

# Задание №1 3-й поток по курсу «Суперкомпьютерное моделирование и технологии»

октябрь 2024 - декабрь 2024

## Содержание

Содержание	1
1 Введение	1
2 Математическая постановка дифференциальной задачи	1
3 Численный метод решения задачи	2
4 Программная реализация	3
5 Варианты заданий	4
6 Требования к отчету	5
7 Материалы, сдаваемые на проверку	6
Список литературы	7

## 1 Введение

В качестве модельной задачи предлагается задача для трехмерного гиперболического уравнения в области, представляющей из себя прямоугольный параллелепипед. Индивидуальные варианты заданий отличаются типом граничных условий. Данное уравнение часто применяется в теории тепло и массопереноса, гидро и аэромеханике, электростатике. Поэтому поиск решения данной задачи в различных областях является весьма актуальным.

Задание необходимо выполнить на следующих ПВС Московского университета:

1. IBM Polus [1],

## 2 Математическая постановка дифференциальной задачи

В трехмерной замкнутой области

$$\Omega = [0 \leq x \leq L_x] \times [0 \leq y \leq L_y] \times [0 \leq z \leq L_z]$$

для  $(0 < t \leq T]$  требуется найти решение  $u(x, y, z, t)$  уравнения в частных производных

$$\frac{\partial^2 u}{\partial t^2} = \Delta u \quad (1)$$

с начальными условиями

$$u|_{t=0} = \varphi(x, y, z), \quad (2)$$

$$\left. \frac{\partial u}{\partial t} \right|_{t=0} = 0, \quad (3)$$

при условии, что на границах области заданы однородные граничные условия первого рода

$$u(0, y, z, t) = 0, \quad u(L_x, y, z, t) = 0, \quad (4)$$

$$u(x, 0, z, t) = 0, \quad u(x, L_y, z, t) = 0, \quad (5)$$

$$u(x, y, 0, t) = 0, \quad u(x, y, L_z, t) = 0, \quad (6)$$

либо периодические граничные условия

$$u(0, y, z, t) = u(L_x, y, z, t), \quad u_x(0, y, z, t) = u_x(L_x, y, z, t), \quad (7)$$

$$u(x, 0, z, t) = u(x, L_y, z, t), \quad u_y(x, 0, z, t) = u_y(x, L_y, z, t), \quad (8)$$

$$u(x, y, 0, t) = u(x, y, L_z, t), \quad u_z(x, y, 0, t) = u_z(x, y, L_z, t). \quad (9)$$

Конкретная комбинация граничных условий определяется индивидуальным вариантом задания (см. п. 5).

### 3 Численный метод решения задачи

Содержание данного пункта основано на материале книги [2].

Для численного решения задачи введем на  $\Omega$  сетку  $\omega_{h\tau} = \bar{\omega}_h \times \omega_\tau$ , где

$$T = T_0,$$

$$L_x = L_{x_0}, L_y = L_{y_0}, L_z = L_{z_0}$$

$$\bar{\omega}_h = \{(x_i = ih_x, y_j = jh_y, z_k = kh_z), i, j, k = 0, 1, \dots, N, h_x N = L_x, h_y N = L_y, h_z N = L_z\},$$

$$\omega_\tau = \{t_n = n\tau, n = 0, 1, \dots, K, \tau K = T\}.$$

Через  $\omega_h$  обозначим множество внутренних, а через  $\gamma_h$  — множество граничных узлов сетки  $\bar{\omega}_h$ .

Для аппроксимации исходного уравнения (1) с однородными граничными условиями (4)–(6) и начальными условиями (2)–(3) воспользуемся следующей системой уравнений:

$$\frac{u_{ijk}^{n+1} - 2u_{ijk}^n + u_{ijk}^{n-1}}{\tau^2} = \Delta_h u^n, \quad (x_i, y_j, z_k) \in \omega_h, \quad n = 1, 2, \dots, K-1,$$

Здесь  $\Delta_h$  — семиточечный разностный аналог оператора Лапласа:

$$\Delta_h u^n = \frac{u_{i-1,j,k}^n - 2u_{i,j,k}^n + u_{i+1,j,k}^n}{h^2} + \frac{u_{i,j-1,k}^n - 2u_{i,j,k}^n + u_{i,j+1,k}^n}{h^2} + \frac{u_{i,j,k-1}^n - 2u_{i,j,k}^n + u_{i,j,k+1}^n}{h^2}.$$

Приведенная выше разностная схема является явной — значения  $u_{ijk}^{n+1}$  на  $(n+1)$ -м шаге можно явным образом выразить через значения на предыдущих слоях.

Для начала счета (т.е. для нахождения  $u_{ijk}^2$ ) должны быть заданы значения  $u_{ijk}^0, u_{ijk}^1$ ,  $(x_i, y_j, z_k) \in \omega_h$ . Из условия (2) имеем

$$u_{ijk}^0 = \varphi(x_i, y_j, z_k), \quad (x_i, y_j, z_k) \in \omega_h. \quad (10)$$

Простейшая замена начального условия (3) уравнением  $(u_{ijk}^1 - u_{ijk}^0)/\tau = 0$  имеет лишь первый порядок аппроксимации по  $\tau$ . Аппроксимацию второго порядка по  $\tau$  и  $h$  дает разностное уравнение

$$\frac{u_{ijk}^1 - u_{ijk}^0}{\tau} = \frac{\tau}{2} \Delta_h \varphi(x_i, y_j, z_k), \quad (x_i, y_j, z_k) \in \omega_h. \quad (11)$$

$$u_{ijk}^1 = u_{ijk}^0 + \frac{\tau^2}{2} \Delta_h \varphi(x_i, y_j, z_k) \quad (12)$$

Разностная аппроксимация для периодических граничных условий выглядит следующим образом

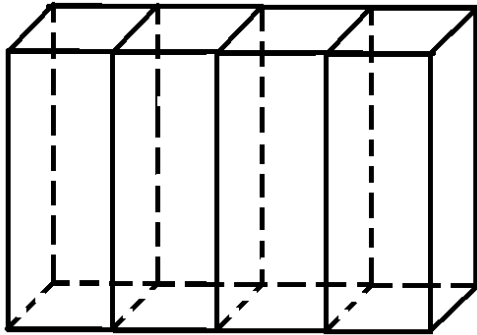
$$\begin{aligned} u_{0jk}^{n+1} &= u_{Njk}^{n+1}, & u_{1jk}^{n+1} &= u_{N+1jk}^{n+1}, \\ u_{i0k}^{n+1} &= u_{iNk}^{n+1}, & u_{i1k}^{n+1} &= u_{iN+1k}^{n+1}, \\ u_{ij0}^{n+1} &= u_{ijN}^{n+1}, & u_{ij1}^{n+1} &= u_{ijN+1}^{n+1}, \end{aligned}$$

$i, j, k = 0, 1, \dots, N$ .

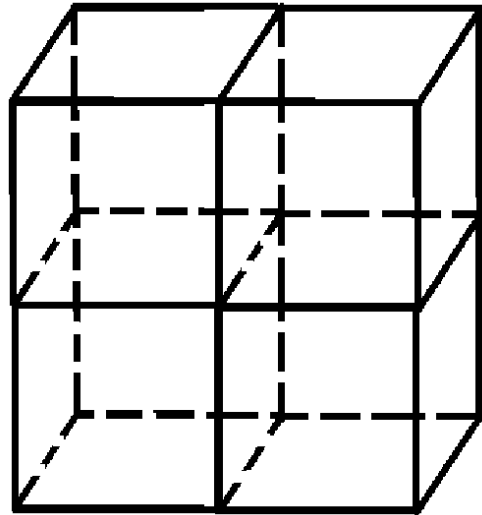
Для вычисления значений  $u^0, u^1 \in \gamma_h$  допускается использование аналитического значения  $u$ , которое задается в программе еще для вычисления погрешности решения задачи.

## 4 Программная реализация

Требуется реализовать блочное разбиение области между процессами, поскольку в этом случае предполагается меньше межпроцессорных коммуникаций, по сравнению с ленточным.



Ленточное разбиение 3d.



Блочное разбиение 3d.

Рис. 1: Виды разбиений 3d

Число областей, на которое необходимо произвести разбиение может быть произвольным, как четным, так и нечетным.

На рисунке 1 изображен схематично двумерный ленточный и блочный вариант разбиения квадратной области на 8 частей.

В случае ленточного разбиения получаем  $S_{strip} = 8L$  межпроцессорных коммуникаций. В блочном случае  $S_{block} = 8(L/2) + 4(L/4) = 5L$  коммуникаций, то есть в 1.6 раз меньше. С ростом числа разбиений (и процессов, соответственно) это число будет только увеличиваться. Данные рассуждения можно обобщить и на трехмерный случай.

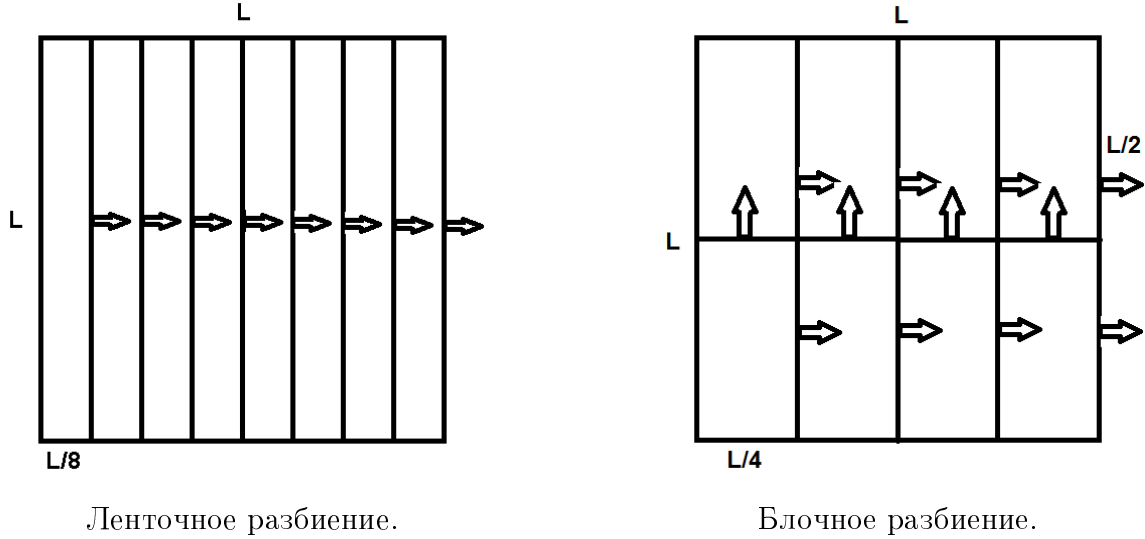


Рис. 2: Виды разбиений

Таким образом, алгоритм решения задачи, выглядит следующим образом:

1. Исходя из варианта, рассчитывается точное аналитическое решение  $u_{analytical}$  в узлах сетки
2. Проводим разбиение области  $\Omega$  между процессами.
3. Фиксируем временной слой (начиная с 0).
4. Используя формулы (10) и (12). находим значения  $u^0$  и  $u^1$ .
5. Пользуясь найденными  $u^0$  и  $u^1$  и разностным представлением уравнения (1), находим значения  $u$  в локальной области разбиения
6. Передаем посчитанные граничные значения блокам-соседям.
7. Повторяем шаги 5-6 для внутренних блоков.
8. Определяем максимальную погрешность на сетке между посчитанным и аналитическим решением.
9. Переходим на следующий слой по времени и повторяем шаги 2-9.

## 5 Варианты заданий

Индивидуальные варианты заданий отличаются комбинацией граничных условий. Варианты приведены в следующей таблице (1P - условие первого рода, П - периодическое условие) 1.

Таблица 1: Варианты заданий

Вариант	$x$	$y$	$z$	$u_{analytical}$
1	1P	1P	1P	$\sin(\frac{\pi}{L_x}x) \cdot \sin(\frac{\pi}{L_y}y) \cdot \sin(\frac{\pi}{L_z}z) \cdot \cos(a_t \cdot t), a_t = \pi\sqrt{\frac{1}{L_x^2} + \frac{1}{L_y^2} + \frac{1}{L_z^2}}$
2	1P	1P	П	$\sin(\frac{\pi}{L_x}x) \cdot \sin(\frac{\pi}{L_y}y) \cdot \sin(\frac{2\pi}{L_z}z) \cdot \cos(a_t \cdot t + 2\pi), a_t = \pi\sqrt{\frac{1}{L_x^2} + \frac{1}{L_y^2} + \frac{4}{L_z^2}}$
3	1P	П	1P	$\sin(\frac{\pi}{L_x}x) \cdot \sin(\frac{2\pi}{L_y}y) \cdot \sin(\frac{3\pi}{L_z}z) \cdot \cos(a_t \cdot t), a_t = \pi\sqrt{\frac{1}{L_x^2} + \frac{4}{L_y^2} + \frac{9}{L_z^2}}$
4	1P	П	П	$\sin(\frac{3\pi}{L_x}x) \cdot \sin(\frac{2\pi}{L_y}y) \cdot \sin(\frac{2\pi}{L_z}z) \cdot \cos(a_t \cdot t + 4\pi), a_t = \pi\sqrt{\frac{9}{L_x^2} + \frac{4}{L_y^2} + \frac{4}{L_z^2}}$
5	П	1P	1P	$\sin(\frac{2\pi}{L_x}x) \cdot \sin(\frac{\pi}{L_y}y) \cdot \sin(\frac{\pi}{L_z}z) \cdot \cos(a_t \cdot t + 2\pi), a_t = \pi\sqrt{\frac{4}{L_x^2} + \frac{1}{L_y^2} + \frac{1}{L_z^2}}$
6	П	1P	П	$\sin(\frac{2\pi}{L_x}x) \cdot \sin(\frac{\pi}{L_y}y + \pi) \cdot \sin(\frac{2\pi}{L_z}z + 2\pi) \cdot \cos(a_t \cdot t + \pi), a_t = \pi\sqrt{\frac{4}{L_x^2} + \frac{1}{L_y^2} + \frac{4}{L_z^2}}$
7	П	П	1P	$\sin(\frac{2\pi}{L_x}x + 3\pi) \cdot \sin(\frac{2\pi}{L_y}y + 2\pi) \cdot \sin(\frac{\pi}{L_z}z) \cdot \cos(a_t \cdot t + \pi), a_t = \pi\sqrt{\frac{4}{L_x^2} + \frac{4}{L_y^2} + \frac{1}{L_z^2}}$
8	П	П	П	$\sin(\frac{2\pi}{L_x}x) \cdot \sin(\frac{4\pi}{L_y}y) \cdot \sin(\frac{6\pi}{L_z}z) \cdot \cos(a_t \cdot t), a_t = \pi\sqrt{\frac{4}{L_x^2} + \frac{16}{L_y^2} + \frac{36}{L_z^2}}$

Значениям «1-го рода» и «периодические» в столбце  $x$  отвечают формулы (4) и (7), в столбце  $y$  — (5) и (8), в столбце  $z$  — (6) и (9).

Функция  $u_{analytical}$  напрямую зависит от введенных  $L_x, L_y, L_z$ , которые являются входными аргументами программы. Функция  $\varphi(x, y, z)$  зависит напрямую от  $u_{analytical}$  (2).

## 6 Требования к отчету

Для того, чтобы успешно сдать задание, необходимо

- понимать семантику всех используемых в коде функций MPI и директив OpenMP;
- представить отчет с результатами исследования параллельных характеристик программы;
- представить программный код.

Также необходимо провести исследование параллельных характеристик гибридной программы MPI/OpenMP и сравнить полученные результаты с программой, не использующей директивы OpenMP.

Отчет о выполнении задания должен содержать

- математическую постановку задачи;
- численные метод ее решения;
- краткое описание проделанной работы по созданию гибридной реализации MPI/OpenMP;
- результаты расчетов (см. ниже).

Расчеты проводятся для разных размеров задач и на разном числе процессоров. Результаты расчетов заносятся в таблицу. Значениями в ячейках таблицы являются время решения и ускорение (таблица 2).

Следует выполнить около 20 шагов по времени.

Таблица 2: Пример оформления таблицы с результатами расчетов на ПВС IBM Polus (MPI код)

Число MPI процессов $N_p$	Число точек сетки $N^3$	Время решения $T$	Ускорение $S$	Погрешность $\delta$
1	128 <sup>3</sup>			
4	128 <sup>3</sup>			
8	128 <sup>3</sup>			
16	128 <sup>3</sup>			
32	128 <sup>3</sup>			
1	256 <sup>3</sup>			
8	256 <sup>3</sup>			
4	256 <sup>3</sup>			
16	256 <sup>3</sup>			
32	256 <sup>3</sup>			
1	512 <sup>3</sup>			
4	512 <sup>3</sup>			
8	512 <sup>3</sup>			
16	512 <sup>3</sup>			
32	512 <sup>3</sup>			

Таблица 3: Пример оформления таблицы с результатами расчетов на ПВС IBM Polus (MPI + OpenMP код)

Число MPI процессов $N_p$	Число OpenMP нитей в процессе	Число то- чек сетки $N^3$	Время ре- шения $T$	Ускорение $S$	Погрешность $\delta$
1	4	128 <sup>3</sup>			
2	4	128 <sup>3</sup>			
4	4	128 <sup>3</sup>			
8	4	128 <sup>3</sup>			
1	4	256 <sup>3</sup>			
2	4	256 <sup>3</sup>			
4	4	256 <sup>3</sup>			
8	4	256 <sup>3</sup>			
1	4	512 <sup>3</sup>			
2	4	512 <sup>3</sup>			
4	4	512 <sup>3</sup>			
8	4	512 <sup>3</sup>			

Требуется провести расчеты как минимум для двух случаев:  $L_x = L_y = L_z = 1$ ,  $L_x = L_y = L_z = \pi$

## 7 Материалы, сдаваемые на проверку

Каждый студент, помимо очной сдачи своему преподавателю и сдачи отчета, должен сдать следующий пакет материалов:

- Исходные коды программы (\*.c, \*.cpp и т.д.)
- make/СMake файлы для сборки программы или \*.txt файл, содержащий команды

для сборки и запуска программы на Polus

- Файл, содержащий основную информацию, выдаваемую запускаемой на Polus программой такую, как время выполнения программы, максимальная полученная погрешность. Отладочную информацию в файл включать не нужно. Файл должен содержать подобную информацию для всех типов запусков (последовательный код, OpenMP, MPI, OpenMP + MPI для всех необходимых чисел узлов).

## IBM Polus

На Polus провести исследование только MPI-реализации для 1, 4, 8, 16, 32 MPI-процессов. Расчеты должны быть проведены на сетках  $128^3$ ,  $256^3$ ,  $512^3$ .

## Список литературы

- [1] Суперкомпьютер IBM Polus. — <http://hpc.cs.msu.su/polus>.
- [2] Самарский А.А., Гулин А.В. Численные методы. — М.: Наука. Гл. ред. физ-мат. лит., 1989.