Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра системного анализа

Отчет по практикуму по суперкомпьютерному моделированию

Вариант 4

Студент 6 курса, 615 группы:
А. В. Горбачёв

Преподаватель:
Н. Н. Попова

Москва, 2024

Содержание

1 Постановка задачи

Область $D \subset \mathbb{R}^2$ - трапеции с вершинами в точках $A(0,0), B(3,0), C(2,3), D(0,3)$, ограниченной контуром γ, рассматривается дифференциальное уравнение Пуассона:

$$-\Delta u = f(x,y), \tag{1.1}$$

где оператор Лапласса имеет вид:

$$-\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}.$$

Функция $f(x,y)$ считается известной. Для выделения единственного решения уравнение дополняется граничным условием Дирихле (см. [1]):

$$u(x,y) = 0, \quad (x,y) \in \gamma \tag{1.2}$$

Требуется найти функцию $u(x,y)$, удовлетворяющую уравнению (1.1) в области D и краевому условию (1.2) на её границе.

2 Численный метод решения

Для приближенного решения задачи (1.1), (1.2) предлагается воспользоваться методом фиктивных областей [2].

Пусть область D принадлежит прямоугольнику
$\Pi = \{(x,y) | A_1 < x < B_1, A_2 < y < B_2\}$ (в нашем случае $A_1 = 0, B_1 = 3, A_2 = 0, B_2 = 3$). Обозначим через $\bar{D}$ и $\bar{\Pi}$ замыкания области D и прямоугольника Π соответственно, через Γ - границу прямоугольника Π. Разность множеств

$$\hat{D} = \Pi \setminus \bar{D}$$

называется *фиктивной областью.*

Выберем и зафиксируем малое $\varepsilon > 0$. В прямоугольнике Π рассматривается задача

Дирихле:

$$
\begin{aligned}
-\frac{\partial}{\partial x}\left(k(x,y)\frac{\partial v}{\partial x}\right)-\frac{\partial}{\partial y}\left(k(x,y)\frac{\partial v}{\partial y}\right)&=F(x,y)\\
v(x,y)=0,\quad (x,y)&\in\Gamma
\end{aligned}
\tag{2.1}
$$

с кусочно-постоянным коэффициентом

$$
k(x,y)=\begin{cases}1, & (x,y)\in D\\ \dfrac{1}{\varepsilon}, & (x,y)\in\hat{D}\end{cases}
\tag{2.2}
$$

и правую часть

$$
F(x,y)=\begin{cases}f(x,y), & (x,y)\in D\\ 0, & (x,y)\in\hat{D}\end{cases}
\tag{2.3}
$$

Требуется найти непрерывную в прямоугольнике Π функцию $v(x,y)$, удовлетворяющую дифференциальному уравнению задачи (2.1) всюду в $\Pi\setminus\gamma$, равную нулю на границе Γ прямоугольника, и такую, чтобы вектор потока

$$
W(x,y)=-k(x,y)\left(\frac{\partial v}{\partial x},\frac{\partial v}{\partial y}\right)
$$

имел непрерывную нормальную компоненту на общей части криволинейной границы области D и прямоугольника Π. Последнее означает, что в каждой точке $(x_0,y_0)\in\gamma\cap\Pi$ должно выполняться равенство

$$
\lim_{\substack{(x,y)\to(x_0,y_0)\\(x,y)\in D}}\left(W(x,y),n(x_0,y_0)\right)=\lim_{\substack{(x,y)\to(x_0,y_0)\\(x,y)\in\hat{D}}}\left(W(x,y),n(x_0,y_0)\right),
\tag{2.4}
$$

где $n(x,y)$ - вектор единичной нормали к границе γ в точке (x,y), определенный всюду или почти всюду на кривой.

Известно [2], что функция $v(x,y)$ равномерно приближает решение $u(x,y)$ задачи (1.1), (1.2) в области D, а именно,

$$
\max_{P\in\bar{D}}|v(x,y)-u(x,y)|<C\varepsilon,\;\; C>0
\tag{2.5}
$$

В частности, $|v(x,y)|<C\varepsilon$ во всех точках кривой γ. Этот результат позволяет получить искомую функцию $u(x,y)$ с любой наперед заданной точностью $\varepsilon>0$, решая задачу (2.1), (2.4), вместо задачи (1.1), (1.2). Тем самым, задача Дирихле в криволинейной области приближенно заменяется задачей Дирихле в прямоугольнике с кусочно-постоянным коэффициентом $k(x, y)$.

3 Разностная схема решения задачи

Краевую задачу (2.1), (2.4) предлагается решать численно методом конечных разностей [3]. В замыкании прямоугольника П определяется равномерная прямоугольная сетка $\bar{\omega_h} = \bar{\omega_1} \times \bar{\omega_2}$, где

$$\bar{\omega_1} = \{x_i = A_1 + ih_1, i = 0, 1, \ldots, M\}\,, \;\; \bar{\omega_2} = \{y_i = A_2 + jh_2, j = 0, 1, \ldots, N\}$$

Здесь

$$h_1 = \frac{B_1 - A_1}{M}, \;\; h_2 = \frac{B_2 - A_2}{N}$$

Через ω_h обозначим множество внутренних узлов сетки $\bar{\omega_h}$, т. е. множество узлов сетки прямоугольника, не лежащих на границе Г.

Рассмотрим линейное пространство H функций, заданных на сетке ω_h. Обозначим через ω_{ij} значение сеточной функции $\omega \in H$ в узле сетки $(x_i, y_j) \in \omega_h$. Будем считать, что в пространстве H задано скалярное произведение и евклидова норма

$$(u, v) = \sum_{i=1}^{M-1} \sum_{j=1}^{N-1} h_1 h_2 u_{ij} v_{ij}, \;\; ||u||_E = \sqrt{(u, u)} \tag{3.1}$$

В методе конечных разностей дифференциальная задача математической физики заменяется конечно-разностной операторной задачей вида

$$A\omega = B, \tag{3.2}$$

где $A : H \to H$ - оператор, действующий в пространстве сеточных функций, $B \in H$ - известная правая часть. Задача (3.2) называется *разностной схемой*. Решение этой задачи считается численным решением исходной дифференциальной задачи.

При построении разностной схемы следует аппроксимировать (приближенно заменить) все уравнения краевой задачи их разностными аналогами - сеточными уравнениями, связывающими значения искомой сеточной функции в узлах сетки. Полученные таким образом уравнения должны быть функционально независимыми, а их общее количество - совпадать с числом неизвестных, т.е. с количеством узлов сетки.

Дифференциальное уравнение задачи (2.1) во внутренних точках сетки аппроксимируется разностным уравнением

$$-\frac{1}{h_1}\left(a_{i+1j}\frac{\omega_{i+1j}-\omega_{ij}}{h_1}-a_{ij}\frac{\omega_{ij}-\omega_{i-1j}}{h1}\right)$$
$$-\frac{1}{h_2}\left(b_{ij+1}\frac{\omega_{ij+1}-\omega_{ij}}{h_2}-b_{ij}\frac{\omega_{ij}-\omega_{ij-1}}{h2}\right)=F_{ij},\ \ i=\overline{1,M-1},\ \ j=\overline{1,N-1},\qquad(3.3)$$

в котором коэффициенты

$$a_{ij}=\frac{1}{h_2}\int\limits_{y_{j-1/2}}^{y_{j+1/2}}k(x_{i-1/2},t)dt,\ \ b_{ij}=\frac{1}{h_1}\int\limits_{x_{i-1/2}}^{x_{i+1/2}}k(t,y_{j-1/2})dt\qquad(3.4)$$

при всех $i=\overline{1,M}$, $j=\overline{1,N}$. Здесь полуцелые узлы

$$x_{i\pm1/2}=x_i\pm0,5h_1,\ \ y_{j\pm1/2}=y_j\pm0,5h_2.$$

Правая часть разностного уравнения

$$F_{ij}=\frac{1}{h_1h_2}\iint\limits_{\Pi_{ij}}F(x,y)dxdy,$$
$$\Pi_{ij}=\left\{(x,y):x_{i-1/2}\leqslant x\leqslant x_{i+1/2},y_{j-1/2}\leqslant y\leqslant y_{j+1/2}\right\}\qquad(3.5)$$

при всех $i=\overline{1,M-1}$, $j=\overline{1,N-1}$.

Введем обозначения правой и левой разностных производных по переменным x,y соответственно:

$$\omega_{x,ij}=\frac{\omega_{i+1j}-\omega_{ij}}{h_1},\omega_{\bar{x},ij}=\omega_{x,i-1j}=\frac{\omega_{ij}-\omega_{i-1j}}{h_1},$$
$$\omega_{y,ij}=\frac{\omega_{ij+1}-\omega_{ij}}{h_2},\omega_{\bar{y},ij}=\omega_{y,ij-1}=\frac{\omega_{ij}-\omega_{ij-1}}{h_2}.$$

С учетом принятых обозначений разностное уравнение (3.3) можно представить в более компактном и удобном виде:

$$-\left(a\omega_{\bar{x}}\right)_{x,ij}-\left(b\omega_{\bar{y}}\right)_{y,ij}=F_{ij},\ \ i=\overline{1,M-1},\ j=\overline{1,N-1} \tag{3.6}$$

Краевые условия Дирихле задачи (2.1), (2.4) апроксимируется точным равенством

$$\omega_{ij}=\omega(x_i,y_j)=0,\ \ (x_i,y_j)\in\Gamma \tag{3.7}$$

Переменные заданные ω_{ij}, заданные равенством (3.7), исключаются из системы уравнений (3.6). В результате остаются неизвестными значения ω_{ij} при $i=\overline{1,M-1}$, $j=\overline{1,N-1}$ и их количество совпадает с числом уравнений. Система является линейной относительно неизвестных величин и может быть представлена в виде (3.2) с самосопряженным и положительно определенным оператором

$$A\omega=-\left(a\omega_{\bar{x}}\right)_x-\left(b\omega_{\bar{y}}\right)_y$$

и правой частью F, определенной равенством (3.5). Таким образом, построенная разностная схема (3.6), (3.7) линейна и имеет единственное решение при любой правой части (см. [5]).

Замечание. Интегралы (3.4) от кусочно-постоянной функции $k(x,y)$ следует вычислять аналитически. Нетрудно видеть, что если отрезок, соединяющий точки $P_{ij}=\left\{x_{i-1/2},y_{j-1/2}\right\}$ и $P_{ij+1}=\left\{x_{i-1/2},y_{j+1/2}\right\}$, целиком расположен в области D, то $a_{ij}=1$. Если же указанный отрезок находится в фиктивной области $\hat{D}$, то $a_{ij}=1/\varepsilon$. В противном случае

$$a_{ij}=h_2^{-1}l_{ij}+(1-h_2^{-1}l_{ij})/\varepsilon,$$

где l_{ij} - длина той части отрезка $[P_{ij},P_{ij+1}]$, которая принадлежит области D. Аналогичным образом вычисляются коэффициенты b_{ij}.

Очевидно, правая часть схемы F_{ij} равна нулю при всех $(i,j):\Pi_{ij}\subset\hat{D}$. Если $\Pi_{ij}\subset D$, то правую часть предлагается приближенно заменить значением $f(x_i,y_j)$. В противном случае, когда прямоугольник Π_{ij} содержит точки оригинальной области D и фиктивной области $\hat{D}$, величина F_{ij} может быть вычислена приближенно как приближенно как произведение

$$(h_1h_2)^{-1}S_{ij}f(x_i^*,y_j^*),$$

где (x_i^*, y_j^*) - любая точка пересечения $\Pi_{ij} \cap D$, $S_{ij} = \mathrm{mes}\,(\Pi_{ij} \cap D)$ - площадь пересечения областей, при вычислении которой криволинейную часть границы можно заменить отрезком прямой.

4 Метод решения системы линейных алгебраических уравнений

Приближенное решение разностной схемы (3.6), (3.7) может быть получено итерационным методом скорейшего спуска [4]. Этот метод позволяет получить последовательность сеточных функций $\omega^{(k)} \in H$, $k = 1, 2, \ldots$, сходящуюся по норме пространства H к решению разностной схемы, т.е.

$$||\omega - \omega^{(k)}||_E \to 0, \;\; k \to +\infty.$$

Начальное приближение $\omega^{(0)}$ можно выбрать любым способом, например, равным нулю во всех точках расчетной сетки.

Метод является одношаговым. Итерация $\omega^{(k+1)}$ вычисляется по итерации $\omega^{(k)}$ согласно равенствам:

$$\omega_{ij}^{(k+1)} = \omega_{ij}^{(k)} - \tau_{k+1} r_{ij}^{(k)}, \tag{4.1}$$

где невязка $r^{(k)} = A\omega^{(k)} - B$, итерационный параметр

$$\tau_{k+1} = \frac{\left(r^{(k)}, r^{(k)}\right)}{Ar^{(k)}, r^{(k)}}.$$

В качестве условия остановки итерационного процесса следует использовать неравенство

$$||\omega^{(k+1)} - \omega^{(k)}||_E < \delta,$$

где δ - положительное число, определяющее точность итерационного метода. Оценку точности приближенного решения можно проводить в других нормах пространства сеточных функций, например, в максимум норме

$$||\omega||_C = \max_{x \in \bar{\omega_h}} |\omega(x)|\,. \tag{4.2}$$

Константу δ для данной задачи предлагается выбрать так, чтобы итерационный процесс укладывался в отведенное для него время.

5 Программная реализация

5.1 Последовательная программа

В программе реализован алгоритм, описанный выше.

5.2 OpenMP

Для эффективного применения OpenMP мы добавили соответствующие директивы (#pragma omp parallel for reduction, #pragma omp parallel for) при подсчете скалярного произведения, вычитания матриц, домножения матрицы на число и применения линейного оператора A к матрице.

5.3 MPI

Для выполнения вычислений на системе с распределенной памятью мы применяем MPI. В зависимости от количества доступных процессов, область разбивается на несколько неперекрывающихся прямоугольников, охватывающих всю исходную область. Каждый из этих прямоугольников назначается отдельному процессу, и для работы с матрицами меньшего размера выполняются операции, аналогичные тем, что используются в последовательном коде. Основная сложность заключается в управлении граничными узлами этих миниппрямоугольников, которые могут находиться на других процессах. Поэтому, перед применением линейного оператора, необходимо обеспечить доступ ко всем "граничным"значениям, это делаем с помощью пересылок. Скалярное произведение также требуют взаимодействия между процессами, это обеспечивается с помощью операции редукции суммы.

5.4 MPI+OpenMP

OpenMP применялся на каждом MPI процессе, аналогично предпоследнему пункту.

6 Результаты работы программы

Численные эксперименты проводились на вычислительном комплексе Polus. Для расчётов значение δ фиксировалось равным 1e-6 для сетки 160×180, 1e-7 - для сетки 90×80.

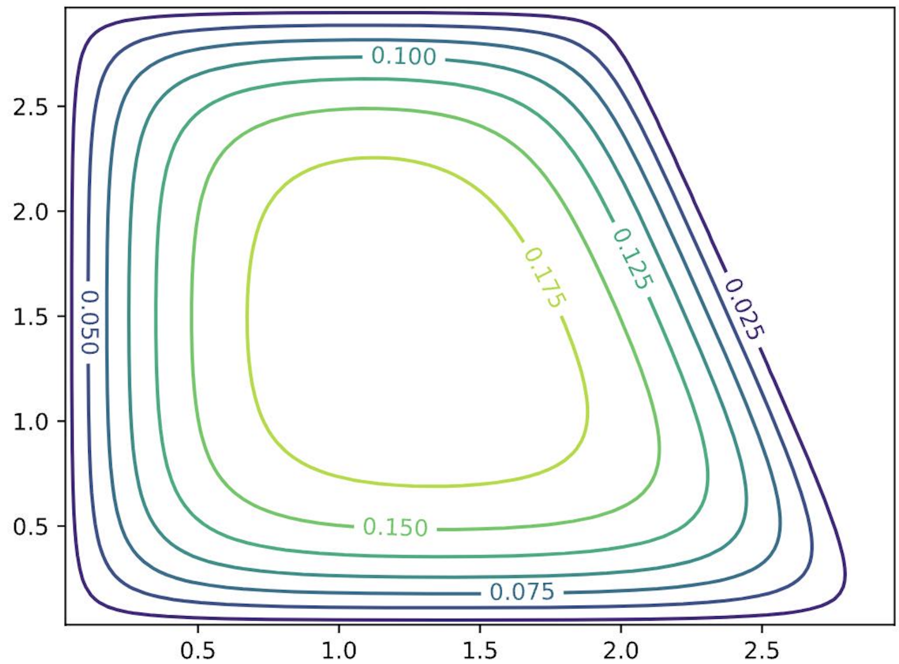

Рис. 1: Визуализация полученного решения

Количество OpenMP-нитей	Число точек сетки (M $\times N$)	Число итераций	Время решения (сек)	Ускорение
1	90×80	10163	54.4446	-
2	90×80	10163	51.0589	1.066
4	90×80	10163	15.4232	3.53
8	90×80	10163	12.1214	4.49
16	90×80	10163	6.51317	8.359
32	90×80	10163	7.16019	7.60379
1	160×180	2449	40.2252	-
2	160×180	2449	25.1355	1.6
4	160×180	2449	10.0136	4.017
8	160×180	2449	10.555	3.811
16	160×180	2449	3.76451	10.685
32	160×180	2449	3.73268	10.776

Таблица 1: Расчеты на Polus (OpenMP)

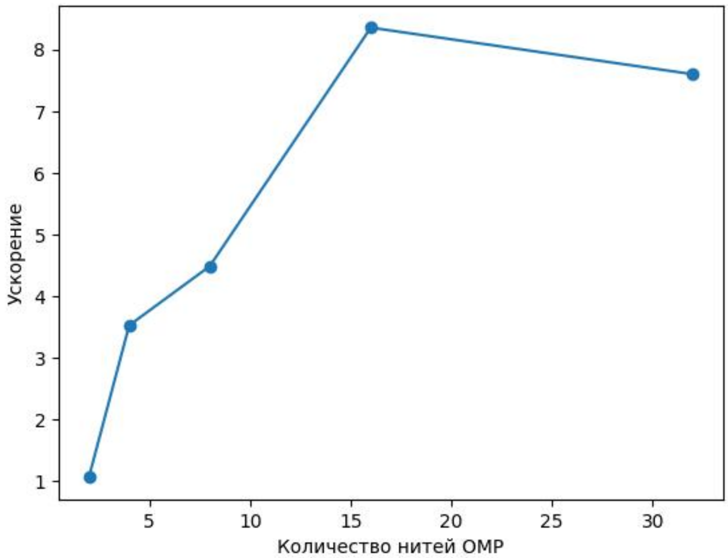

Рис. 2: Ускорение на сетке 80×90 для OMP

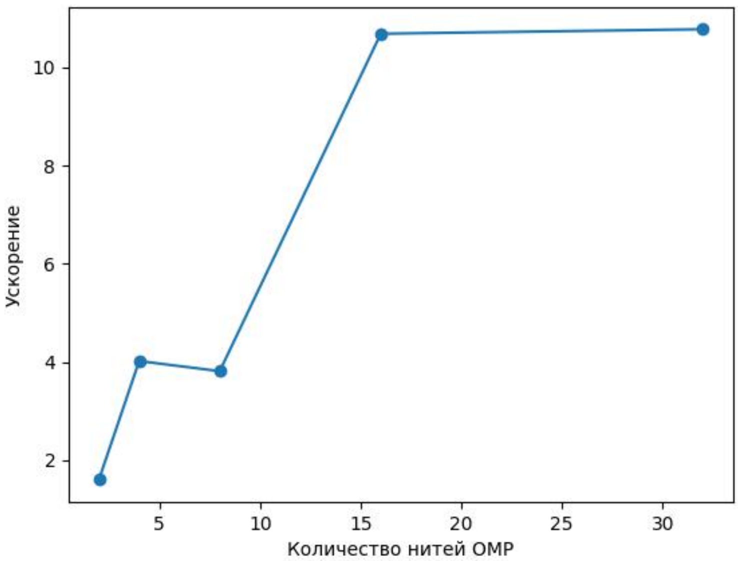

Рис. 3: Ускорение на сетке 160×180 для OMP

Количество процессов MPI	Количество OpenMP-нитей в процессе	Число точек сетки (M $\times N$)	Число итераций	Время решения (сек)	Ускорение
2	1	80 $\times$90	10163	23.6699	1.0
2	2	80 $\times$90	10163	11.7185	2.0199
2	4	80 $\times$90	10163	11.1894	2.11539
2	8	80 $\times$90	10163	13.961	1.3178
4	1	160 $\times$180	2449	12.29	1.0
4	2	160 $\times$180	2449	7.617	1.6135
4	4	160 $\times$180	2449	5.8817	2.0895
4	8	160 $\times$180	2449	5.979	2.05553

Таблица 2: Таблица с результатами расчетов на Polus (MPI+OpenMP код)

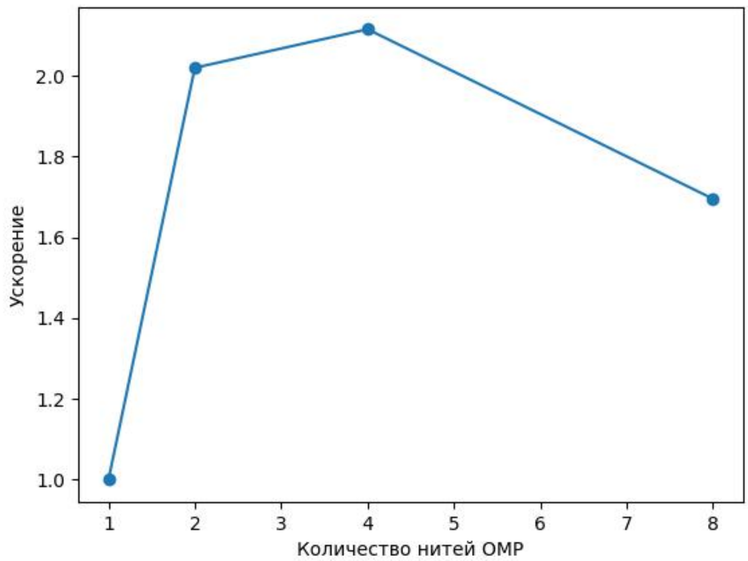

Рис. 4: Ускорение на сетке 800×90 для 2х MPI-процессов

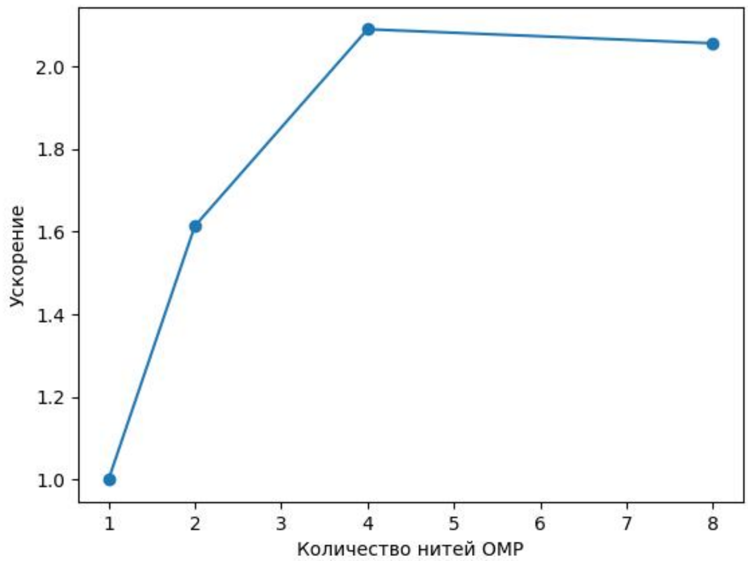

Рис. 5: Ускорение на сетке 160×180 для 4х MPI-процессов

Список литературы

[1] Тихонов А. Н., Самарский А. А. *Уравнение математической физики.* М.: Наука, 1977.

[2] Марчук Г. И. *Методы вычислительной математики.* М.: Наука, 1989.

[3] Самарский А. А. *Теория разностных схем.* М.: Наука, 1989.

[4] Самарский А. А., Гулин А. В. *Численные методы.* М.: Наука, 1989.

[5] Ильин В. А., Ким Г. Д. *Линейная алгебра и аналитическая геометрия* М.: Издательство Московского университета, 2002.

[6] IBM Polus - http://hpc.cmc.msu.ru