Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе № 2 по курсу "Анализ алгоритмов"

Тема Алгоритмы умножения матриц

Студент Цветков И.А.

Группа ИУ7-53Б

Оценка (баллы) ____________

Преподаватель Волкова Л. Л.

Москва — 2021 г.

Содержание

Введение

В математике и программировании часто приходится прибегать к использованию матриц. Существует огромное количество областей их применения в этих сферах. Например, матрицы активно исполюуются при выводе различных формул в физике:

- градиент;
- дивергенция;
- ротор.

Также часто применяются и операции над матрицами - сложение, возведение в степень, умножение. При различных задачах размеры матрицы могут достигать больштх значений. Поэтому оптимизация операций работы над матрицами является важной задачей в программировании. Об оптимизации операции умножения пойдет речь в данной лабораторной работе.

Целью данной работы является изучение, реализация и исследование алгоритмов умножения матриц - классический алгоритм, алгоритм Винограда и оптимизированный алгоритм Винограда. Для достижения поставленной цели необходимо выполнить следующие задачи:

- изучить и реализовать алгоритмы - классический, Винограда и его оптимизацию;
- провести тестирование по времени и по памяти для алгоритмов лабораторной работы;
- провести сравнительный анализ по времени классического алгоритма и алгоритма Винограда;
- провести сравнительный анализ по времени алгоритма Винограда и его оптимизации;
- описать и обосновать полученные результаты в отчете о выполненной лабораторной работе, выполненного как расчётно-пояснительная записка к работе.

1 Аналитическая часть

В этом рзаделе будут представлены классический алгоритм умножения матриц и алгоритм Винограда.

1.1 Матрица

Матрица [1] - математический объект, который представляет собой двумерный массив, в котором элементы располагаются по строкам и столбцам.

Пусть A - матрица, тогда $A_{i,j}$ - элемент этой матрицы, который находится на *i-ой* строке и *j-ом* столбце.

Можно выделить следующие операции над матрицами:

- матрицы одинакового размера можно складывать и вычитать;
- количество столбцов одной матрицы равно количеству строк другой матрицы - их можно перемножить, причем количество строк будет, как у первой матрицы, а столбцов - как у второй.

Замечание: операция умножения матриц некоммутативна - если A и B - квадтраные матрицы, а C - результат их перемножения, то произведение AB и BA дадут разный результат C.

1.2 Стандартный алгоритм

Пусть даны две матрицы

$$A_{lm} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \dots & a_{lm} \end{pmatrix}, \quad B_{mn} = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \dots & b_{mn} \end{pmatrix}, \tag{1.1}$$

тогда матрица C

$$C_{ln} = \begin{pmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{l1} & c_{l2} & \dots & c_{ln} \end{pmatrix}, \tag{1.2}$$

где

$$c_{ij} = \sum_{r=1}^{m} a_{ir} b_{rj} \quad (i = \overline{1,l}; j = \overline{1,n}) \tag{1.3}$$

будет называться произведением матриц A и B.

Стандартный алгоритм реализует данную формулу.

1.3 Алгоритм Копперсмита-Винограда

Алгоритм Копперсмита-Винограда [2] — алгоритм умножения квадратных матриц. Начальная версия имела асимптотическую сложность алгортма около $O(n^{2,3755})$, где n - размер стороны матрицы, но после доработки он стал обаладть лучшей асимптотикой среди всех алгоритмов умножения матриц.

Рассмотрим два вектора $V = (v_1, v_2, v_3, v_4)$ и $W = (w_1, w_2, w_3, w_4)$. Их скалярное произведение равно: $V \cdot W = v_1 w_1 + v_2 w_2 + v_3 w_3 + v_4 w_4$, что эквивалентно (1.4):

$$V \cdot W = (v_1 + w_2)(v_2 + w_1) + (v_3 + w_4)(v_4 + w_3) - v_1 v_2 - v_3 v_4 - w_1 w_2 - w_3 w_4. \tag{1.4}$$

Прирост производительности заключается в идее предварительной обработки - полученное выражение требует большего количества операций, чем стандартное умножение матриц, но выражение в правой части крайнего равенства можно вычислить заранее и запомнить для каждой строки первой матрицы и каждого столбца второй матрицы. Это позволит выполнить лишь два умножения и пять сложений, при учете, что потом будет сложено только с двумя предварительно посчитанными суммами соседних элементов текущих строк и столбцов. Операция сложения выполняется быстрее, поэтому на практике алгоритм должен работать быстрее обычного алогритма перемножения матриц.

Но стоит упомянуть, что при нечетном значении размера матрицы нужно дополнительно добавить произведения крайних элементов соответствующих строк и столбцов.

Вывод

В данном разделе были рассмотрены алгоритмы умножения матриц - стандартного и Винограда, который имеет большую эффективность за счет предварительных вычислений.

Список литературы

[1] Матрица [Электронный ресурс]. Режим доступа: https://terme.ru/termin/matrica.html (дата обращения: 23.10.2021).

[2] Умножение матриц [Электронный ресурс]. Режим доступа: http://algolib.narod.ru/Math/Matrix.html (дата обращения: 23.10.2021).