

УДК 519.676

Алгоритмы стохастического моделирования для итерационного решения уравнения Ламе*

И.А. Аксюк, А.Е. Киреева, К.К. Сабельфельд, Д.Д. Смирнов

Институт вычислительной математики и математической геофизики Сибирского отделения Российской академии наук, просп. Акад. Лаврентьева, 6, Новосибирск, 630090

E-mails: i.aksyuk@g.nsu.ru (Аксюк И.А.), kireeva@ssd.sccc.ru (Киреева А.Е.), karl@osmf.sccc.ru (Сабельфельд К.К.), smirnovdd@mail.ru (Смирнов Д.Д.)

Английская версия этой статьи печатается в журнале “Numerical Analysis and Applications” № 4, Vol. 16, 2023.

Аксюк И.А., Киреева А.Е., Сабельфельд К.К., Смирнов Д.Д. Алгоритмы стохастического моделирования для итерационного решения уравнения Ламе // Сиб. журн. вычисл. математики / РАН. Сиб. отд-ние. — Новосибирск, 2023. — Т. 26, № 4. — С. 357–377.

В статье построены итерационные алгоритмы стохастического моделирования для решения системы уравнений Ламе, описывающей смещения изотропного упругого тела. Предложено три различных стохастических метода. Первый основан на глобальном алгоритме блуждания по сферам для вычисления решения и производных для анизотропного уравнения диффузии. Он является бессеточным и не требует большой оперативной памяти. Второй метод основан на рандомизированном алгоритме для решения больших систем линейных уравнений и требует введения сеток. Третий метод также является сеточным, он использует алгоритм случайного блуждания по сетке. Все три метода реализуют итерационный процесс, на каждом шаге которого решаются уравнения анизотропной диффузии. В работе проводится сравнительный анализ предложенных методов и обсуждаются границы применимости каждого из них.

DOI: 10.15372/SJNM20230402

EDN: ATRIYU

Ключевые слова: бессеточный алгоритм, блуждание по сферам, глобальный алгоритм случайного блуждания, рандомизированный алгоритм решения линейных уравнений.

Aksyuk I.A., Kireeva A.E., Sabelfeld K.K., Smirnov D.D. Stochastic simulation algorithms for iterative solution of the Lamé equation // Siberian J. Num. Math. / Sib. Branch of Russ. Acad. of Sci. — Novosibirsk, 2023. — Vol. 26, № 4. — P. 357–377.

In this paper, iterative stochastic simulation algorithms for the Lamé equation describing the displacements of an isotropic elastic body are constructed. Three different stochastic methods are proposed: the first one is based on a global algorithm of random walk on spheres to compute the solution and its derivatives for an anisotropic diffusion equation. It does not use grids and does not require large amounts of RAM. The second method is based on a randomized algorithm for solving large systems of linear equations and requires the introduction of a grid. The third method is also grid-based and uses a random walk algorithm. All three methods implement an iterative process, at each step of which anisotropic diffusion equations are solved. The paper provides a comparative analysis of the proposed methods and discusses the limits of applicability of each of them.

Keywords: meshless stochastic algorithm, random walk on spheres, global random walk algorithm, randomized algorithm for solving linear equations.

*Работа выполнена при поддержке РФФИ (проект № 19-11-00019) и в рамках госзадания ИВМиМГ СО РАН (проект № 0251-2022-0002).

1. Введение

Известно, что традиционные стохастические методы типа алгоритмов блуждания по сферам неприменимы для решения систем эллиптических уравнений, связанных через свои вторые производные. Это связано с фундаментальным ограничительным свойством классических вероятностных представлений в пространстве винеровских процессов [9]. Отдельные попытки прямого обобщения метода блуждания по сферам на случай системы уравнений Ламе оказались ошибочными [8, 13], а введение ветвящихся процессов блуждания по сферам [22, 23] позволяет получать решение лишь с небольшой точностью.

В данной работе впервые строятся стохастические методы решения уравнения Ламе, основанные на стохастическом итерационном методе, в котором на каждой итерации решаются уравнения анизотропной диффузии. Предложено три различных стохастических итерационных алгоритма для решения системы уравнений Ламе, описывающей упругие смещения изотропной среды под произвольными нагрузками. Первый алгоритм бессеточный, он основан на глобальном методе блуждания по сферам для уравнения анизотропной диффузии [18, 19], который позволяет вычислять решение и его первые и вторые производные в произвольном множестве точек области. Второй и третий алгоритмы используют сетки. Во втором алгоритме конечно-разностная аппроксимация уравнения анизотропной диффузии решается с помощью рандомизированного алгоритма для решения систем линейных алгебраических уравнений (СЛАУ), предложенного в [14]. Третий алгоритм основан на дискретных случайных блужданиях, организованных таким образом, чтобы каждая траектория давала вклады в каждый узел сетки, который она посещает до обрыва на границе области [21]. В работе проводится сравнительный анализ точности и трудоёмкости этих трёх алгоритмов. Обсуждаются границы применимости каждого из методов, анализируются их сравнительные преимущества и недостатки. Этот анализ сопровождается серией численных экспериментов на тестовых задачах, имеющих известное точное решение.

2. Глобальный алгоритм блуждания по сферам для итерационного решения системы уравнений Ламе

В данном пункте строится бессеточный итерационный метод решения уравнения Ламе с помощью глобального алгоритма случайного блуждания по сферам, предложенного в [18, 19]. Здесь на каждой итерации решаются два линейных уравнения анизотропной диффузии, каждое из которых зависит от решения другого уравнения, с помощью метода глобального блуждания по сферам.

Для простоты изложения рассмотрим двумерную систему уравнений Ламе с нулевыми граничными условиями Дирихле:

$$\mu\Delta\mathbf{u}(x, y) + (\lambda + \mu)\nabla(\nabla \cdot \mathbf{u}) = -\mathbf{f}_c(x, y), \quad (x, y) \in \Omega, \quad (1)$$

$$\mathbf{u}(x, y) = 0, \quad (x, y) \in \Gamma, \quad (2)$$

где для вектор-столбцов принята запись $\mathbf{u} = (u_1, u_2)$, $\mathbf{f}_c(x, y) = (f_1(x, y), f_2(x, y))$. Область Ω ограничена границей Γ , $\lambda, \mu > 0$, и предполагаются выполненными условия существования и единственности решения данной краевой задачи [4, 5].

Обобщение задачи в трёхмерном пространстве не представляет труда.

Перепишем систему уравнений (1), (2) в виде

$$\begin{aligned} \frac{\partial^2 u_1}{\partial x^2} + \frac{\mu}{\lambda + 2\mu} \frac{\partial^2 u_1}{\partial y^2} &= -F_1(x, y), \\ \frac{\partial^2 u_2}{\partial x^2} + \frac{\lambda + 2\mu}{\mu} \frac{\partial^2 u_2}{\partial y^2} &= -F_2(x, y), \\ \mathbf{u}(x, y) &= 0, \quad (x, y) \in \Gamma, \end{aligned} \quad (3)$$

где

$$F_1(x, y) = \frac{\lambda + \mu}{\lambda + 2\mu} \frac{\partial^2 u_2}{\partial x \partial y} + \frac{1}{\lambda + 2\mu} f_1(x, y), \quad (4)$$

$$F_2(x, y) = \frac{\lambda + \mu}{\mu} \frac{\partial^2 u_1}{\partial x \partial y} + \frac{1}{\mu} f_2(x, y). \quad (5)$$

Данную систему уравнений будем решать с помощью итерационного процесса:

$$\begin{aligned} \frac{\partial^2 u_1^i}{\partial x^2} + \frac{\mu}{\lambda + 2\mu} \frac{\partial^2 u_1^i}{\partial y^2} &= -F_1^{i-1}(x, y), \\ \frac{\partial^2 u_2^i}{\partial x^2} + \frac{\lambda + 2\mu}{\mu} \frac{\partial^2 u_2^i}{\partial y^2} &= -F_2^{i-1}(x, y), \\ \mathbf{u}^i(x, y) &= 0, \quad (x, y) \in \Gamma. \end{aligned} \quad (6)$$

Запись процесса в виде системы (6) означает, что на каждой итерации $i = 1, \dots, i_{\max}$ мы решаем два линейных уравнения анизотропной диффузии с помощью глобального метода блуждания по сферам, используя на текущем шаге итерации в правой части вычисленное на предыдущем шаге приближённое значение функций $\frac{\partial^2 u_j^i}{\partial x \partial y}$. Этот итерационный процесс реализуется также в двух других стохастических методах в пунктах 3 и 4. Описанный итерационный процесс является частным случаем универсальной итерационной процедуры, описанной и исследованной в [3], где для уравнения Ламе получена геометрическая скорость сходимости с оценкой ошибки на n -й итерации в виде $\delta_n \leq Cq^n$, где $q = \sqrt{1 - \mu^2 / (\lambda m + 2\mu)^2}$, m — размерность задачи, в нашем случае $m = 2$.

2.1. Решение уравнения анизотропной диффузии с помощью глобального алгоритма блуждания по сферам

Рассмотрим скалярное уравнение анизотропной диффузии с нулевыми граничными условиями Дирихле в ограниченной области Ω с границей области Γ :

$$\begin{aligned} \frac{\partial^2 u(x, y)}{\partial x^2} + \alpha \frac{\partial^2 u(x, y)}{\partial y^2} &= -f(x, y), \\ u(x, y) &= 0, \quad (x, y) \in \Gamma, \end{aligned} \quad (7)$$

где $\alpha > 0$.

Решение этого уравнения может быть представлено в виде интеграла [6, 24]

$$u(x, y) = \int_{\Omega} G((x, y), \boldsymbol{\xi}) f(\boldsymbol{\xi}) d\boldsymbol{\xi},$$

где $G((x, y), \boldsymbol{\xi})$ — функция Грина, $\boldsymbol{\xi} = (\xi_1, \xi_2)$. Используя симметричность функции Грина, получаем

$$u(x, y) = \int_{\Omega} G((x, y), \boldsymbol{\xi}) f(\boldsymbol{\xi}) d\boldsymbol{\xi} = \int_{\Omega} G(\boldsymbol{\xi}, (x, y)) f(\boldsymbol{\xi}) d\boldsymbol{\xi}.$$

Функция Грина строится из представления

$$G(\boldsymbol{\xi}, (x, y)) = E(\boldsymbol{\xi}, (x, y)) + w(\boldsymbol{\xi}, (x, y)),$$

где $w(\boldsymbol{\xi}, (x, y))$ — решение краевой задачи

$$\frac{\partial^2 w(\boldsymbol{\xi}, (x, y))}{\partial \xi_1^2} + \alpha \frac{\partial^2 w}{\partial \xi_2^2} = 0, \quad (8)$$

$$w(\boldsymbol{\xi}^*, (x, y)) = -E(\boldsymbol{\xi}^*, (x, y)), \quad \boldsymbol{\xi}^* \in \Gamma. \quad (9)$$

Здесь $E(\boldsymbol{\xi}, (x, y))$ — фундаментальное решение анизотропного уравнения диффузии (8)

(см. [6]); $E(\boldsymbol{\xi}, (x, y)) = \frac{1}{2\pi\sqrt{\alpha}} \log\left(\frac{1}{r}\right)$, где $r = \sqrt{(x - \xi_1)^2 + \frac{(y - \xi_2)^2}{\alpha}}$.

Решение неоднородного уравнения $u(x, y)$ находим как математическое ожидание:

$$u(x, y) = \left\langle \frac{f(\boldsymbol{\xi})}{q(\boldsymbol{\xi})} G(\boldsymbol{\xi}, (x, y)) \right\rangle, \quad (10)$$

где $q(\boldsymbol{\xi})$ — плотность распределения случайной величины $\boldsymbol{\xi}$ в области Ω . Эта плотность может быть произвольной при предположении, что $q(\boldsymbol{\xi}) \neq 0$, если $f(\boldsymbol{\xi}) \neq 0$.

2.2. Плотность распределения точки выхода на окружность

Для метода блуждания по сферам (в нашем случае по окружностям) необходимо иметь аналитический вид функции Грина для круга, однако в случае анизотропной диффузии функция Грина нам не известна. Поэтому мы строим приближённую плотность для точки выхода на окружность, полученную с помощью блуждания по сетке в круге из центра круга радиуса R . Нетрудно показать, что угловое распределение точки выхода на окружность не зависит от радиуса круга, поэтому достаточно вычислять его для единичного круга.

Найдем вероятности перехода из точки на сетке с координатами (i, j) в соседнюю точку для дискретного случайного блуждания для уравнения

$$\frac{\partial^2 u(x, y)}{\partial x^2} + \alpha \frac{\partial^2 u(x, y)}{\partial y^2} = 0$$

на основе разностного приближения

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \alpha \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{h^2} = 0,$$

откуда

$$u_{i,j} = \frac{1}{2+2\alpha} u_{i+1,j} + \frac{1}{2+2\alpha} u_{i-1,j} + \frac{\alpha}{2+2\alpha} u_{i,j+1} + \frac{\alpha}{2+2\alpha} u_{i,j-1}.$$

Последнее соотношение определяет вероятностную интерпретацию: начиная блуждание из центра круга и блуждая по сетке с вероятностями $p_{i+1,j} = p_{i-1,j} = \frac{1}{2+2\alpha}$, $p_{i,j+1} = p_{i,j-1} = \frac{\alpha}{2+2\alpha}$, частица пересекает границу круга либо её ε -границу. При таком пересечении ближайшая точка для частицы на окружности имеет координаты (x^*, y^*) , которые соответствуют углу ϕ^* в полярной системе координат. Для каждой из N блуждающих частиц мы получим набор углов $\{\phi_i^*\}$, с помощью которого строим аппроксимирующую плотность, используя метод гистограмм. На рисунке 1 представлена вычисленная плотность распределения точек выхода на окружность для различных значений коэффициента α .

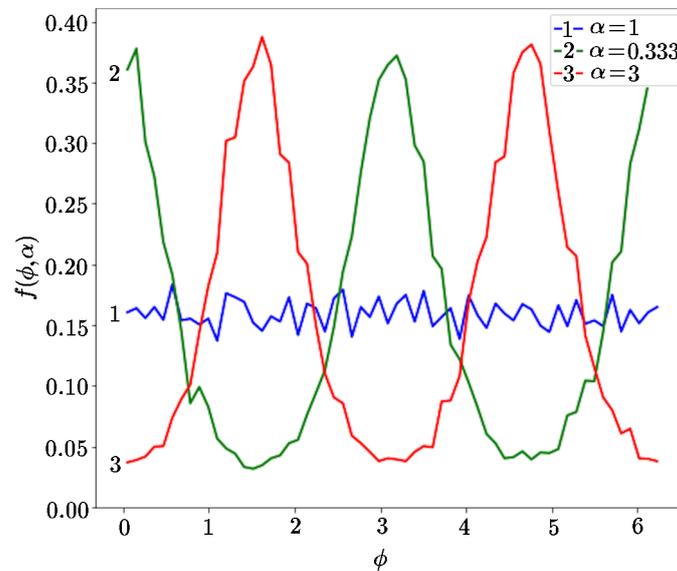


Рис. 1. Рассчитанная плотность распределения точек выхода на окружность для различных значений α . Видно, что случай уравнения Лапласа ($\alpha = 1$) соответствует равномерному распределению

При моделировании удобно использовать следующие свойства плотности распределения точки выхода на окружности:

- плотность $f(\phi, \alpha)$ не зависит от радиуса окружности;
- $f(\phi, \alpha) = f\left(\tilde{\phi}, \frac{1}{\alpha}\right)$, где $\tilde{\phi} = \phi + \frac{\pi}{2}$, если $\phi \in \left[0, \frac{3}{2}\pi\right]$, иначе $\tilde{\phi} = \phi - \frac{3}{2}\pi$;
- $f(\phi, 1) = \frac{1}{2\pi}$;
- $f\left(\frac{\pi}{2} - \phi, \alpha\right) = f\left(\frac{\pi}{2} + \phi, \alpha\right) = f\left(\frac{3\pi}{2} - \phi, \alpha\right) = f\left(\frac{3\pi}{2} + \phi, \alpha\right)$, где $\phi \in \left[0, \frac{\pi}{2}\right]$.

2.3. Приближение функции смешанной производной с помощью случайных оценок

Для решения линейного уравнения анизотропной диффузии на каждой итерации нужно знать значение смешанной производной $\frac{\partial^2 u_j^i(x, y)}{\partial x \partial y}$:

$$\frac{\partial^2 u_j^i(x, y)}{\partial x \partial y} = \frac{\partial^2 \left(\int_{\Omega} G(\xi, (x, y)) f(\xi) d\xi \right)}{\partial x \partial y} = \int_{\Omega} \frac{\partial^2 G(\xi, (x, y))}{\partial x \partial y} f(\xi) d\xi.$$

Для этого использовалось два подхода. В первом применялось прямое дифференцирование глобальной оценки в методе блуждания по сферам, во втором — приближение решения с помощью полинома на основе рассчитанных в узлах решений методом блуждания по сферам (см. п. 2.4).

В первом подходе смешанная производная $\frac{\partial^2 u_j^i(x, y)}{\partial x \partial y}$ вычисляется на заданной равномерной сетке размера S в области Ω как математическое ожидание случайной величины

$$\zeta_n(x, y) = \left[\frac{F_j^i(\tilde{x}_{\text{start}})}{q(x_{\text{start}})} \left(\frac{\partial^2 E(x_{\text{start}}, (x, y))}{\partial x \partial y} - \frac{\partial^2 E(x_{\text{end}}, (x, y))}{\partial x \partial y} \right) \right]_n,$$

где x_{start} — точка начала блуждания частицы с номером n в глобальном методе блуждания по сферам (эта точка разыгрывается согласно плотности распределения $q(\cdot)$ в области Ω), \tilde{x}_{start} — ближайшая точка сетки S для точки x_{start} , x_{end} — точка окончания блуждания частицы с номером n в глобальном методе блуждания по сферам, $F_j^i(\tilde{x}_{\text{start}})$ — функция источника:

$$F_1^i(\tilde{x}_{\text{start}}) = \frac{\lambda + \mu}{\lambda + 2\mu} \frac{\partial^2 u_2^{i-1}}{\partial x \partial y} + \frac{1}{\lambda + 2\mu} f_1(x_{\text{start}}),$$

$$F_2^i(\tilde{x}_{\text{start}}) = \frac{\lambda + \mu}{\mu} \frac{\partial^2 u_1^i}{\partial x \partial y} + \frac{1}{\mu} f_2(x_{\text{start}}).$$

Тогда

$$\frac{\partial^2 u(x, y)}{\partial x \partial y} \approx \langle \zeta_n(x, y) \rangle = \left\langle \left[\frac{F_j^i(\tilde{x}_{\text{start}})}{q(x_{\text{start}})} \left(\frac{\partial^2 E(x_{\text{start}}, (x, y))}{\partial x \partial y} - \frac{\partial^2 E(x_{\text{end}}, (x, y))}{\partial x \partial y} \right) \right]_n \right\rangle.$$

Замечание. Редкие траектории блуждания, для которых выполняется условие $r < 10^{-3}$, где $r = (\xi_1 - x)^2 + \frac{(\xi_2 - y)^2}{\alpha}$, ξ — точка начала или окончания блуждания, не учитываются при усреднении, поскольку вносят малый вклад в решение, но влияют заметно на дисперсию и погрешность.

Перейдем к описанию алгоритма.

1. Зададим N , ϵ и сетку размера S , в узлах которой мы вычисляем значения смешанных производных.
2. Положим $\frac{\partial^2 u_2^0(x, y)}{\partial x \partial y} = 0$ в каждом узле сетки S .
3. Для каждого из уравнений системы смоделируем серию траекторий блуждания по сферам, где каждая частица имеет точку начала блуждания x_{start} с плотностью распределения $q(x)$ в области Ω и точку окончания блуждания x_{end} . Для каждого из уравнений мы получим два массива: $array_{\text{start}}^j$, $array_{\text{end}}^j$, $j = 1, \dots, N$.
4. Зададим номер итерации $i = 1$.
5. Для каждой точки в сетке S вычислим значение смешанной производной $\frac{\partial^2 u_1^i(x, y)}{\partial x \partial y}$ как среднее случайной оценки ζ_n , используя значения $F_1^i(\tilde{x}_{\text{start}}, \frac{\partial^2 u_2^{i-1}(x, y)}{\partial x \partial y})$.

6. Для каждой точки в сетке S вычислим значение смешанной производной $\frac{\partial^2 u_2^i(x, y)}{\partial x \partial y}$ как среднее случайной оценки ζ_n , используя значения $F_2^i\left(\tilde{x}_{\text{start}}, \frac{\partial^2 u_1^i(x, y)}{\partial x \partial y}\right)$.
7. Если изменение смешанных производных функций u_1^i и u_2^i становится меньше заданной величины, то итерационный процесс останавливается. Вычисленные значения смешанных производных используются для дальнейшего решения уравнения. Иначе переходим к следующему шагу.
8. $i = i + 1$, переходим к шагу 5.

2.4. Приближение смешанной производной с помощью аппроксимирующего полинома

Для улучшения свойств гладкости и повышения точности вычисления производных будем аппроксимировать полученные на каждом шаге итерации решения линейных уравнений с помощью полиномов степени P в каждой ячейке Ω_s ($\Omega = \bigcup_{s=1}^S \Omega_s$, $\mu(\Omega_{s_1} \cap \Omega_{s_2}) = 0$, $s_1 \neq s_2$):

$$u_j^i(x, y) \approx \tilde{u}_j^i(x, y) = \sum_{p_1=0}^P \sum_{p_2=0}^P c_{p_1, p_2}^{j, s} X_{p_1}(x) Y_{p_2}(y),$$

где $(x, y) \in \Omega_s$, $X_p(x)$, $Y_p(y)$ — мономы или полиномы Чебышева степени p .

Опишем построение локальной СЛАУ $\tilde{A}\tilde{c} = \tilde{b}$ для нахождения коэффициента полинома для ячейки Ω_s :

1. Строим локальную сетку узлов размером $(P^* + 1)^2$, где $P^* \geq P$, в которых вычисляются приближённые значения $u_j^i(x, y)$, и сохраняем их в векторе правой части \tilde{b} .
2. Для каждого узла (x_q, y_q) заполняем соответствующую строку в матрице \tilde{A} , $\tilde{A}_{q, P \cdot p_1 + p_2} = X_{p_1}(x_q) Y_{p_2}(y_q)$, где q — номер узла.
3. При необходимости гладкости аппроксимирующей функции решения и аппроксимирующей функции смешанной производной решения используем условия согласования между ячейками: если ячейка Ω_s имеет общую границу с другой ячейкой Ω_c , то задаём набор точек на этой границе (x_l, y_l) , для каждой точки в матрице сохраняются условия согласования решений, такие как равенство первых производных, равенство вторых производных, равенство третьих производных.

Глобальная переопределённая СЛАУ $Ac = b$ получается из объединения локальных СЛАУ $\tilde{A}\tilde{c} = \tilde{b}$. Для решения переопределённой СЛАУ используем метод QR-разложения. Значение смешанной производной функции u_j^i находим как

$$\frac{\partial^2 u_j^i(x, y)}{\partial x \partial y} \approx \frac{\partial^2 \tilde{u}_j^i(x, y)}{\partial x \partial y} = \sum_{p_1=0}^P \sum_{p_2=0}^P c_{p_1, p_2}^{j, s} X'_{p_1}(x) Y'_{p_2}(y).$$

Опишем алгоритм итерационного метода решения уравнений Ламе с применением глобального метода блуждания по сферам с использованием приближения функции смешанной производной с помощью аппроксимирующего полинома:

1. Зададим N , ϵ , P , S .
2. Зададим начальные значения $c_{p_1, p_2}^{j, s} = 0$.

3. Построим глобальную матрицу A .
4. Для каждого из уравнений системы произведём блуждание по сферам, где каждая частица имеет точку начала блуждания x_{start} с плотностью распределения $q(x)$ в области Ω и точку окончания блуждания x_{end} , для каждого из уравнений мы получим два массива: $array_{\text{start}}^j, array_{\text{end}}^j$.
5. Зададим номер итерации $i = 1$.
6. Посчитаем значения $u_1^i(x, y)$ при заданных значениях $\frac{\partial^2 \tilde{u}_2^{i-1}(x, y)}{\partial x \partial y}$ в узлах сетки, которые использовались при построении глобальной СЛАУ, сохраним эти значения в векторе правых частей \mathbf{b} .
7. Решим глобальную СЛАУ, найдем коэффициенты $c_{p_1, p_2}^{1, s}$.
8. Используя коэффициенты $c_{p_1, p_2}^{1, s}$, найдем значения $\frac{\partial^2 \tilde{u}_1^i(x, y)}{\partial x \partial y}$.
9. Посчитаем значения $u_2^i(x, y)$ при заданных значениях $\frac{\partial^2 \tilde{u}_1^i(x, y)}{\partial x \partial y}$ в узлах сетки, которые использовались при построении глобальной СЛАУ, сохраним эти значения в векторе правых частей \mathbf{b} .
10. Решим глобальную СЛАУ, найдем коэффициенты $c_{p_1, p_2}^{2, s}$.
11. Используя коэффициенты $c_{p_1, p_2}^{2, s}$, найдем значения $\frac{\partial^2 \tilde{u}_2^i(x, y)}{\partial x \partial y}$.
12. Если максимальная разница коэффициентов между двумя итерациями достаточно мала, то итеративный процесс останавливается, иначе переходим к следующему шагу.
13. $i = i + 1$, переходим к шагу 6.

2.5. Эксперименты

Решалась система уравнений Ламе при $\lambda = \mu = 1$ в единичном квадрате Ω :

$$\mu \Delta u_1 + (\lambda + \mu) \frac{\partial}{\partial x} (u_{1x} + u_{2y}) = -2 \left(\mu (y - y^2 + x - x^2) + (\lambda + \mu) (3y - y^2 - 1 + 2x - 4xy) \right),$$

$$\mu \Delta u_2 + (\lambda + \mu) \frac{\partial}{\partial y} (u_{1x} + u_{2y}) = -4\mu (y - y^2 + x - x^2) - (\lambda + \mu) (6x - 4x^2 - 1 + 2y - 4xy),$$

$$\mathbf{u}(x, y) = 0, \quad (x, y) \in \Gamma.$$

Точное решение имеет вид

$$u_1(x, y) = x(1 - x)y(1 - y),$$

$$u_2(x, y) = 2x(1 - x)y(1 - y).$$

На рис. 2 и 3 представлены решение уравнения Ламе и смешанные производные от решения, полученные с помощью алгоритма из п. 2.3 для $N = 2 \cdot 10^5$, $\epsilon = 10^{-6}$, размер сетки $S = 1600$, $q((x, y)) = 1$ — плотность распределения точки начала блуждания, средняя относительная погрешность 1%, среднее время одной итерации ≈ 37 секунд.

На рис. 4 и 5 представлены решение уравнения Ламе и смешанные производные от решения, полученные с помощью алгоритма из п. 2.4 для $N = 10^6$, $S = 1$, $P^* = P = 3$, $\epsilon = 10^{-6}$, $q((x, y)) = 1$ — плотность распределения точки начала блуждания, средняя относительная погрешность составила 1%, среднее время одной итерации ≈ 45 секунд.

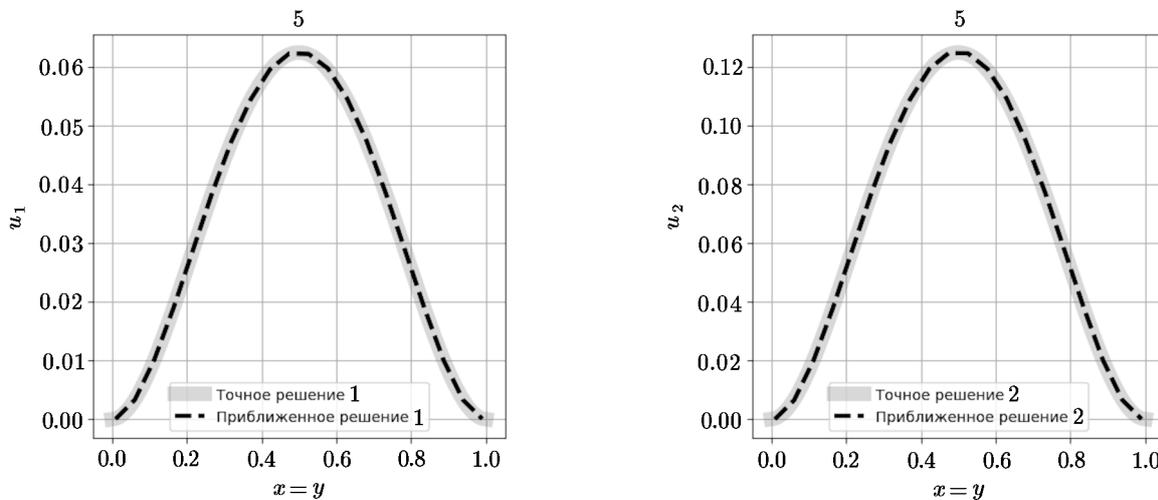


Рис. 2. Итерация 5, графики функций u_1 (слева) и u_2 (справа) на диагонали $x = y$ единичного квадрата Ω

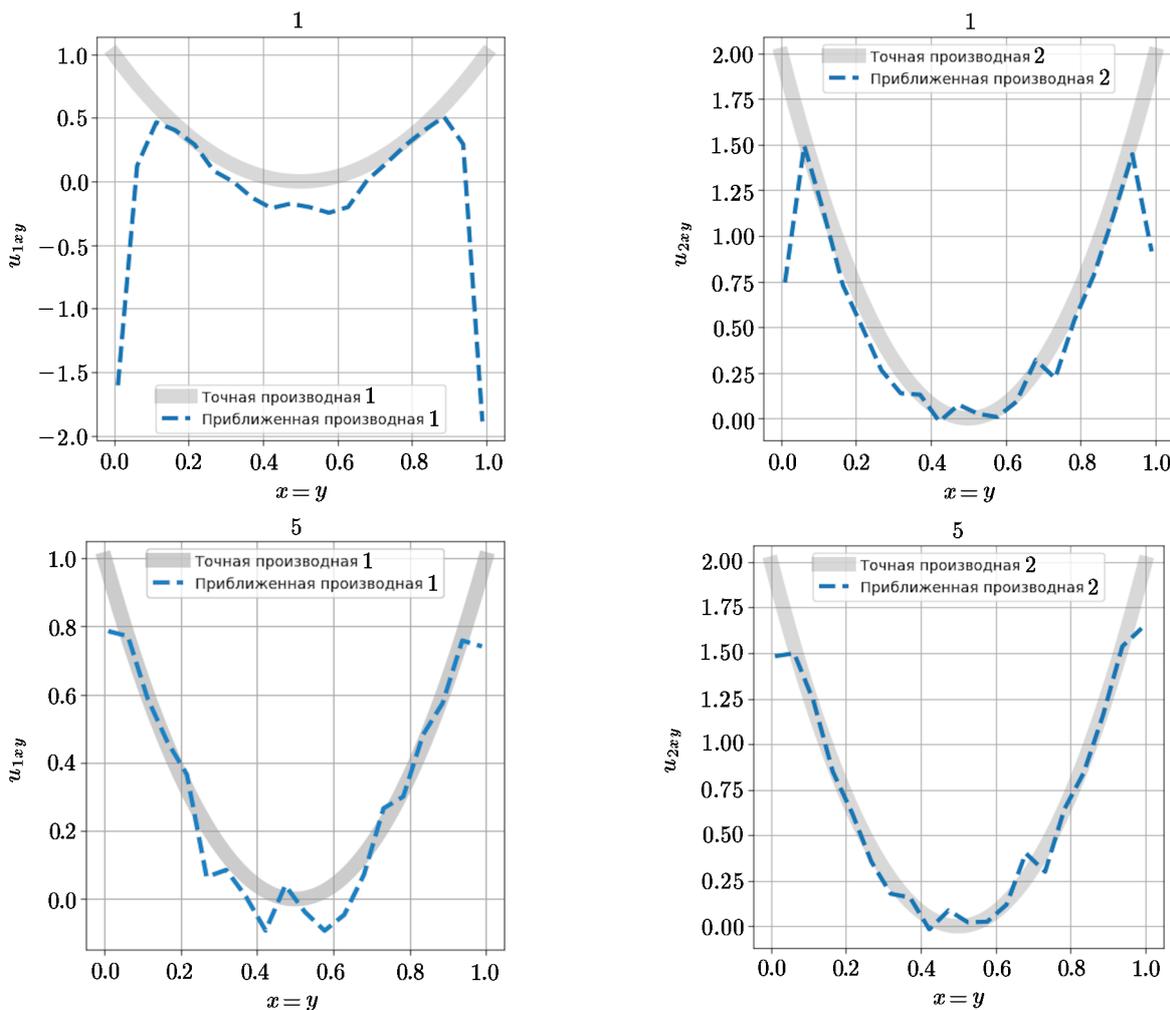


Рис. 3. Графики смешанных производных u_1 (слева) и u_2 (справа), на диагонали единичного квадрата Ω , итерация 1 (сверху), итерация 5 (снизу)

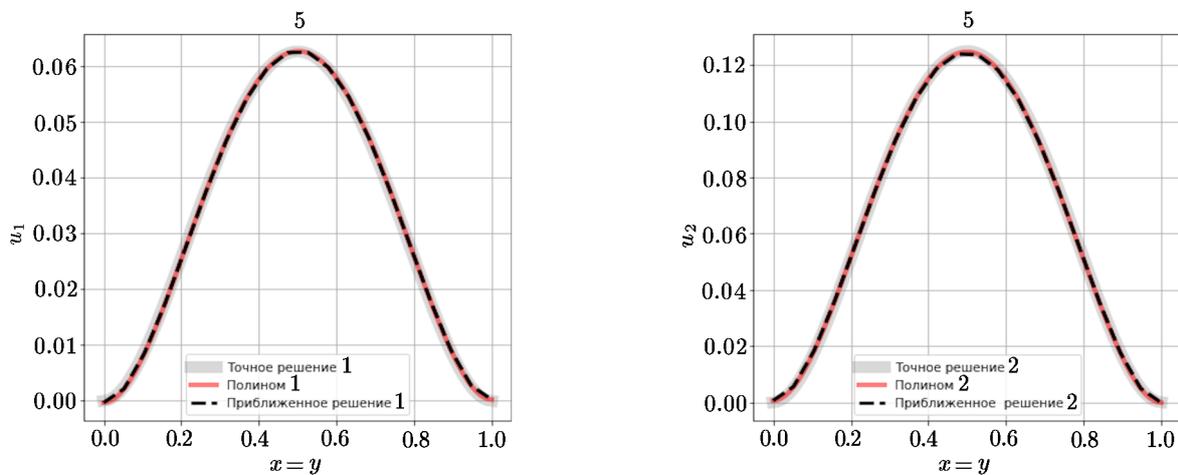


Рис. 4. Итерация 5, графики функций u_1 (слева) и u_2 (справа) на диагонали единичного квадрата Ω

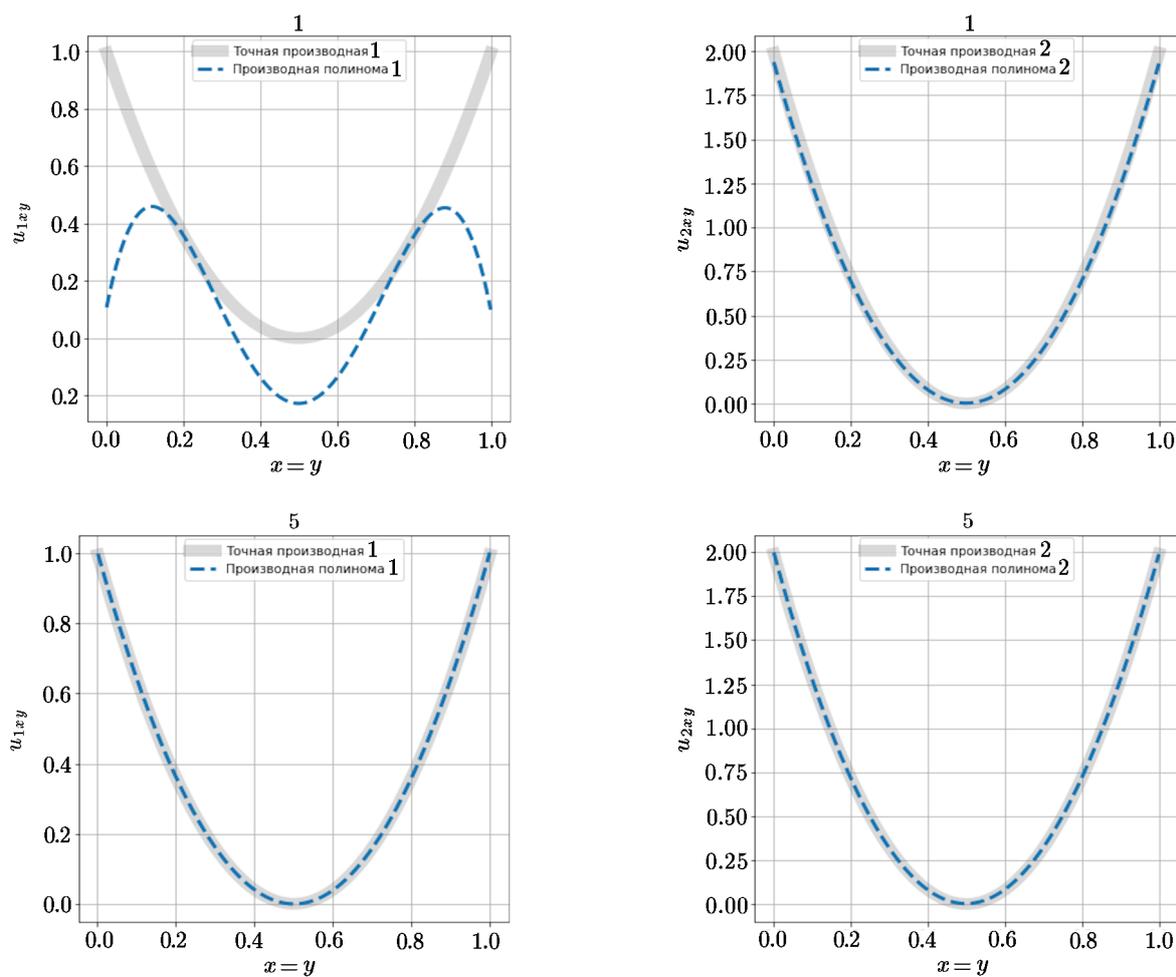


Рис. 5. Графики смешанных производных u_1 (слева) и u_2 (справа), на диагонали единичного квадрата Ω , итерация 1 (сверху), итерация 5 (снизу)

3. Итерационный рандомизированный алгоритм решения уравнения Ламе

3.1. Уравнение Ламе

Рассмотрим то же самое стационарное уравнение Ламе, что и в п. 2:

$$\begin{aligned} \mu \Delta \mathbf{u}(\mathbf{r}) + (\lambda + \mu) \nabla(\nabla \cdot \mathbf{u}(\mathbf{r})) &= -\mathbf{f}_c(\mathbf{r}), \quad \mathbf{r} \in G, \\ \mathbf{u}(\mathbf{r})|_{\Gamma} &= \mathbf{g}(\mathbf{r}), \end{aligned} \quad (11)$$

где $\mu \in R^+$ и $\lambda \in R^+$ — коэффициенты Ламе, $\mathbf{u}(\mathbf{r})$ — вектор перемещений в точке \mathbf{r} . Компоненты вектора $\mathbf{g}(\mathbf{r})$ задаются непрерывными функциями, определёнными на границе Γ .

В двумерном случае уравнение (11) может быть записано в следующем виде

$$\begin{cases} (\lambda + 2\mu) \frac{\partial^2 u_1(\mathbf{r})}{\partial x^2} + \mu \frac{\partial^2 u_1(\mathbf{r})}{\partial y^2} = -f_1(\mathbf{r}) - (\lambda + \mu) \frac{\partial^2 u_2(\mathbf{r})}{\partial x \partial y}, \\ \mu \frac{\partial^2 u_2(\mathbf{r})}{\partial x^2} + (\lambda + 2\mu) \frac{\partial^2 u_2(\mathbf{r})}{\partial y^2} = -f_2(\mathbf{r}) - (\lambda + \mu) \frac{\partial^2 u_1(\mathbf{r})}{\partial x \partial y}, \\ u_1(\mathbf{r})|_{\Gamma} = g_1(\mathbf{r}), \quad u_2(\mathbf{r})|_{\Gamma} = g_2(\mathbf{r}), \end{cases} \quad (12)$$

где $u_1(\mathbf{r})$ и $u_2(\mathbf{r})$ — компоненты вектора перемещений в точке \mathbf{r} вдоль осей X и Y . Здесь смешанные производные перенесены в правую часть уравнений.

3.2. Дискретизация уравнения Ламе

Уравнение Ламе (12) решается в двумерной квадратной области G размером $L \times L$. Для дискретизации области вводится равномерная сетка, состоящая из $n \times n$ узлов: $\omega = \{(x_i, y_j), x_i = ih, i = 0, \dots, n, y_j = jh, j = 0, \dots, n\}$, где шаг $h = L/(n-1)$.

Для получения разностного аналога уравнения Ламе (12) непрерывные функции $u_k(\mathbf{r})$, $f_k(\mathbf{r})$ и $g_k(\mathbf{r})$ заменяются сеточными функциями $\bar{u}_k(i, j)$, $\bar{f}_k(i, j)$ и $\bar{g}_k(i, j)$, $k = 1, 2$. Выражения для производных заменяются конечно-разностными операторами со вторым порядком точности.

В результате дискретизации исходная задача (12) сводится к СЛАУ. Конечно-разностная аппроксимация уравнения Ламе (12) для внутренних узлов сетки $\{(x_i, y_j) : i, j = 1, \dots, n-1\}$ записывается в виде

$$c_1(\bar{u}_1(i-1, j) + \bar{u}_1(i+1, j)) + b_1(\bar{u}_1(i, j-1) + \bar{u}_1(i, j+1)) + a_1 \bar{u}_1(i, j) = F_1(i, j), \quad (13)$$

$$c_1 = \frac{(2\mu + \lambda)}{h^2}, \quad b_1 = \frac{\mu}{h^2}, \quad a_1 = \frac{(6\mu + 2\lambda)}{h^2},$$

$$F_1(i, j) = -\bar{f}_1(i, j) - \frac{(\lambda + \mu)}{h^2}(\bar{u}_2(i, j) - \bar{u}_2(i, j+1) - \bar{u}_2(i+1, j) + \bar{u}_2(i+1, j+1));$$

$$c_2(\bar{u}_2(i-1, j) + \bar{u}_2(i+1, j)) + b_2(\bar{u}_2(i, j-1) + \bar{u}_2(i, j+1)) + a_2 \bar{u}_2(i, j) = F_2(i, j), \quad (14)$$

$$c_2 = \frac{\mu}{h^2}, \quad b_2 = \frac{(2\mu + \lambda)}{h^2}, \quad a_2 = \frac{(6\mu + 2\lambda)}{h^2},$$

$$F_2(i, j) = -\bar{f}_2(i, j) - \frac{(\lambda + \mu)}{h^2}(\bar{u}_1(i, j) - \bar{u}_1(i, j+1) - \bar{u}_1(i+1, j) + \bar{u}_1(i+1, j+1)).$$

Значения сеточных функций $\bar{u}_k(\mathbf{r})$ в граничных узлах сетки $\{(x_i, y_j) : i = \{0, n\} \text{ или } j = \{0, n\}\}$ определяются граничными значениями $\bar{u}_k(i, j) = \bar{g}_k(i, j)$, $k = 1, 2$.

Уравнения (13), (14) можно записать также в матричном виде

$$\begin{cases} A_1 \bar{\mathbf{u}}_1 = \mathbf{F}_1, \\ A_2 \bar{\mathbf{u}}_2 = \mathbf{F}_2, \end{cases} \quad (15)$$

где компоненты вектора решения $\bar{\mathbf{u}}_k$ и вектора правой части \mathbf{F}_k , $k = 1, 2$, упорядочены следующим образом: $\bar{\mathbf{u}}_k = [\bar{u}_k(1, 1), \dots, \bar{u}_k(1, n-1), \dots, \bar{u}_k(n-1, 1), \dots, \bar{u}_k(n-1, n-1)]$, $\mathbf{F}_k = [F_k(1, 1), \dots, F_k(1, n-1), \dots, F_k(n-1, 1), \dots, F_k(n-1, n-1)]$. Матрица коэффициентов A_k , $k = 1, 2$, — симметричная пятидиагональная матрица размера $p \times p$, где $p = (n-1)^2$.

3.3. Метод Гаусса–Зейделя с красно-чёрным упорядочиванием узлов

В данном пункте для решения СЛАУ используется рандомизированный векторный алгоритм, предложенный в [14]. Этот метод основан на представлении решения СЛАУ

$$\mathbf{x} = A\mathbf{x} + \mathbf{b} \quad (16)$$

в виде ряда Неймана

$$\mathbf{x} = \sum_{m=0}^{\infty} A^m \mathbf{b}, \quad (17)$$

который сходится при условии, что спектральный радиус матрицы $\rho(A) < 1$.

Для приведения каждой СЛАУ системы (15) к нужному виду (16) и улучшению спектральных свойств матрицы A_k , $k = 1, 2$, используется красно-чёрное упорядочивание узлов сетки и преобразование к матричному виду метода Гаусса–Зейделя [1, 7].

При красно-чёрном упорядочивании вначале нумеруются узлы, сумма индексов $i + j$ которых даёт чётное число (такие узлы называются красными узлами), потом нумеруются узлы, сумма индексов которых даёт нечётное число (такие узлы называются чёрными узлами). Элементы матриц A_k и векторов $\bar{\mathbf{u}}_k$, \mathbf{F}_k переупорядочиваются в соответствии с новой нумерацией узлов. Например, рассмотрим сетку ω размером 3×3 . В этом случае матрица коэффициентов, соответствующая красно-чёрному разбиению, имеет вид

$$A_k^{\text{RB}} = \left(\begin{array}{ccccc|cccc} a_k & 0 & 0 & 0 & 0 & b_k & c_k & 0 & 0 \\ 0 & a_k & 0 & 0 & 0 & b_k & 0 & c_k & 0 \\ 0 & 0 & a_k & 0 & 0 & c_k & b_k & b_k & c_k \\ 0 & 0 & 0 & a_k & 0 & 0 & c_k & 0 & b_k \\ 0 & 0 & 0 & 0 & a_k & 0 & 0 & c_k & b_k \\ \hline b_k & b_k & c_k & 0 & 0 & a_k & 0 & 0 & 0 \\ c_k & 0 & b_k & c_k & 0 & 0 & a_k & 0 & 0 \\ 0 & c_k & b_k & 0 & c_k & 0 & 0 & a_k & 0 \\ 0 & 0 & c_k & b_k & b_k & 0 & 0 & 0 & a_k \end{array} \right). \quad (18)$$

Таким образом, для любого размера сетки n матрица коэффициентов является симметричной блочной (состоящей из четырёх блоков) разрежённой матрицей с диагональным преобладанием [7], и каждое СЛАУ системы (15) преобразуется к виду

$$\begin{pmatrix} a_k E & C_k \\ C_k^\top & a_k E \end{pmatrix} \begin{pmatrix} \bar{\mathbf{u}}_k^R \\ \bar{\mathbf{u}}_k^B \end{pmatrix} = \begin{pmatrix} \mathbf{F}_k^R \\ \mathbf{F}_k^B \end{pmatrix}, \quad (19)$$

где $a_k E$ — диагональная матрица, элементы которой равны a_k . Матрица C_k равна верхнему правому блоку матрицы A_k^{RB} . Матрица C_k^\top — это транспонированная матрица C_k . Векторы $\bar{\mathbf{u}}_k^R$, \mathbf{F}_k^R , $\bar{\mathbf{u}}_k^B$, \mathbf{F}_k^B состоят из элементов векторов $\bar{\mathbf{u}}_k$ и \mathbf{F}_k в красных и чёрных узлах соответственно.

В матричном представлении метода Гаусса–Зейделя каждое СЛАУ системы (15) записывается в следующем виде:

$$\bar{\mathbf{u}}_k^{t+1} = \tilde{A}_k \bar{\mathbf{u}}_k^t + \mathbf{b}_k, \quad (20)$$

$$\tilde{A}_k = (E - D_k^{-1} L_k)^{-1} D_k^{-1} U_k, \quad (21)$$

$$\mathbf{b}_k = (E - D_k^{-1} L_k)^{-1} D_k^{-1} \mathbf{F}_k, \quad (22)$$

где D_k , L_k и U_k — матрицы, состоящие из элементов матрицы A_k^{RB} : D_k — диагональная, L_k — строго нижняя треугольная, U_k — строго верхняя треугольная матрица, E — единичная матрица. При красно-чёрном упорядочивании обратная матрица $(E - D_k^{-1} L_k)^{-1} = (E + D_k^{-1} L_k)$ [20, гл. 9, стр. 176]. Вычисленная по формуле (21) матрица \tilde{A}_k содержит нулевые элементы, соответствующие красным узлам. Это является следствием представления метода Гаусса–Зейделя для красно-чёрного разбиения. Итерация метода Гаусса–Зейделя для системы в матричной форме (19) может быть записана в виде

$$\begin{cases} \bar{\mathbf{u}}_R^{(t+1)} = \frac{1}{a} (\bar{\mathbf{F}}_R - C \bar{\mathbf{u}}_B^{(t+1)}), \\ \bar{\mathbf{u}}_B^{(t+1)} = \frac{1}{a} (\bar{\mathbf{F}}_B - \frac{1}{a} C^\top (\bar{\mathbf{F}}_R - C \bar{\mathbf{u}}_B^{(t)})). \end{cases} \quad (23)$$

Таким образом, итерации метода Гаусса–Зейделя преобразуются в последовательные итерации метода Якоби. Сначала находится решение СЛАУ в чёрных узлах ($\bar{\mathbf{u}}_B$). Затем решение в красных узлах $\bar{\mathbf{u}}_R$ явным образом выражается через вычисленные значения $\bar{\mathbf{u}}_B$.

3.4. Итерационный метод решения уравнения Ламе

В результате красно-чёрного упорядочивания узлов и применения формул (20)–(22) исходная система (15) преобразуется в систему

$$\begin{cases} \bar{\mathbf{u}}_1^{t+1} = \tilde{A}_1 \bar{\mathbf{u}}_1^t + \mathbf{b}_1^t, \\ \bar{\mathbf{u}}_2^{t+1} = \tilde{A}_2 \bar{\mathbf{u}}_2^t + \mathbf{b}_2^t. \end{cases} \quad (24)$$

Как следует из формул (13), (14), правые части \mathbf{b}_1 и \mathbf{b}_2 зависят от решений $\bar{\mathbf{u}}_1$ и $\bar{\mathbf{u}}_2$ соответственно, поэтому для решения системы (24) используется итерационный алгоритм. На первой итерации значение правой части \mathbf{b}_2^0 выбирается произвольным образом, для этих значений вычисляются решение $\bar{\mathbf{u}}_1^0$ и правая часть $\mathbf{b}_2^0(\bar{\mathbf{u}}_1^0)$. Далее на каждой итерации вычисляются решение $\bar{\mathbf{u}}_2^t$ и правая часть $\mathbf{b}_1^t(\bar{\mathbf{u}}_2^t)$, а также решение $\bar{\mathbf{u}}_1^t$ и правая часть $\mathbf{b}_2^t(\bar{\mathbf{u}}_1^t)$.

3.5. Векторный рандомизированный алгоритм решения СЛАУ

Решение СЛАУ (16) можно представить в виде ряда Неймана (17). Согласно [2, 14], ряд Неймана сходится, если спектральный радиус матрицы $\rho(A) < 1$. В [14] предложен векторный рандомизированный алгоритм вычисления произведений $A^m \mathbf{b}$. Опишем алгоритм сначала для стохастических матрицы и вектора. Вектор $\mathbf{b} \in R^n$ называется стохастическим, если $\sum_{i=1}^n b_i = 1$ и $b_i \geq 0$. Матрица A размера $n \times n$ называется стохастической по столбцам, если $a_{ij} \geq 0$ и $\sum_{i=1}^n a_{ij} = 1$, $j = 1, 2, \dots, n$.

Оценка произведения $A\mathbf{b}$ вычисляется как математическое ожидание по столбцам матрицы A , выбранным в соответствии с распределением вероятности \mathbf{b} : $A\mathbf{b} = E\xi_{j_1} = E\{A_j | \mathbf{p} = \mathbf{b}\}$, где A_j обозначает j -й столбец матрицы A . Оценка произведения $A^2\mathbf{b}$ вычисляется как двойное математическое ожидание по столбцам матрицы A , выбранным за два этапа: сначала, согласно распределению \mathbf{b} , выбирается столбец A_j , затем, согласно распределению A_j , выбирается столбец A_i : $A^2\mathbf{b} = E\xi_{j_2, j_1} = AE\xi_{j_1} = EE\{A_i | \mathbf{p}_2 = A_j; \mathbf{p}_1 = \mathbf{b}\}$. Аналогично вычисляются оценки произведений $A^m\mathbf{b}$ для более высоких степеней m . Каждый следующий столбец выбирается в соответствии с распределением вероятности столбца, выбранного на предыдущем шаге.

Для выбора случайного индекса очередного столбца в соответствии с распределением вероятности выбранного столбца используется алгоритм Уолкера (Walker's alias method) [25]. Согласно этому алгоритму, для всех столбцов матрицы A и для вектора \mathbf{b} строятся таблицы вероятности и меток размера n , которые далее используются для вычисления всех оценок $A^m\mathbf{b}$.

Для произвольного вектора \mathbf{b} оценка произведения $A^m\mathbf{b}$ вычисляется как математическое ожидание $E\left\{\frac{b_{j_1}}{p_{j_1}} \xi_{j_k, \dots, j_2, j_1}\right\}$, где \mathbf{p} — плотность распределения индексов $j = 1, 2, \dots, n$, удовлетворяющая условию $p_j \neq 0$, если $b_j \neq 0$, $j = 1, \dots, n$.

В [14] также приведён алгоритм вычисления произведений $A^m\mathbf{b}$ для произвольной неотрицательной матрицы A , который основан на приведении A к стохастической по столбцам матрице $S = \frac{1}{\lambda} ZAZ^{-1}$, где $\lambda > 0$ — собственное число матрицы A^\top , z — собственный вектор, соответствующий собственному числу λ : $A^\top z = \lambda z$, $Z = \text{diag}\{z_1, \dots, z_n\}$ — диагональная матрица, составленная из элементов собственного вектора z .

В качестве собственного числа λ выбирается максимальное собственное число матрицы A^\top , которое вычисляется с помощью степенного метода [12] следующим образом:

$$\lambda = \frac{(\mathbf{r}_k, \mathbf{r}_{k-1})}{(\mathbf{r}_{k-1}, \mathbf{r}_{k-1})}, \quad r_k = \frac{A^\top \mathbf{r}_{k-1}}{\|A^\top \mathbf{r}_{k-1}\|}. \quad (25)$$

В качестве начального приближения $\mathbf{r}_0 \in R^n$ выбирается единичный вектор. Вычисления продолжаются до выполнения условия сходимости $\|\mathbf{r}_k - \mathbf{r}_{k-1}\| > \epsilon$, где $\epsilon \in R$ — погрешность метода. Степенной метод позволяет вычислить одновременно максимальное собственное число λ и соответствующий ему собственный вектор $\mathbf{z} = \mathbf{r}_k$.

Степени исходной матрицы A^m выражаются через степени стохастической по столбцам матрицы S^m следующим образом: $A^m = \lambda^m Z^{-1} S^m Z$, $m = 1, 2, \dots$.

В результате преобразования матрицы A к стохастической по столбцам матрице S исходная СЛАУ (16) преобразуется следующим образом [14]:

$$A = \lambda Z^{-1} S Z, \quad \mathbf{x} = \lambda Z^{-1} S Z \mathbf{x} + \mathbf{b}, \quad Z \mathbf{x} = \lambda S Z \mathbf{x} + Z \mathbf{b},$$

сделаем замену переменных $\mathbf{y} = Z \mathbf{x}$ и $\tilde{\mathbf{b}} = Z \mathbf{b}$, получим новую систему

$$\mathbf{y} = \lambda S \mathbf{y} + \tilde{\mathbf{b}}. \quad (26)$$

Аналогично (16), решение системы (26) ищется в виде ряда Неймана. Очевидно, для вычисления решения СЛАУ с точностью ϵ число членов ряда Неймана M должно быть пропорционально $|\log(\epsilon)|$.

Таким образом, с помощью векторных рандомизированных оценок вычисляются произведения $S^m \tilde{\mathbf{b}}$, $m = 1, \dots, M$, для стохастической матрицы S и произвольного вектора $\tilde{\mathbf{b}}$. Оценки вычисляются как математическое ожидание по N траекториям. Траекторией здесь называется множество случайных индексов столбцов матрицы, которые выбираются при вычислении одной реализации случайных оценок $S^m \tilde{\mathbf{b}}$. На каждой траектории вычисляются вклады q_m во все оценки $S^m \tilde{\mathbf{b}}$, $m = 1, \dots, M$, которые суммируются в вектор решения \mathbf{y} . Значение вклада вычисляется по формуле $q_m = w \lambda^m S_{j_k}$, где $w = \frac{\tilde{b}_{j_1}}{p_{j_1}}$, j_1 — случайный индекс столбца матрицы S , выбранный в соответствии с распределением \mathbf{p} , j_k — случайный индекс, выбранный, согласно распределению столбца $S_{j_{k-1}}$, выбранного на предыдущем шаге.

3.6. Результаты моделирования

Реализована параллельная версия векторного рандомизированного алгоритма решения СЛАУ путём распределения траекторий между доступными ядрами суперкомпьютера с помощью библиотеки MPI. Вычислительные эксперименты проводились на кластере “МВС-10П” межведомственного суперкомпьютерного центра РАН [10].

Для верификации реализованной программы решается уравнение Ламе (12) с правыми частями $f_k(x, y) = a_k 2y(y-1) + b_k 2x(x-1) + 2(\lambda + \mu)(2x-1)(2y-1)$, $k = 1, 2$, $a_1 = b_2 = (2\mu + \lambda)$, $a_2 = b_1 = \mu$, и значениями коэффициентов Ламе $\lambda = 2$, $\mu = 6$. Решение ищется в квадрате G со стороной $L = 1$, на границах области заданы нулевые условия: $u_1(\mathbf{r})|_{\Gamma} = 0$, $u_2(\mathbf{r})|_{\Gamma} = 0$. Решением системы является функция $u_1(x, y) = u_2(x, y) = x(x-1)y(y-1)$. Для дискретизации области G используется сетка размером $n = 52 \times 52$ узла, с шагом $h = 0.0196$. Спектральный радиус матриц \tilde{A}_k (21) для таких значений параметров равен $\rho(\tilde{A}_k) = 0.996$, $k = 1, 2$. Число членов ряда Неймана $M = 1300$, число траекторий $N = 10^6$.

В результате моделирования с помощью векторного рандомизированного алгоритма вычислена максимальная дисперсия оценки решения СЛАУ $D = 0.0098$ и относительная погрешность решения СЛАУ $\delta = 0.007$. На рис. 6 а представлены решение СЛАУ, полученное численно с помощью рандомизированного алгоритма, и точное решение системы для приграничного значения $y = h$ и $x \in [h, L - h]$. Видно, что численное решение u находится внутри доверительного интервала $[u - 3\sigma; u + 3\sigma]$, $\sigma = \sqrt{D/N}$, и совпадает с точным решением СЛАУ u^* .

Итерационный алгоритм решения уравнения Ламе с приведёнными выше значениями параметров позволяет получить решение системы (12) с относительной погрешностью $\delta = 0.01$ за 4 итерации. На рис. 6 б–6 г представлены результаты моделирования: решение уравнения Ламе $u_{_}1e6$, правая часть $b_{_}1e6$ и производная $d_{_}1e6$ от решения, вычисленная по формуле (13), для значений $y = 0.25$ и $x \in [h, L - h]$. Видно, что решение $u_{_}1e6$ практически совпадает с точным значением u^* , вычисленным аналитически. В значениях правой части $b_{_}1e6$ и производной $d_{_}1e6$ присутствует шум, который уменьшается при увеличении числа траекторий N . Например, для $N = 10^7$ значения $b_{_}1e7$ и $d_{_}1e7$ ближе к точным значениям b^* , d^* . Однако увеличение числа траекторий приводит к увеличению времени вычислений. При использовании 48 MPI-процессов, запускаемых на вычислительном узле “Cascade_lake”, при $N = 10^6$ время решения одной СЛАУ составляет 0.4 с, время решения уравнения Ламе (12) — 3.6 с. При $N = 10^7$ время решения

СЛАУ — 3.56 с, время решения уравнения Ламе — 30.2 с. Время работы последовательной программы при $N = 10^6$ составляет 14.7 с для решения одной СЛАУ и 1.97 минут для решения уравнения Ламе.

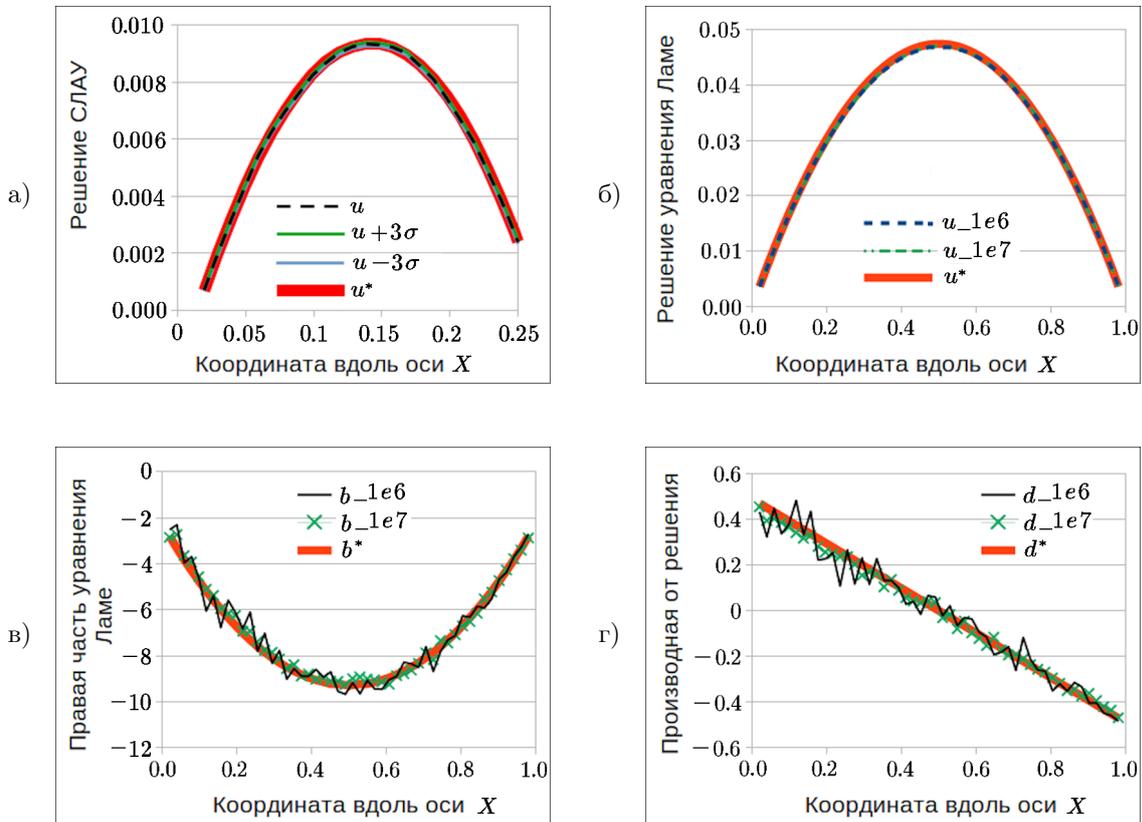


Рис. 6. Сравнение точных значений u^* , b^* , d^* и результатов моделирования: а — решение СЛАУ при $y = 0.01$; б-г — решение уравнения Ламе, правая часть уравнения и производная от решения, полученные для числа траекторий $N = 10^6$ и $N = 10^7$ при $y = 0.25$

4. Глобальный алгоритм блуждания по сетке

В данном пункте уравнение Ламе решается с помощью того же итерационного процесса, что был применен в п. 1 и п. 2, но система разностных уравнений для двух связанных анизотропных уравнений диффузии, представленная в п. 2, решается здесь с помощью глобального алгоритма блуждания по сетке, подробно описанного в [21].

Пусть сетка выбрана точно так же, как и в п. 2. Общая схема итерационного решения системы уравнений Ламе с помощью глобального алгоритма блуждания по сетке может быть описана следующим образом.

1. Выбираем начальное приближение $f_1 = 0$.
2. Решаем первое уравнение анизотропной диффузии с помощью глобального алгоритма блуждания по сетке, находим решение u_1 во всех узлах сетки.
3. Вычисляем $f_2 \approx \frac{u_{1_{i+1j+1}} - u_{1_{ij+1}} - u_{1_{i+1j}} + u_{1_{ij}}}{h^2}$.

4. Решаем второе уравнение с помощью алгоритма блуждания, вычисляем решение u_2 во всех узлах сетки.
5. Вычисляем $f_1 \approx \frac{u_{2_{i+1j+1}} - u_{2_{ij+1}} - u_{2_{i+1j}} + u_{2_{ij}}}{h^2}$.
6. Переходим к шагу 2 и повторяем все действия, пока не выполняются условия:
 $\max_{i,j} |u_{1_{ij}}^{(k+1)} - u_{1_{ij}}^{(k)}| < \varepsilon, \max_{i,j} |u_{2_{ij}}^{(k+1)} - u_{2_{ij}}^{(k)}| < \varepsilon$.

Для реализации алгоритма блуждания по сетке остается только выписать вероятности перехода “влево–вправо” и “вверх–вниз”.

Введём обозначения для вероятностей перехода в соседний узел: p_1 — вероятность перехода в узел справа, p_2 — в узел слева, p_3 — в узел сверху, p_4 — в узел снизу. Тогда для первого уравнения:

$$p_1 = p_2 = \frac{\lambda + 2\mu}{2\lambda + 6\mu}, \quad p_3 = p_4 = \frac{\mu}{2\lambda + 6\mu},$$

для второго уравнения:

$$p_1 = p_2 = \frac{\mu}{2\lambda + 6\mu}, \quad p_3 = p_4 = \frac{\lambda + 2\mu}{2\lambda + 6\mu}.$$

Статистический вес правой части для обоих уравнений $q = \frac{\lambda + \mu}{2\lambda + 6\mu} h^2$, где h — шаг сетки.

4.1. Численные эксперименты

В первой тестовой задаче решаем уравнение Ламе с нулевой правой частью при $\mu = 6$, $\lambda = 2$ в единичном квадрате. В качестве точного решения берём полиномы второй степени:

$$u_1(x, y) = 6x^2 - 24xy + 2y^2, \quad u_2(x, y) = 8x^2 - 19xy + 2y^2,$$

след которых на границе задаёт нам граничные условия Дирихле.

На рис. 7 показаны точное решение $u_1(x, 0.5)$ (сплошная линия) и численное решение (пунктир) для количества итераций $k = 5$; моделировалось 10^6 траекторий, шаг сетки $h = 0.02$. Средняя относительная погрешность составила 1.9%, время расчёта на одном процессоре 20 мин.

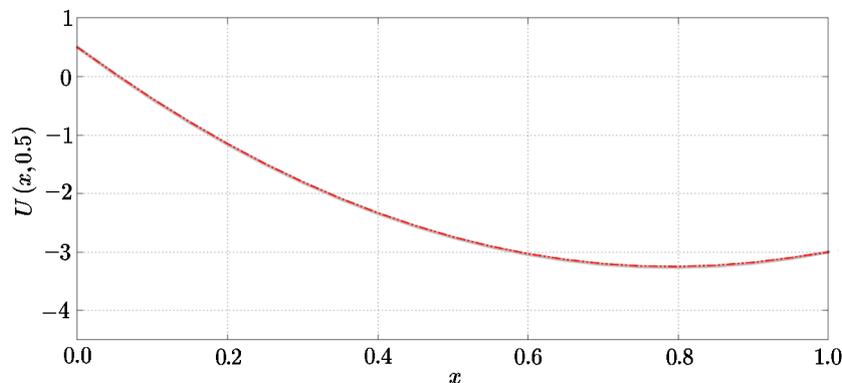


Рис. 7. Решение $U(x, 0.5) = u_1(x, 0.5)$ как функция x при фиксированном $y = 0.5$: сплошная линия — точное решение, пунктирная — численное решение

Второй пример взят аналогичным задаче из п. 1 и п. 2. В качестве точного решения выбираем функцию $u_1(x, y) = u_2(x, y) = x(x - 1)y(y - 1)$, удовлетворяющую нулевым граничным условиям для единичного квадрата. Правая часть уравнения Ламе при этом, с учётом переноса смешанной производной в правую часть, будет иметь вид

$$f_1 = - \left(\frac{\lambda + 2\mu}{\lambda + \mu} (2y^2 - 2y) + \frac{\mu}{\lambda + \mu} (2x^2 - 2x) + (4xy - 2x - 2y + 1) - \frac{\partial^2 V}{\partial x \partial y} \right),$$

$$f_2 = - \left(\frac{\mu}{\lambda + \mu} (2y^2 - 2y) + \frac{\lambda + 2\mu}{\lambda + \mu} (2x^2 - 2x) + (4xy - 2x - 2y + 1) - \frac{\partial^2 U}{\partial x \partial y} \right).$$

В расчётах, как и в примере из п. 2, $\lambda = 2$, $\mu = 6$.

На рис. 8 показаны точное решение $V(x, 0.5) = u_2(x, 0.5)$ (сплошная линия) и численное решение для количества итераций $k = 5$ (пунктирная). Число траекторий 10^6 , шаг сетки $h = 0.02$. Средняя относительная погрешность составила 1.7%, время расчёта на одном процессоре 16 мин.

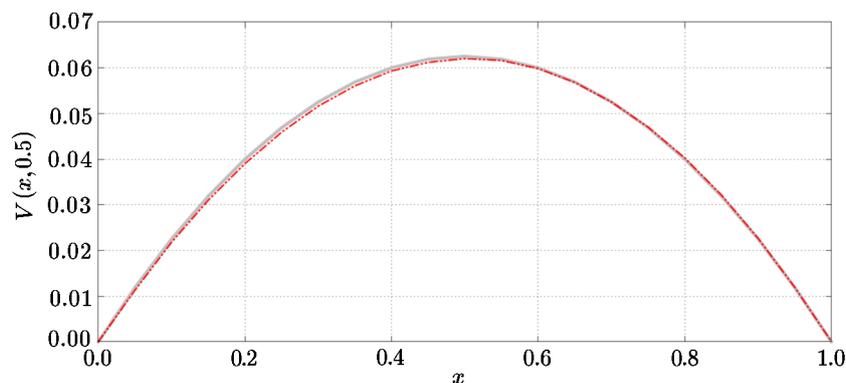


Рис. 8. Решение $V(x, 0.5) = u_2(x, 0.5)$ как функция x при фиксированном $y = 0.5$: сплошная линия — точное решение, пунктирная — численное решение

5. Заключение и обсуждение

В работе построены три стохастических метода для итерационного решения системы уравнений Ламе, описывающей смещения изотропного упругого тела. Первый метод является бессеточным, он основан на решении уравнения анизотропной диффузии с помощью алгоритма случайного блуждания по сферам, в двумерном случае блуждания по окружностям. Этот алгоритм требует минимума памяти, он легко распараллеливается, особенно эффективен для решения краевой задачи с нулевыми граничными условиями и несколько более трудоёмкий при ненулевых граничных условиях. Отметим, что, хотя при вычислении вектора решений сетка не требуется, поскольку используется процесс блуждания по окружностям, для вычисления производных вводится сетка, однако она может быть достаточно редкой и нерегулярной. Второй метод основан на рандомизированном вычислении итераций матриц и решении больших систем линейных алгебраических уравнений. Этот метод требует введения сетки, однако без труда работает с очень мелкими сетками. Он также легко распараллеливается, и в данной работе это продемонстрировано на решении модельной задачи. Из трёх предложенных методов он показал самую высокую эффективность: решение уравнения анизотропной диффузии на

одном шаге итерации с учётом распараллеливания при использовании 48 MPI-процессов и моделировании $N = 10^6$ траекторий заняло всего 0.4 с, решение всей задачи для уравнения Ламе получено за 3.6 с. Однако у первого метода есть следующее преимущество: в алгоритме блуждания по сферам область и её граница могут быть очень сложными, это не сильно влияет на реализацию алгоритма, тогда как для второго метода такая задача представляет существенные трудности, как и для любого разностного метода. Наконец, третий метод основан на введении случайного блуждания по сетке. Этот метод оказался самым трудоёмким из трёх предложенных, однако и у него есть определённые преимущества: хотя этот метод реализован на стандартной равномерной сетке, он допускает и облачную реализацию, когда строится облако хаотически расположенных узлов, и по ним осуществляется случайное блуждание. Как и два первых метода, третий метод также легко и эффективно распараллеливается. Все три метода эффективно масштабируются, поскольку желаемая точность решения просто определяется числом моделируемых траекторий.

Литература

1. **Еремеев В.А.** Решение систем линейных алгебраических уравнений для больших разреженных матриц. Учебно-методическое пособие. — Ростов-на-Дону, 2008.
2. **Ермаков С.М., Михайлов Г.А.** Статистическое моделирование. 2-е изд., дополн. — М.: Наука, 1982.
3. **Кошелёв А.И.** Применение универсального итерационного процесса для некоторых задач механики // Вестник СПбГУ. Сер. 1. — 2008. — Вып. 2. — С. 47–55.
4. **Купрадзе В.Д., Гегелиа Т.Г., Башелейшвили М.О., Бурчуладзе Т.В.** Трёхмерные задачи теории упругости. — М.: Наука, 1976.
5. **Лурье А.И.** Теория упругости. — М.: Наука, 1970.
6. **Полянин А.Д.** Справочник по линейным уравнениям математической физики. — М.: Изд-во физ.-мат. литература, 2001.
7. **Старченко А.В., Берцун В.Н.** Методы параллельных вычислений. Учебник. Серия “Суперкомпьютерное образование”. — Томск: Изд-во Томского ун-та, 2013.
8. **Budaev V.V. and Bogy D.B.** Probabilistic approach to the Lamé equations of linear elastostatics // Intern. J. of Solids and Structures. — 2003. — Vol. 40, iss. 23. — P. 6285–6306. — DOI:10.1016/S0020-7683(03)00364-0.
9. **Dynkin E.V.** The Theory of Markov Processes. — NY: Pergamon Press, 1961.
10. Joint Supercomputer Center of the Russian Academy of Sciences. — <http://www.jscc.ru>.
11. **Muller M.E.** Some continuous Monte Carlo methods for the Dirichlet problem // Ann. Math. Statist. — 1956. — Vol. 27, № 3. — P. 569–589.
12. **O’Leary D.P., Stewart G.W., Vandergraft J.S.** Estimating the largest eigenvalue of a positive definite matrix // Mathematics of Computation. — 1979. — Vol. 33, № 148. — P. 1289–1292.
13. **Roux S.** Generalized Brownian motion and elasticity // J. of Statistical Physics. — 1987. — Vol. 48. — P. 201–213.
14. **Sabelfeld K.K.** A new randomized vector algorithm for iterative solution of large linear systems // Appl. Math. Letters. — 2022. — Vol. 126. — ID-number 107830. — DOI: 10.1016/j.aml.2021.107830.
15. **Sabelfeld K.K.** Monte Carlo Methods in Boundary Value Problems. — Berlin: Springer, 1991.

16. **Sabelfeld K.K.** Random walk on spheres method for solving drift-diffusion problems // Monte Carlo Methods Appl. — 2016. — Vol. 22, iss. 4. — P. 199–223. — DOI:10.1515/mcma-2016-0118.
17. **Sabelfeld K.K.** Random walk on spheres algorithm for solving transient drift-diffusion-reaction problems // Monte Carlo Methods Appl. — 2017. — Vol. 23, iss. 3. — P. 189–212. — DOI:10.1515/mcma-2017-0113.
18. **Sabelfeld K.K. and Kireeva A.** A new Global Random Walk algorithm for calculation of the solution and its derivatives of elliptic equations with constant coefficients in an arbitrary set of points // Appl. Math. Letters. — 2020. — Vol. 107. — ID-number 106466. — <https://doi.org/10.1016/j.aml.2020.106466>.
19. **Sabelfeld K.K. and Kireeva K.** A Global Random Walk on spheres algorithm for calculating the solution and its derivatives of drift-diffusion-reaction equations // Mathematical Methods in the Applied Sciences. — 2022. — Vol. 45, № 3. — P. 1420–1431. — DOI:10.1002/MMA.7861.
20. **Sabelfeld K.K. and Shalimova I.A.** Spherical and Plane Integral Operators for PDEs: Construction, Analysis, and Applications. — Berlin: de Gruyter, 2013. — DOI:10.1515/9783110315332.
21. **Sabelfeld K.K. and Smirnov D.** A global random walk on grid algorithm for second order elliptic equations // Monte Carlo Methods Appl. — 2021. — Vol. 27, iss. 3. — P. 211–225. — DOI:10.1515/mcma-2021-2092.
22. **Sabelfeld K.K. and Talay D.** Integral formulation of the boundary value problems and the method of random walk on spheres // Monte Carlo Methods Appl. — 1995. — Vol. 1, iss. 1. — P. 1–34.
23. **Shalimova I.A. and Sabelfeld K.K.** Development and implementation of branching random walk on spheres algorithms for solving the 2D elastostatics Lamé equation // Monte Carlo Methods Appl. — 2023. — Vol. 29, iss. 1. — P. 79–93.
24. **Stakgold I.** Green's Functions and Boundary Value Problems. — NY: Wiley, 1979.
25. **Walker A.J.** New fast method for generating discrete random numbers with arbitrary frequency distributions // Electr. Lett. — 1974. — Vol. 10. — P. 127–128.

Поступила в редакцию 13 апреля 2023 г.

После исправления 02 июня 2023 г.

Принята к печати 05 сентября 2023 г.

Литература в транслитерации

1. **Eremeev V.A.** Reshenie sistem lineinykh algebraicheskikh uravnenii dlya bol'shikh razrezhennykh matrits. Uchebno-metodicheskoe posobie. — Rostov-na-Donu, 2008.
2. **Ermakov S.M., Mikhailov G.A.** Statisticheskoe modelirovanie. 2-e izd., dopoln. — M.: Nauka, 1982.
3. **Koshelev A.I.** Primenenie universal'nogo iteratsionnogo protsessa dlya nekotorykh zadach mekhaniki // Vestnik SPbGU. Ser. 1. — 2008. — Вып. 2. — С. 47–55.
4. **Kupradze V.D., Gegelia T.G., Bashaileshvili M.O., Burchuladze T.V.** Trekhmernye zadachi teorii uprugosti. — M.: Nauka, 1976.
5. **Lur'e A.I.** Teoriya uprugosti. — M.: Nauka, 1970.
6. **Polyanin A.D.** Spravochnik po lineinym uravneniyam matematicheskoi fiziki. — M.: Izd-vo fiz.-mat. literatura, 2001.
7. **Starchenko A.V., Bertsun V.N.** Metody parallel'nykh vychislenii. Uchebnik. Seriya "Superkomp'yuternoe obrazovanie". — Tomsk: Izd-vo Tomskogo un-ta, 2013.

8. **Budaev B.V. and Bogy D.B.** Probabilistic approach to the Lamé equations of linear elastostatics // Intern. J. of Solids and Structures. — 2003. — Vol. 40, iss. 23. — P. 6285–6306. — DOI:10.1016/S0020-7683(03)00364-0.
9. **Dynkin E.B.** The Theory of Markov Processes. — NY: Pergamon Press, 1961.
10. Joint Supercomputer Center of the Russian Academy of Sciences. — <http://www.jscs.ru>.
11. **Muller M.E.** Some continuous Monte Carlo methods for the Dirichlet problem // Ann. Math. Statist. — 1956. — Vol. 27, № 3. — P. 569–589.
12. **O’Leary D.P., Stewart G.W., Vandergraft J.S.** Estimating the largest eigenvalue of a positive definite matrix // Mathematics of Computation. — 1979. — Vol. 33, № 148. — P. 1289–1292.
13. **Roux S.** Generalized Brownian motion and elasticity // J. of Statistical Physics. — 1987. — Vol. 48. — P. 201–213.
14. **Sabelfeld K.K.** A new randomized vector algorithm for iterative solution of large linear systems // Appl. Math. Letters. — 2022. — Vol. 126. — ID-number 107830. — DOI: 10.1016/j.aml.2021.107830.
15. **Sabelfeld K.K.** Monte Carlo Methods in Boundary Value Problems. — Berlin: Springer, 1991.
16. **Sabelfeld K.K.** Random walk on spheres method for solving drift-diffusion problems // Monte Carlo Methods Appl. — 2016. — Vol. 22, iss. 4. — P. 199–223. — DOI:10.1515/mcma-2016-0118.
17. **Sabelfeld K.K.** Random walk on spheres algorithm for solving transient drift-diffusion-reaction problems // Monte Carlo Methods Appl. — 2017. — Vol. 23, iss. 3. — P. 189–212. — DOI:10.1515/mcma-2017-0113.
18. **Sabelfeld K.K. and Kireeva A.** A new Global Random Walk algorithm for calculation of the solution and its derivatives of elliptic equations with constant coefficients in an arbitrary set of points // Appl. Math. Letters. — 2020. — Vol. 107. — ID-number 106466. — <https://doi.org/10.1016/j.aml.2020.106466>.
19. **Sabelfeld K.K. and Kireeva K.** A Global Random Walk on spheres algorithm for calculating the solution and its derivatives of drift-diffusion-reaction equations // Mathematical Methods in the Applied Sciences. — 2022. — Vol. 45, № 3. — P. 1420–1431. — DOI:10.1002/MMA.7861.
20. **Sabelfeld K.K. and Shalimova I.A.** Spherical and Plane Integral Operators for PDEs: Construction, Analysis, and Applications. — Berlin: de Gruyter, 2013. — DOI:10.1515/9783110315332.
21. **Sabelfeld K.K. and Smirnov D.** A global random walk on grid algorithm for second order elliptic equations // Monte Carlo Methods Appl. — 2021. — Vol. 27, iss. 3. — P. 211–225. — DOI:10.1515/mcma-2021-2092.
22. **Sabelfeld K.K. and Talay D.** Integral formulation of the boundary value problems and the method of random walk on spheres // Monte Carlo Methods Appl. — 1995. — Vol. 1, iss. 1. — P. 1–34.
23. **Shalimova I.A. and Sabelfeld K.K.** Development and implementation of branching random walk on spheres algorithms for solving the 2D elastostatics Lamé equation // Monte Carlo Methods Appl. — 2023. — Vol. 29, iss. 1. — P. 79–93.
24. **Stakgold I.** Green’s Functions and Boundary Value Problems. — NY: Wiley, 1979.
25. **Walker A.J.** New fast method for generating discrete random numbers with arbitrary frequency distributions // Electr. Lett. — 1974. — Vol. 10. — P. 127–128.

