**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по лабораторной работе №4**

**по дисциплине «Параллельные алгоритмы»**

Тема: П**араллельное умножение матриц**

|  |  |  |
| --- | --- | --- |
| Студент гр. 0303 |  | Парамонов В.В. |
| Преподаватель |  | Сергеева Е. И. |

Санкт-Петербург

2023

## Цель работы.

Исследовать реализации масштабируемого и “быстрого” умножения матриц (Штрассена). Проверить корректность реализации и скорость выполнения.

**Постановка задачи.**

Выполняется на основе работы 1.

* Реализовать параллельный алгоритм умножения матриц с масштабируемым разбиением по потокам (2).
* Исследовать масштабируемость выполненной реализации с реализацией из работы 1 (1).
* Реализовать параллельный алгоритм “быстрого” умножения матриц (Штрассена или его модификации) (3).
* Проверить, что результаты вычислений (2) и (3) совпадают.
* Сравнить производительность (3) с (2) на больших размерностях данных (порядка 10^4 – 10^6)

**Выполнение задач.**

1. **Изменения структуры решения в сравнении с лабораторной 1**:
2. Были убраны решения задачи умножения матриц с разделением по задачам на потоки и процессы.
3. Добавлены 2 новых функции параллельного умножения матриц:

* Масштабируемое умножение (2) [void threadsScalableMul(Matrix& first, Matrix& second, Matrix& mulResult, int threadInd, int threadsNum)], которое заключается в вычислении каждым потоком отдельных значений матрицы результата.
* “Быстрое” умножение Штрассена (3) [threadsMulStrassen(Matrix first, Matrix second, Matrix& mulResult, int recursionDepth, int allowRecursionDepth)], которое соответствует стратегии “разделяй и властвуй”. Умножаемые матрицы делятся на четыре равные части и каждая рекурсивно вычисляется тем же алгоритмом, далее в 7 умножений находится искомая матрица результат.

1. Класс матрица претерпел изменения. Добавились операции сложения, вычитания и обычного не параллельного умножения.
2. При каждом запуске программы есть выбор на использование нескольких вариантов перемножения матриц из (1), (2) и (3). Результаты данных умножений сравниваются на идентичность.

1. **Исследование полученных с использованием программы результатов:**
2. Исследуем скорость работы параллельного умножения матриц, реализованной в 1 лабораторной по строкам (1) с масштабируемым умножением с вычислением элемента результирующей матрицы (2). Будем рассматривать умножение матриц (x, 1000) на (1000, 1000) в 7 потоков (см. таблицу 1):

Таблица 1 – Измерение времени работы параллельных умножений матриц (1) и (2)

|  |  |  |
| --- | --- | --- |
| Размеры матриц | Время выполнения для параллельного умножения матриц по строкам (1) (с) | Время выполнения для масштабируемого параллельного умножения матриц (2) (с) |
| (1000, 1000); (1000, 1000) | **1.56842** | 2.15794 |
| (100, 1000); (1000, 1000) | **0.189192** | 0.212244 |
| (10, 1000); (1000, 1000) | 0.0347434 | **0.029875** |
| (1, 1000); (1000, 1000) | 0.0190915 | **0.00706199** |

Исходя из полученных данных в таблице 1 масштабируемое умножение матриц (2) гораздо быстрее, чем умножение по строкам при уменьшении количества строк в результирующей матрице, так как в случае параллельного умножения матриц по строкам (1) не все потоки получают равный объем работы из-за проблем с масштабированием такого подхода.

1. Теперь исследуем скорости работы масштабируемого умножения матриц (2) и “быстрого” умножения матриц Штрассена (3). Будем рассматривать умножение 2-х квадратных матриц размера (size, size); (size, size) в 7 потоков (см. таблицу 2).

Таблица 2 - Измерение времени работы параллельных умножений матриц (2) и (3)

|  |  |  |
| --- | --- | --- |
| Размеры матриц size | Время выполнения для масштабируемого параллельного умножения матриц (2) (с) | Время выполнения для “быстрого” параллельного умножения матриц Штрассена (3) (с) |
| 64 | **0.000703902** | 0.00268239 |
| 128 | **0.00556726** | 0.00965697 |
| 256 | **0.0414657** | 0.0477349 |
| 512 | 0.30188 | **0.223769** |
| 1024 | 2.50183 | **1.37716** |
| 2048 | 39.3093 | **10.6346** |
| 4096 | 261.531 | **85.3452** |
| 8192 | 2251.61 | **654.763** |

Исходя из полученных данных в таблице 2 масштабируемое умножение матриц (2) быстрее, чем “быстро” умножение Штрассена (3) только при маленьких размерах перемножаемых матриц. При увеличении размера матриц (3) начинает выигрывать за счёт меньшей асимптотической сложности у (2), как и ожидалось.

**Заключение.**

В ходе работы были изучены алгоритмы масштабируемого умножения матриц (2) и “быстрого” умножения матриц Штрассена (3). Была практически доказана масштабируемость параллельного умножения матриц (2) и эффективность при больших размерах матриц (3).