

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

Отчет
По курсу: «Суперкомпьютерное моделирование и технологии»
«Итоговый отчет»

Выполнила:
Казанцева Варвара Денисовна
Группа: 608

12 декабря 2025 г.

Содержание

1 Введение	2
2 Математическая постановка задачи	2
3 Метод фиктивных областей	2
4 Разностная схема решения задачи	3
4.1 Определение коэффициентов разностной схемы	3
4.2 Определение коэффициентов разностной схемы аналитически	3
5 Метод решения системы линейных алгебраических уравнений	4
6 Программная реализация	4
7 Программная реализация	4
7.1 Последовательная программа	4
7.2 OpenMP-программа	5
7.3 MPI-программа	5
7.4 Гибридная программа MPI+OpenMP	5
8 Результаты расчетов	6
8.1 Графики	7
8.2 График приближенного решения	7
8.3 Графики ускорений	7
9 Заключение	10

1 Введение

В данной работе рассматривается приближенное решение двумерной задачи Дирихле для уравнения Пуассона в криволинейной области. В качестве расчетной области выбран **остроугольный треугольник с вершинами в точках С(3,0), А(3,0), В(0,4)** (вариант 2).

Целью работы является разработка и исследование эффективности различных реализаций программы для решения поставленной задачи:

1. Последовательная программа
2. Параллельная версия с использованием OpenMP
3. Параллельная версия с использованием MPI
4. Гибридная реализация MPI+OpenMP

2 Математическая постановка задачи

В области $D \subset \mathbb{R}^2$, ограниченной кусочно-гладким контуром Γ , рассматривается дифференциальное уравнение Пуассона:

$$-\Delta u = f(x, y), \quad (x, y) \in D \quad (1)$$

где оператор Лапласа:

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \quad (2)$$

функция $f(x, y) = 1$ задана. Для выделения единственного решения уравнение дополняется граничным условием Дирихле:

$$u(x, y) = 0, \quad (x, y) \in \Gamma \quad (3)$$

3 Метод фиктивных областей

Для приближенного решения задачи (1)-(3) используется метод фиктивных областей. Область D принадлежит прямоугольнику $\Pi = \{(x, y) : A_1 < x < B_1, A_2 < y < B_2\}$. В прямоугольнике Π рассматривается задача Дирихле с кусочно-постоянным коэффициентом:

$$k(x, y) = \begin{cases} 1, & (x, y) \in D \\ \frac{1}{\varepsilon}, & (x, y) \in \hat{D} \end{cases} \quad (4)$$

и правой частью:

$$F(x, y) = \begin{cases} f(x, y), & (x, y) \in D \\ 0, & (x, y) \in \hat{D} \end{cases} \quad (5)$$

где $\hat{D} = \Pi \setminus D$ - фиктивная область, $\varepsilon > 0$ - малый параметр.

4 Разностная схема решения задачи

Краевая задача решается численно методом конечных разностей. В замыкании прямоугольника Π определяется равномерная прямоугольная сетка $\bar{\omega}_h = \bar{\omega}_1 \times \bar{\omega}_2$, где:

$$\begin{aligned}\bar{\omega}_1 &= \{x_i = A_1 + ih_1, i = 0, 1, \dots, M\}, \\ \bar{\omega}_2 &= \{y_j = A_2 + jh_2, j = 0, 1, \dots, N\}\end{aligned}$$

с шагами $h_1 = (B_1 - A_1)/M$, $h_2 = (B_2 - A_2)/N$.

Дифференциальное уравнение аппроксимируется разностным уравнением:

$$-\frac{1}{h_1}(a_{i+1,j}w_{x,ij} - a_{i,j}w_{\bar{x},ij}) - \frac{1}{h_2}(b_{i,j+1}w_{y,ij} - b_{i,j}w_{\bar{y},ij}) = F_{ij} \quad (6)$$

где коэффициенты a_{ij} , b_{ij} вычисляются через интегралы от функции $k(x, y)$.

4.1 Определение коэффициентов разностной схемы

Коэффициенты a_{ij} вычисляются по формуле:

$$a_{ij} = \frac{1}{h_2} \int_{y_{j-1/2}}^{y_{j+1/2}} k(x_{i-1/2}, t) dt \quad (7)$$

где $x_{i\pm 1/2} = x_i \pm 0.5h_1$, $y_{j\pm 1/2} = y_j \pm 0.5h_2$.

Коэффициенты b_{ij} вычисляются по формуле:

$$b_{ij} = \frac{1}{h_1} \int_{x_{i-1/2}}^{x_{i+1/2}} k(t, y_{j-1/2}) dt \quad (8)$$

Правая часть F_{ij} вычисляется как:

$$F_{ij} = \frac{1}{h_1 h_2} \iint_{\Pi_{ij}} F(x, y) dx dy \quad (9)$$

где $\Pi_{ij} = \{(x, y) : x_{i-1/2} \leq x \leq x_{i+1/2}, y_{j-1/2} \leq y \leq y_{j+1/2}\}$.

4.2 Определение коэффициентов разностной схемы аналитически

Для коэффициента a_{ij} :

$$a_{ij} = \begin{cases} 1, & \text{если отрезок } [P_{ij}, P_{ij+1}] \subset D \\ \frac{1}{\varepsilon}, & \text{если отрезок } [P_{ij}, P_{ij+1}] \subset \hat{D} \\ \frac{l_{ij}}{h_2} + \frac{1}{\varepsilon}(1 - \frac{l_{ij}}{h_2}), & \text{иначе} \end{cases} \quad (10)$$

где l_{ij} - длина части отрезка $[P_{ij}, P_{ij+1}]$, принадлежащей области D , $P_{ij} = (x_{i-1/2}, y_{j-1/2})$.

Аналогично для коэффициента b_{ij} :

$$b_{ij} = \begin{cases} 1, & \text{если отрезок } [P_{ij}, P_{i+1j}] \subset D \\ \frac{1}{\varepsilon}, & \text{если отрезок } [P_{ij}, P_{i+1j}] \subset \hat{D} \\ \frac{l_{ij}}{h_1} + \frac{1}{\varepsilon}(1 - \frac{l_{ij}}{h_1}), & \text{иначе} \end{cases} \quad (11)$$

где l_{ij} - длина части отрезка $[P_{ij}, P_{i+1j}]$, принадлежащей области D .

Для правой части F_{ij} :

$$F_{ij} = \begin{cases} 0, & \text{если } \Pi_{ij} \subset \hat{D} \\ 1, & \text{если } \Pi_{ij} \subset D \\ \frac{S_{ij}}{h_1 h_2}, & \text{иначе} \end{cases} \quad (12)$$

где S_{ij} - площадь пересечения $\Pi_{ij} \cap D$.

$\varepsilon = h^2$ - малый параметр, $h = \max(h_1, h_2)$.

5 Метод решения системы линейных алгебраических уравнений

Приближенное решение разностной схемы находится итерационным методом сопряженных градиентов с диагональным предобуславливанием. Алгоритм метода:

1. Выбирается начальное приближение $w^{(0)}$
2. Вычисляется невязка $r^{(0)} = B - Aw^{(0)}$
3. Решается система $Dz^{(0)} = r^{(0)}$
4. Задается направление $p^{(1)} = z^{(0)}$
5. Для $k = 0, 1, 2, \dots$ до сходимости:

$$\begin{aligned}\alpha_{k+1} &= \frac{(z^{(k)}, r^{(k)})}{(Ap^{(k+1)}, p^{(k+1)})} \\ w^{(k+1)} &= w^{(k)} + \alpha_{k+1}p^{(k+1)} \\ r^{(k+1)} &= r^{(k)} - \alpha_{k+1}Ap^{(k+1)} \\ Dz^{(k+1)} &= r^{(k+1)} \\ \beta_{k+1} &= \frac{(z^{(k+1)}, r^{(k+1)})}{(z^{(k)}, r^{(k)})} \\ p^{(k+2)} &= z^{(k+1)} + \beta_{k+1}p^{(k+1)}\end{aligned}$$

Условие остановки итерационного процесса:

$$\|w^{(k+1)} - w^{(k)}\|_E < \varepsilon \quad (13)$$

6 Программная реализация

В работе выполнена поэтапная разработка программы для решения задачи Дирихле для уравнения Пуассона в треугольной области методом фиктивных областей. Были созданы четыре версии программы: последовательная, OpenMP, MPI и гибридная MPI+OpenMP.

7 Программная реализация

7.1 Последовательная программа

Базовой основой для всех параллельных версий является последовательная программа, реализующая:

- `get_x_inter()`, `get_y_inter()` – вычисление пересечений с прямыми, ограничивающими треугольник
- `is_right()`, `is_left()` – проверка положения точки относительно сторон треугольника
- `compute_a()`, `compute_b()` – вычисление коэффициентов для разностной схемы

- `compute_F()` – вычисление правой части с учетом площади пересечения ячеек с областью
- `applyOperator()` – применение пятиточечного разностного оператора
- `conjugateGradient()` – реализация метода сопряженных градиентов

Эти функции реализуют метод фиктивных областей и итерационный алгоритм решения без распараллеливания.

7.2 OpenMP-программа

Для распараллеливания на данном этапе в последовательную программу добавлены директивы OpenMP:

- `#pragma omp parallel for collapse(2)` – для параллелизации вложенных циклов
- `#pragma omp parallel for reduction(+:result)` – для параллельного вычисления скалярных произведений в `dot()` и норма в `normDifference()`
- `omp_set_num_threads()` – установка числа потоков

7.3 MPI-программа

Для распределения задачи между MPI-процессами используются:

- Структура `Domain` – хранит границы локальной подобласти (`start_i, end_i, size_M, size_N`) и соседние процессы
- Функции MPI для топологии: `MPI_Dims_create()`, `MPI_Cart_create()` – создание декартовой сетки процессов
- Функции MPI для коммуникации: `MPI_Isend()`, `MPI_Irecv()`, `MPI_Waitall()` – неблокирующий обмен граничными значениями между соседями
- Функции MPI для синхронизации: `MPI_Allreduce()` – глобальная редукция скалярных произведений, `MPI_Barrier()` – синхронизация процессов

MPI-программа обеспечивает декомпозицию области на подобласти и межпроцессное взаимодействие.

7.4 Гибридная программа MPI+OpenMP

Гибридная версия объединяет обе технологии:

- MPI – распределяет задачу между процессами
- OpenMP – распараллеливает вычисления внутри каждого MPI-процесса

Гибридный подход позволяет оптимально использовать ресурсы кластера.

8 Результаты расчетов

Количество OpenMP-нитей	Число точек сетки ($M \times N$)	Число итераций	Время решения, с	Ускорение
0	400×600	1077	112.59	1.0
2	400×600	1077	60.32	1.87
4	400×600	1077	34.92	3.22
8	400×600	1077	26.93	4.18
16	400×600	1077	16.09	7.00
0	800×1200	2047	857.14	1.0
4	800×1200	2047	250.45	3.42
8	800×1200	2047	169.35	5.06
16	800×1200	2047	122.98	6.97
32	800×1200	2047	51.91	16.51

Таблица 1: Таблица с результатами расчетов на ПВС IBM Polus (OpenMP)

Количество процессов MPI	Число точек сетки ($M \times N$)	Число итераций	Время решения, с	Ускорение
0	400×600	1077	112.59	1.0
2	400×600	1077	40.71	2.77
4	400×600	1077	22.45	5.02
8	400×600	1077	13.54	8.31
16	400×600	1077	7.07	15.93
0	800×1200	2047	857.14	1.0
4	800×1200	2047	160.45	5.34
8	800×1200	2047	89.44	9.58
16	800×1200	2047	52.51	16.32
32	800×1200	2047	23.68	36.20

Таблица 2: Таблица с результатами расчетов на ПВС IBM Polus (MPI)

Количество процессов MPI	Количество OpenMP-нитей в процессе	Число точек сетки ($M \times N$)	Число итераций	Время решения, с	Ускорение
0	0	400×600	1077	112.59	1.0
2	1	400×600	1077	6.49	17.36
2	2	400×600	1077	3.38	33.31
2	4	400×600	1077	1.79	62.90
2	8	400×600	1077	0.99	113.73
0	0	800×1200	2047	857.14	1.0
4	1	800×1200	2047	24.93	34.38
4	2	800×1200	2047	13.16	65.13
4	4	800×1200	2047	7.21	118.88
4	8	800×1200	2047	4.87	176.00

Таблица 3: Таблица с результатами расчетов на ПВС IBM Polus (MPI + OpenMP)

8.1 Графики

8.2 График приближенного решения

На рисунке 1 представлено приближенное решение задачи Дирихле для уравнения Пуассона в треугольной области, полученное на сетке 800×1200 .

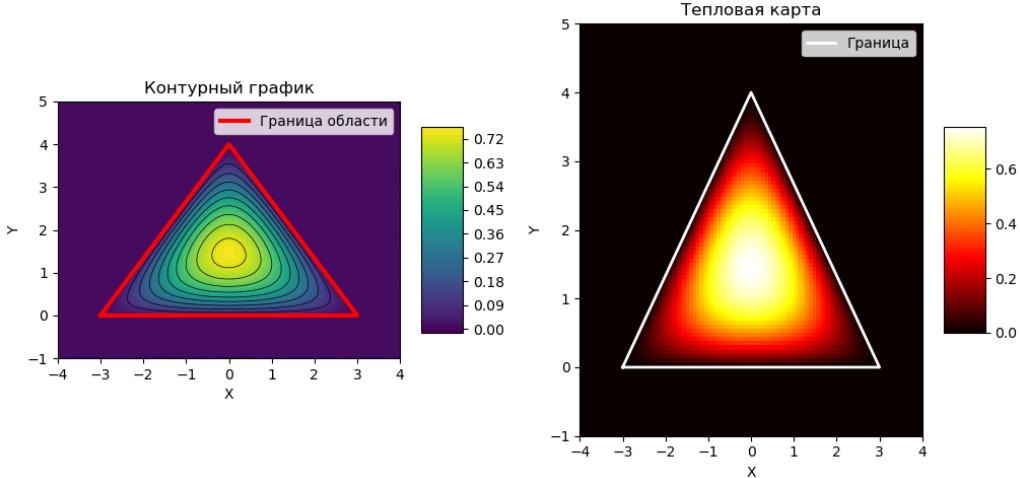


Рис. 1: Приближенное решение уравнения Пуассона в треугольной области на сетке 800×1200

Решение демонстрирует ожидаемое поведение – максимальные значения в центре треугольной области с плавным уменьшением к границам, где задано нулевое граничное условие. Форма решения соответствует геометрии расчетной области.

8.3 Графики ускорений

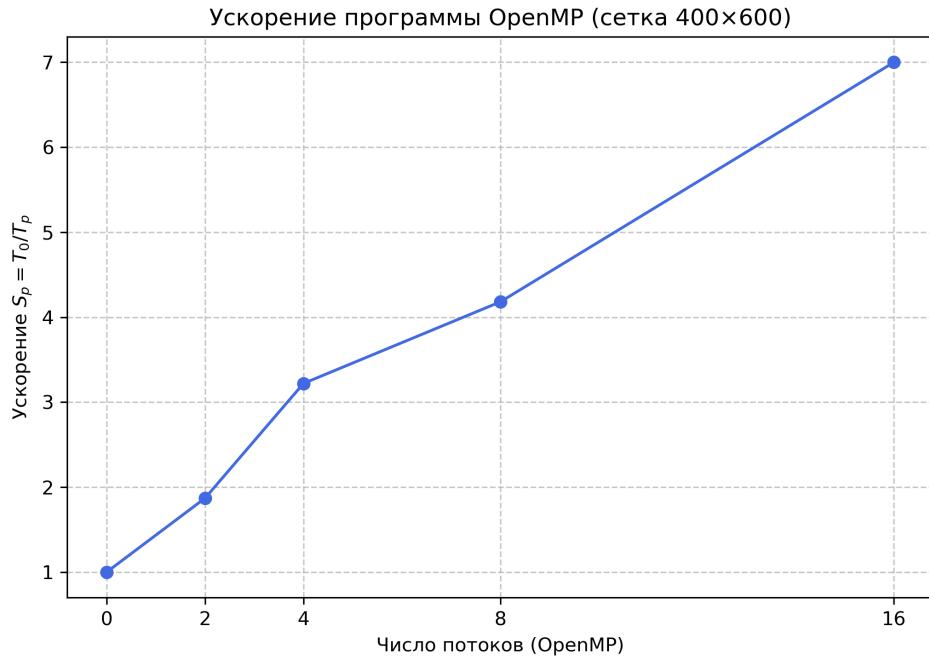


Рис. 2: Ускорение программы OpenMP для сетки 400×600

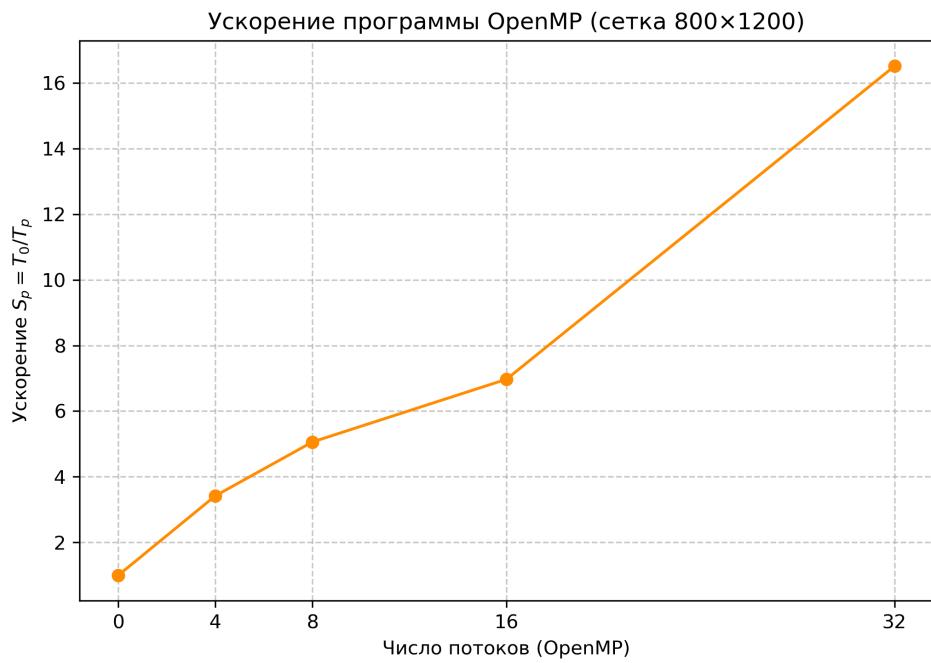


Рис. 3: Ускорение программы OpenMP для сетки 800×1200

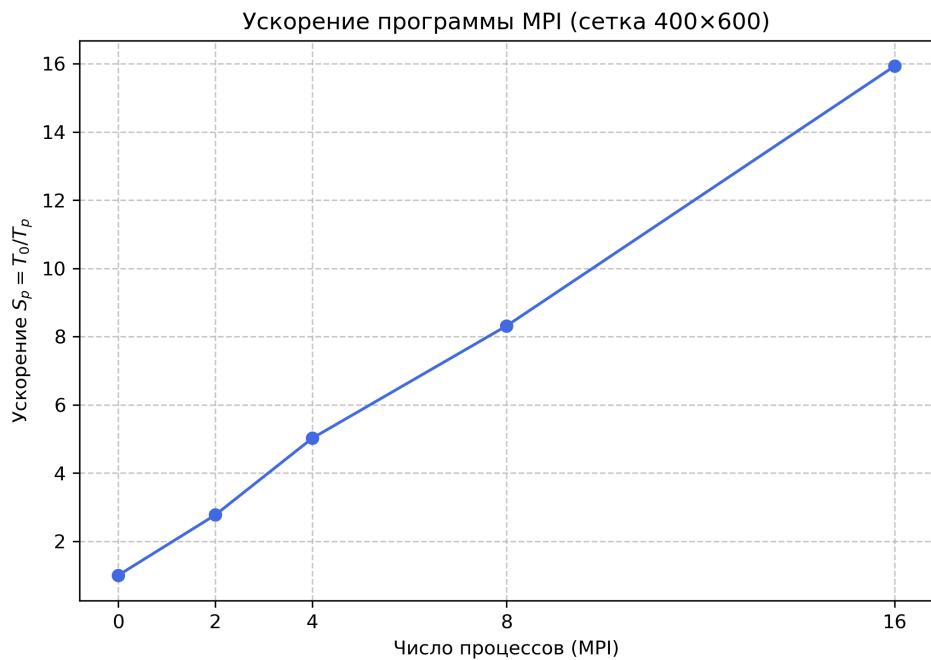


Рис. 4: Ускорение программы MPI для сетки 400×600

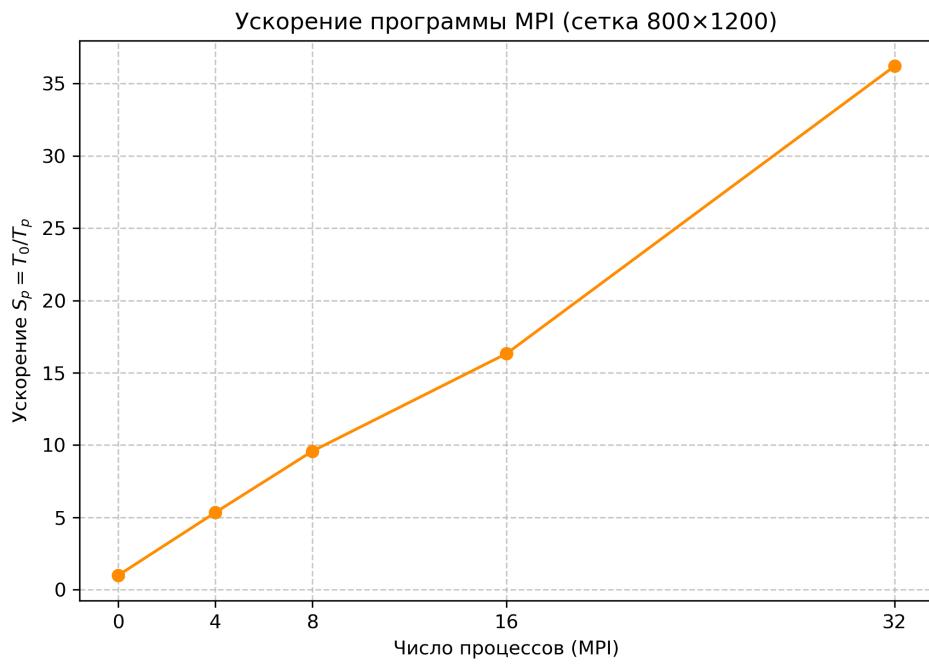


Рис. 5: Ускорение программы MPI для сетки 800×1200

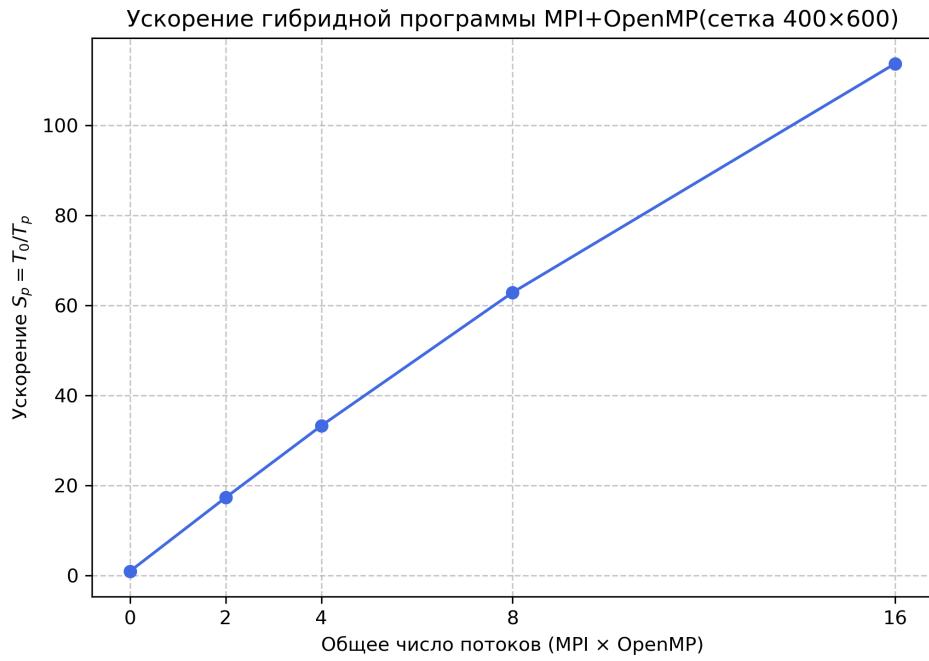


Рис. 6: Ускорение гибридной программы (MPI+OpenMP) для сетки 400×600

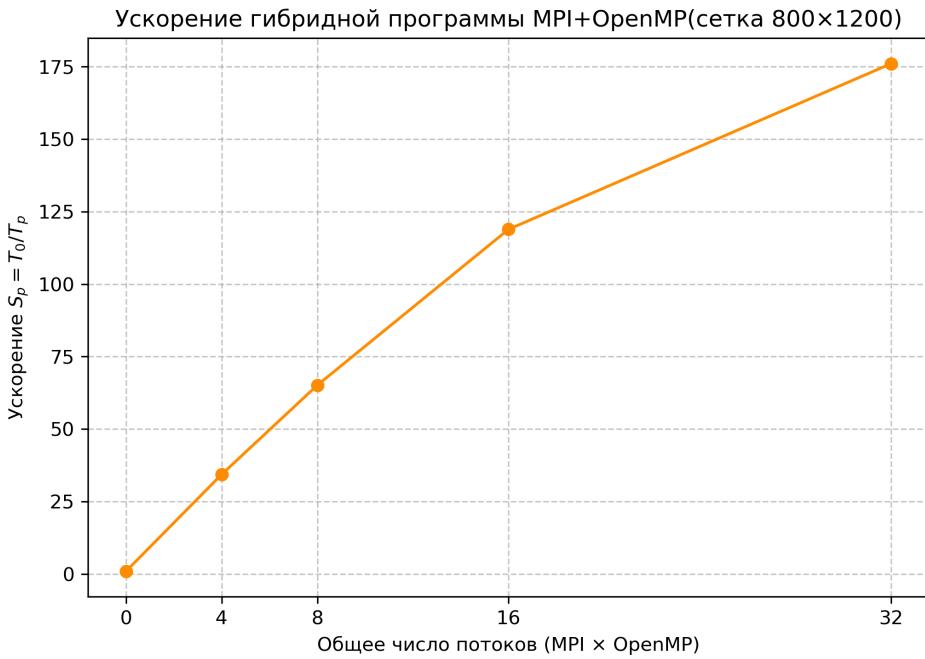


Рис. 7: Ускорение гибридной программы (MPI+OpenMP) для сетки 800×1200

9 Заключение

В ходе работы последовательно разработаны программы для решения уравнения Пуассона в треугольной области: последовательная, OpenMP, MPI и гибридная MPI+OpenMP. Все версии сохраняют точность алгоритма и дают одинаковое количество итераций сходимости.

OpenMP показала ускорение до 16.51 раза, MPI — до 36.20 раза, а гибридная реализация достигла наилучшего результата — ускорения 176.00 раза при использовании 32 потоков. Гибридный подход MPI+OpenMP оптимально использует ресурсы кластерных систем.

Эффективность параллелизации возрастает с увеличением размера задачи, так как объем вычислений превосходит накладные расходы на коммуникацию. Полученные результаты подтверждают целесообразность использования вышеупомянутых методов.