

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет информационных технологий  
Кафедра параллельных вычислений

Основы параллельного программирования

## Отчет

О выполнении работы № 4

**Работу**  
**выполнил:**  
Е. И. Биточкин  
Группа: 22209  
**Преподаватель:**  
А. А. Артюхов

Новосибирск  
2024

# Содержание

<b>1. Цель</b>	<b>3</b>
<b>2. Задание</b>	<b>3</b>
<b>3. Описание работы</b>	<b>4</b>
3.1. Реализация . . . . .	4
3.2. Профилирование . . . . .	5
3.3. Замеры . . . . .	7
3.3.1. Зависимость времени выполнения от числа процессов . . . . .	7
<b>4. Заключение</b>	<b>8</b>
<b>5. Приложение</b>	<b>8</b>
5.1. Исходный код . . . . .	8

## 1. Цель

- Реализовать алгоритм Якоби используя MPI.
- Реализовать передачу данных, параллельную вычислениям.
- Исследовать разработанный алгоритм.

## 2. Задание

- Реализация алгоритма Якоби.
- Выявления зависимости времени работы алгоритма от числа процессов

## **3. Описание работы**

### **3.1. Реализация**

Алгоритм был реализован на C++. В качестве системы сборки был выбран CMake.

## 3.2. Профилирование

За счет того, что коммуникация происходит на фоне вычислений, время на коммуникацию минимально:

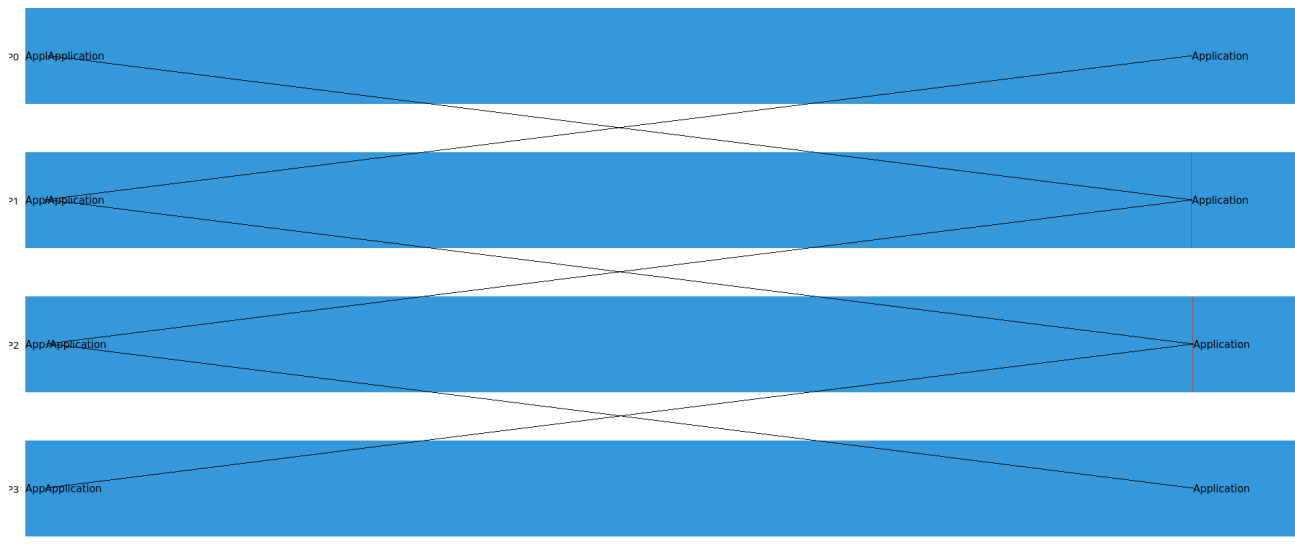


Рисунок 3.1. Работа

Если мы посмотрим на статистику, то увидим, что время, потраченное на коммуникацию минимально:

### Ratio

This section represents a ratio of all MPI calls to the rest of your code in the application.

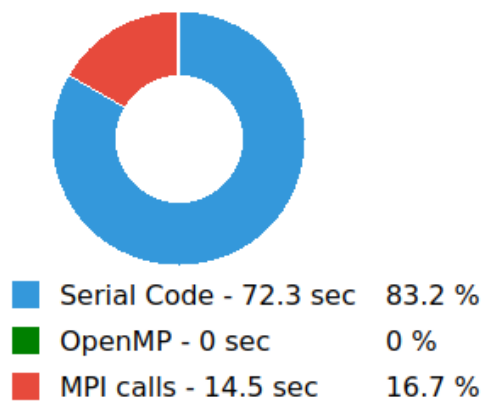


Рисунок 3.2. Статистика по времени

Более того, 14.4 из 14.5 секунд было потрачено на сборку результата в одном процессе (MPI\_Gather):

## Top MPI functions

---

This section lists the most active MPI functions from all MPI calls in the application.


MPI_Gatherv		14.4 sec (9.29 %)
MPI_Allreduce		0.0335 sec (0.0216 %)
MPI_Waitall		0.0177 sec (0.0114 %)
MPI_Isend		0.00135 sec (0.000871 %)
MPI_Finalize		0.000539 sec (0.000347 %)

Рисунок 3.3. Gather

Из чего можно сделать вывод, что коммуникация получилась эффективной/

### 3.3. Замеры

#### 3.3.1. Зависимость времени выполнения от числа процессов

На кластере был произведен замер для выявления зависимости времени работы от числа процессов:

число процессов	время, с	ускорение, раз	эффективность, %
1	3,90757	1	100,00
2	3,49905	1,12	55,84
4	1,78751	2,19	54,65
8	1,87174	2,09	26,10
16	0,984717	3,97	24,80

Рисунок 3.4.  $N_x = N_y = N_z = 600$

Графики:

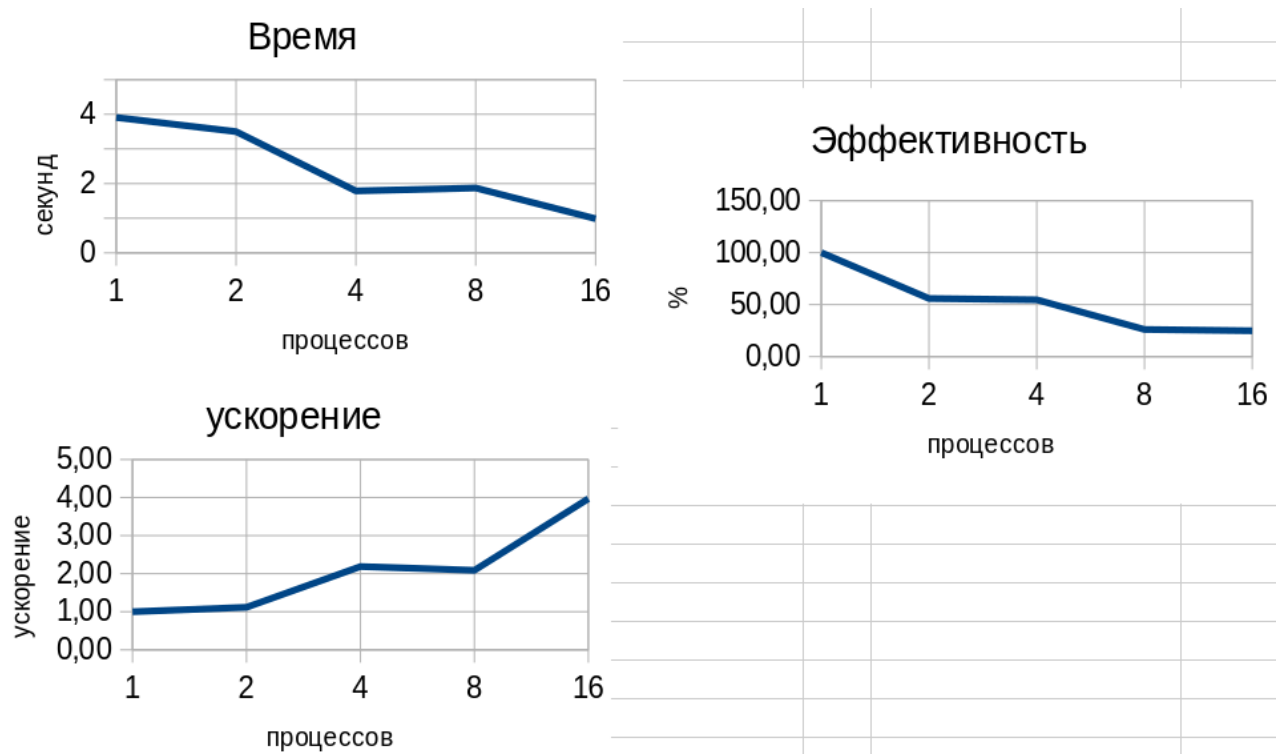


Рисунок 3.5. Время, Ускорение, Эффективность

## 4. Заключение

Был реализован алгоритм Якоби на нескольких процессах. Рассмотрена эффективность изложенного подхода.

- Зависимость времени работы алгоритма от числа процессов рассмотрена.
- Была написана программа, вычисляющая метод Якоби в трехмерной области

## 5. Приложение

### 5.1. Исходный код

[https://github.com/BigCubeCat/bpp\\_labs.git](https://github.com/BigCubeCat/bpp_labs.git)