидуал на решението за матрицата $B \in \mathbb{R}^{100\times 100}$

N	RIMC	t,s	WE	t,s	IWE	t,s
2	5.877e-03	0.04	4.295e-02	0.47	2.388e-02	0.06
5	3.729e-03	0.21	1.662e-01	1.23	2.414e-04	0.28
10	2.616e-03	0.54	2.491e-05	2.68	1.565e-08	0.59
15	2.726e-03	0.88	3.064e-07	4.11	3.201e-10	0.89
20	2.134e-03	1.24	8.342e-08	5.53	2.611e-13	1.31
30	1.722e-03	2.54	9.511e-11	11.69	2.682e-16	2.59

до IWE като по-голямото предимство на новия алгоритъм е по отношение на бързината - при един и същ брой итерации IWE дава над 5 пъти по-малко изчислително време от WE, което дори за малки размерности е съществено подобрение. Броят на траекториите е $10n$. IWE постига по-добра точност от WE с 2 порядъка при $N > 15$ - виж Таблица 2.4. Също така е интересно да се отбележи, че RIMC е бавно сходящ, освен в случаите, когато системата има тривиалното единичното решение - виж Фиг. 2.2. Може да се направи извода, че за малки размерности IWE и WE дават сходен порядък на относителна грешка.

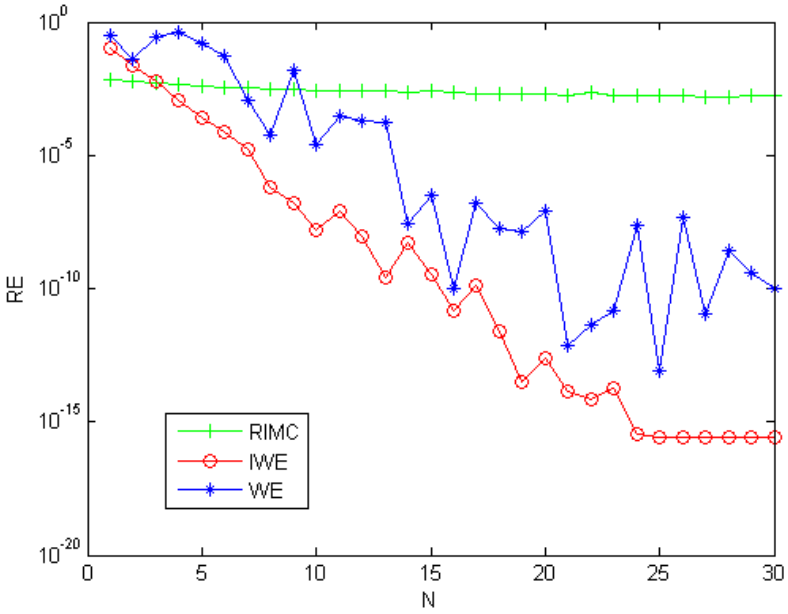
Пример 2

Нека $\mathbf{B}$ е плътна матрица 100×100 с елементи в интервала (0,1), а $f \in \mathbb{R}^{100}, f_i = 1,\ i = 1, \ldots, 100$. Броят на случайните траектории е $5n$.

За плътна матрица с размер 100 на системата предимството на IWE в сравнение с WE е 3 до 5 порядъка по отношение на точност, което е вече съществено подобрение. По отношение на изчислителното време отново новият метод е над 5 пъти по-бърз в сравнение с оригиналния WE и постига по-добра точност - виж Таблица 2.3. Рафинираният метод Монте Карло е бавно сходящ и дава по-голяма относителна грешка за разглежданата система - виж Фиг. 2.3

Пример 3

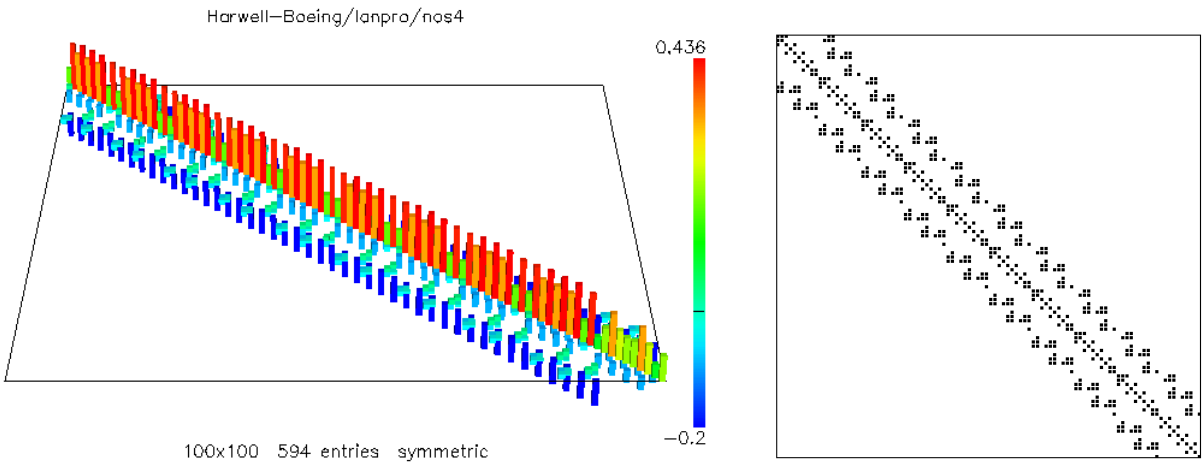
Нека $\mathbf{B}$ е симетричната положително определена матрицата NOS4 от колекцията на Harwell-Boeing [233] от групата матрици LANPRO, а $f \in \mathbb{R}^{100}, b_i = 1,\ i = 1, \ldots, 100$. Тази матрица е взета от приложения, свързани с апроксимация по метода на крайните елементи на модел, описваща гредова структура в



Фигура 2.3: Теглови резидуал на решението за матрица $B \in \mathbb{R}^{100\times100}$

конструктивната механика. Матрицата NOS4 е разредена матрица, има точно 594 ненулеви елемента, 100 ненулеви елемента по главния диагонал, и 247 елемента под и над главния диагонал, или средно 5.9 ненулеви елемента във всяка колона и всеки ред. В [233] е дадено, че l_2-нормата на разглежданата матрица е 0.85, а числото на обусловеност - 2700. Структурата на матрицата NOS4 е взета от [233] и е дадена на Фиг. 2.4.

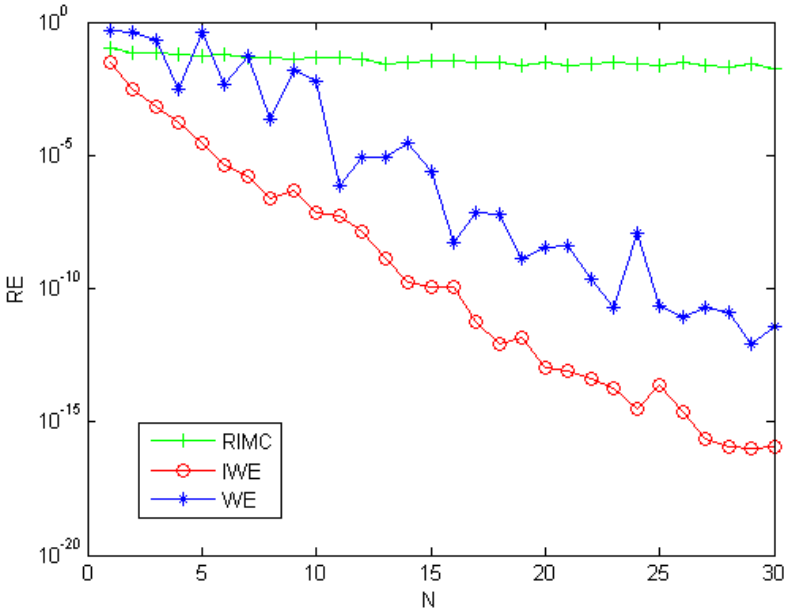
За разредена матрица с размер 100 предимството на IWE в сравнение с WE е 3 до 5 порядъка по отношение на точност, което е вече съществено подобрение - виж Фиг. 2.5. По отношение на изчислителното време отново новият метод е няколко пъти по-бърз в сравнение с оригиналния WE и постига по-добра точност - виж Таблица 2.6. Рафинираният метод Монте Карло е бавно сходящ и дава по-голяма относителна грешка за разглежданата система. Съществено е, че алгоритмите не се влияят от плътността на матрицата. Матрицата NOS4 е подбрана, така че да има локални минимуми, близо до глобалния. Известно е, че когато детерминистичните алгоритми попаднат в локален минимум, който е съседен на глобален минимум, не могат да излязат от него, алгоритъмът остава



Фигура 2.4: Структура на матрицата NOS4

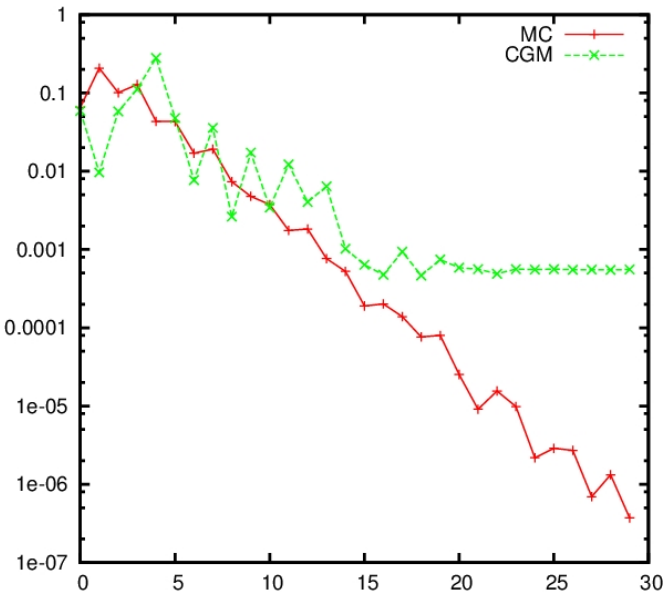
Таблица 2.6: Теглови резидуал на решението за матрицата $NOS4 \in \mathbb{R}^{100\times100}$.

N	RIMC	t,s	WE	t,s	IWE	t,s
2	7.253e-02	0.05	4.178e-01	0.84	3.028e-03	0.08
5	5.449e-02	0.22	4.148e-01	2.37	3.071e-05	0.24
10	4.319e-02	0.56	5.943e-03	5.31	7.461e-08	0.61
15	3.520e-02	0.78	2.419e-06	9.1	1.217e-10	0.89
20	3.197e-02	1.11	3.336e-09	13.5	1.022e-13	1.26
30	1.835e-02	2.15	3.660e-12	18.6	1.109e-16	2.29



Фигура 2.5: Теглови резидуал на решението за матрицата NOS4.

в този локален минимум и зацикля, докато методите Монте Карло, поради стохастичната си природа, дори да попаднат в локални минимуми, лесно излизат от тях [62]. Затова за матрицата $NOS4$ сходимостта на Монте Карло метода е по-добра от тази на метода на спрегнатия градиент (PCG). Линията на грешката отива до 10^{-6} или 10^{-7} при IWE, докато тази на PCG достига точност 10^{-3}, както е показано на Фиг. 2.6. При експериментите сме поискали точност от 10^{-8}. Този резултат може да се обясни по следния начин. 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 $NOS4$. Метода 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. Това е



Фигура 2.6: Сравнение на метода Монте Карло IWE и метода PCG за матрицата $NOS4$ от колекцията Harwell-Boeing.

обект на по-дълбок анализ и не е тривиален проблем. Не може да се гарантира, че такъв ефект се случва при всяка матрица, но има случаи, в които методът WE и съответно подобреният IWE, е по-добър от PCG.

Пример 4

Интересно е да се види поведението на алгоритмите за разредени матрици с голяма размерност. Нека B е положително определена разредена матрица 1000×1000 със случайни числа в интервала $(0, 1)$ и $f \in \mathbb{R}^{1000}, f_i = 1,\ i = 1, \ldots, 1000$.

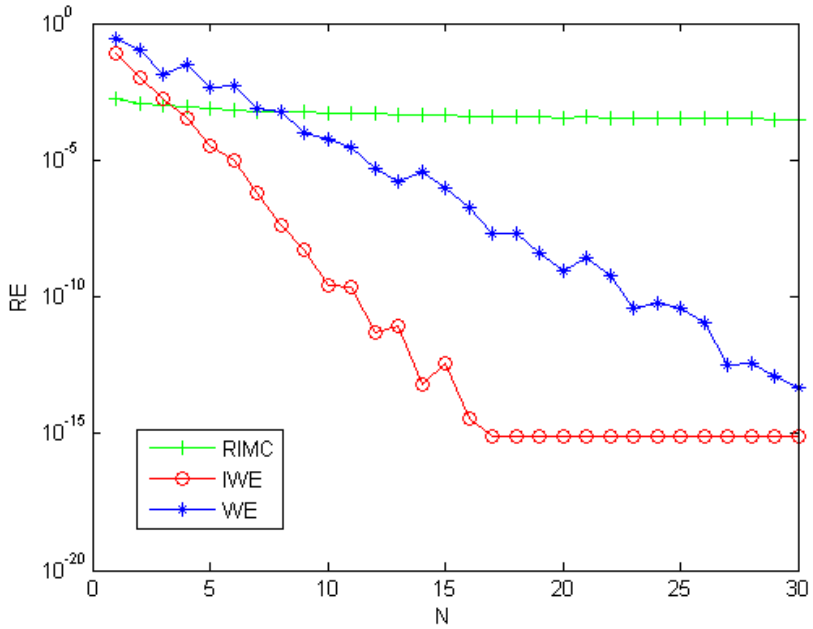
След 15 итерации методът IWE достига точност 10^{-14}, докато методът WE достига същата точност при 30 итерации - виж Таблица 2.7. Броят на случайните траектории е избран да бъде равен на размерността на системата. Предимството на подобрения алгоритъм спрямо оригиналния е показано на Фиг. 2.7. Докато преобладаващият главен диагонал оказва силно влияние върху резултатите на WE, при подобрения алгоритъм IWE това влияние е по-слабо изразено.

Пример 5

Интересно е да се види поведението на алгоритмите и за плътни матрици с голяма размерност. Нека B е положително определена матрица 5000×5000

Таблица 2.7: Теглови резидуал за матрицата $B \in \mathbb{R}^{1000\times 1000}$.

N	RIMC	t	WE	t	IWE	t
2	1.212e-03	2.17	1.035e-01	0.75	1.064e-02	0.11
5	7.397e-04	13.1	4.621e-03	2.34	3.468e-05	0.45
10	5.343e-04	33.3	5.819e-05	6.3	2.545e-10	0.97
15	4.420e-04	65	9.309e-07	13.5	3.823e-13	2.3
20	3.554e-04	169	8.670e-10	36	7.731e-16	6.7
30	3.063e-04	357	4.436e-14	87	7.325e-16	17



Фигура 2.7: Теглови резидуал на решението за матрицата $B \in \mathbb{R}^{1000\times 1000}$.

Таблица 2.8: Теглови резидуал за матрицата $B \in \mathbb{R}^{5000\times5000}$.

N	RIMC	t	WE	t	IWE	t
2	5.438e-03	10.05	4.304e-02	3.95	2.931e-02	0.15
5	3.875e-03	60.2	1.217e-01	13.3	1.816e-04	0.9
10	2.866e-03	130.5	2.301e-05	32.3	1.235e-07	2.4
15	2.367e-03	310.7	6.486e-09	67.8	1.833e-10	5.1
20	1.941e-03	811	3.205e-09	171.5	1.054e-14	11.1
30	1.701e-03	2135	1.126e-07	418.6	2.481e-16	25.2

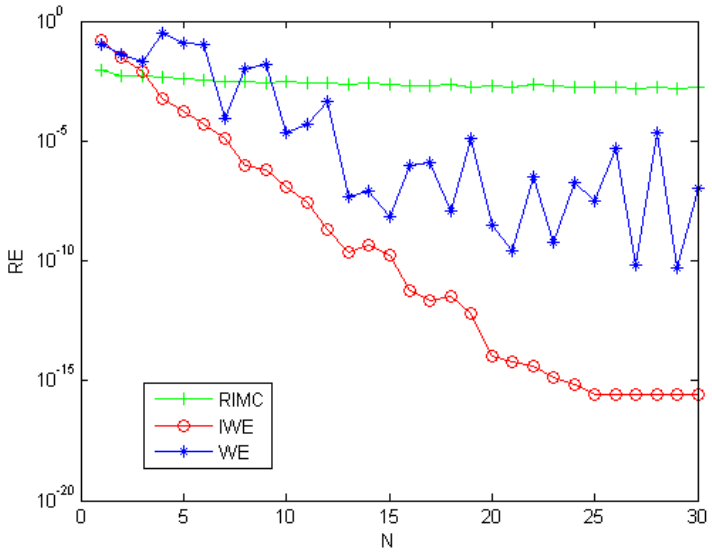
със случайни числа в интервала $(0, 1)$ и $f \in \mathbb{R}^{5000}, f_i = 1,\ i = 1, \ldots, 5000$. Броят на случайните траектории отново е избран да бъде равен на размерността на системата.

За по-големи размерности превъзходството на новия метод IWE над стандартния WE е още по-голямо. За 30 итерации методът WE достига точност 10^{-9}, докато методът IWE достига 10^{-16} при зададена точност $tol = 10^{-16}$ - виж Фиг. 2.8. Това съчетано с над 15 пъти по-голямото изчислително време на IWE спрямо WE - виж Таблица 2.8, води до още по-голямо предимство за подобрения алгоритъм.

2.2.5 Заключение

- Разработен е нов метод Монте Карло, който е реализиран на базата на метода WE с използване на дискретна верига на Марков. Матрицата B и дясната част f са нормирани, за да се ускори сходимостта на стохастичния процес. Направен е оптимален избор на релаксационния параметър γ, което води до балансиране на итерационната матрица A и до повишаване на точността на алгоритъма.

- Поради реализацията си методът е в пъти по-бърз от Монте Карло метода WE, като експериментите показват и предимство по отношение на точността, което се вижда особено за матрици с голяма размерност. Новият метод постига много по-добра точност за по-малко време и по-малък



Фигура 2.8: Теглови резидуал на решението за матрицата $B \in \mathbb{R}^{5000\times5000}$.

брой предварително зададени итерации.

- Методът може успешно да се конкурира с градиентните методи, като съществуват матрици като NOS4, за които методът Монте Карло е по-добър от метода на спрегнатия градиент.

- Направено е сравнение със стандартния рафиниран метод Монте Карло за линейни системи и с оригинални метод WE, като получените резултати утвърждават новия метод Монте Карло като един от най-добре известните методи Монте Карло за линейни системи.

- Поведението на алгоритъма Монте Карло за линейни системи не зависи от плътността на матрицата. Матрицата NOS4 има само 5.9 средно ненулеви елемента в колона и ред. Предимството на алгоритъма е в сила както за разредени, така и за плътни матрици.

- Подобреният метод Монте Карло за линейни системи IWE, конструиран в тази глава, може да се използва за решаването на системи от линейни алгебрични уравнения, получени след дискретизация на системи от частни диференциални уравнения, които се разглеждат в следващата глава.

Глава 3

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

3.1 Въведение

В много области на природните и инженерните науки, параболичните ЧДУ винаги се използват за описание на еволюционни процеси, затова числените методи, базирани на диференчни схеми, имат важно значение, виж [43, 141, 176, 182]. В този контекст, стандартните диференчни схеми от втори ред се нуждаят от по-фина мрежа за да постигнат апроксимация на решението с исканата точност. В резултат на това се получават алгебрични системи с голяма размерност, за чието решаване се изискват съвременни високопроизводителни изчислителни ресурси. Един възможен подход да се намали изчислителната сложност в математическите модели и симулации с голяма размерност е да се използват методи на дискретизация с по-голяма точност. Друг важен фактор, отнасящ се до изчислителната ефективност на методите за дискретизация, е да се решат получените линейни и нелинейни системи от алгебрични уравнения. Методите с висок ред на точност обикновено генерират алгебрични системи от много по-малък ред, сравнени с тези, получени с дискретизация с по-нисък ред на точност.

В последните години се наблюдава нов и все по-нарастващ интерес към компактните (с минимален шаблон) диференчни схеми с висок ред на точност за

решаване на ЧДУ [130, 175, 197, 202, 210, 216]. 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 [90] 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 висок ред в локалната грешка на апроксимация (local truncation error, LTE) [198, 217]. Повече детайли за компактните диференчни схеми за задачи от тип адвекция-дифузия може да се намерят в [130, 197, 202, 217].

В [121, 226, 227] модела на далечен пренос на замърсители във въздуха (UNI-DEM) е добре описан. Там е предложен итерационен метод с преубословител за нелинейната параболична система. Използва се неявен метод на Ойлер по времето и дискретизация по метода на крайните елементи по пространството, и след това външно-вътрешни итерации и преобуславяне. Добре известен подход за повишаване на реда на точността на диференчните схеми, е да се използва екстраполация по Ричардсон [141]. Компактни диференчни схеми с четвърти ред на точност и схема от четвърти ред, базирана на екстраполация по Ричардсон за моделно едномерно елиптично уравнение са получени в [60, 217].

Напоследък има голям интерес в създаването на диференчни схеми от шести ред. С използването на ред на Тейлър, Spotz и Garey [197] разработват компактна схема за уравнението на Poisson, която може да постигне шести ред на точност, само когато производните могат да се пресметнат аналитично. Разработени са подобни схеми с шести ред на точност, виж [130, 210], но всички те имат няколко слаби качества като:

(1) производните на решението се появяват в дясната страна, което изисква аналитични форми на апроксимация на производните от даден ред на точност;

(2) компактни схеми, които могат да създадат трудности във възли близо до границата;

(3) получават се сложни линейни системи алгебрични уравнения, които увеличават трудността в използването на ефективни алгоритми за линейни системи.

Както знаем, не съществуват явни диференчни схеми от шести ред върху

единична мрежа [216].

3.2 Двумерен модел на процес за далечен пренос на замърсители във въздуха

Симулацията на различни процеси в химията, физиката и инженерните науки използва модели, описвани с параболични уравнения. Тази глава е посветена на конструирането на нови компактни диференчни схеми (CFDS) с висок ред на точност за полулинейни параболични системи. Модели на преноси на замърсители във въздуха със свързани нелинейни химични реакции [121, 226, 227], са от основен интерес, а именно:

$$\frac{\partial u_l}{\partial t} - K\triangle u_l + \mathbf{b}_l \nabla u_l = R_l(x, y, u_1, \ldots, u_L), \quad (x, y, t) \in \Omega \times (0, T], \tag{3.2.1}$$

$$\mathbf{u} = 0, \quad (x, y, t) \in \partial\Omega \times (0, T], \tag{3.2.2}$$

$$\mathbf{u} = \mathbf{u}_0(x, y), \qquad (x, y) \in \Omega, \tag{3.2.3}$$

където $\mathbf{u} = (u_1, u_2, ..., u_L)$, $u_l = u_l(x, y, t)$, $l = 1, ..., L$ 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 числения подход. Това съответства на физичен модел, описан в [85, 121, 227].

Основната цел е приложението на по-горе споменатите диференчни апроксимации за следната реална физична параболична транспортна система, описана в [85]. Следвайки [85, 121, 228] адвекцията в (3.2.1) е

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

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

модела е следната:

$$\begin{aligned}
R_1(u_1,...,u_{10}) &= k_5u_2 - (k_6u_5 + k_4u_7 + k_3u_8)u_1, \\
R_2(u_1,...,u_{10}) &= (k_6u_5 + k_4u_7 + k_3u_8)u_1 - (k_5 + k_9u_9)u_2, \\
R_3(u_1,...,u_{10}) &= -k_1u_3u_9, \\
R_4(u_1,...,u_{10}) &= 2k_1u_3u_9 + k_3u_1u_8 - k_2u_4, \\
R_5(u_1,...,u_{10}) &= k_2u_5 \\
R_6(u_1,...,u_{10}) &= k_9u_2u_9, \\
R_7(u_1,...,u_{10}) &= 2k_2u_4 + k_3u_1u_8 + k_{10}u_9 - k_4u_1u_7, \\
R_8(u_1,...,u_{10}) &= 4k_1u_3u_9 - k_3u_1u_8, \\
R_9(u_1,...,u_{10}) &= k_4u_1u_7 + 2k_8u_{10} - (k_1u_3 - k_9u_2 + k_{10})u_9, \\
R_{10}(u_1,...,u_{10}) &= k_7u_5 - k_8u_{10}.
\end{aligned} \tag{3.2.4}$$

Химичните реакции на модела са дадени в Таблица 3.1 за пълнота. Скоростните

Таблица 3.1: Химичните реакции на модела

1	$HC + OH \to 4RO_2 + 2ALD$	6	$NO + O_3 \to NO_2 + O_2$
2	$ALD + h\nu \to 2HO_2 + CO$	7	$O_3 + h\nu \to O_2 + O(^1D)$
3	$RO_2 + NO \to NO_2 + ALD + HO_2$	8	$O(^1D) + H_2O \to 2OH$
4	$NO + HO_2 \to NO_2 + OH$	9	$NO_2 + OH \to HNO_3$
5	$NO_2 + h\nu \to NO + O_33$	10	$CO + OH \to CO_2 + HO_2$

константи на химическите реакции са дадени в Таблица 3.2. Някои от константите 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. Химическите вещества, включени в опростените реакции, са дадени в Таблица 3.3.

От практическа и математическа гледна точка, интерес представлява съществуването и качественото поведение (неотрицателността) на решението на задачата (3.2.1)-(3.2.3). Коректността на начално-граничната задача за по-обща

Таблица 3.2: 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}	$2.9e-13$

Таблица 3.3: Химичните вещества в 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	$O(^1D)$

система от (3.2.1) е получена в [171]. Ще допускаме до края съществуване и единственост на класическо решение на задачата (3.2.1)-(3.2.3), което означава, че функцията принадлежи на $C([0,T]\times\overline{\Omega})\bigcap C^1((0,T);C(\overline{\Omega}))\bigcap(C(0,T);C^2(\overline{\Omega}))$ и удовлетворява (3.2.1)-(3.2.3) поточково.

Тъй като сме заинтересовани от система, описваща химични концентрации, неотрицателността на решението трябва да се запази. Доказано е в [42], че ако

1. $\mathbf{u}_0(x,y)\geq 0$;

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

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

Лесно е да се провери, че химичните реакции $R_l(u_1,u_2,\ldots,u_{10})$, $l=1,...,10$ зададени от (3.2.4) удовлетворяват точка 2. и решението на задачата (3.2.1)-(3.2.3) с химична част (3.2.4) е неотрицателно по времето $t>0$, ако началните данни $\mathbf{u}_0(x,y)\geq 0$.

3.3 Компактни схеми в едномерния случай

За по-голяма яснота най-напред ще представим конструирането на компактни диференчни схеми (CFDS) за едномерна параболична система от 2 уравнения

$$\begin{aligned} \frac{\partial u}{\partial t} - a(x)\frac{\partial^2 u}{\partial x^2} + b(x)\frac{\partial u}{\partial x} &= f(x,t,u,v), \\ \frac{\partial v}{\partial t} - c(x)\frac{\partial^2 v}{\partial x^2} + d(x)\frac{\partial v}{\partial x} &= g(x,t,u,v), \end{aligned} \tag{3.3.1}$$

дефинирана в областта $Q_T = \Omega \times (0,T)$, където $\Omega \subset R$ е ограничена област. Нелинейните функции f и g са достатъчно гладки относно техните аргументи. Коефициентите $a(x)$, и $c(x)$ са положителни в Ω. Ще разглеждаме гранични условия на Дирихле:

$$u(x,t) = \overline{\phi}(x,t), \quad v(x,t) = \overline{\overline{\phi}}(x,t), \quad (x,t) \in \partial\Omega \times (0,T) \tag{3.3.2}$$

с начално условие

$$u(x,0) = \overline{\psi}(x), \quad v(x,0) = \overline{\overline{\psi}}(x), \quad (x) \in \Omega, \tag{3.3.3}$$

където $\overline{\phi}$, $\overline{\overline{\phi}}$, $\overline{\psi}$ и $\overline{\overline{\psi}}$ са дадени.

Да въведем мрежата $\Omega_h = \{x_i = ih, \quad i = 0,1,\ldots,M, \quad h = 1/M\}$ и централните диференчни оператори от втори ред $\delta_x\varphi_i = (\varphi_{i+1} - \varphi_{i-1})/2h$, $\delta_x^2\varphi_i = (\varphi_{i+1} - 2\varphi_i + \varphi_{i-1})/h^2$ за произволна мрежова функция φ_i, $i = 1,\ldots,M-1$. Прилагаме тези оператори към елиптичната част на системата и получаваме:

$$-a_i\delta_x^2 u_i + b_i\delta_x u_i - e_{1,i} = F_i(x_i,t,u_i,v_i) \equiv f(x_i,t,u_i,v_i) - \frac{\partial u_i}{\partial t}, \tag{3.3.4а}$$

$$-c_i\delta_x^2 v_i + d_i\delta_x v_i - e_{2,i} = G_i(x_i,t,u_i,v_i) \equiv g(x_i,t,u_i,v_i) - \frac{\partial v_i}{\partial t}, \tag{3.3.4б}$$

където локалните грешки от апроксимация (LTE) са:

$$\begin{aligned} e_{1,i} &= \frac{h^2}{12}\left(2b\frac{\partial^3 u}{\partial x^3} - a\frac{\partial^4 u}{\partial x^2}\right)\Bigg|_i + O(h^4), \\ e_{2,i} &= \frac{h^2}{12}\left(2d\frac{\partial^3 v}{\partial x^3} - c\frac{\partial^4 v}{\partial x^4}\right)\Bigg|_i + O(h^4). \end{aligned}$$

След изпускането в (3.3.4) на членовете $e_{1,i}$, $e_{2,i}$ получаваме полу-дискретна схема от втори ред по пространствената променлива. Ще наричаме тази схема стандартна.

За да получим диференчна схема от четвърти ред, без да разширяваме шаблона, трябва да елиминираме членовете от втори ред в LTE, като използваме диференциалните уравнения на системата. Диференцираме първото уравнение в (3.9.1) два пъти по x и получаваме

$$\begin{cases} a\frac{\partial^3 u}{\partial x^3} = \left(b - \frac{da}{dx}\right)\frac{\partial^2 u}{\partial x^2} - \frac{db}{dx}\frac{\partial u}{\partial x} - \frac{\partial F}{\partial x} \\ a\frac{\partial^4 u}{\partial x^4} - 2b\frac{\partial^3 u}{\partial x^3} = \left(2\frac{db}{dx} - \frac{d^2 a}{dx^2}\right)\frac{\partial^2 u}{\partial x^2} + \frac{d^2 b}{dx^2}\frac{\partial u}{\partial x} - \left(b + 2\frac{da}{dx}\right)\frac{\partial^3 u}{\partial x^3} - \frac{\partial^2 F}{\partial x^2}\,. \end{cases}$$

За да повишим реда на грешката до $O(h^4)$ в (3.3.4a) използваме, че

$$\begin{aligned} \left(a\frac{\partial^4 u}{\partial x^4} - 2b\frac{\partial^3 u}{\partial x^3}\right)\Big|_i = & - (\delta_x^2 a_i - \widetilde{a}_i(\delta_x a_i - b_i) - 2\delta_x b_i)\delta_x^2 u_i \\ & + (\delta_x b_i - \widetilde{a}_i.\delta_x c_i)\delta_x u_i - \delta_x^2 F_i + \widetilde{a}_i F_i + O(h^2), \end{aligned}$$

където $\widetilde{a}_i = (b_i + 2\delta_x a_i)/a_i$, $i = 1, \ldots, M-1$. Нека $\alpha_i = (\delta_x^2 a_i - \widetilde{a}_i(\delta_x a_i - b_i) - 2\delta_x b_i)$, $\widetilde{\alpha}_i = a_i + \frac{h^2}{12}\alpha_i$, $\widetilde{\widetilde{\alpha}}_i = b_i + \frac{h^2}{12}(\delta_x^2 b_i - \widetilde{a}_i \delta_i b_i)$. Дефинираме диференчните оператори

$$l_i^h = -\widetilde{\alpha}_i\delta_x^2 + \widetilde{\widetilde{\alpha}}_i\,\delta_x, \quad \nu_i^h = 1 + \frac{h^2}{12}(\delta_x^2 - \widetilde{a}_i\delta_x), \quad \overline{\mathcal{P}}_i^h = 6h^2 l_i^h, \quad \overline{\mathcal{Q}}_i^h = 6h^2\nu_i^h\,.$$

Нека също $\overline{P} = tridiag(p_{i,i-1}, p_{i,i}, p_{i,i+1})$ и $\overline{Q} = tridiag(q_{i,i-1}, q_{i,i}, q_{i,i+1})$ е тридиагонална матрица (съответстваща на $\overline{\mathcal{P}}$, $\overline{\mathcal{Q}}$) с елементи $p_{i,i} = 12a_i + h^2\alpha$, $p_{i,i\pm1} = -6a_i \pm \widetilde{\alpha}_i - 0.5h^2\alpha$, $q_{ii} = 5h^2 \quad q_{i,i\pm1} = 0.25h^2(2 \mp \widetilde{\alpha}_i h)$. Накрая, ако $U_i \approx u(x_i, t)$, $i = 0, ..., M$ и $U = (U_0, ..., U_M)^T$, тогава полудискретизацията на (3.3.4a) от ред $O(h^4)$ е както следва:

$$\overline{\mathcal{P}}_i^h U_i = \overline{\mathcal{Q}}^h F_i \;\; i = 1, \ldots, M-1 \;\text{ and }\; U_0 = \overline{\phi}(x_0, t) \;\; U_M = \overline{\phi}(x_M, t)\,, \tag{3.3.5}$$

$U^0 = \overline{\Psi} = (\overline{\psi}(x_0), ..., \overline{\psi}(x_M))$.

По същия начин правим с (3.3.4б). Аналогично на $\widetilde{a}_i$, α_i, $\widetilde{\alpha}_i$, $\widetilde{\widetilde{\alpha}}_i$, $\overline{P}$ и $\overline{Q}$ дефинираме $\widetilde{c}_i$, β_i, $\widetilde{\beta}_i$, $\widetilde{\widetilde{\beta}}_i$, $\overline{\overline{P}}$ and $\overline{\overline{Q}}$, като заменяме $a \leftrightarrow c$, $b \leftrightarrow d$ и $U \leftrightarrow V$.

Презаписваме системата ODE (3.3.5) в канонична форма

$$\frac{\partial U}{\partial t} = L_1 U + f(U, V) \qquad \frac{\partial V}{\partial t} = L_2 V + g(U, V)\,, \tag{3.3.6}$$

където $L_1 = (\overline{Q})^{-1}\overline{P}$, $L_2 = (\overline{\overline{Q}})^{-1}\overline{\overline{P}}$ и $U = (U_0, U_1, \ldots, U_M)^T$, $V = (V_0, V_1, \ldots, V_M)^T$, $f(U, V) = (f(U_0, V_0), f(U_1, V_1), \ldots f(U_M, V_M))^T$. Като бъдеща работа ще бъде изучена сходимостта и устойчивостта на неявно-явния (IMEX) дискретен метод по

времето. Нека $\Omega_\tau = \{t_j = j\tau, \quad j = 0, 1, \ldots, N, \quad \tau = T/N\}$ е равномерна мрежа по времето. За пълната дискретизация ще използваме схема с тегло θ.

$$\frac{U^{j+1} - U^j}{\tau} = (L_1 U)^{j,\theta} + (f(U,V))^{j,\theta}, \ \frac{V^{j+1} - V^j}{\tau} = (L_2 V)^{j,\theta} + (g(U,V))^{j,\theta}, \quad (3.3.7)$$

където $W^{j,\theta} = \theta W^{j+1} + (1-\theta)W^j$, $0 \leq \theta \leq 1$. За $\theta = 0.5$ схемата е известна като Кранк-Никълсън (Crank-Nicolson). Като приложим тази схема към *стандартната* полудискретизация получаваме LTE от ред $\mathcal{O}(\tau^2 + h^2)$, докато компактната схема е от ред $\mathcal{O}(\tau^2 + h^4)$.

3.4 Екстраполация на Ричардсон

Екстраполацията на Ричардсон е мощно изчислително средство, което успешно може да се използва в усилията за подобряване на точността на приближените решения на системите ЧДУ, получени по метода на диференчните схеми и крайните елементи.

Следователно, друг начин за получаване на диференчни схеми с висок ред на точност, е да се използва метода на екстраполация на Ричардсон. Главната идея [141] 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:

$$U_h^\tau = U_{(i,j)}^n = u(x_i, y_j, t^n) + C_1 h^\sigma + \chi(h,\tau), \quad (x_i, y_j, t_n) \in \Omega_{h,\tau}, \quad (3.4.1)$$

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

$$\begin{aligned} \gamma_1 + \gamma_2 &= 1 \\ \gamma_1 + \frac{\gamma_2}{2^\sigma} &= 0 \end{aligned} \quad (3.4.2)$$

- получаваме ново числено решение на грубата мрежа

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

От (3.4.2) имаме за случая на стандартна схема на Кранк-Никълсън ($\sigma = 2$), че коефициентите на екстраполацията по Ричардсон са

$$\gamma_1 = -1/3 \qquad \gamma_2 = 4/3. \tag{3.4.3}$$

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

$$\gamma_1 = -1/15 \qquad \gamma_2 = 16/15. \tag{3.4.4}$$

Ако в(3.4.1) се направи по-детайлен анализ на LTE, тогава продължение на идеята на пространствено-времевата екстраполация по Ричардсон [175] може да се приложи.

3.5 Централни диференчни схеми в двумерния случай

В тази секция за яснота е описана конструкцията на централна диференчна схема от втори ред (CDS) за слабо свързаната система от две уравнения

$$\frac{\partial u}{\partial t} - a(x,y)\frac{\partial^2 u}{\partial x^2} - b(x,y)\frac{\partial^2 u}{\partial y^2} + c(x,y)\frac{\partial u}{\partial x} + d(x,y)\frac{\partial u}{\partial y} = r(x,y,t,u,v), \tag{3.5.1а}$$

$$\frac{\partial v}{\partial t} - e(x,y)\frac{\partial^2 v}{\partial x^2} - f(x,y)\frac{\partial^2 v}{\partial y^2} + g(x,y)\frac{\partial v}{\partial x} + h(x,y)\frac{\partial v}{\partial y} = s(x,y,t,u,v), \tag{3.5.1б}$$

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

$$u(x,y,t) = \overline{\phi}(x,y,t), \quad v(x,y,t) = \overline{\overline{\phi}}(x,y,t), \quad (x,y,t) \in \partial\Omega \times (0,T] \tag{3.5.2}$$

и начални условия

$$u(x,y,0) = \overline{\psi}(x,y), \quad v(x,y,0) = \overline{\overline{\psi}}(x,y), \quad (x,y) \in \Omega, \tag{3.5.3}$$

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

Ще използваме индексираната двойка (i, j) да представим мрежовата точка (x_i, y_j) и дефинираме

$$u_{i,j} = u(x_i, y_j, t), \;\; v_{i,j} = v(x_i, y_j, t), \;\; r_{i,j} = r(x_i, y_j, t, u_{i,j}, v_{i,j}), \quad \text{ect.}$$

За $w = u, v$ представяме централния диференчен оператор

$$\delta_x w_{i,j} = (w_{i+1,j} - w_{i-1,j})/(2h_x), \qquad \delta_y w_{i,j} = (w_{i+1,j} - w_{i-1,j})/(2h_y), \quad (3.5.4)$$

$$\delta_x^2 w_{i,j} = (w_{i+1,j} - 2w_{i,j} + w_{i-1,j})/h_x^2, \qquad \delta_y^2 w_{i,j} = (w_{i,j+1} - 2w_{i,j} + w_{i,j-1})/h_y^2. \quad (3.5.5)$$

3.5.1 Втори ред полудискретизация по пространството

Прилагането на диференчните оператори (3.5.4) към системата (3.5.1) за всяка точка $(i, j) \in \Omega_h$ води до

$$\begin{aligned} \left.\frac{\partial u}{\partial t}\right|_{(x_i,y_j)} - a_{i,j}\delta_x^2 u_{i,j} - b_{i,j}\delta_y^2 u_{i,j} + c_{i,j}\delta_x u_{i,j} + d_{i,j}\delta_y u_{i,j} + \chi_{i,j,1} &= r_{i,j}, \\ \left.\frac{\partial v}{\partial t}\right|_{(x_i,y_j)} - e_{i,j}\delta_x^2 v_{i,j} - f_{i,j}\delta_y^2 v_{i,j} + g_{i,j}\delta_x v_{i,j} + h_{i,j}\delta_y v_{i,j} + \chi_{i,j,2} &= s_{i,j}, \end{aligned}$$

където локалните грешки от апроксимация (LTE) $\chi_{i,j,1}$ и $\chi_{i,j,2}$ са

$$\chi_{i,j,1} = \frac{h_x^2}{12}\left(2c\frac{\partial^3 u}{\partial x^3} - a\frac{\partial^4 u}{\partial x^4}\right)_{i,j} + \frac{h_y^2}{12}\left(2d\frac{\partial^3 u}{\partial y^3} - b\frac{\partial^4 u}{\partial y^4}\right)_{i,j} + \mathcal{O}(h_x^4 + h_y^4), \quad (3.5.6)$$

$$\chi_{i,j,2} = \frac{h_x^2}{12}\left(2g\frac{\partial^3 v}{\partial x^3} - e\frac{\partial^4 v}{\partial x^4}\right)_{i,j} + \frac{h_y^2}{12}\left(2h\frac{\partial^3 v}{\partial y^3} - f\frac{\partial^4 v}{\partial y^4}\right)_{i,j} + \mathcal{O}(h_x^4 + h_y^4). \quad (3.5.7)$$

След изпускането на членовете с грешките от апроксимация полу-дискретна централна диференчна апроксимация от втори ред за (3.5.1) приема вида:

$$\begin{aligned} \left.\frac{\partial u^h}{\partial t}\right|_{(x_i,y_j)} - a_{i,j}\delta_x^2 u_{i,j}^h - b_{i,j}\delta_y^2 u_{i,j}^h + c_{i,j}\delta_x u_{i,j}^h + d_{i,j}\delta_y u_{i,j}^h &= r_{i,j}^h, \quad (3.5.8) \\ \left.\frac{\partial v^h}{\partial t}\right|_{(x_i,y_j)} - e_{i,j}\delta_x^2 v_{i,j}^h - f_{i,j}\delta_y^2 v_{i,j}^h + g_{i,j}\delta_x v_{i,j}^h + h_{i,j}\delta_y v_{i,j}^h &= s_{i,j}^h, \end{aligned}$$

където за $(i,j) \in \Omega_h$

$$u^h_{i,j} \approx u(x_i, y_j, t), \qquad v^h_{i,j} \approx v(x_i, y_j, t),$$
$$r^h_{i,j} \approx r(x_i, y_j, t, u^h_{i,j}, v^h_{i,j}), \qquad s^h_{i,j} \approx s(x_i, y_j, t, u^h_{i,j}, v^h_{i,j}).$$

Сега разглеждаме матрично представяне на системата (3.5.8). Подреждаме точките на мрежата лексикографски отляво надясно по посока на x и отдолу нагоре по посока на y. Като изключим граничните точки $(i,j) \in \partial\Omega_h$, for $j = 1, 2, ..., N_y - 1$ дефинираме следните $(N_x - 1)$ мерни вектори:

$$U^h_j = \left(u^h_{1,j}, u^h_{2,j}, ..., u^h_{N_x-1,j}\right), \qquad V^h_j = \left(v^h_{1,j}, v^h_{2,j}, ..., v^h_{N_x-1,j}\right),$$
$$R_j(U^h_j, V^h_j) = \left(R_{1,j}, R_{2,j}, ..., R_{N_x-1,j}\right), \qquad S_j(U^h_j, V^h_j) = \left(S_{1,j},\ S_{2,j}, ..., S_{N_x-1,j}\right)$$

и тогава

$$U = \left(U^h_1, U^h_2, ..., U^h_{N_y-1}\right)^T, \qquad V = \left(V^h_1, V^h_2, ..., V^h_{N_y-1}\right)^T,$$
$$R = \left(R_1, R_2, ..., R_{N_y-1}\right)^T, \qquad S = \left(S_1, S_2, ..., S_{N_y-1}\right)^T.$$

Тогава пренаписваме системата (3.5.8) като система от ОДУ:

$$\frac{d}{dt}U + \overline{P}U = R + \overline{\Phi}, \quad t \in (0,T), \tag{3.5.9}$$
$$\frac{d}{dt}V + \overline{\overline{P}}V = S + \overline{\overline{\Phi}}, \quad t \in (0,T) \tag{3.5.10}$$

с начални условия $U(0)$ и $V(0)$, получени от $\overline{\psi}$ и $\overline{\overline{\psi}}$ за $(i,j) \in \Omega_h$ след пренареждането. В (3.5.9) матрицата $\overline{P}$ е $(N_y - 1) \times (N_y - 1)$ блочно тридиагонална матрица, $\overline{P} = tridiag(\overline{P}_{k,k-1}, \overline{P}_{k,k}, \overline{P}_{k,k+1})$ и $\overline{P}_{k,l}$, $l = k-1, k, k+1$ са тридиагонални матрици за $l = k$ и диагонални за $l = k \pm 1$ с размер $(N_x - 1) \times (N_x - 1)$. Нека за две естествени числа m и M, $m < M$ означаваме $m : M = m, m+1, ..., M$ и $\mathbf{p}_{k,m:M}$ е вектор с елементи $\mathbf{p}_{k,m:M} = (p_{k,m}, p_{k,m+1}, ..., p_{k,M})$. Тогава от (3.6.5) елементите на $\overline{P}_{k,l}$ са

$$\overline{P}_{k,l} = tridiag(\mathbf{p}^{(-1,\varepsilon)}_{k,2:N_x-1}, \mathbf{p}^{(0,\varepsilon)}_{k,2:N_x}, \mathbf{p}^{(1,\varepsilon)}_{k,1:N_x-2}) \qquad l = k + \varepsilon, \ \ \varepsilon = 0, \pm 1\,, \tag{3.5.11}$$

където

$$\begin{aligned} p_{i,j}^{(\pm 1,0)} &= \pm\frac{c(i,j)}{2h_x} - \frac{a(i,j)}{h_x^2}\,, \\ p_{i,j}^{(0,\pm 1)} &= \pm\frac{d(i,j)}{2h_y} - \frac{b(i,j)}{h_y^2}\,, \\ p_{i,j}^{(0,0)} &= 2\frac{a(i,j)}{h_x^2} + 2\frac{b(i,j)}{h_y^2}\,. \end{aligned} \tag{3.5.12}$$

Заменяйки $a \leftrightarrow e$, $b \leftrightarrow f$, $c \leftrightarrow g$ и $d \leftrightarrow h$, по същия начин получаваме елементите на $\overline{\overline{P}}$.

Векторите $\overline{\Phi}$ и $\overline{\overline{\Phi}}$ в (3.5.9)-(3.5.10) са свързани с граничните функции и също зависят от времето t.

3.5.2 Пълна дискретизация

За дискретизация по времето използваме схема с тегло θ. Нека $\omega_\tau = \{t_n = n\tau,\ \ n = 0, 1, \ldots, N,\ \ \tau = T/N\}$ са равномерни мрежи по времето със стъпка τ. Тогава тегловата θ-дискретизация на (3.5.9), (3.5.10) е както следва,

$$\begin{aligned} \frac{U^{n+1} - U^n}{\tau} + \overline{P}U^{n,\theta} &= R^{n,\theta} + \overline{\Phi}^{n,\theta}, \quad t \in (0,T), \\ \frac{V^{n+1} - V^n}{\tau} + \overline{\overline{P}}V^{n,\theta} &= S^{n,\theta} + \overline{\overline{\Phi}}^{n,\theta}, \quad t \in (0,T), \end{aligned} \tag{3.5.13}$$

където $Z^{n,\theta} = \theta Z^{n+1} + (1-\theta)Z^n$ for $Z = U, V, R, S, \overline{\Phi}, \overline{\overline{\Phi}}$, $Z^n \approx Z(t_n)$ and $0 \le \theta \le 1$, $n = 0, 1, \ldots N-1$. За $\theta = 1$ се получава напълно неявна диференчна схема, за $\theta = 0$ - явна и за $\theta = 1/2$ - схема на Кранк-Никълсън. Последната схема има предимството, че е от втори ред на точност по времето и затова в числените експерименти използваме основно $\theta = 1/2$.

За $\theta > 0$ диференчните схеми изискват решаването на нелинейни алгебрични системи. Накратко описваме приложението на метода на Нютон за задачата (3.6.13). За да приложим класическия метод на Нютон записваме системата (3.6.13) във формата $\Upsilon(W) = 0$, където $W = [U^T, V^T]^T$ е вектор с дължина $2(N_x-1)(N_y-1)$. Поставяме $\overset{0}{W}{}^{n+1}$ като начално приближение на новия слой по времето $t = t_{n+1}$ да бъде численото решение на стария слой по времето $t = t_n$. Тогава, за да намерим решението на $t = t_{n+1}$, итерационен процес с подходящ

стоп критерий е използван:

$$\begin{cases} \Upsilon'(\overset{k}{W}{}^{n+1}) \overset{k}{\Delta} = -\Upsilon(\overset{k}{W}{}^{n+1}), \\ \overset{k+1}{W}{}^{n+1} = \overset{k}{W}{}^{n+1} + \overset{k}{\Delta}. \end{cases} \tag{3.5.14}$$

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

$$\Upsilon'(\overset{k}{W}{}^{n+1}) = \frac{\partial \Upsilon}{\partial W} = \left(\begin{array}{cc} \frac{1}{\tau} I + \frac{1}{2}\overline{P} - \frac{1}{2}\frac{\partial R}{\partial U} & \frac{1}{2}\frac{\partial R}{\partial V} \\ \frac{1}{2}\frac{\partial S}{\partial U} & \frac{1}{\tau} I + \frac{1}{2}\overline{\overline{P}} - \frac{1}{2}\frac{\partial S}{\partial V} \end{array} \right) \Bigg|_{(U,V)=(\overset{k}{U},\overset{k}{V})}. \tag{3.5.15}$$

В числените експерименти за да решим първия ред в (3.6.14), който е линейна система от $2(N_x - 1)(N_y - 1)$ уравнения, използване така наречения „неточен метод“ на Нютон (inexact Newton) [49], т.е. решаваме системата приближено с вградената MatLab функция bicgstab(l) (biconjugate gradients stabilized (l) method), който дава най-добрите резултати за числените експерименти по отношение на брой вътрешни операции и изчислително време. Това е един от най-бързите известни градиентни методи за линейни системи, съответно най-добрия вграден метод за СЛАУ в Matlab, и е подробно разгледан в редица публикации [187, 188, 222]. В числените експерименти може да се използва и конструирания нов метод Монте Карло за линейни системи в предишната глава, но е значително по бавен за системата от 10 уравнения и това води до голямо нарастване на изчислителното време при увеличаване на броя слоеве по времето. В последната секция на тази глава се разглеждат примери, в които е приложен и Монте Карло алгоритъма за линейни системи.

3.6 Компактни диференчни схеми в двумерния случай

В тази секция се описва построяването на компактни диференчни схеми (CFDS) в двумерния случай за системата от две уравнения (3.5.1).

3.6.1 Дискретизация по пространството

За да елиминираме членовете от ред $\mathcal{O}(h_x^2 + h_y^2)$ в (3.5.6) диференцираме уравнението (3.5.1a) два пъти по x и получаваме изрази 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 по-големи подробности тази процедура е описана в [217].

Нека

$$\tilde{a}_{i,j} = (c_{i,j} + 2\delta_x a_{i,j})/a_{i,j}, \quad \tilde{b}_{i,j} = (d_{i,j} + 2\delta_y b_{i,j})/b_{i,j}, \quad (i,j) \in \Omega_h \ .$$

Нека също

$$\begin{aligned}
\alpha_{i,j} &= a_{i,j} + \frac{h_x^2}{12}\left(\delta_x^2 a_{i,j} - \tilde{a}_{i,j}(\delta_x a_{i,j} - c_{i,j}) - 2\delta_x c_{i,j}\right) + \frac{h_y^2}{12}\left(\delta_y^2 a_{i,j} - \tilde{b}_{i,j}\delta_y a_{i,j}\right), \\
\beta_{i,j} &= b_{i,j} + \frac{h_x^2}{12}\left(\delta_x^2 b_{i,j} - \tilde{a}_{i,j}\delta_x b_{i,j}\right) + \frac{h_y^2}{12}\left(\delta_y^2 b_{i,j} - \tilde{b}_{i,j}(\delta_y b_{i,j} - d_{i,j}) - 2\delta_y d_{i,j}\right), \\
\tilde{\alpha}_{i,j} &= c_{i,j} + \frac{h_x^2}{12}\left(\delta_x^2 c_{i,j} - \tilde{a}_{i,j}\delta_x c_{i,j}\right) + \frac{h_y^2}{12}\left(\delta_y^2 c_{i,j} - \tilde{b}_{i,j}\delta_y c_{i,j}\right), \\
\tilde{\beta}_{i,j} &= d_{i,j} + \frac{h_x^2}{12}\left(\delta_x^2 d_{i,j} - \tilde{a}_{i,j}\delta_x d_{i,j}\right) + \frac{h_y^2}{12}\left(\delta_y^2 d_{i,j} - \tilde{b}_{i,j}\delta_y d_{i,j}\right),
\end{aligned}$$

и

$$\begin{aligned}
\theta_{i,j} &= \frac{h_y^2}{12}c_{i,j} - \frac{h_x^2}{12}(2\delta_x b_{i,j} - \tilde{a}_{i,j}b_{i,j}), \quad \tilde{\theta}_{i,j} = \frac{h_x^2}{12}d_{i,j} - \frac{h_y^2}{12}(2\delta_y a_{i,j} - \tilde{b}_{i,j}a_{i,j}), \\
\gamma_{i,j} &= \frac{h_x^2}{12}b_{i,j} + \frac{h_y^2}{12}a_{i,j}, \quad \tilde{\gamma}_{i,j} = \frac{h_x^2}{12}(2\delta_x - \tilde{a}_{i,j}d_{i,j}) + \frac{h_y^2}{12}(2\delta_y c_{i,j} - \tilde{b}_{i,j}c_{i,j}).
\end{aligned}$$

Дефинираме диференчните оператори

$$\begin{aligned}
l_{i,j}^h &= -\alpha_{i,j}\delta_x^2 - \beta_{i,j}\delta_y^2 + \tilde{\alpha}_{i,j}\delta_x + \tilde{\beta}_{i,j}\delta_y - \gamma_{i,j}\delta_x^2\delta_y^2 + \theta_{i,j}\delta_x\delta_y^2 + \tilde{\theta}_{i,j}\delta_x^2\delta_y + \tilde{\gamma}_{i,j}\delta_x\delta_y \\
\nu_{i,j}^h &= 1 + \frac{h_x^2}{12}(\delta_x^2 - \tilde{a}_{i,j}\delta_x) + \frac{h_y^2}{12}(\delta_y^2 - \tilde{b}_{i,j}\delta_y).
\end{aligned}$$

Прилагайки тези оператори към (3.5.1a) получаваме

$$l_{i,j}^h u_{i,j} = \nu_{i,j}^h (r_{i,j} - u_{t,i,j}) + \mathcal{O}(h_x^4 + h_x^2 h_y^2 + h_y^4). \tag{3.6.1}$$

За удобство, въвеждаме операторите

$$\overline{\mathcal{P}}_{i,j}^h = 6h_x^2 l_{i,j}^h, \quad \overline{\mathcal{Q}}_{i,j}^h = 6h_x^2 \nu_{i,j}^h, . \tag{3.6.2}$$

Нека $\sigma = h_x/h_y$ е отношението на мрежовите стъпки. Тогава

$$\overline{\mathcal{P}}^h_{i,j} u_{i,j} = \sum_{k_1=-1}^{1} \sum_{k_2=-1}^{1} p^{(k_1,k_2)}_{i,j} u_{i+k_1,j+k_2}, \tag{3.6.3}$$

$$\overline{\mathcal{Q}}^h_{i,j} u_{i,j} = \sum_{k_1=-1}^{1} \sum_{k_2=-1}^{1} q^{(k_1,k_2)}_{i,j} u_{i+k_1,j+k_2}, \tag{3.6.4}$$

където

$$\begin{aligned}
p^{(\pm1,-1)}_{i,j} &= -\frac{a_{i,j}+\sigma^2 b_{i,j}}{2} \pm \frac{1}{4}\left(c_{i,j} - \sigma^2(2\delta_x b_{i,j} - \tilde{a}_{i,j} b_{i,j}) \mp \sigma d_{i,j} \pm \frac{1}{\sigma}(2\delta_y a_{i,j} - \tilde{b}_{i,j} a_{i,j})\right) h_x \\
&\quad \mp \frac{1}{8}\left(\sigma(2\delta_x - \tilde{a}_{i,j} d_{i,j}) + \frac{1}{\sigma}(2\delta_y c_{i,j} - \tilde{b}_{i,j} c_{i,j})\right) h_x^2, \\
p^{(\pm1,1)}_{i,j} &= -\frac{a_{i,j}+\sigma^2 b_{i,j}}{2} \pm \frac{1}{4}\left(c_{i,j} - \sigma^2(2\delta_x b_{i,j} - \tilde{a}_{i,j} b_{i,j}) \pm \sigma d_{i,j} \mp \frac{1}{\sigma}(2\delta_y a_{i,j} - \tilde{b}_{i,j} a_{i,j})\right) h_x \\
&\quad \pm \frac{1}{8}\left(\sigma(2\delta_x - \tilde{a}_{i,j} d_{i,j}) + \frac{1}{\sigma}(2\delta_y c_{i,j} - \tilde{b}_{i,j} c_{i,j})\right) h_x^2, \\
p^{(\pm1,0)}_{i,j} &= \sigma^2 b_{i,j} - 5a_{i,j} \pm \left(3\tilde{\alpha}_{i,j} - \frac{1}{2} c_{i,j} + \frac{\sigma^2}{2}(2\delta_y c_{i,j} - \tilde{b}_{i,j} c_{i,j})\right) h_x \\
&\quad -\frac{1}{2}\left(\delta_x^2 a_{i,j} - \tilde{a}_{i,j}(\delta_x a_{i,j} - c_{i,j}) - 2\delta_x c_{i,j} + \frac{1}{\sigma^2}(\delta_y^2 a_{i,j} - \tilde{b}_{i,j}\delta_y a_{i,j})\right) h_x^2 \\
p^{(0,\pm1)}_{i,j} &= a_{i,j} - 5\sigma^2 b_{i,j} \pm \left(3\sigma\tilde{\beta}_{i,j} - \frac{\sigma}{2} d_{i,j} + \frac{1}{2\sigma}(2\delta_y a_{i,j} - \tilde{b}_{i,j} a_{i,j})\right) h_x \\
&\quad -\frac{1}{2}\left(\sigma^2(\delta_x^2 b_{i,j} - \tilde{a}_{i,j}\delta_x b_{i,j}) + \delta_y^2 b_{i,j} - 2\delta_y d_{i,j} - \tilde{b}_{i,j}(\delta_y b_{i,j} - d_{i,j})\right) h_x^2, \\
p^{(0,0)}_{i,j} &= 10(a_{i,j} + \sigma^2 b_{i,j}) + \left(\delta_x^2 a_{i,j} - \tilde{a}_{i,j}(\delta_x a_{i,j} - c_{i,j}) - 2\delta_x c_{i,j} + \frac{1}{\sigma^2}(\delta_y^2 a_{i,j} - \tilde{b}_{i,j}\delta_y a_{i,j})\right) h_x^2 \\
&\quad + \left(\sigma^2(\delta_x^2 b_{i,j} - \tilde{a}_{i,j}\delta_x b_{i,j}) + \delta_y^2 b_{i,j} - 2\delta_y d_{i,j} - \tilde{b}_{i,j}(\delta_y b_{i,j} - d_{i,j})\right) h_x^2
\end{aligned} \tag{3.6.5}$$

и

$$q^{(\pm1,\pm1)}_{i,j} = 0,\ q^{(\pm1,0)}_{i,j} = \frac{1}{4}(2 \mp \tilde{a}_{i,j} h_x) h_x^2,\ q^{(0,\pm1)}_{i,j} = \frac{1}{4}(2 \mp \frac{\tilde{a}_{i,j}}{\sigma} h_x) h_x^2,\ q^{(0,0)}_{i,j} = 4h_x^2. \tag{3.6.6}$$

С тези означения, след като изпуснем члена $\mathcal{O}(h_x^4 + h_x^2 h_y^2 + h_y^4)$ в (3.6.1), полудискретната компактна апроксимация на (3.5.1a) е, както следва:

$$\begin{cases}
\overline{\mathcal{P}}^h_{i,j} u^h_{i,j} = \overline{\mathcal{Q}}^h_{i,j}\left(r^h_{i,j} - \frac{d}{dt} u^h_{i,j}\right), & (i,j) \in \Omega_h, \quad t \in (0,T], \\
u^h_{i,j} = \overline{\phi}_{i,j}, & (i,j) \in \partial\Omega_h, \quad t \in (0,T], \\
u^h_{i,j} = \overline{\psi}_{i,j}, & (i,j) \in \overline{\Omega}_h, \quad t = 0.
\end{cases} \tag{3.6.7}$$

По-същия начин се прави с уравнението (3.5.1б). Заменяйки $a_{i,j}$, $b_{i,j}$, $c_{i,j}$, $d_{i,j}$ с $e_{i,j}$, $f_{i,j}$, $g_{i,j}$, $h_{i,j}$ и $\overline{\mathcal{P}}^h_{i,j}$, $\overline{\mathcal{Q}}^h_{i,j}$ с $\overline{\overline{\mathcal{P}}}^h_{i,j}$, $\overline{\overline{\mathcal{Q}}}^h_{i,j}$ получаваме втората част на полудискретната нелинейна система

$$\begin{cases} \overline{\overline{\mathcal{P}}}^h_{i,j} v^h_{i,j} = \overline{\overline{\mathcal{Q}}}^h_{i,j} \left(s^h_{i,j} - \frac{d}{dt} v^h_{i,j} \right), & (i,j) \in \Omega_h, \quad t \in (0,T], \\ v^h_{i,j} = \overline{\overline{\phi}}_{i,j}, & (i,j) \in \partial\Omega_h, \quad t \in (0,T], \\ v^h_{i,j} = \overline{\overline{\psi}}_{i,j}, & (i,j) \in \overline{\Omega}_h, \quad t = 0. \end{cases} \tag{3.6.8}$$

Сега ще бъде представен матричен запис на системата (3.6.7), (3.6.8). Получаваме следната система ОДУ

$$\overline{Q} \frac{d}{dt} U^h + \overline{P} U^h = \overline{Q} R + \overline{\Phi}, \quad t \in (0,T], \tag{3.6.9}$$

$$\overline{\overline{Q}} \frac{d}{dt} V^h + \overline{\overline{P}} V^h = \overline{\overline{Q}} S + \overline{\overline{\Phi}}, \tag{3.6.10}$$

с начални условия $U^h(0)$ и $V^h(0)$ получени от $\overline{\psi}$ и $\overline{\overline{\psi}}$ за $(i,j) \in \Omega_h$ след преназначаването. В системата (3.6.9), (3.6.10) матрицата $\overline{P}$ (аналогично $\overline{\overline{P}}$) е $(N_y - 1) \times (N_y - 1)$ блочно-тридиагонална матрица $\overline{P} = tridiag(\overline{P}_{k,k-1}, \overline{P}_{k,k}, \overline{P}_{k,k+1})$ и $\overline{P}_{k,l}$, $l = k-1, k, k+1$ са също тридиагонални матрици от ред $(N_x - 1) \times (N_x - 1)$. Тогава от (3.6.5) елементите на $\overline{P}_{k,l}$ са

$$\overline{P}_{k,l} = tridiag(p^{(-1,\varepsilon)}_{k,2:N_x-1}, p^{(0,\varepsilon)}_{k,2:N_x}, p^{(1,\varepsilon)}_{k,1:N_x-2}) \qquad l = k + \varepsilon, \;\; \varepsilon = 0, \pm 1 \, . \tag{3.6.11}$$

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

$$\overline{Q}_{k,l} = tridiag(q^{(-1,\varepsilon)}_{k,2:N_x-1}, \; q^{(0,\varepsilon)}_{k,2:N_x}, \; q^{(1,\varepsilon)}_{k,1:N_x-2}) \qquad l = k + \varepsilon, \;\; \varepsilon = 0, \pm 1 \tag{3.6.12}$$

със забележката, че за $\varepsilon = \pm 1$ матриците $\overline{Q}_{k,l}$ са диагонални (вместо тридиагонални) матрици, виж (3.6.6).

Векторите $\overline{\Phi}$ и $\overline{\overline{\Phi}}$ са свързани с граничните функции и също зависят от времето t.

3.6.2 Дискретизация по времето

За дискретизация по времето е използвана схема с тегла с $\theta = 1/2$. Тогава пълната дискретизация по Кранк-Никълсън за (3.6.9), (3.6.10) е както следва:

$$\overline{Q} \frac{U^{n+1} - U^n}{\tau} + \overline{P} U^{n,\theta} = \overline{Q} R^{n,\theta} + \overline{\Phi}^{n,\theta}, \quad n = 1, ..., N-1, \tag{3.6.13}$$

$$\overline{\overline{Q}} \frac{V^{n+1} - V^n}{\tau} + \overline{\overline{P}} V^{n,\theta} = \overline{\overline{Q}} S^{n,\theta} + \overline{\overline{\Phi}}^{n,\theta}, \quad n = 1, ..., N-1.$$

Аналогично на случая на централната диференчна схема се прилага „неточния метод“ на Нютон (inexact Newton), т.е. решаваме системата приближено с вградената MatLab функция bicgstab(l) [236], който дава най-добрите резултати за числените експерименти по отношение на брой вътрешни операции и изчислително време и е няколко пъти по-бърз от метода Монте Карло за линейни системи IWE. Сравнение между двата метода е дадено в последната секция на тази глава.

Системата (3.6.13) 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 $\overset{0}{W}{}^{n+1}$ 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 подходящ стоп критерии е използван:

$$\begin{cases} \Upsilon'(\overset{k}{W}{}^{n+1}) \overset{k}{\Delta} = -\Upsilon(\overset{k}{W}{}^{n+1}), \\ \overset{k+1}{W}{}^{n+1} = \overset{k}{W}{}^{n+1} + \overset{k}{\Delta}. \end{cases} \tag{3.6.14}$$

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

$$\Upsilon'(\overset{k}{W}{}^{n+1}) = \left(\begin{array}{cc} \frac{1}{\tau}\overline{Q}I + \frac{1}{2}\overline{P} - \frac{1}{2}\overline{Q}\frac{\partial R}{\partial U} & \frac{1}{2}\overline{Q}\frac{\partial R}{\partial V} \\ \frac{1}{2}\overline{\overline{Q}}\frac{\partial S}{\partial U} & \frac{1}{\tau}\overline{\overline{Q}}I + \frac{1}{2}\overline{\overline{P}} - \frac{1}{2}\overline{\overline{Q}}\frac{\partial S}{\partial V} \end{array} \right) \Bigg|_{(U,V)=(\overset{k}{U},\overset{k}{V})}. \tag{3.6.15}$$

3.7 Числени експерименти за моделна задача от Датския Ойлеров модел

В тази секция се разглеждат два числени експеримента за потвърждение на теоретичните резултати. Първата задача е с точно решение, а втората е опростена двумерна моделна задача за пренос на замърсители, от Датския Ойлеров модел, описана в [85, 227].

3.7.1 Пример 1 (известно аналитично решение)

Тук разглеждаме задачата:

$$\frac{\partial u_l}{\partial t} - K\triangle u_l + \mathbf{b}_l.\nabla u_l = R_l(x, y, \mathbf{u}) + \xi_l(x, y, t), (x, y, t) \in \Omega \times (0, T]. \tag{3.7.1}$$

Функциите ξ_l, $l = 1, ..., 10$, и началните и граничните условия са избрани, така че точното решение да бъде

$$u_l = \exp(-t/T) sin(\frac{\pi x}{X}) sin(\frac{\pi y}{Y}), \quad l = 1, ..., 10, \quad (x, y, t) \in \partial\Omega \times [0, T]. \quad (3.7.2)$$

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 (максимална) 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 $ratio$:

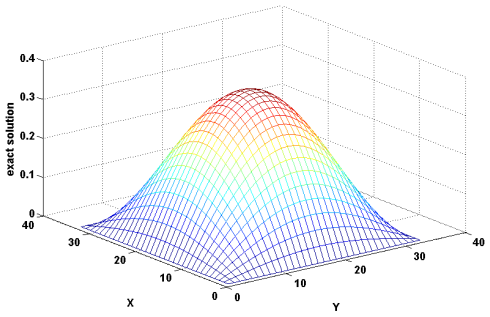
$$ratio = ratio_{M,l/2M,l} =: error_{M,l}/error_{2M,l}.$$

В Таблица 3.4 са представени анализи със сгъстяване на мрежата(the mesh refinement analyses) с използването на CDS и CFDS. Резултатите потвърждават теоретичния ред на сходимост, т.е. отношението близо до 4 потвърждава втори ред за CDS и отношението близо до 16 - четвърти ред за CFDS. Също, тъй като CFDS има грешка $O(h^4 + \tau^2)$, за да се наблюдава четвърти ред, когато удвояваме броя на мрежовите точки по пространството, трябва да се очетворяват мрежовите точки по времето. Предимството на CFDS се вижда от представеното CPU време - необходимо е по-малко време на CFDS за получаването на резултати с по-добра точност вместо използването на повече слоеве по времето. На Фиг. 3.1 точното решение за финалното време T за u_1 и мрежови параметри $M_x = M_y = 32$, $N = 32$ е представено. На Фиг. 3.2 грешката, получена за a) CDS за $M_x = M_y = 32$, $N = 256$ и за b) CFDS $M_x = M_y = 32$, $N = 256$, е представена.

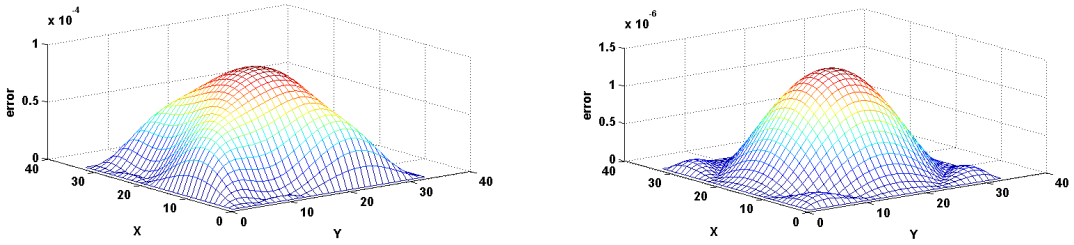
В Таблица 3.5 са представени анализи със сгъстяване на мрежата с използването на CDS и CFDS с екстраполация по Ричардсон (RE) по пространството (с използването на съответните тегла от (3.4.3) и (3.4.4)). Отново, за да се наблюдава четвърти и шести ред на CDSRE и CFDSRE, удвоявайки мрежовите

Таблица 3.4: Сравнение на абсолютна грешка равномерна норма 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

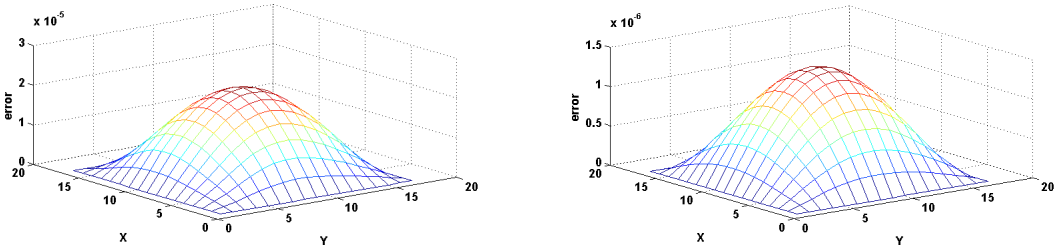


Фигура 3.1: Точното решение



(a) (b)

Фигура 3.2: Грешката в равномерна норма за *Пример 1*: (a) CDS с параметри $M_x = M_y = 32$, $N = 32$; (b) CFDS за $M_x = M_y = 32$, $N = 256$



(a) (b)

Фигура 3.3: Грешка в равномерна норма за *Пример 1*: (a) CDS с RE по пространството $M_x = M_y = 16$, $N = 64$; (b) CFDS с RE по пространството и $M_x = M_y = 16$, $N = 256$

точки по пространството, трябва да бъдат взети слоеве по времето четири или осем пъти повече в сравнение с предишния експеримент. Резултатите потвърждават очаквания ред на сходимост за конструираните числени методи. Отношение близо до 64 съответства на шести ред за CFDSRE. Сравняването на CPU времето за Таблица 3.4 и Таблица 3.5 показва предимство в използването на RE с получаването на по-малки грешки за по-кратко време, независимо че RE има нужда да изчисли численото решение на две последователни мрежи. Предимството на CFDS с RE също е очевидно. На Фиг. 3.3 грешката, получена със a) CDS с RE за $M_x = M_y = 16$, $N = 64$ и с b) CFDS с RE по пространството и $M_x = M_y = 16$, $N = 256$, са представени.

Таблица 3.5: Сравнение на грешките в равномерна норма за **Пример 1** за CDS и CFDS с екстраполация по Ричардсон 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	ред	CPU	M_x	M_y	N	err_N	ред	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

В Таблица 3.8 са представени анализи със сгъстяване на мрежата с използването на CDS и CFDS с (RE) по пространството и времето. Отново, за да се наблюдава четвърти и шести ред на CDSRE и CFDSRE, удвоявайки мрежовите

точки по пространството, трябва да се вземе брой слоеве по времето два и осем пъти повече в сравнение с предишния експеримент. Това може да доведе до много голямо нарастване на CPU времето за случая на CFDS и затова тук вземаме само четири пъти (вместо осем пъти) по-малко интервали по времето. Резултатите потвърждават теоретичния ред на сходимост за двата числени метода. От сравняването на CPU времето за Таблици 3.4 , 3.5 и 3.8 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. 3.4 показва грешката в равномерна норма за *Пример 1* (a) със CDS и RE по пространството и времето за $M_x = M_y = 16$, $N = 16$; (b) с CFDS и RE по пространството и времето за $M_x = M_y = 16$, $N = 64$ и е в съответствие с резултатите в Таблица 3.8.

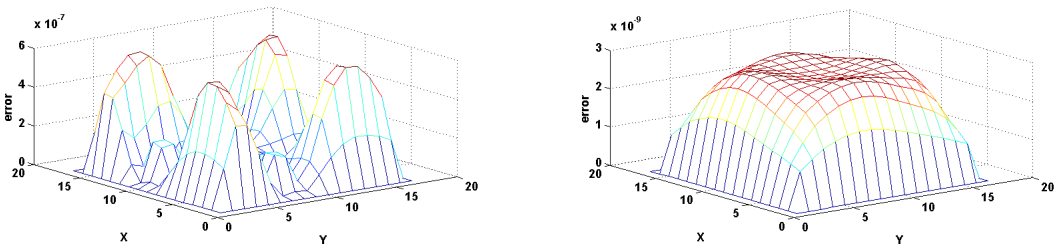
Таблица 3.6: Абсолютна грешка в равномерна норма за **Пример 1** за CDS и CFDS в RE по пространството и времето

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

3.7.2 Пример 2 (без точно решение)

В този случай е разгледан по-реалистичен вариант на задачата (3.2.1)-(3.2.3) 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 броя на уравненията е $L = 10$. Началните условия на първия слой по времето $t = 0$ са константни функции

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



(a) (b)

Фигура 3.4: Грешка в равномерна норма за *Пример 1*: (a) CDS с RE по пространството и времето при $M_x = M_y = 16$, $N = 16$; (b) CFDS с RE по пространството и времето при $M_x = M_y = 16$, $N = 64$

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

$$\gamma_l(t) = const_l(sin(t/C) + 2),$$

където C е константа и константите $const_l$, $l = 1, ..., L$ са избрани по такъв начин, че да е осигурена съгласуваност на началните и граничните условия. Коефициентът на дифузия е избран да бъде $K = 1.8km^2/min$ и коефициентът μ е $\mu = 2\pi/(60 * T)$.

В този случай няма аналитично решение. Един възможен начин за пресмятане на реда на сходимост е метода на Рунге на три вложени мрежи. Тук използваме друга идея. Като „точно" решение се взема решението, получено с най-малките размери на мрежата по пространството. В следващите таблици това решение е означено с удебелен шрифт. Също в този случай представяме относителната грешка. За l-тата субстанция с $relerror_{M,l}$ означаваме относителната грешка в равномерна норма, получена на последния слой по времето $t_N = T$ за брой подинтервали по времето $M_x = M_y = M$:

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

Следим реда на сходимост, означен с $order$ и пресметнат с помощта на 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 сгъстяванията на мрежата.

В Таблица 3.7 са представени резултатите, получени с CDS за брой слоеве по времето $N = 256$ за първото и петото вещество u_1 и u_5 в централния възел с координати $(x_{M/2}, y_{M/2}) = (X/2, Y/2) = (250, 250)$. Вторият ред на сходимост е потвърден. Интересно е да се отбележи, че u_1 и u_5 нямат едни и същи стойности, но относителните грешки са приблизително едни и същи за двата замърсителя. Подобни резултати са представени в Таблица 3.8, но за точката $(x, y) = (X/6, Y/6) = (83.33, 83.33)$. Отново вторият ред по пространството на CDS може да се види.

Таблица 3.7: Числените стойности, относителната грешка и реда на сходимост за **Пример 2** за CDS в централния възел $(x, y) = (X/2, Y/2)$ със стъпки по времето $N = 256$ за първото и петото вещество u_1 и u_5

U_1					U_5				
M_x	M_y	числ. стойност	отн. грешка	ред	M_x	M_y	числ. стойност	отн. грешка	ред
8	8	1975.8824812	1.001 e-02	-	8	8	4523.2929772	1.001 e-03	-
16	16	1991.1436076	2.366 e-03	2.08	16	16	4558.2293785	2.366 e-03	2.08
24	24	1993.8130112	1.028 e-03	2.05	24	24	4564.3402809	1.028 e-03	2.05
32	32	1994.7306173	5.685 e-04	2.06	32	32	4566.4408997	5.684 e-04	2.06
40	40	1995.1523236	3.572 e-04	2.08	40	40	4567.4062858	3.572 e-04	2.08
48	48	1995.3806072	2.428 e-04	2.11	48	48	4567.9288813	2.428 e-04	2.11
56	56	1995.5179890	1.739 e-04	2.16	56	56	4568.2433808	1.740 e-04	2.16
64	64	1995.6070489	1.293 e-04	2.21	64	64	4568.4472602	1.293 e-04	2.21
192	192	**1995.8651853**			192	192	**4569.0381956**		

Експериментите със същите параметри са повторени с използването на CFDS. Резултатите са представени в Таблица 3.9 и Таблица 3.10. Четвъртият ред на сходимост в двата случая (централен възел (x,y)=(X/2,Y/2) и възел (x,y)=(X/6,Y/6)

Таблица 3.8: Числените стойности, относителната грешка и ред на сходимост за **Пример 2** за CDS за възела $(x, y) = (X/6, Y/6)$ с $N = 256$ за u_1 и u_5

U_1					U_5				
M_x	M_y	числ. стойност	отн. грешка	ред	M_x	M_y	числ. стойност	отн. грешка	ред
6	6	1068.4732730	4.271 e-02	-	6	6	2447.733406	4.203 e-02	-
12	12	1110.5572844	5.007 e-03	3.09	12	12	2542.3695918	4.998 e-03	3.07
24	24	1115.5372163	5.451 e-04	3.19	24	24	2553.7466839	5.450 e-04	3.20
48	48	1116.0563728	7.994 e-05	2.76	48	48	2554.9350740	7.992 e-05	2.77
96	96	1116.1455939	1.783 e-05	2.16	96	96	2555.1392795	1.782 e-05	2.16
192	192	**1116.1654919**	-		192	192	**2555.1848192**	-	

за двете вещества u_1 и u_5 е потвърден. Отново в централния възел относителните грешки са почти еднакви.

В Таблица 3.11 и Таблица 3.12 са показани резултатите, получени в CDSRE и CFDSRE по пространството. Броят на слоевете по времето е $N = 256$ и представените стойности са числените стойности на последния слой по времето $t_N = T$ в централния възел $(x, y) = (X/2, Y/2)$. Резултатите потвърждават четвърти ред по пространството за CDSRE и шести ред по пространството за CFDSRE.

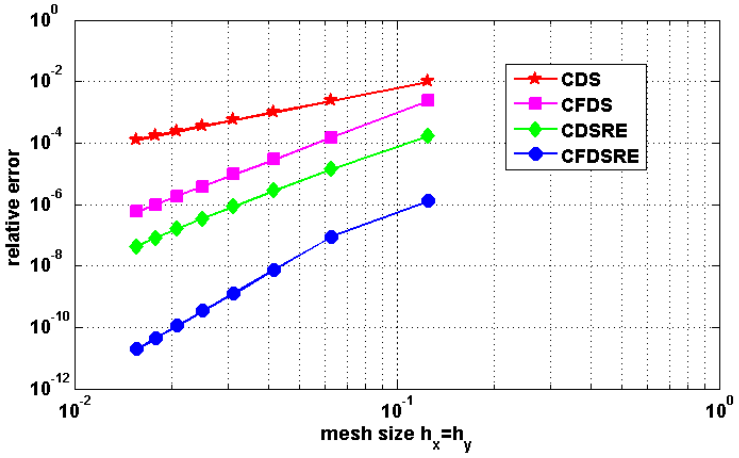
На Фиг. 3.5 е представена логаритмична графика на грешките и пространствените мрежови възли за *Пример 2*, получена с: CDS - червена линия; CFDS - виненочервена линия; CDSRE - зелена линия; CFDSRE - синя линия. Нарастването на наклона на линиите съответства на нарастването на реда на сходимост. Най-наклонената линия съответства на предимството на CFDS в комбинация с екстраполация по Ричардсон.

На Фиг. 3.6 е показано числеността 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. 3.7 е показано числеността решение получено с CFDS.

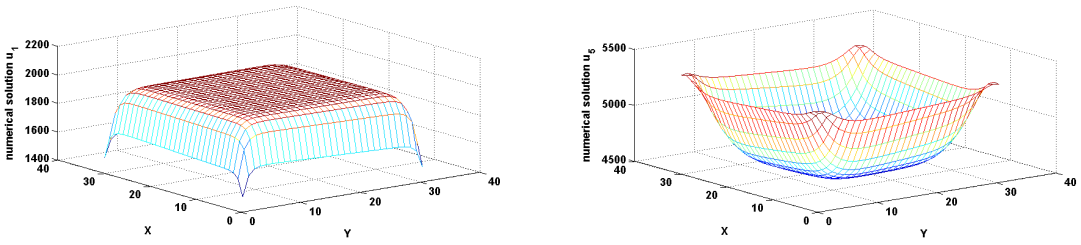
Проведени са и други експерименти. Интересно е да се види поведението на решението, ако коефициентите μ на конвекцията са $\mu = 2\pi/(X)$, както е в [121] вместо $\mu = 2\pi/(60T)$ както е в [85]. Това е представено на Фиг. 3.8. Нараст-

Таблица 3.9: Числените стойности, относителната грешка и ред 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. стойност	отн. грешка	ред	M_x	M_y	числ. стойност	отн. грешка	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**		

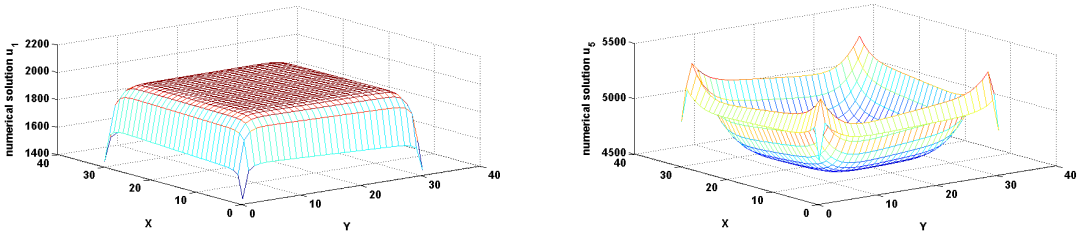


Фигура 3.5: Логаритмична графика на грешките и пространствените мрежови възли за *Пример 2*, получен с: CDS - червена линия, $-\star-$; CFDS - виненочервена линия, $-\blacksquare-$; CDSRE - зелена линия, $-\blacklozenge-$; CFDSRE - синя линия, $-\bullet-$.



(a) (b)

Фигура 3.6: Численото решение получено с CDS за $\mu = 2\pi/(60T)$ с мрежови параметри $M_y = M_y = 32$, $N = 256$ за *Пример 2*: (a) за u_1; (b) за u_5



(a) (b)

Фигура 3.7: 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

Таблица 3.10: Числените стойности, относителната грешка и реда на сходимост за **Пример 2** за CFDS за възела $(x, y) = (X/6, Y/6)$ със слоеве по времето $N = 256$ за u_1 и u_5

U_1					U_5				
M_x	M_y	числ. стойност	отн. грешка	ред	M_1	M_y	числ. стойност	отн. грешка	ред
6	6	1043.2932910	6.529 e-02	-	6	6	2257.9483166	1.163 e-01	-
12	12	1118.0804590	1.710 e-03	5.25	12	12	2550.1122599	1.991 e-03	5.86
24	24	1116.0780123	8.411 e-05	4.34	24	24	2554.9992916	7.834 e-05	4.66
48	48	1116.1660548	5.229 e-06	4.00	48	48	2555.1860847	5.236 e-06	3.91
96	96	1116.1715505	3.054 e-07	4.09	96	96	2555.1986800	3.068 e-07	4.09
192	192	**1116.1718914**	-		192	192	**2555.1994639**	-	

Таблица 3.11: Числените стойности, относителната грешка и ред на сходимост за **Пример 2** за CDS с RE по пространството за централния възел $(x, y) = (X/2, Y/2)$ със слоеве по времето $N = 256$

U_1					U_5				
M_x	M_Y	числ. стойност	отн. грешка	ред	M_x	M_Y	числ. стойност	отн. грешка	ред
8	8	1996.2306498	1.669 e-04	-	8	8	4569.8748455	1.669 e-04	-
16	16	1995.926287	1.446 e-05	3.529	16	16	4569.1780734	1.446 e-05	3.529
24	24	1995.9031392	2.862 e-06	3.995	24	24	4569.1250814	2.863 e-06	3.994
32	32	1995.8991928	8.856 e-07	4.079	32	32	4569.1160470	8.855 e-07	4.078
40	40	1995.8981287	3.524 e-07	4.129	40	40	4569.1136111	3.524 e-07	4.129
48	48	1995.8977510	1.632 e-07	4.225	48	48	4569.1127463	1.631 e-07	4.224
56	56	1995.8975904	8.268 e-08	4.409	56	56	4569.1123786	8.267 e-08	4.409
64	64	1995.8975129	4.386 e-08	4.748	64	64	4569.1122012	4.385 e-08	4.749
96	96	**1995.8974254**			96	96	**4569.1120009**		

ването на конвективните членове води до съществено изменение на численото решение близо до границата. Може да се види, че константните начални стойности остават относително непроменени в средата на областта, но са разгънати

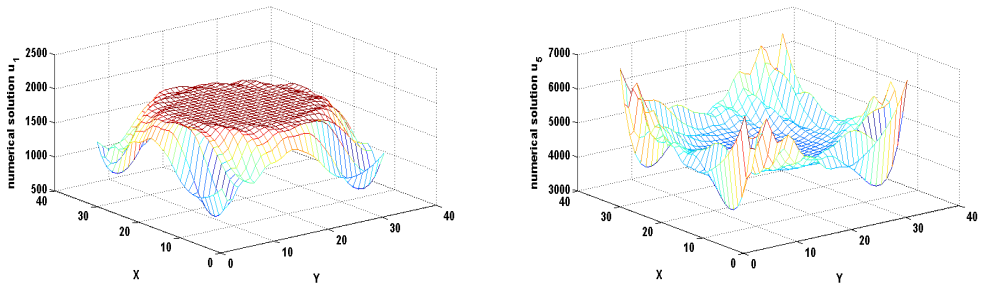
Таблица 3.12: Числените стойности, относителната грешка и ред на сходимост за **Пример 2** за CFDS с RE по пространството за централния възел $(x, y) = (X/2, Y/2)$ със стъпки по времето $N = 256$

U_1					U_5				
M_x	M_y	числ. стойност	отн. грешка	ред	M_x	M_Y	числ. стойност	отн. грешка	ред
8	8	1995.89999599	1.299 e-06	-	8	8	4569.10452815	1.624 e-06	-
16	16	1995.89757927	8.779 e-08	3.887	16	16	4569.11235256	8.767 e-08	4.212
24	24	1995.89741917	7.582 e-09	6.040	24	24	4569.11198657	7.565 e-09	6.042
32	32	1995.89740668	1.300 e-09	6.077	32	32	4569.11195799	1.311 e-09	6.092
40	40	1995.89740473	3.428 e-10	6.041	40	40	4569.11195355	3.380 e-10	6.075
48	48	1995.89740427	1.144 e-10	6.018	48	48	4569.11195251	1.115 e-10	6.080
56	56	1995.89740413	4.505 e-11	6.045	56	56	4569.11195220	4.326 e-11	6.144
64	64	1995.89740408	1.968 e-11	6.204	64	64	4569.11195209	1.855 e-11	6.339
96	96	**1995.89740404**			96	96	**4569.11195200**		

близо до границата от синусоидалните гранични условия.

В Таблица 3.13 са представени средният брой итерации за **Пример 1** за външната итерация (Newton) и за вътрешната итерация (bicgstabl), част от „неточния“ метод на Нютон за CDS и CFDS са представени. За да отидем от n-тия слой по времето до следващия $(n + 1)$ слой, се нуждаем от приблизително три итерации за външната (Newton) част за двете диференчни схеми. На вътрешната (bicgstabl) част за случая на CDS се нуждаем от три итерации и за случая на CFDS се наблюдава намаляващ брой итерации от 3.40 до 2.05, когато броя на мрежовите точки по пространството се увеличава. Аналогични резултати са представени в Таблица 3.14 за **Пример 2**, получен с брой стъпки по времето $N = 256$. Броя на външните итерации е 3 за CDS и намалява от 3.80 до 3.17 за CFDS. Противно, броя на вътрешните итерации (bicgstabl) се увеличава за CDS от 1.75 до 6.54 и намалява от 4.70 до 2.50 за CFDS в резултат на по-добра локална апроксимация.

Въпреки всички предимства на CFDS по отношение на точност и CPU време, има и някои недостатъци. Шаблонът на CFDS е 9 точков и условието на зна-



(a) (b)

Фигура 3.8: Числеността 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

Таблица 3.13: Осреднен брой итерации за **Пример 1** за външна (Newton) и вътрешна (bicgstabl) части на неточния метод на Нютон за CDS и CFDS

CDS					CFDS				
M_x	M_y	N	Newton	bicgstabl	M_x	M_y	N	Newton	bicgstabl
8	8	8	3	2.67	8	8	16	3	3.40
16	16	16	3	2.67	16	16	64	2.98	2.57
32	32	32	3	2.67	32	32	256	2.96	2.15
64	64	64	2.95	3.31	64	64	1024	2.65	2.05

Таблица 3.14: Осреднен брой итерации за **Пример 2** за външна (Newton) и вътрешна (bicgstabl) части на неточния метод на Нютон за CDS и CFDS за брой стъпки по времето $N = 256$

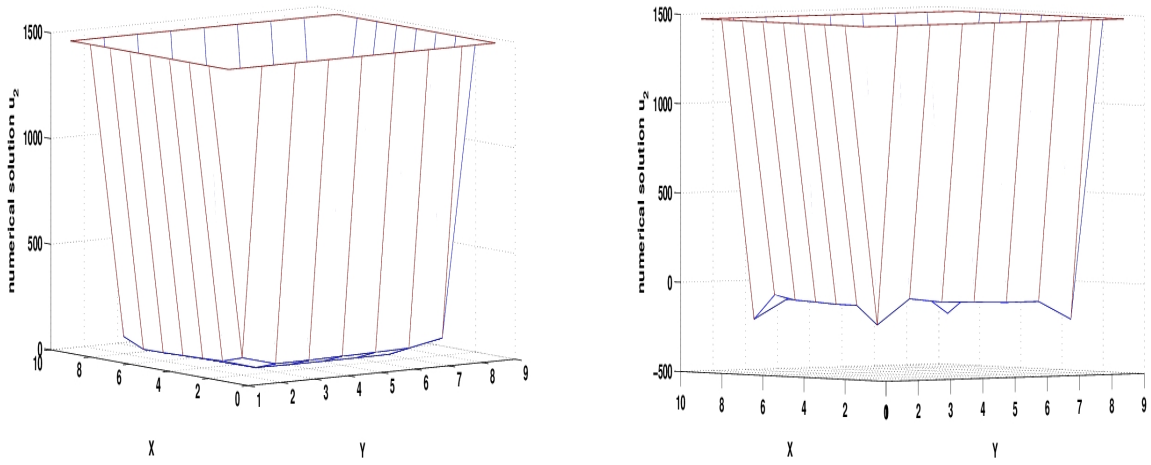
CDS				$CFDS$			
M_x	M_y	Newton	bicgstabl	M_x	M_y	Newton	bicgstabl
8	8	3	1.75	8	8	3.80	4.70
16	16	3	2.48	16	16	3.96	4.36
32	32	3	3.86	32	32	3.32	3.67
64	64	3	6.54	64	64	3.17	2.50

щите на дискретния принцип на максимума не е изпълнено. В резултат на това положителността на числено решение е нарушена за някои стойности на параметрите на мрежата по времето и пространството. На Фиг. 3.9 е представено числено 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. Схемата CDS запазва положителността на числено решение, докато CFDS не го запазва - близо до ъглите числено решение е отрицателно и няма теоретично обяснение.

3.8 Числени експерименти за атмосферен модел на базата на цикъла на Чапман

Използваме опростен, но реалистичен, модел на три компонентна $n = 3$ химична система, която моделира химични процеси, протичащи в атмосферата. В тази секция се концентрираме на системата (3.2.1)-(3.2.3) за $L = 3$ с коефициенти, реакции и източници, които отговарят на модел в атмосферата, базиран на цикъла на Чапман [122, 144]. Докато в един реалистичен атмосферен модел в екологията могат да участват концентрации на много реагиращи вещества [227, 85], опростеният модел тук притежава основните свойства на големите практически модели.

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



(a) (b)

Фигура 3.9: Численото решение за замърсителя NO_2 - u_2 за $\mu = 2\pi/(60T)$ с параметри на мрежата $M_x = M_y = 8$, $N = 256$, получено с: (a) CDS; (b) CFDS

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

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

където k_1, k_2 са скоростите на реакциите.

Пример 1. Случай на точно аналитично решение. Тук разглеждаме следната задача:

$$\frac{\partial u_l}{\partial t} - K \triangle u_l + \mathbf{b}_l . \nabla u_l = R_l(x, y, \mathbf{u}) + \xi_l(x, y, t), (x, y, t) \in \Omega \times (0, T]. \qquad (3.8.1)$$

Функциите ξ_l, $l = 1, 2, 3$, и началните и граничните условия са избрани, така че точното решение да е:

$$u_l = \exp(-t) sin(\pi x) sin(\pi y), \quad l = 1, 2, \quad (x, y, t) \in \overline{\Omega} \times [0, T],$$
$$u_3 = 1 + \exp(-t) sin(\pi x) sin(\pi y), \qquad (x, y, t) \in \overline{\Omega} \times [0, T].$$

Другите параметри са, както следва: $\overline{\Omega} = [0, 500] \times [0, 500]$, $T = 1440$, $b_l = (0.1, 0.1)$, за $l = 1, 2, 3$, $K_1 = 1$, $K_2 = K_3 = 5$. Коефициентите пред конвективните членове са $c_1 = c_2 = c_3 = -0.1, d_1 = d_2 = d_3 = 0$, където $b_i = (c_i, d_i)$, $i = 1, 2, 3$.

За l-тата субстанция с $error_{M,l}$ означаваме грешката (разликата между точното и числено решение) в равномерна норма, получена на последния слой

Таблица 3.15: Грешката в равномерна норма за първото вещество

CDS $O(h^2+\tau^2)$						CDS RE $O(h^4+\tau^2)$					
M_1	M_2	N	err_N	ratio	CPU	M_1	M_2	N	err_N	ratio	CPU
8	8	8	1.025 e-03		1.46	8	8	8	1.820 e-06		3.08
16	16	16	2.567 e-04	3.991	3.33	16	16	16	1.814 e-07	10.03	20.70
32	32	32	6.421 e-05	3.998	20.48	32	32	32	1.219 e-08	14.88	477
64	64	64	1.605 e-05	3.999	442	64	64	64	7.642 e-10	15.95	9871

Таблица 3.16: Грешката в равномерна норма за първото вещество

CFDS $O(h^4+\tau^2)$						CFDS RE $O(h^6+\tau^2)$					
M_1	M_2	N	err_N	ratio	CPU	M_1	M_2	N	err_N	ratio	CPU
8	8	8	5.223 e-06		1.34	8	8	8	2.512 e-09		2.95
16	16	32	3.293 e-07	15.86	15.26	16	16	32	3.945 e-11	64.31	38.27
32	32	128	2.062 e-08	15.972	84.87	32	32	128	6.145 e-13	64.19	373
64	64	512	1.289 e-09	15.991	6384	64	64	512	9.598 e-15	64.03	15510

по времето $t_N = T$ за брой подинтервали по пространството $M_x = M_y = M$:

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

Отношението между грешките, получено на две последователни сгъстявания на мрежата (обикновено с удвояване), е означено с *ratio*:

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

Резултатите с грешката в равномерна норма за първото, второто и третото вещество са дадени в таблиците по-долу. В Таблици 3.15 и 3.16 за азотния оскид, и 3.17 и 3.18 за озона, е направен анализ на грешката при сгъстявания на мрежата, получени с CDS и CFDS, и с CDS и CFDS, комбинирани с екстраполация по Ричардсон за $k_1 = 1000, k_2 = 2000$. Могат да се направят същите изводи, както и в случая на системата с 10 уравнения.

Пример 2. Задача с локални източници. В този пример се фокусираме върху локални източници, които се описват от Делта - функции. Тук разглеждаме задачата (3.8.1), където функциите ξ_l са точкови източници от вида

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

Таблица 3.17: Грешката в равномерна норма за третото вещество

CDS $O(h^2+\tau^2)$						CDS RE $O(h^4+\tau^2)$					
M_1	M_2	N	err_N	ratio	CPU	M_1	M_2	N	err_N	ratio	CPU
8	8	8	3.364 e-03		1.46	8	8	8	3.002 e-06		3.08
16	16	16	8.441 e-04	3.985	3.33	16	16	16	1.924 e-07	15.60	22.70
32	32	32	2.112 e-04	3.997	20.48	32	32	32	1.204 e-08	15.98	477
64	64	64	5.282 e-05	3.999	442	64	64	64	7.529 e-10	15.99	9871

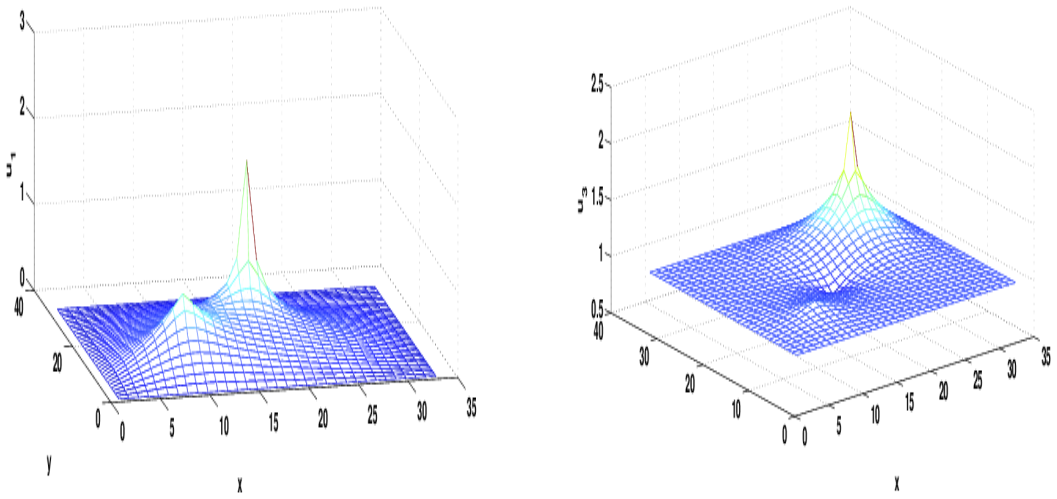
Таблица 3.18: Грешката в равномерна норма за третото вещество

CFDS $O(h^4+\tau^2)$						CFDS RE $O(h^6+\tau^2)$					
M_1	M_2	N	err_N	ratio	CPU	M_1	M_2	N	err_N	ratio	CPU
8	8	8	1.719 e-05		1.34	8	8	8	8.225 e-09		2.95
16	16	32	1.083 e-06	15.87	15.26	16	16	32	1.243 e-10	66.13	38.27
32	32	128	6.784 e-08	15.971	84.87	32	32	128	2.202 e-12	56.48	373
64	64	512	4.241 e-09	15.993	6384	64	64	512	3.454 e-14	63.75	15510

Параметрите са, както следва: $\overline{\Omega} = [0,500] \times [0,500]$, $T = 1440$, $b_l = (-0.1, 0)$, за $l = 1,2,3$, $K_1 = 1$, $K_2 = K_3 = 5$, $k_1 = 1000, k_2 = 2000$, $(\overline{x}_1, \overline{y}_1) = (0.5, 0.5)$, $(\overline{x}_2, \overline{y}_2) = (0.25, 0.25)$, $(\overline{x}_3, \overline{y}_3) = (0.75, 0.75)$, $f_1(t) = 7$, $f_2(t) = 11$, $f_3(t) = 13$. На Фиг. 3.10 е представено числено решение получено с CFDS с параметри $M_x = M_y = 32$, $N = 256$: (a) за NO - u_1; (b) за O_3 - u_3. Влиянието на точковите източници се вижда ясно.

3.9 Сравнение с метода Монте Карло за линейни системи

В тази секция са разгледани примери на различни параболични ЧДУ, за които е направено сравнение между изчислителното време за bicgstabl, означено с CPU(b) и изчислителното време на Монте Карло метода за линейни системи, конструиран в предишната глава, означено с CPU(m). Наблюдава се над четири пъти по-малко време за най добрия алгоритъм за решаване на линейни системи в Матлаб bicgstabl. Като бъдеща работа са предвидени за изследване и



(a) (b)

Фигура 3.10: Численото решение с параметри на мрежата $M_x = M_y = 32$, $N = 256$ за *Пример 2*: (a) за NO - u_1; (b) за O_3 - u_3

реализиране техники за ускоряване на метода Монте Карло за линейни системи. Трябва да се отбележи, че сравнението е с най-бързия алгоритъм bicgstabl и методът Монте Карло за линейни системи постига едни от най-добрите резултати за стохастичен метод за линейни системи. В предишната глава видяхме, че има матрици като NOS4, за които методът Монте Карло дава по-добри резултати от метода на спрегнатия градиент (pcg).

Числените експерименти включват едномерен Lotka-Volterra модел в популационната биология, и двумерна тестова задача с точни аналитични решения.

3.9.1 Едномерен Лотка-Волтера модел в популационната биология

Разглеждаме задачата:

$$\begin{aligned} \frac{\partial u}{\partial t} - a(x)\frac{\partial^2 u}{\partial x^2} + b(x)\frac{\partial u}{\partial x} &= f(x,t,u,v), \\ \frac{\partial v}{\partial t} - c(x)\frac{\partial^2 v}{\partial x^2} + d(x)\frac{\partial v}{\partial x} &= g(x,t,u,v), \end{aligned} \tag{3.9.1}$$

в областта $\Omega = (0,1) \times (0,T)$, с параметри $a = b = c = d = 1$, $f(x,t,u,v) = u(1-u-v)+\xi_1(x,t)$, $g(x,t,u,v) = v(1-u-v)+\xi_2(x,t)$, $T = 1$, където функциите ξ_1 и ξ_2 са избрани, така че точните аналитични решения да са $u = e^{-t}\sin(\pi x)$ и $v = e^{-t}x(1-x)$.

В Таблица 3.19 е представен анализ на грешката за стандартната и компактната схема. С err_M е означена грешката в равномерна норма на последния слой по времето $t_N = T$:

$$err_M = \max_i \|U_i^N - u(x_i, t_N)\|, \quad i = 1, ..., M-1.$$

Дадено е отношението между две последователни мрежи:

$$ratio = err_M / err_{2M}.$$

Резултатите (отношение близо до 4 и 16 съответно) потвърждават съответно втори ред $\mathcal{O}(\tau^2 + h^2)$ на стандартната схема по времето и пространството и четвърти ред по пространството $\mathcal{O}(\tau^2 + h^4)$ на CFDS. Изчислителното CPU време също е показано и е установено предимството на CFDS.

Таблица 3.19: Грешката в равномерна норма за едномерния числен пример.

Стандартна схема $O(h^2+\tau^2)$						Компактна схема $O(h^4+\tau^2)$					
M	N	err_N	ratio	CPU(b)	CPU(m)	M	N	err_N	ratio	CPU(b)	CPU(m)
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						

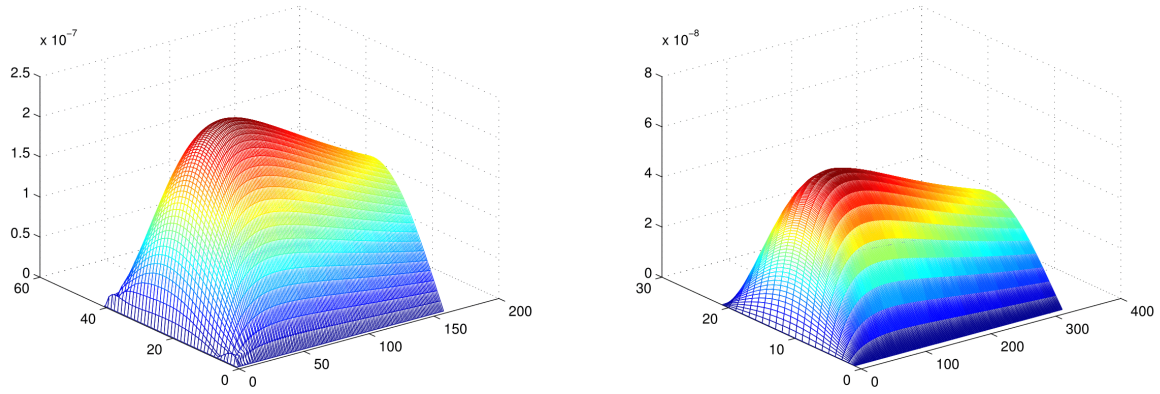
На същата едномерна моделна система прилагаме и екстраполация по Ричардсон по пространствената променлива x за стандартната и компактната схема. Резултатите са представени в Таблица 3.20. Тъй като стандартната схема е от втори ред по времето и пространството, то съответната схема с Ричардсон екстраполация е от четвърти ред по пространството и втори по времето. За да потвърдим четвъртия ред на схемата удвояваме точките на мрежата по пространството и очетворяваме точките по времето. Аналогично за компактната схема с Ричардсон екстраполация използваме осем пъти по-фина мрежа

по времето, за да сравним грешките и да получим отношение около 64, което потвърждава шестия ред на новата схема, която сме конструирали.

Таблица 3.20: Грешката в равномерна норма за едномерния числен пример с екстраполация по Ричардсон (RE)

Стандартна схема с RE $O(h^4+\tau^2)$						Компактна схема с RE $O(h^6+\tau^2)$					
M	N	err_N	ratio	CPU(b)	CPU(m)	M	N	err_N	ratio	CPU(b)	CPU(m)
10	10	4.18 e-05	-	0.89	3.02	5	5	1.61 e-04	-	2.59	9.78
20	40	2.46 e-06	17.03	1.42	5.21	10	40	2.18 e-06	73.85	1.42	5.23
40	160	1.16 e-07	16.21	26.95	112.1	20	320	3.36 e-08	64.88	7.86	27.19
80	640	9.46 e-09	16.02	223.7	923.1	40	2560	5.24 e-10	64.12	47.9	203.2

На Фиг. 3.11 е представена абсолютната грешка на численото решение, получено с екстраполация по Ричардсон за U_h^τ: (a) със стандартната схема с $M = 40$, $N = 160$; и (b) с компактната схема с $M = 20$ и $N = 320$.



(a) (b)

Фигура 3.11: Абсолютната грешка в равномерна норма за екстраполираното числено решение за едномерния числен пример: (a) получена за стандартната схема с Ричардсон за $M = 40$ и $N = 160$; (b) получено за компактната схема с Ричардсон за $M = 20$ and $N = 320$

3.9.2 Двумерна система от 2 уравнения

Нека да разгледаме задачата (3.5.1) с параметри $a = b = c = d = 1$ и десни страни $r(x, y, t, u, v) = u(1 - u - v) + \xi_1(x, y, t)$, $s(x, y, t, u, v) = v(1 - u - v) + \xi_2(x, y, t)$. Функциите ξ_1 и ξ_2, както и функциите $\overline{\phi}(x, y, t)$, $\overline{\overline{\phi}}(x, y, t)$, $\overline{\psi}(x, y)$, $\overline{\overline{\psi}}(x, y)$ са избрани, така че точните решения да са $u = t\sin(\pi x)\sin(\pi y)$ и $v = tx(1 - x)y(1 - y)$. В Таблица 3.21 представяме числените резултати за CFDS и CFDS с екстраполация по Ричардсон, а в Таблица - 3.22 числените резултати за стандартната схема и стандартната схема с екстраполация по Ричардсон. Предимството на схемите с RE по точност и изчислително време е неоспоримо.

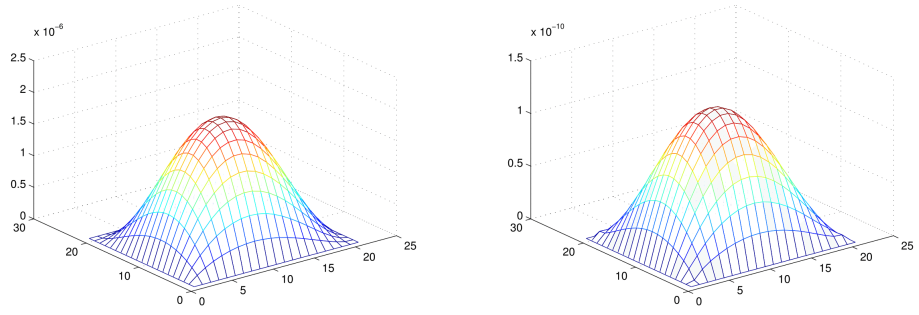
На Фиг. 3.12 е представена грешката на численото решение на последния слой по времето U_h^τ: (a) получено с CFDS за $M_1 = M_2 = 20$ и $N = 40$; (b) получено CFDS с RE за $M_1 = M_2 = 20$ и $N = 80$.

Таблица 3.21: Грешката в равномерна норма за за CFDS и CFDS с RE за двумерната система от 2 уравнения

Компактна схема $O(h^4 + \tau^2)$							Компактна схема с Ричардсон $O(h^6 + \tau^2)$						
M_1	M_2	N	err_N	ratio	CPU(b)	CPU(m)	M_1	M_2	N	err_N	ratio	CPU(b)	CPU(m)
10	10	10	2.90 e-05	-	4.05	18.11	5	5	5	4.85 e-07	-	0.24	1.12
20	20	40	2.01 e-06	14.41	6.38	26.12	10	10	20	8.34 e-09	58.15	2.18	9.31
40	40	160	1.28 e-07	15.70	146.4	754.1	20	20	80	1.33 e-10	62.70	61.82	386.5
80	80	640	8.03 e-09	15.95	4160	25134	40	40	320	2.05 e-12	64.87	2336	14132

Таблица 3.22: Грешката в равномерна норма за CDS и CDS с RE за двумерната система от 2 уравнения

Стандартна схема $O(h^2 + \tau^2)$							Стандартна схема с Ричардсон $O(h^4 + \tau^2)$						
M_1	M_2	N	err_N	ratio	CPU(b)	CPU(m)	M_1	M_2	N	err_N	ratio	CPU(b)	CPU(m)
10	10	10	7.02 e-03	-	0.75	3.21	5	5	5	7.87 e-05	-	0.99	4.16
20	20	20	1.86 e-03	3.79	3.17	14.23	10	10	20	5.86 e-06	13.45	6.02	26.53
40	40	40	4.78 e-04	3.89	29.98	167.1	20	20	80	3.80 e-07	15.44	90.75	511.3
80	80	80	1.20 e-04	3.98	1096	5406	40	40	320	2.40e-08	15.85	4384	29512



(a) (b)

Фигура 3.12: Грешката в равномерна норма за численото решение за двумерната система от 2 уравнения: (a) получено с компактна схема за $M_1 = M_2 = 20$ и $N = 40$; (b) получено с компактна и Ричардсон за $M_1 = M_2 = 20$ и $N = 80$

3.10 Заключение

Разработени са компактни диференчните схеми с четвърти ред на точност по пространствената променлива за модели в екологията за далечен пренос на замърсители във въздуха и други области. Направена е екстраполация по Ричардсон за повишаване на точността на численото решение. За първи път е приложена схема с шести ред на точност по пространствената променлива за реалната физична параболична транспортна система, описваща далечен пренос на замърсители във въздуха, от Датския Ойлеров модел и за атмосферния модел, базиран на цикъла на Чапман.

За получената след дискретизация СЛАУ за системата от Датския Ойлеров модел е използван най-бързия алгоритъм ("linear solver") на Матлаб bicgstabl, който е над четири пъти по-бърз от разработения нов метод Монте Карло за линейни системи в предишната глава. Вместо bicgstabl в експериментите може да се използва и методът Монте Карло за линейни системи, но това съчетано с бавния процесор и малкото количество памет, ще доведе до още по-голямо изчислително време.

В обобщение са получени следните научни приноси:

- Разработена е нова компактна диференчна схема от четвърти ред по пространствената променлива за нелинейна параболична система ЧДУ.

- Получена е схема с шести ред на точност по пространството с помощта на екстраполация по Ричардсон на компактната схема и с диференчна схема с четвърти ред на точност с екстраполация по Ричардсон върху централната схема.

- Дискретизацията по времето е реализирана с θ-схема, като в числените експерименти е приложен Кранк-Никълсон/Нютон алгоритъм.

- За решаване на линейната система, получена след дискретизация на моделната задача от Датския Ойлеров модел, се използва най-бързия в Matlab алгоритъм bicgstabl, но може да се използва и конструирания метод Монте Карло за линейни системи, като изчислителното време нараства поне четири пъти. Като бъдеща работа се цели прилагането на оптимизационни техники за ускоряване на изислителното време за Монте Карло алгоритъма.

- За числените експерименти при решаване на СЛАУ, получени след дискретизация на системи от едномерни и двумерни параболични ЧДУ с две уравнения, е приложен разработения метод Монте Карло за линейни системи.

- Получените числени резултати потвърждават теоретичните.

- Направено е сравнение с централна диференчна схема с ред $O(h^4 + \tau^2)$ и същата с екстраполация по Ричардсон от ред $O(h^4 + \tau^2)$. Компактната диференчна схема е $O(h^4 + \tau^2)$, и в комбинация с екстраполация по Ричардсон е $O(h^6 + \tau^2)$.

- Схемата CDS запазва положителността на численото решение, докато CFDS не го запазва. Това ще бъде обект на допълнително изследване и бъдеща работа.

- Компактната схема има предимства по отношение на време и точност в сравнение със стандартната схема с екстраполация по Ричардсон. Числено е потвърдено предимството на CDFS в сравнение с CDS едновременно по точност и изчислително време.

- Прилагането на екстраполация по Ричардсон играе важна роля в получаването на добри резултати в реално време с малък брой възли по мрежата, въпреки големите размери на областта по пространството и по времето на модели, свързани със замърсители във въздуха.

- От числените експерименти най-важно значение има моделът на далечен пренос на замърсители във въздуха, от Датския Ойлеров модел, описан от система параболични ЧДУ с 10 уравнения без точно аналитично решение и моделът на базата на цикъла на Чапман само с 3 уравнения с точкови източници, които се описват от делта функции.

Заключение

Настоящата дисертация е посветена на конструирането и сравнението на различни Монте Карло методи за интегрални уравнения, линейни системи и многомерни интеграли. Разработен е и нов числен метод с висок ред на точност на базата на диференчните схеми за модели в екологията. Направено е сравнение на алгоритми от тип Монте Карло и квази-Монте Карло за многомерни интеграли от гладки подинтегрални функции с различна размерност и са направени изводи кой от алгоритмите е най-добър за големи и малки размерности. Някои от алгоритмите са приложени за решаване на реални проблеми – за оценка на опции във финансите и интеграли в статистиката, за приближено пресмятане на ядрото на Вигнер в квантовата механика. Конструиран е нов алгоритъм Монте Карло за интегрални уравнения и е приложен за различни интегрални уравнения описващи процеси на обучение на невронни мрежи и взаимодействие между твърди тела. Разработен е нов метод Монте Карло за линейни системи на базата на метода „случайно блуждаене по уравненията“, който може успешно да се конкурира с най-добре известните градиентни методи за големи линейни системи. Специална глава е посветена и на разработването на нови компактни диференчни схеми с висок ред на точност за системи от параболични ЧДУ с приложение в екологията. Повишена е скоростта на сходимост с екстраполация по Ричардсон и е получена схема с шести ред на точност. Разработените схеми са приложени за модел на далечен пренос на замърсители във въздуха, където за системата от нелинейни параболични ЧДУ от 10 уравнения, е постигната много малка относителна грешка и са потвърдени теоретично четвърти и шести ред на сходимост на конструираните схеми. За първи път е приложена диференчна схема с шести ред на точност за атмосферния модел на базата на цикъла на Чапман и за моделната задача от Датския Ойлеров модел.

Списък на публикациите по дисертацията

Основните резултати по дисертацията са публикувани в следните статии:

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, ISSN: 0302-9743, **SJR (2015): 0.252**, DOI : 10.1007/978 − 3 − 319 − 15585 − 2_5.

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: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:0.198**.

5. V. Todorov. Computing high dimensional integrals with Monte Carlo methods, Journal Scientific and Applied Research, Vol.10, (2016), 11-16, Konstantin Preslavsky Publishing House, ISSN 1314-6289.

Втората статия е цитирана в

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.

Апробация на резултатите

Резултати, включени в дисертацията, са докладвани на следните семинари:

- съвместен семинар на секции „Паралелни алгоритми“ и „Научни пресмятания“ на ИИКТ-БАН;

Част от резултатите са представени на следните конференции:

- 9th International Conference on „Large-Scale Scientific Computations“ (LSSC'13), Созопол, България, 2013;
- 8th International Conference on Numerical Methods and Applications: NMA'14, Borovetz.
- 10th IMACS Seminar on Monte Carlo Methods, (MCM 2015), Linz, Austria, July 6-10,2015
- Advanced Computing for Innovation, AComIn 2015, 10-11 November, 2015, Sofia.
- Международната конференцията Numerical Methods for Scientific Computations and Advanced Applications, NMSCAA16, 29 май- 2 юни в Хисаря, 2016.
- 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.

Резултатите, описани в представените публикации, са получени в рамките на 4 научноизследователски проекта:

1. Проект за млади учени на БАН, Договор #ДФНП 91-А1/04.05.2016 (Ръководител: Венелин Тодоров, ИИКТ-БАН, научен консултант: Доц. д-р Цветан Остромски, ИИКТ-БАН)
2. Ефективни паралелни алгоритми за големи изчислителни задачи, ФНИ, Договор #ФНИ И-02- 20/2014 (Ръководител: Проф. дн Ив. Димов, от 2016 г. - проф. Ст. Фиданова, ИИКТ-БАН)

3. Съвременните пресмятания в полза на иновацията (AComIn), "Капацитети" в 7-ма Рамкова програма на Европейската комисия (ЕК), "Научно-изследователски потенциал в конвергентните райони", конкурс FP7-REGPOT-2012-2013-1, договор 316087, 2012 – 2015(Ръководител: Проф. дн Галя Ангелова, ИИКТ-БАН)

4. Разработване и изследване на нови методи Монте Карло за моделиране на сложни системи, НФНИ, Договор #ДМУ 03/61, 2011-2013 (Ръководител: Доц. д-р Райна Георгиева, ИИКТ-БАН)

Основни научни и научно-приложни приноси

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 интеграли 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

Изказвам своята искрена признателност и благодарност на научния си ръководител проф. дтн 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 неоценима морална подкрепа и за проявеното търпение.

Благодаря изключително много на доц. д-р Райна Георгиева за нейните ценни напътствия, готовността й за съдействие във всеки момент и неоценимата 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 на целия екип

на Института по информационни и комуникационни технологии и към неговия Директор за осигурените качествени условия за работа и обучение.

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

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] Атанасов, Е. Монте-Карло и квази-Монте Карло методи, Дисертация, София, 2002.

[2] Бахвалов, Н.С.: Численнье методь. Москва, Наука, 1973.

[3] Боянов, Б.: *Лекции по числени методи.* Дарба, София, 1998.

[4] Георгиева, Р.: Изчислителна сложност на алгоритми Монта Карло за многомерни интеграли и интегрални уравнения, Дисертация, София, 2003.

[5] Гюров, Т.: Монте Карло алгоритми за някои задачи за пренос, Дисертация, София, 1999.

[6] Димитров, Б.: *Вериги на Марков.* Наука и изкуство, София, 1974.

[7] Ивановска, С.: Квази Монте Карло методи за интегрални уравнения, Дисертация, София, 2007.

[8] Караиванова, А.: *Стохастични числени методи и симулации*, София, 2012.

[9] Папанчева, Р.: Ефективни немрежови Монте Карло алгоритми за решаване на гранични задачи с локално интегрално представяне на решението, Дисертаци, София, 2003.

[10] Никольский, С.: Квадратурные формулы. Наука, Москва, 1988.

[11] Самарский, А.А.: Теория разностниьх схем. Москва, Наука, 1977.

[12] Самарский, А.А., Анреев, В.В.: Разностнье методь для елиптических уравнений. Москва, Наука, 1976.

[13] Самарский, А.А., Гулин, В.: Устойивост разностньх схем. Москва, Наука, 1973.

[14] Стоилова, С.: Ортонормирани функционални системи и равномерно разпределение на редици, Дисертация, София, 2003.

[15] Черногорова, Т.: Теория на диференчните схеми, София, 2005. https://www.fmi.uni-sofia.bg/econtent/tds.pdf

[16] Женсыкбаев, А.: О наилучших квадратурных формулах для некоторых классов непериодических функций. In Докл. АН СССР, volume 236, pp 531–534, 1977.

[17] Женсыкбаев, А.: Характеристические свойства наилучших квадратурных формул. Сиб. мат. журн., 20:49–68, 1979.

[18] Харалампиев, К. 2007. За парадигмите в статистиката–бейсовска статистика. Международна научна конференция „Актуални проблеми на статистическата теория и практика", Равда. http://kaloyan-haralampiev.info/wp-content/uploads/2010/03/doklad-1.pdf

[19] Янев, Н., Димитров, Б.: *Вероятности и статистика.* Наука и изкуство, София, 1990.

[20] Amiri-Jaghargh A., Roohi E., Niazmand H., Stefanov S.: DSMC Simulation of Low Knudsen Micro/Nanoflows Using Small Number of Particles per Cells. J Heat Trans-T Asme 135(10) (2013).

[21] Antonov, I.A., Saleev, V.M.: An Economic Method of Computing LP Tau-Sequences, USSR Computational Mathematics and Mathematical Physics, Volume 19, 1980, pages 252-256.

[22] Atanassov, E., Dimov, I.: A new optimal Monte Carlo method for calculating integrals of smooth functions, Journal of Monte Carlo Methods and Applications, Vol. 5, (1999), No 2, pp. 149-167.

[23] Bahvalov, N.: On the Approximate Computation of Multiple Integrals. In *Vestnik Moscow State University, Ser. Mat., Mech.*, volume 4, pages 3–18, 1959.

[24] Bahvalov, N.: Average Estimation of the Remainder Term of Quadrature Formulas. *USSR Comput. Math. and Math. Phys.*, 1(1):64–77, 1961.

[25] Baker, R.C.: On irregularities of distribution II. J. London Math. Soc., 2(59):50-64, 1999.

[26] Bayes, T.: Essay towards Solving a Problem in the Doctrine of Chances. In: Biometrika, 45, 1958.

[27] Bellman, R.: *Adaptive Control Processes: A Guided Tour.* Princeton University Press, 1961

[28] Berntsen, J., Espelid, T.O., Genz, A.: An adaptive algorithm for the approximate calculation of multiple integrals, *ACM Trans. Math. Softw.* **17** (1991) 437–451.

[29] Black, F., Scholes, M.: The Pricing of Options and Corporate Liabilities. J. Pol. and Econ. 81 (1973) 637-659.

[30] Boyle, P.P.: Options: a Monte Carlo Approach. J. Finan. Econ. 4 (1977) 323-338

[31] Boyle, P.P., Broadie, M., Glasserman, P.: Monte Carlo methods for security pricing. Journal of Economic Dynamics and Control, 21:1267-1321, 1997.

[32] Boyle, P., Lai, Y., Tan, K.: *Using Lattice Rules to Value Low-Dimensional Derrivative Contracts*, 2001

[33] Bratley, P., Fox, B.: Algorithm 659: Implementing Sobol's Quasirandom Sequence Generator, ACM Transactions on Mathematical Software, Volume 14, Number 1, pages 88-100, 1988.

[34] Bretthorst, L: An Introduction of Parameter Estimation Using Bayesian Probability. In: Maximum Entropy and Bayesian Methods, P. Fougere (ed.), Kluwer Academic Publishers, Dordrecht the Netherlands., 1990.

[35] Broadie M., Glasserman P.: Pricing American-style Securities Using Simulation, J. of Economic Dynamics and Control 21, 1323-1352,1997.

[36] Bull, J.M., Freeman, T.L.: Parallel globally adaptive quadrature on the KSR-1, *Advances in Comp. Mathematics*, 2, 357–373, 1994.

[37] Bull, J.M., Freeman, T.L.: Parallel algorithms for multi-dimensional integration, *Parallel and Distributed Computing Practices*, 1(1), 89–102, 1998.

[38] Buslenko, N., Golenko, D., Shreider, Yu., Sobol, I., Sragovich, V.: *The Monte Carlo Method: the Method of Statistical Trials*. Pergamon Press, 1962. Translated from the Russian by G. J. Tee, 1966.

[39] Caflish, R.E.: Monte Carlo and quasi-Monte Carlo methods, Acta Numerica, 7:1-49, 1998.

[40] Caffisch, R.E., Morokoff, W., Owen, A.: Valuation of mortgage-backed securities using Brownian bridges to reduce effective dimension. Journal of Computational Finance, 1(1):27-46, 1997.

[41] Chance, D.M.: An Introduction to Derivatives. (third edition) The Dryden Press 1995

[42] Chen W., Li C., Wright E.: On a nonlinear parabolic system-modeling chemical reactions in rivers, Comm. on Pure and Appl. Anal., 4(4), 2005, pp. 889–899.

[43] Cheney W., Kincard, D.: *Numerical Mathematics and Computing*, 4th Ed., (Brooks/Cole Publishing, Pacific Grove, CA, 1999)

[44] Chickering D.M. and Heckerman. D.: Efficient approximations for the margin al likelihood of bayesian networks with hidden variables. Machine Learning, 29:181–212, 1997. Microsoft Research Report, MSR-TR-96-08.

[45] Cox, J.C., Rubinstein, M.: Options Markets. Prentice Hall 1985

[46] Curtiss, J.H.: Monte Carlo Methods for The Iteration of Linear Operators. *J. Math. Phys.* 32, 209–232, 1954.

[47] Davis, P., Rabinowitz. P.: *Methods of Numerical Integration.* 2nd ed., Academic Press, New York, 1984.

[48] Davis, P.J., Rabinowitz, P.: Methods of Numerical Integration. Academic Press, London, 2nd edition, (1984).

[49] Dembo R.S., Eisenstat, S.C., Steihaug, T.: Inexact Newton methods. SIAM Journal on Numerical Analysis, **19**(2), 400-408 (1982)

[50] Dimov, I.: *Monte Carlo Methods for Applied Scientists*, New Jersey, London, Singapore, **World Scientific** (2008), 291 p., ISBN-10 981-02-2329-3.

[51] Dimov, I.T.: *Minimization of the Probable Error for Some Monte Carlo methods.* Proc. Int. Conf. on Mathematical Modeling and Scientific Computation, Albena, Bulgaria, Sofia, **Publ. House of the Bulgarian Academy of Sciences**. 1991, pp. 159–170.

[52] Dimov, I.: Efficient and overconvergent Monte Carlo Methods. Parallel algorithms., *Advances in Parallel Algorithms, I.Dimov, O.Tonev (Eds.), Amsterdam, IOS Press*, 100–111, 1994.

[53] Dimov, I., Alexandrov, V., Karaivanova, A.: Parallel resolvent Monte Carlo algorithms for linear algebra problems. *Mathematics and Computers in Simulation*, **55** (2001), 25-35.

[54] Dimov, I.T., Atanassov, E.: Exact error estimates and optimal randomized algorithms for integration, *LNCS* **4310** (2007) 131–139.

[55] Dimov, I.T., Georgieva, R.: Complexity of Monte Carlo Algorithms for a Class of Integral Equations. In *LNCS*, Part I, volume 4487, pages 731–738. Springer-Verlag, 2007.

[56] Dimov, I., R. Georgieva: Monte Carlo Algorithms for Evaluating Sobol' Sensitivity Indices. Math. Comput. Simul. **81** (2010) (3), 506-514.

[57] Dimov, I.T., Georgieva, R., Todorov, V.: 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 (Dimov, I., S. Fidanova, and I. Lirkov - Eds.), LNCS 8962, Springer, 2015, 44–51. DOI : $10.1007/978-3-319-15585-2_5$. ISSN: 0302-9743.

[58] 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.

[59] 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.

[60] 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), doi: 10.1063/1.4968448.

[61] Dimov I., Karaivanova A., Georgieva R., Ivanovska S.: Parallel Importance Separation and Adaptive Monte Carlo Algorithms for Multiple Integrals, 5th Int. conf. on NMA, August, 2002, Borovets, Bulgaria, Springer Lecture Notes in Computer Science, 2542, (2003), Springer-Verlag, Berlin, Heidelberg, New York, pp. 99 - 107.

[62] 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.

[63] Dimov, I., Nedjalkov, M., Sellier, J.M.: An Introduction to Applied Quantum Mechanics in the Wigner Monte Carlo Formalism, *P*hysics Reports vol. 577, pp. 1-–34, 2015.

[64] Dimov, I.T., Philippe, B., Karaivanova, A., Weihrauch, C.: *Robustness and applicability of Markov chain Monte Carlo algorithms for eigenvalue problems*, **Applied Mathematical Modelling**, **Vol. 32** (2008) pp. 1511-–1529.

[65] Dimov I., Stoilova S. and Mitev N., Diaphony of Uniform Samples over Hemisphere and Sphere, Springer LNCS, 5434, 2009, pp. 257-264.

[66] Dimov, I., Todorov, V.: Error Analysis of Biased Stochastic Algorithms for the Second Kind Fredholm Integral Equation, DOI:10.1007/978-3-319-32207-0_1, Studies in Computational Intelligence 648, 2016, Innovative Approaches and Solutions in Advanced Intelligent Systems, ISSN: 1860949X, pp.3-16.

[67] Dimov, I.T., Tonev, O.: Monte Carlo Numerical Methods With Overconvergent Probable Error. In Proc. 2nd Intern. Conf. on Numerical Methods and Appl., pages 116–120, Sofia, 1989. Publ. house of the Bulg. Acad. Sci.

[68] Disney, S., Sloan, I. H.: (1991), Error bounds for the method of good lattice points, Math. Comp. 56, 257–266.

[69] Dooren, P. van, Ridder, L. de: An adaptive algorithm for numerical integration over an N-dimensional cube, *Journal of Computational and Applied Mathematics*, 2, 207–217, 1976.

[70] Doucet, A., Freitas, N. de, Gordan, N.: *Sequential Monte Carlo methods in Practice*. Springer-Verlag, New York, 2001.

[71] 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).

[72] Dupach, V.: Stochasticke Pocetni Metody. *Cas. pro pest. mat.*, 81(1):55–68, 1956.

[73] Eckhardt, R.: *Stan Ulam John von Neumann and the Monte Carlo Method*. 1987.

[74] Eglajs, V., Audze P.: New approach to the design of multifactor experiments. Problems of Dynamics and Strengths. 35 (in Russian). Riga: Zinatne Publishing House: 104–107, 1977.

[75] Ermakov, S., Mikhailov, G.: *Statistical Modeling*, Nauka, Moscow, 1982.

[76] Ermakov, S.: On summation of series connected with integral equation. Vestnik Leningrad Univ. Math. 16 (1984), 57-63.

[77] Ermakov, S.M.: *Monte Carlo Methods and Mixed Problems*, **Nauka**, Moscow, (1985).

[78] Faure, H.: Discrepance de suites associees a un systeme de numeration (en dimensions). Acta Arithmetica, 41:337–351, 1982.

[79] Faure, H.: Monte-Carlo and Quasi-Monte-Carlo Methods for Numerical Integration. *Combinatorial and computational mathematics*, pages 1–12, 2001.

[80] Feynman, R.P.: Space-time approach to non-relativistic quantum mechanics, Rev. Mod. Phys. 20 (1948).

[81] Fox, B.: Algorithm 647: Implementation and Relative Efficiency of Quasirandom Sequence Generators, ACM Transactions on Mathematical Software, Volume 12, Number 4, pages 362-376, 1986.

[82] Freeman, T., Bull, J.: A comparison of parallel adaptive algorithms for multidimensional integration, in: *Proceedings of 8th SIAM Conference on Parallel Processing for Scientific Computing* (1997).

[83] Gemmill, G.: Options Pricing. McGraw-Hill 1992

[84] Genz, A.: Testing multidimensional integration routines. Tools, *Methods and Languages for Scientific and Engineering Computation* (1984) 81–94.

[85] 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)

[86] Georgieva, R., Ivanovska, S.: Importance Separation for Solving Integral Equations. In *LNCS*, volume 2907, pages 144–152. Springer-Verlag, 2004.

[87] Gobet, E., Maire, S.: *Sequential control variates for functionals of Markov processes*, **SIAM Journal on Numerical Analysis 43**, pp 1256-1275 (2005).

[88] Gupta, M.M., Manohar, R.P., Stephenson, J.W.: A single cell high order scheme for the convection-diffusion equation with variable coefficients, Int. J. for Num. Methods in Fluids, **4**, 641-651 (1984)

[89] Gursa, E.: *Course in Mathematical Analysis*, State Science and Technological Publishing House, Moscow, vol. **3**, 1934.

[90] Gustafsson, B., Kreiss, H., Oliger, J.: Time Dependent Problems and Difference Methods, Wiley, New York (1995)

[91] Hall, A.: On an experimental determination of PI, 1873.

[92] Halton, J.H.: On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. Numerische Mathematik, 2:84–90, 1960.

[93] Halton, J.: *Sequential Monte Carlo, Proceedings of the Cambridge Philosophical Society*, **Vol. 58** (1962) pp. 57-78.

[94] Halton, J.: *Sequential Monte Carlo. University of Wisconsin*, Madison, Mathematics Research Center Technical Summary Report No. 816 (1967) 38 pp.

[95] Halton, J.: *Sequential Monte Carlo for linear systems - a practical summary*, **Monte Carlo Methods & Applications, 14** (2008) pp. 1–27.

[96] Halton, J., Zeidman, E.A.: *Monte Carlo integration with sequential stratification.* University of Wisconsin, Madison, Computer Sciences Department Technical Report No. 61 (1969) 31 pp.

[97] Hammersley, J., Handscomb, D.: *Monte Carlo Methods.* John Wiley & Sons, inc., New York, London, Sydney, Methuen, 1964.

[98] Hartmanis, J., Stearns, R.: On the Computational Complexity of Algorithms. *Trans. Amer. Math. Soc.*, 117:285–306, 1965.

[99] Hesterberg, T.: Weighted average importance sampling and defensive mixture distributions, *Technometrics* **37(2)** (1995) 185–194.

[100] Hickernell, F. J.: (1996), Quadrature error bounds with applications to lattice rules, SIAM J. Numer. Anal. 33, 1995–2016. corrected printing of Sections 3-6 in ibid., 34 (1997), 853–866.

[101] Hlawka, E.: Funktionen von beschrankter Variation in der Theorie der Gleichverteilung. Annali di Matematica Pura Applicata, 54:324–334, 1961.

[102] Hlawka, E.: (1962), Zur angenaherten Berechnung mehrfacher Integrale, Monatsh. Math. 66, 140–151.

[103] Hua, L.K., Wang, Y.: Remarks concerning numerical integration, Sci. Record (N.S.) (1960) 4, 8–11.

[104] Hua, L.K., Wang, Y.: *Applications of Number Theory to Numerical analysis*, 1981

[105] Hull, J.C.: Options, Futures, and other Derivative Securities. Prentice-Hall, Inc. 1993

[106] Iman, R.L., Helton, J.C., Campbell, J.E.: An approach to sensitivity analysis of computer models, Part 1. Introduction, input variable selection and preliminary variable assessment. Journal of Quality Technology. 13 (3): 174–183, 1981.

[107] Iman, R.L., Davenport, J.M., Zeigler, D.K.: Latin hypercube sampling (program user's guide), 1980.

[108] Jarosz, Wojciech: Efficient Monte Carlo Methods for Light Transport in Scattering Media, PhD dissertation, UCSD, 2008

[109] Jaynes, E.: Bayesian Methods: General Background. In: Maximum-Entropy and Bayesian Methods in Applied Statistics, J. H. Justice (ed.), Cambridge Univercity Press, Cambridge, 1986.

[110] Ji, H., Mascagni, M., Li, Y.: *Convergence Analysis of Markov Chain Monte Carlo Linear Solvers Using Ulam–von Neumann Algorithm.* **SIAM Journal on Numerical Analysis, 51(4)**, (2013), pp. 2107-2122.

[111] Joy, C., Boyle, P.P., Tan, K.S.: Quasi-Monte Carlo methods in numerical finance. Management Science, 42(6):926-938, 1996.

[112] Kahn, H.: Random Sampling (Monte Carlo) techniques in Neutron Attenuation Problems. *Nucleonics*, 6:27–33, 60–65, 1950.

[113] Kalos, M.A., Whitlock, P.A.: *Monte Carlo Methods, Volume 1: Basics*, Wiley, New York, 1986.

[114] Kantorovich, L., Akilov, G.: *Functional Analysis.* Nauka, Moskow, 1977.

[115] Kantorovich, L.W., Krylov, V.I.: *Approximate Methods of Higher Analysis*, **Interscience**, New York, 1964.

[116] Kantorovich, L., Akilov, G.: *Functional Analysis in Normed Spaces*, Pergamon Press, Oxford, 1964.

[117] Karaivanova, A.: Adaptive Monte Carlo methods for numerical integration, *Mathematica Balkanica* **11** (1997) 391–406.

[118] Karaivanova, A., Dimov, I.: Error analysis of an adaptive Monte Carlo method for numerical integration, *Mathematics and Computers in Simulation* **47** (1998) 201–213.

[119] Karaivanova A., Dimov I., Ivanovska S.: A Quasi-Monte Carlo Method for Integration with Improved Convergence, Lecture Notes in Computer Science, Vol. 2179 (2001), Springer-Verlag, Berlin, pp. 158 – 165.

[120] Karaivanova A., Dimov I.: A Power Method with Monte Carlo Iterations, Recent Advances in Numerical Methods and Applications (O. Iliev, M. Kaschiev, S. Margenov, Bl. Sendov, P. Vassilevski, Eds.), World Scientific, Singapore, 1999, pp. 239 -247.

[121] Karatson, J., Kurics. T.: A preconditioned iterative solution scheme for nonlinear parabolic systems arizing in air pollution modeling. Math. Modell. Anal. **18** (5), 641-653 (2013)

[122] Kim, J., Cho, S.: Computation accuracy and efficiency of the time splitting method. J. Atmosph. Envir. **31**(15), pp. 2215-2224 (1997)

[123] Koksma, J.F.: Een algemeene stelling uit de theorie der gelijkmatige verdeeling modulo 1. Mathematica B (Zutphen), 11:7-11, 1942/43.

[124] Kollman, C., Baggerly, K., Cox, D., Picard, R.: *Adaptive importance sampling on discrete Markov chains*. **Annals of Applied Probability, 9(2)**, pp. 391–412, 1999.

[125] Korobov, N.M.: The approximate computation of multiple integrals, Dokl. Akad. Nauk. SSR 124, (1959) 1207–1210. (Russian).

[126] Korobov, N.M.: Properties and calculation of optimal coefficients. Doklady Akademii Nauk SSSR, 132:1009–1012 (Russian), 1959. English Translation: Soviet Mathematics Doklady, 1, 696–700.

[127] Korobov, N.M.: Number-Theoretical Methods in Approximate Analysis, Fizmatgiz, Moscow, 1963.

[128] Kublanovskaya, V. Application of analytical continuation by substitution of variables in numerical analysis, *Proceedings of the Steklov Institute of Mathematics* **53** (1959), 145-185.

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

[130] Kyei, Y., Roop, J.P., Tang, G.: A family of sixth-order compact finite- difference schemes for the three-dimensional Poisson equation. Advances in Numerical Analysis, **2010**, 1-17 (2010).

[131] 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

[132] Lécot, C.: Low Discrepancy Sequences for Solving the Boltzmann Equation. *J. Comput. Appl. Math.*, 25:237–249, 1989.

[133] L'Ecuyer, P., Blanchet, J.H., Tuffin, B., Glynn, P.W.: *Asymptotic robustness of estima- tors in rare-event simulation. ACM Transactions on Modeling and Computer Simulation*, **20(1)**:Article 6, 2010.

[134] L'Ecuyer, P., Tuffin, B.: *Approximate zero-variance simulation.* In *Proceedings of the 2008 Winter Simulation Conference*, pp. 170–181. IEEE Press, 2008.

[135] Lewerenz, M.: Monte Carlo Methods: Overview and Basics. *In Quantum Simulations of Complex Many-Body Systems: from Theory to Algorithms, Lecture Notes*, volume 10, pages 1–24. NIC Series, 2002.

[136] Lin, S.: "Algebraic Methods for Evaluating Integrals in Bayesian Statistics," Ph.D. dissertation, UC Berkeley, May 2011.

[137] 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.

[138] Lirkov I., Stoilova S., The b-adic Diaphony as a Tool to Study Pseudorandomness of Nets, Springer LNCS, 6046, 2011, pp. 68-76.

[139] Loredo, T. 1990. From Laplace To SN 1987A: Bayesian Inference In Astrophysics. In: Maximum Entropy and Bayesian Methods, P. F. Fougere (ed), Kluwer Academic Publishers, Dordrecht.

[140] Maire, S.: *Reducing variance using iterated control variates*, **Journal of Statistical Computation and Simulation, Vol. 73(1)**, pp. 1-29, 2003.

[141] Marchuk, G.I., Shaidurov, V.V.: *Difference Methods and Their Extrapolations* (Springer-Verlag, New York Inc. 1983).

[142] Mascagni, M., Karaivanova, A.: A Parallel Quasi-Monte Carlo Method for Computing Extremal Eigenvalues. pages 369–380. Springer, 2002.

[143] Matsumoto, M., Nishimura, T., Twister, M.: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Transactions on Modeling and Computer Simulation* **8(3)** 1998.

[144] Mamonov, A. V., Tsai, Y.-H. R.: Point source identification in non-linear advection-diffusion-reaction systems. Inverse Problems **29**(3), pp. 035009 (2012)

[145] 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

[146] Medvedev, I., Mikhailov, G.: *A New Criterion for Finiteness of Weight Estimator Variance in Statistical Simulation*, in: "Monte Carlo and Quasi-Monte Carlo Methods, 2006", (Editors: Alexander Keller, Stefan Heinrich, Harald Niederreiter), 2008, pp. 561–576.

[147] Merton, R.C.: Theory of rational option pricing, Bell J. Econ. 4 (1973) 141-183.

[148] Metropolis, N., Ulam, S.: *The Monte Carlo Method*, **J. of Amer. Statistical Assoc., 44**, (1949), No. 247, pp. 335–341.

[149] Mikhailov, G.A.: *Optimization of Weight Monte Carlo methods*, **Nauka**, Moscow, 1987.

[150] 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.

[151] Minasny B., McBratney B.: Conditioned Latin Hypercube Sampling for Calibrating Soil Sensor Data to Soil Properties, Chapter: Proximal Soil Sensing, Progress in Soil Science, pp. 111-119, 2010.

[152] Miranda C.: Un'osservazione su un teorema di Brouwer, Boll. UMI 2 Vol.3 (1940-1941).

[153] 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.

[154] Mohammadzadeh A., Roohi E., Niazmand H., Stefanov S., R. S. Myong R.: Thermal and second-law analysis of a micro- or nanocavity using direct-simulation Monte Carlo, (2012) Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 85 (5).

[155] Morokoff, W., Caflisch, R.: A Quasi-Monte Carlo Approach to Particle Simulation of the Heat Equation. *Siam J. Num. Anal.*, 30(6):1558–1573, 1993.

[156] Nedjalkov, M., Ferry, D.K., Vasileska, D., Dollfus, P., Querlioz, D., Dimov, I., Schwaha, P., Selberherr, S.: Physical Scales in the Wigner-Boltzmann Equation, *Annals of Physics*, Vol.328, pp.220-237, 2013.

[157] Niederreiter, H., Kuipers, L.: Uniform distribution of sequences. John Wiley & sons, New York, 1974.

[158] Niederreiter, H.: Existence of good lattice points in the sense of Hlawka, Monatsh. Math. 86, 1978, 203–219.

[159] Niederreiter, H.: Point sets and sequences with small discrepancy. Monatshefte fur Mathematik, 104:273–337, 1987.

[160] Niederreiter, H.: Low-discrepancy and low-dispersion sequences. Journal Number Theory, 30:51–70, 1988.

[161] Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. S.I.A.M., Conf.Ser.Appl.Math. Vol.63, Philadelphia, 1992.

[162] Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods* **CBSM 63** (1992).

[163] Niederreiter, H.: Improved error bounds for lattice rules, J. Complexity 9, (1993) 60–75.

[164] Niederreiter, H., Xing, C.P.: Explicit global function fields over the binary field with many rational points. Acta Arithmetica, 75:383–396, 1996.

[165] Niederreiter, H., Xing, C.P.: Global function fields with many rational points over the ternary field. Acta Arithmetica, 83:65–86, 1998.

[166] Ninomiya, S., Tezuka, S.: Toward real-time pricing of complex financial derivatives. Applied Mathematical Finance, 3:1-20, 1996.

[167] Novak, E., Ritter, K.: High Dimensional Integration of Smooth Functions Over Cubes. *Numerishche Mathematik*, pages 1–19, 1996.

[168] Owen, A.: Orthogonal Arrays for Computer Experiments, Integration and Visualization. *Statistica Sinica*, 2(2):439—452, 1992.

[169] Owen, A.: Scrambling Sobol and Niederreiter-Xing Points. Journal of Complexity, 14(4):466-489, 1998.

[170] Owen, A., Zhou, Y. Safe and effective importance sampling, Technical report, Stanford University, Statistics Department, 1999.

[171] Pao C. V.: Nonlinear Parabolic and Elliptic Equations, Springer, US, 1992.

[172] Paskov, S.: Computing High Dimensional Integrals with Applications to Finance. preprint Columbia Univ. (1994)

[173] Paskov, S., Traub, J.: Faster Evaluation of Financial Derivatives. *J. Portfolio Management*, 22:113–120, 1995

[174] Paskov, S.H., Traub, J.F.: Faster valuation of financial derivatives. Journal of Portfolio Management, 22(1):113-120, 1995.

[175] Richards, S.: Completed Richardson extrapolation in space and time. Commun. Numer. Meth. Engineering **13**, 573–582 (1997).

[176] Richtmyer, R.D., Morton, K.W.: *Difference Methods for Intial-Value Problems* (Krieger, Malabar, FL., 1994).

[177] Rota, Gian-Carlo: *Indiscrete Thoughts*, 1996

[178] Roth, K.F.: On irregularities of distribution. Mathematica, 1:73-79, 1954.

[179] Rubinstein, R.: Simulation and the Monte Carlo Method. John Wiley & Sons, Inc., New York, USA, 1981.

[180] Sabelfeld, K.: Methods of numerical construction of the resolvent when solving integral and differential equations by the Monte Carlo method, *Theory and applications of the statistical simulation*, Computing Center, Novosibirsk (1985), 1-10.

[181] Sabelfeld, K.: *Monte Carlo Methods in Boundary Value Problems*, Springer-Verlag, Berlin - Heidelberg - New York - London, 1991.

[182] Samarskii, A.A.: *The Theory of Difference Schemes* (Marcel Dekker, Inc. New York, NY 2001).

[183] Schmidt, W.M.: Irregularities of distribution. vii, Acta Arith 21, (1972) 42–50.

[184] Sellier, J.M.: A signed particle formulation of non-relativistic quantum mechanics, Journal of Computational Physics 297(2015) 254–265.

[185] Sendov, Bl., Andreev, A., Kjurkchiev, N.: *Numerical Solution of Polynomial Equations (Handbook of Numerical Analysis), Solution of Equations in R^n (Part 2).* North-Holland, Amsterdam, New York, 1994.

[186] Sharygin, I.F.: A lower estimate for the error of quadrature formulas for certain classes of functions, Zh. Vychisl. Mat. i Mat. Fiz. 3, (1963) 370–376.

[187] Sleijpen, G.; Vorst, H.; Fokkema, D.: BiCGstab(l) and other hybrid Bi-CG methods, Numerical Algorithms, vol. 7, no. 1, pp. 75–109, 1994, DOI: 10.1007/BF02141261

[188] Sleijpen, G.L.G. and Fokkema, D.R.: BiCGSTAB(l) for linear equations involving unsymmetric matrices with complex spectrum, Electronic Transactions on Numerical Analysis. Kent, OH: Kent State University. 1: 11–32, 1993.

[189] Sloan, I.H., Joe, S.: Lattice Methods for Multiple Integration, Oxford University Press, Oxford (1994).

[190] Sloan, I.H., Kachoyan, P.J.: Lattice methods for multiple integration: Theory, error analysis and examples, SIAM J. Numer. Anal. 24, (1987) 116-128.

[191] Sloan, I.H., Lyness, J.N.: The representation of lattice quadrature rules as multiple sums. Mathematics of Computation, 52:8194, 1989.

[192] Sloan, I., Wozniakowski, H.: When Are Quasi-Monte Carlo Algorithms Efficient for High Dimensional Integrals? *J. Complexity*, 14:1–33, 1998.

[193] Sobol, I.M.: The distribution of points in a cube and the approximate evaluation of integrals. U.S.S.R Computational Mathematics and Mathematical Physics, 7(4):86–112, 1967.

[194] Sobol, I.M.: *Monte Carlo Numerical Methods*, Nauka, Moscow, 1973, (in Russian).

[195] Sobol, I.M., Levitan: The Production of Points Uniformly Distributed in a Multidimensional Cube (in Russian), Preprint IPM Akad. Nauk SSSR, Number 40, Moscow 1976.

[196] Sobol, I.M.: On Quadratic Formulas for Functions of Several Variables Satisfying a General Lipschitz Condition. *USSR Comput. Math. and Math. Phys.*, 29(6):936–941, 1989.

[197] Spotz, W.F., Carey, G.F.: A high-order compact formulation for the 3D Poisson equation. Numer. Meth. PDEs **12**, 235-243 (1996).

[198] Spotz, W.F., Carey, G.F.: Extension of high-order compact schemes to time-dependent problems. Numer. Meth. PDE **17**(6), 657–672 (2001)

[199] Stefanov S. and Cercignani C.: Monte Carlo Simulation of the Propagation of a Disturbance in the Channel Flow of a Rarefied Gas, Special issue "Simulation

methods in Kinetic Theory" of Computers and Mathematics with Applications, Vol. 35, No 1-2, pp. 41-53, 1998.

[200] Stefanov S., Gospodinov P. and Cercignani C.: Monte Carlo Simulation and Navier-Stokes Finite Differnce Solution of Rarefied Gas Flow Problems, Physics of Fluids, Vol. 10, No 1, pp. 289-300, 1998.

[201] Strassburg, J., Alexandrov, V.N.: *A Monte Carlo Approach to Sparse Approximate Inverse Matrix Computations.* **Procedia Computer Science, 18**, (2013), pp. 2307-2316.

[202] Sun, H., Zhang, J.: A high-order finite difference discretization strategy based on extrapolation for convection diffusion equations, Numer. Meth. PDEs **20**, 18–32 (2004).

[203] Suykens, F.: *On Robust Monte Carlo Algorithms for Multi-Pass Global Illumination.* PhD thesis, Katholieke Universiteit Leuven, 1997.

[204] Takev, M.: On Probable Error of the Monte Carlo Method for Numerical Integration. *Mathematica Balkanica (New Series)*, 6:231–235, 1992.

[205] Tan, K.S., Boyle, P.P.: Applications of randomized low discrepancy sequences to the valuation of complex securities. Journal of Economic Dynamics and Control, 24:1747-1782, 2000.

[206] Tanaka, H., Nagata, H.: Quasi-random Number Method for the Numerical Integration. *Supplement of the progress of theoretical physics*, 56:121–131, 1974.

[207] Tezuka, S.: Uniform Random Numbers: Theory and Practice. Kluwer Academic Publishers, Norwell, Massachusetts, 1995.

[208] Todorov, V., Dimov, I.: Monte Carlo methods for multidimensional integration for European option pricing, DOI:10.1063/1.4965003, AIP Conf. Proc. 1773, 100009, ISSN 0094243X, (2016).

[209] Todorov, V.: Computing high dimensional integrals with Monte Carlo methods, Journal Scientific and Applied Research, Vol.10, 2016, 11-16, ISSN 1314-6289.

[210] Turkel, E., Gordon, D., Gordon, R., Tsynkov, S.: Compact 2D and 3D sixth order schemes for the Helmholtz equation with variable wave number. J. Comput. Phys. **232**, 272-287 (2013).

[211] Van der Corput, J.G.: Verteilungsfunktionen I-VIII. Proc. Akad. Amsterdam, Vol. 38 (1935) 813-821, 1058-1066, Vol.39 (1936) 10-19, 19-26, 149-153, 339-344, 489-494, 579-590.

[212] Veach, E.: *Robust Monte Carlo Methods for Light Transport Simulation*, Ph.D. dissertation, Stanford University, 1997.

[213] Veach, E., Guibas, L.L Optimally combining sampling techniques for Monte Carlo rendering, in: *Computer Graphics Proceedings* (1995) 419–428.

[214] Vose, D.: The pros and cons of Latin Hypercube sampling, (2014).

[215] Vrahatis M.: A short proof and a generalization of Miranda's existence theorem, Proceeding of the American Mathematical Society, Volume 107, Number 3, November 1989, pp. 701-703.

[216] Wang, Y.: High accuracy multiscale multigrid computation for partial differential equations, Ph.D. thesis, University of Kentucky, Lexington, KY, 2010.

[217] 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)

[218] Wang, Y., Hickernell, F.J.: *An historical overview of lattice point sets*, (2002).

[219] Wang, X., Sloan, I.: Why Are High-Dimensional Finance Problems Often of Low Effective Dimension? *SIAM J. Sci. Comput.*, 27:159–183, 2005.

[220] Wigner, E.: On the quantum correction for thermodynamic equilibrium, Phys. Rev. 40 (1932) 749.

[221] Wilmott, P., Dewynne, J., Howison, S.: Option Pricing: Mathematical Models and Computation. Oxford University Press 1995

[222] Yeung, M.: ML(n)BiCGStab: Reformulation, Analysis and Implementation, submitted to Numerical Mathematics: Theory, Methods and Applications. Available at http://www.uwyo.edu/mathmyeung/r17.pdf.

[223] Zaremba, S. K.: Good lattice points, discrepancy, and numerical integration, Ann. Mat. Pura Appl. 73, (1966) 293–317.

[224] Zaremba, S.K.: The mathematical basis of Monte carlo and quasi-Monte Carlo methods. SIAM review, 10:303-314, 1968.

[225] Zaremba, S.K.: La methode des "Bons Treillis" pour le calcul des integrales multiples. In S.K. Zaremba, editor, Applications of Number Theory to Numerical Analysis, pages 39–119. Academica Press, New York, 1972.

[226] Zlatev, Z.: *Computer Treatment of Large Air Pollution Models* (Kluwer Academic Publishers, 1995).

[227] Zlatev, Z., Dimov, I.: *Computational and Numerical Challenges in Air Polution Modelling.* Elsevier Science, Amsterdam-Boston-...-Tokyo, (2006).

[228] 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)

[229] MATH 3795 Lecture 6. Sensitivity of the Solution of a Linear System: http://www.math.uconn.edu/~leykekhman/courses/MATH3795/Lectures/Lecture_6_Linear_system_error.pdf

[230] Linear systems: www.math.umd.edu/~petersd/466/linsysterrn.pdf

[231] Sobol quasirandom vector: https://people.sc.fsu.edu/~jburkardt/m_src/sobol/i4_sobol.m

[232] Mersenne Twister pseudorandom number generator: http://www.math.sci.hiroshima-u.ac.jp/m-mat/MT/emt.html

[233] 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

[234] Bufffon needle: http://www.webspace.ship.edu/deensley/mathdl/stats/Buffon.html

[235] Wolfram Mathematica: http://www.wolfram.com/products/mathematica/index.html

[236] bicgstabl: https://www.mathworks.com/help/matlab/ref/bicgstabl.html

[237] MATLAB - MathWorks: http://www.mathworks.com/