

**Московский Государственный Университет имени
М.В. Ломоносова
Факультет вычислительной математики и
кибернетики
Введение в численные методы
Отчёт по практическому заданию**

Студент Кибизов Кирилл, группа 207

2024

Оглавление

Оглавление	1
1 Постановка задачи	2
2 Описание используемых числовых методов	3
3 Анализ применимости используемых числовых методов	4
4 Результаты	6
Заключение	7
Приложения	8
Литература	10

Постановка задачи

Дано:

1. Уравнение в частных производных с граничными условиями:

$$\begin{cases} k_x \frac{\partial^2 u}{\partial x^2} + k_y \frac{\partial^2 u}{\partial y^2} = 0, & (x, y) \in [0, 1] \times [0, 1], \\ u(x, 0) = 0, & x \in [0, 1] \\ u(0, y) = 0, & y \in [0, 1] \\ u(x, 1) = \sin(\pi x), & x \in [0, 1] \\ u(1, y) = 0, & y \in [0, 1] \end{cases}$$

2. Разностная схема:

$$\begin{cases} k_x \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + k_y \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h^2} = 0, & i = \overline{1, N-1}, j = \overline{1, N-1}, \\ u_{i,0} = 0, & i = \overline{0, N}, \\ u_{0,j} = 0, & j = \overline{0, N}, \\ u_{i,N} = 0, & i = \overline{0, N}, \\ u_{N,j} = 0, & j = \overline{0, N}. \end{cases}$$

где

$$u_{i,j} \approx u(x_i, y_j), \quad x_i = \frac{i}{N}, \quad y_j = \frac{j}{N}, \quad h = \frac{1}{N}.$$

3. Аналитическое решение данной задачи:

$$u(x, y) = \frac{\sinh(\pi y / \sqrt{k_y})}{\sinh(\pi / \sqrt{k_y})} \sin(\pi x)$$

Задача:

Требуется решить данную СЛАУ с помощью итерационного метода Якоби (где он применим) для $N = 100$, рассматривая следующие случаи:

1. $k_x = k_y = 1$,
2. $k_x = 1, k_y = 10^6$.

В случае неприменимости итерационного метода Якоби предложить рабочий альтернативный метод.

Описание используемых числовых методов

Итерационные алгоритмы

При применении итерационных методов решения СЛАУ $Ax = f$ ответ получается в процессе построения последовательных приближений (итераций) $x_k = \{x_1^k, x_2^k, \dots, x_n^k\}$, сходящихся к решению исходной системы в пространстве E_n с евклидовой нормой $\|x\|$: $\lim_{k \rightarrow \infty} x_k = x$, где i - номер компонент, а k - номер итерации.

Сходимость обеспечивает принципиальную возможность получить в процессе итераций ответ с любой наперед заданной степенью точности.

Если очередной член последовательности x_{k+1} может выражаться только через предыдущий $x_{k+1} = F(x_k)$. Такие итерационные алгоритмы называют одношаговыми. Обычно линейно одношаговые алгоритмы записывают в стандартной канонической форме: $B_{k+1} \frac{x_{k+1} - x_k}{\tau_{k+1}} + Ax_k = f$ и $\det B_{k+1} \neq 0$ и $\tau_{k+1} > 0$. В такой записи процесс характеризуется последовательностью матриц B_{k+1} и числовых параметров τ_{k+1} , которые называют итерационными параметрами.

Метод Якоби

Метод Якоби является простым итерационным методом решения системы линейных уравнений, основанным на разностной схеме. В этом методе значения функции в каждой точке сетки обновляются независимо, используя значения с предыдущей итерации. Метод Якоби можно записать в виде:

$$u_{i,j}^{(k+1)} = \frac{k_x(u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)}) + k_y(u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)})}{2(k_x + k_y)},$$

где $u_{i,j}^{(k)}$ — значение функции в точке (i, j) на k -й итерации, а k_x и k_y — коэффициенты.

Метод Якоби обладает сравнительно низкой скоростью сходимости, поскольку обновления в каждой точке сетки не зависят от результатов на предыдущих итерациях. Это может привести к большому количеству итераций для достижения заданной точности. С увеличением числа итераций погрешность метода уменьшается, притом увеличивается вычислительная сложность алгоритма.

Метод верхней релаксации (SOR)

Метод верхней релаксации (SOR) является модификацией метода Гаусса-Зейделя, в котором значения функции обновляются с дополнительной корректировкой с использованием параметра релаксации ω . Этот параметр регулирует скорость сходимости метода. Итерационная схема для метода SOR выглядит следующим образом:

$$u_{i,j}^{(k+1)} = (1 - \omega)u_{i,j}^{(k)} + \omega \left(\frac{k_x(u_{i+1,j}^{(k)} + u_{i-1,j}^{(k+1)}) + k_y(u_{i,j+1}^{(k)} + u_{i,j-1}^{(k+1)})}{2(k_x + k_y)} \right),$$

где $u_{i,j}^{(k)}$ — значение функции в точке (i, j) на k -й итерации, а k_x и k_y — коэффициенты, ω — параметр релаксации, принимающий значения от 1 до 2. Параметр ω регулирует скорость сходимости итераций: при оптимальном выборе ω можно добиться минимального числа итераций для достижения заданной точности.

Метод SOR, в отличие от метода Якоби, сразу использует данные, обновлённые во время текущей итерации. Выбор значения параметра ω оказывает значительное влияние на минимизацию погрешности и скорость сходимости.

Анализ применимости используемых числовых методов

Перед тем как применять итерационные методы для решения системы линейных алгебраических уравнений (СЛАУ), необходимо убедиться, что они сходятся в рассматриваемом случае. Это включает в себя проверку структуры и свойств матрицы системы, а также оценку выполнения достаточных условий сходимости итерационных методов.

Достаточные условия сходимости итерационного процесса

Самосопряжённость матрицы

В одномерном случае в направлении x вторая производная $\frac{\partial^2 u}{\partial x^2}$ аппроксимируется по формуле:

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_x^2}.$$

Важно, что коэффициенты при $u_{i+1,j}$ и $u_{i-1,j}$ одинаковы (оба равны $\frac{1}{h_x^2}$). Аналогично производится аппроксимация второй производной по оси y . Благодаря симметрии разностной схемы все связи между узлами в матрице A получаются парными и "зеркальными". Это говорит о симметричности матрицы. Известные значения на границах области не делают матрицу несимметричной, так как они просто выносятся в вектор правой части f . Таким образом, можно сделать вывод, что матрица A — самосопряжённая. В случае вещественной матрицы (все элементы матрицы A — вещественные) понятия самосопряжённости и симметричности совпадают.

Положительно определённая матрица

Чтобы доказать, что матрица A положительно определённая, нужно показать, что для любого ненулевого вектора v выполняется неравенство: $v^T A v > 0$. Рассмотрим выражение $v^T A v$. Это скаляр, который можно записать как:

$$v^T A v = \sum_{i,j} v_{i,j} (A v)_{i,j}.$$

Для матрицы A , полученной из разностной аппроксимации второго порядка, можно записать, что A действует на вектор v следующим образом:

$$(A v)_{i,j} = \frac{k_x}{h^2} (v_{i+1,j} - 2v_{i,j} + v_{i-1,j}) + \frac{k_y}{h^2} (v_{i,j+1} - 2v_{i,j} + v_{i,j-1}).$$

Подставим это в $v^T A v$ и раскроем сумму:

$$v^T A v = \sum_{i,j} v_{i,j} \left(\frac{k_x}{h^2} (v_{i+1,j} - 2v_{i,j} + v_{i-1,j}) + \frac{k_y}{h^2} (v_{i,j+1} - 2v_{i,j} + v_{i,j-1}) \right).$$

При раскрытии суммы оказывается, что многие члены сокращаются. Получаем:

$$v^T A v = \sum_{i,j} \frac{k_x}{h^2} (v_{i+1,j} - v_{i,j})^2 + \frac{k_y}{h^2} (v_{i,j+1} - v_{i,j})^2.$$

В выражении $v^T A v$ остались только суммы квадратов разностей значений v в соседних узлах. Так как $k_x > 0$, $k_y > 0$ и $h > 0$, каждый член суммы неотрицателен. Так как $v^T A v$ является суммой строго неотрицательных слагаемых, и каждое из них положительно, если $v \neq 0$, то $v^T A v > 0$. Это доказывает, что матрица A положительно определённая.

Теорема Самарского

Пусть A — самосопряжённая положительно определённая матрица: $A = A^T$, $A > 0$, и $B = A - \frac{\tau}{2}A$ — положительно определённая матрица, τ — положительное число: $B = A - \frac{\tau}{2}A > 0$.

Можно утверждать, что для матрицы A , которая:

- симметрична ($A^T = A$),
- положительно определённая ($v^T A v > 0$ для любого $v \neq 0$),

выполняются достаточные условия сходимости итерационных методов, таких как методы Якоби и верхней релаксации (SOR).

Сходимость методов

В итоге методы Якоби и верхней релаксации (SOR) применимы, к данной задаче, однако важно также учитывать:

- **Точность решения:** Точность определяется выбранным критерием остановки (например, достижением малого значения невязки или изменения решения между итерациями).
- **Быстрота сходимости:** Для улучшения быстроты сходимости можно:
 - уменьшить шаг h ,
 - использовать "ускоряющие" параметры, такие как ω в методе верхней релаксации,

Результаты

В данном разделе представлены результаты численного решения уравнения с использованием метода Якоби для одного из тестов.

Для каждой точки вычислены значения, полученные численно и аналитически, с целью сравнения точности решения.

Пример вывода для первого теста представлен ниже.

1ый столбец - координаты точек,

2ой столбец - численное решение,

3ий столбец - аналитическое решение,

```
1 ./a.out
2 Please, input amount of tests (max 10): 1
3 Leave 3rd argument as 0 (for Jacobi) or as w (w = 1 for Gauss-Seidel; 1 < w < 2 for SOR)
4 Input kx and ky and w; for test #1: 1 1 0
5 ...
6 u(0.250000, 0.500000) | 0.139489 | 0.140904
7 u(0.250000, 0.750000) | 0.319105 | 0.320099
8 u(0.250000, 1.000000) | 0.707107 | 0.707107
9 u(0.500000, 0.000000) | 0.000000 | 0.000000
10 ...
11 u(0.500000, 0.750000) | 0.451283 | 0.452688
12 u(0.500000, 1.000000) | 1.000000 | 1.000000
13 u(0.750000, 0.000000) | 0.000000 | 0.000000
14 u(0.750000, 0.250000) | 0.052184 | 0.053187
15 ...
16 -----
17 Test #1:
18 Iterations = 10247
```

Вывод метода Якоби для 1го теста (для некоторых точек, по координатам кратных 0.25)

Для других тестов результаты можно получить вручную, используя программу, с различными значениями параметра ω , чтобы исследовать его влияние на сходимость метода.

Кроме того, для метода SOR был найден оптимальный параметр $\omega = 1.94$, который обеспечил наилучшую сходимость.

Заключение

Критерий	Метод Якоби	Метод SOR
Реализация	<ul style="list-style-type: none">- Два массива, поэтапное обновление.- В каждой итерации элементы независимы.- Без вспомогательных параметров.	<ul style="list-style-type: none">- Один массив, обновление "на ходу".- В каждой итерации элементы зависимы.- Требуется параметр ω ($0 < \omega \leq 2$), влияющий на сходимость.
Сходимость	Медленнее	Существенно быстрее
Погрешность	Больше	Существенно меньше

Таблица 4.1: Сравнение метода Якоби и метода SOR

В данном отчёте была рассмотрена задача решения уравнения в частных производных с помощью итерационных методов, таких как метод Якоби и метод верхней релаксации. Реализация методов была выполнена на языке программирования C. Были получены результаты со сравнимо высокой точностью относительно предложенного аналитического решения.

Приложения

<https://github.com/kibizoffs/jacobi>

Программа `main.c` решает задачу с использованием методов Якоби и SOR. Изображения графиков, полученные в процессе вычислений, сохраняются в директории `media`.

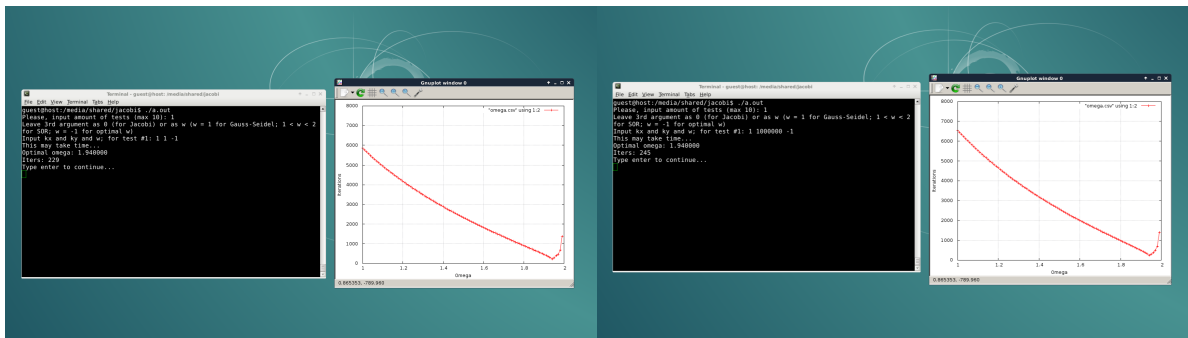


Рис. 4.1: Best ω ($kx = 1, ky = 1$), $\omega = 1.94$

Рис. 4.2: Best ω ($kx = 1, ky = 10^6$), $\omega = 1.94$

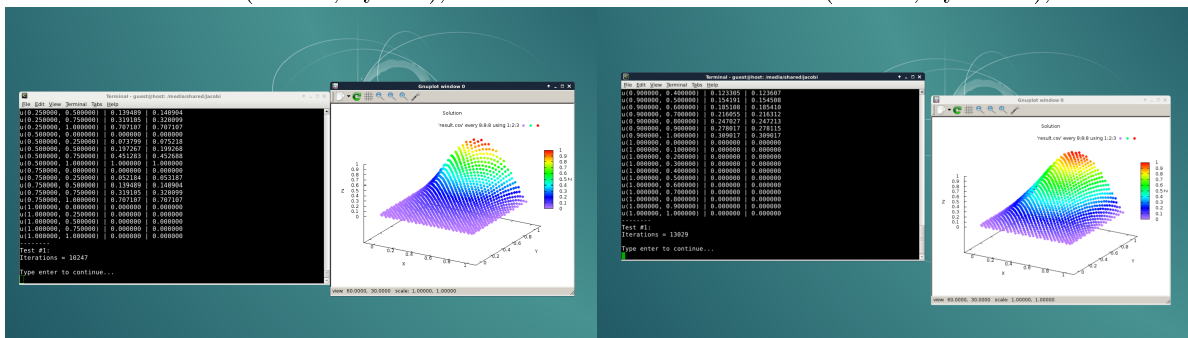


Рис. 4.3: Jacobi ($kx = 1, ky = 1$), $\text{iters} = 10247$

Рис. 4.4: Jacobi ($kx = 1, ky = 10^6$), $\text{iters} = 13029$

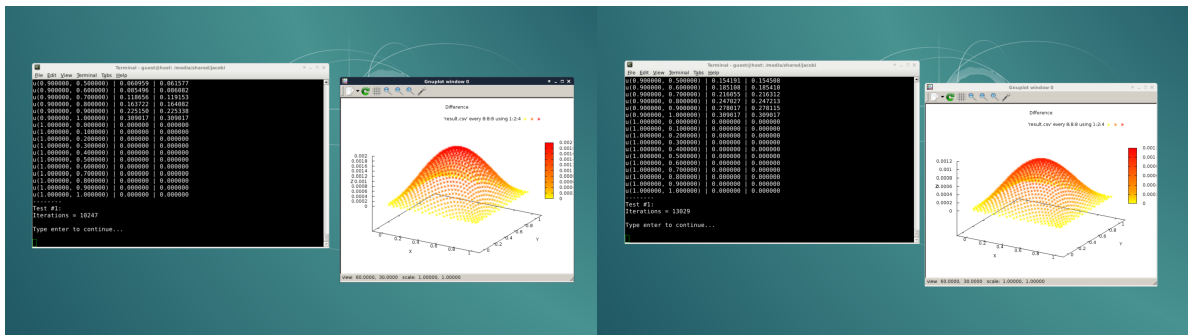


Рис. 4.5: Dif Jacobi ($kx = 1, ky = 1$)

Рис. 4.6: Dif Jacobi ($kx = 1, ky = 10^6$)

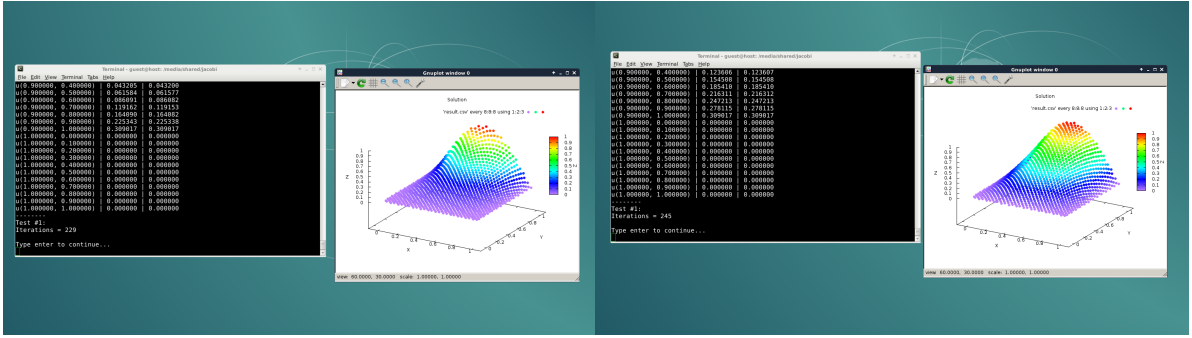


Рис. 4.7: Best res ($kx = 1$, $ky = 1$), iters = 229 Рис. 4.8: Best res ($kx = 1$, $ky = 10^6$), iters = 245

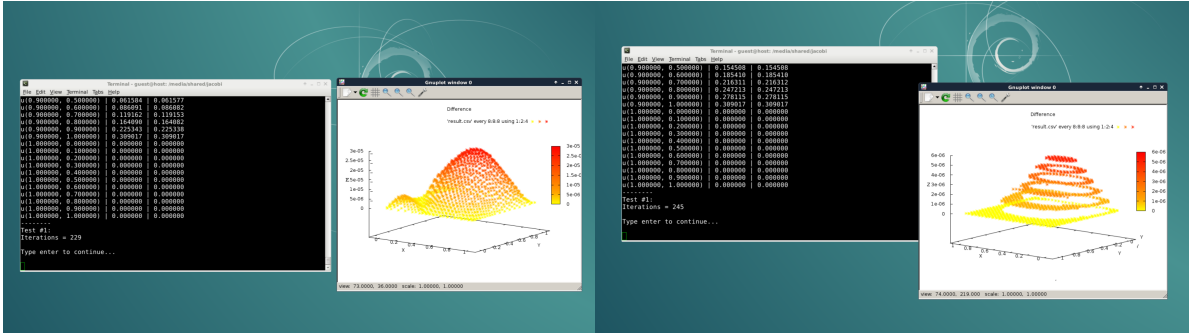


Рис. 4.9: Best dif ($kx = 1$, $ky = 1$) Рис. 4.10: Best dif ($kx = 1$, $ky = 10^6$)

Литература

- [1] Костомаров Д. П., Фаворский А. П. *Вводные лекции по численным методам*. — М.: Логос, 2004. — 184 с.
- [2] Самарский А. А. *Введение в численные методы*. — М.: Наука, 1989. — 416 с.