**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №4**

**по дисциплине «Параллельные алгоритмы»**

Тема: Параллельное умножение матриц

|  |  |  |
| --- | --- | --- |
| Студентка гр. 1304 |  | Чернякова В.А. |
| Преподаватель |  | Сергеева Е.И. |

Санкт-Петербург

2024

## Цель работы.

Изучить умножения матриц с помощью алгоритма Штрассена с разбиением на потоки и сравнить производительность разных алгоритмов.

## Задание.

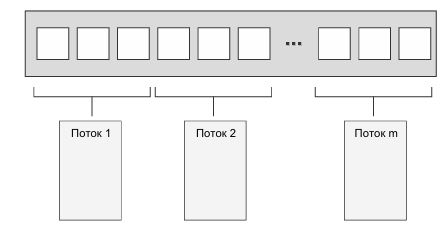
4.1 Реализовать параллельный алгоритм умножения матриц с блочным разбиением по потокам. Исследовать масштабируемость выполненной реализации, сравнить с реализацией из работы 1.

4.2 Реализовать параллельный алгоритм “быстрого” умножения матриц (Штрассена или его модификации). Тестирование: проверить, что результаты вычислений реализаций 4.1 и 4.2 совпадают (в том числе на больших размерностях). Сравнить производительность с реализацией 4.1 на больших размерностях данных (порядка 10^4 – 10^6).

## Выполнение работы.

Задание 4.1

Простейший способ распределения данных заключается в том, чтобы назначить первые N элементов одному потоку, следующие N – другому и так далее.



Как бы ни были распределены данные, каждый поток обрабатывает только назначенные ему элементы, никак не взаимодействуя с другими потоками до тех пор, пока не завершит обработку.

В реализации параллельного умножения матриц общее количество элементов финальной матрицы делится на блоки равного размера между P потоками. Если невозможно нацело разделить количество умножений, остаток операций выполняется в последнем потоке. С помощью этого достигался баланс загруженности каждого потока и равномерное вычисление.

Задание 4.2

Штрассен придумал сложный набор соотношений, которые позволили заменить одно из этих восьми умножений 14 дополнительными сложениями.

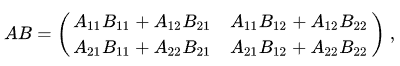
Смысл этого многократного разбиения больших матриц на более мелкие заключается в том, что можно снова и снова применять алгоритм Штрассена к меньшим матрицам и с помощью его метода сокращать количество шагов на каждом этапе. В целом алгоритм Штрассена увеличил скорость умножения матриц с n3 до n2.81 мультипликативных шагов.

Если добавить к матрицам A и B одинаковые нулевые строки и столбцы, их произведение станет равно матрице AB с теми же добавленными строками и столбцами. Поэтому можно рассматривать только матрицы размера , а другие случаи сводить к этому добавлением нулей, отчего n может увеличиться лишь вдвое.

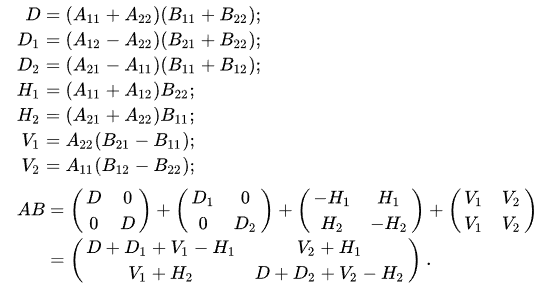
Пусть A, B – матрицы размера Их можно представить как блочные матрицы размера из матриц.



По принципу блочного умножения, матрица AB выражается через их произведение:



где в правой части происходит восемь умножений матриц размера 2k−1×2k−1. Поскольку матрицы образуют кольцо, то для вычисления правой части годится любой алгоритм умножения  (2×2)- матриц, использующий лишь сложения, вычитания и умножения. Штрассен предложил такой алгоритм с семью умножениями:



Каждое умножение можно совершать рекурсивно по той же процедуре, а сложение – тривиально, складывая (2k−1)2 элементов.

Сравнение скорости умножения

Были проведены измерения времени от размеров вычисляемых матриц, результаты приведены в таблице. Количество потоков для простого и параллельного алгоритма 7.

|  |  |  |  |
| --- | --- | --- | --- |
| **Размер** | **Простой алгоритм, мс** | **Параллельный алгоритм, мс** | **Алгоритм Штрассена, мс** |
|  | 1.51015 | 0.55202 | 0.116481 |
|  | 0.655445 | 0.579161 | 0.685582 |
|  | 2.56756 | 1.1307 | 1.46393 |
|  | 9.04588 | 4.31222 | 11.4782 |
|  | 65.2177 | 35.6696 | 31.7894 |
|  | 615.772 | 334.645 | 237.683 |
|  | 4918.72 | 2668.66 | 1796.51 |
|  | 43059.5 | 22126.1 | 12056.3 |
|  | 332325 | 165552 | 81226 |

Как можно заметить, алгоритм Штрассена превосходит параллельный алгоритм по производительности, когда сторона матрицы превосходит 256. До этого размера преимущество имеет параллельный алгоритм. Когда сторона матрицы меньше 256 - в алгоритме Штрассена происходит много копирования и

вставки матриц, из-за чего его производительность падает на фоне параллельного алгоритма, которому копирование не требуется. Когда сторона матрицы больше 256 - алгоритм Штрассена выигрывает за счёт меньшего числа перемножений матриц (7 против 8).

## Выводы.

В ходе работы был исследован алгоритм Штрассена для быстрого умножения матриц.

Было установлено, что алгоритм Штрассена эффективен при больших размерах матриц и неэффективен при малых, т. к. использует большое количество копирований. Алгоритм параллельного умножения работает быстрее до размера 256х256, при больших размерах быстрее оказывается алгоритм Штрассена.