УДК 004.422.632:004.421–048:004

ИССЛЕДОВАНИЕ АЛГОРИТМОВ ЛИНЕЙНОЙ АЛГЕБРЫ В ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЯХ НА ПРИМЕРЕ АЛГОРИТМОВ УМНОЖЕНИЯ МАТРИЦ

Камышев С.В. [1]*, студент гр.353505, Петроченко М.М.* [2]*, студент гр.353502*

Белорусский государственный университет информатики и радиоэлектроники[1]
г. Минск, Республика Беларусь

Примичева З.Н. – канд. физ.-мат. наук

Аннотация. Работа посвящена исследованию различных методов умножения матриц, анализу их эффективности, а также целесообразности применения различных методов к матрицам произвольных размеров.

Ключевые слова. Информационные технологии, матрица, умножение матриц, алгоритм Штрассена.

Введение

На сегодняшний день информационные технологии находят применение в любой сфере человеческой деятельности. В их основе лежит процесс обработки данных, объем которых растет с каждым годом. Наиболее удобным способом хранения связанных между собой данных, представленных числовыми значениями, является матрица.

Поскольку умножение матриц, возведение матрицы в степень и многочлен от матрицы широко применяются в графическом дизайне и компьютерной графике для преобразования объектов (например, масштабирование, поворот, трансляция), в машинном обучении и искусственном интеллекте при решении задач классификации или прогнозирования, в теории сигналов и цифровой обработке сигналов, в физическом моделировании и инженерии для моделирования и анализа физических систем, в теории графов и компьютерных сетей для анализа связности, кратчайших путей и других свойств графа, в динамических системах, описываемых системами линейных дифференциальных уравнений, для предсказывания поведения системы в будущем и анализирования ее стабильности, в цепях Маркова и анализе вероятностей при моделировании физических процессов, при решении матричных полиномиальных уравнений (МПУ), для составления матричного многочлена, возникает необходимость нахождения эффективных методов вычисления и способов ускорения умножения матриц произвольных размеров.

Постановка задачи

Обычно при упоминании умножения матриц подразумевается стандартный алгоритм, суть которого заключается в нахождении суммы произведений элементов строки первой матрицы на соответствующие элементы столбца второй. Этот алгоритм весьма прост и понятен, однако требует длительных вычислений, что накладывает некоторые ограничения на размер входных данных и производительность программы в целом.

Указанный алгоритм можно оптимизировать, используя особенности программного обеспечения и представив матрицу в виде вектора, что позволит ускорить перебор элементов, значительно уменьшая количество переходов по памяти (модифицированный стандартный алгоритм). Учитывая, что в памяти данные хранятся последовательно, полностью сохраняется последовательное чтение матрицы, тем самым увеличиваются количество попаданий в кэш и быстродействие программы. Другим способом оптимизировать умножение матриц является транспонирование правой матрицы. Её суть заключается в превращении переходов по столбцу в последовательное чтение строки, что увеличивает скорость выполнения алгоритма несмотря на временные затраты на транспонирование.

Одним способом ускорения произведения матриц является применение усовершенствованных алгоритмов. Алгоритмом с наименьшей асимптотикой является алгоритм Копперсмита-Винограда, у которого асимптотическая сложность составляет всего $O(n^{2,37})$, и, значит, для матриц больших размеров существенно уменьшается время, необходимое для выполнения операции, и увеличивается производительность программы. Следует отметить, что этот алгоритм не применяется на практике, так как имеет очень большую константу пропорциональности и является быстродействующим по сравнению с другими алгоритмами только для матриц, размеры которых превышают память современных компьютеров. Другим методом является алгоритм Штрассена.

Алгоритм Штрассена

Рассмотрим сначала блочный метод умножения матриц. Для этого разделим каждую матрицу на четыре подматрицы:

$$A = [A_{11}\ A_{12}\ A_{21}\ A_{22}\ ], B = [B_{11}\ B_{12}\ B_{21}\ B_{22}\ ]. \qquad (1)$$

Тогда

$$AB = [A_{11}B_{11} + A_{12}B_{21} \quad A_{11}B_{12} + A_{12}B_{22} \qquad A_{21}B_{11} + A_{22}B_{21} \quad A_{21}B_{12} + A_{22}B_{22} \ ]. \tag{2}$$

Согласно алгоритму Штрассена введем семь матриц:

$$\begin{gathered} M_1 = (A_{11} + A_{22}) \times (B_{11} + B_{22}); \\ M_2 = (A_{21} + A_{22}) \times B_{11}; \\ M_3 = A_{11} \times (B_{12} - B_{22}); \\ M_4 = A_{22} \times (B_{21} - B_{11}); \\ M_5 = (A_{11} + A_{12}) \times B_{22}; \\ M_6 = (A_{21} - A_{11}) \times (B_{11} + B_{12}); \\ M_7 = (A_{12} - A_{22}) \times (B_{21} + B_{22}). \end{gathered} \tag{3}$$

Выразив компоненты матрицы (2) через матрицы, полученные в равенствах (3), произведение матриц А и В будет иметь вид:

$$C = AB = \ [M_1 + M_4 - M_5 + M_7 \ M_3 + M_5 \ M_2 + M_4 \ M_1 - M_2 + M_3 + M_6 \ ]. \tag{4}$$

Нетрудно заметить, что алгоритм Штрассена содержит всего семь произведений матриц за счет увеличенного количества операций их сложения и вычитания. Поэтому алгоритм можно представить как рекурсивное повторение этого действия.

Пусть все квадратные матрицы имеют порядок $N = 2^n$, в противном случае матрица иного порядка сводится к матрице данного добавлением нулевых строк и столбцов. Обозначим $f(n)$, $n \in N$, — количество действий, совершённых при выполнении алгоритма. Тогда получим

$$f(n) = 7f(n-1) + 4^n k, \tag{5}$$

где $k \in N$ — константа, зависящая от количества операций сложений, выполненных на каждом шаге рекурсии.

Упростив формулу (5), получим

$$f(n) \ = \ 7^n \ * \ (4^n \ + \ 4^{n-1} + \ldots + 4^2 \ + \ 4). \tag{6}$$

Найдем предел отношения функции $f(n)$ к функции $g(n) \ = \ 7^n$ при $n \to \infty$, которая является предположительной асимптотикой данной функции:

$$\frac{f(n)}{g(n)} = \frac{7^n + k4^n(1 + \frac{1}{4} + \ldots + \frac{1}{4^{n-1}})}{7^n} = \ (1 \ + \left(\frac{4}{7}\right)^n \ * \ \frac{4}{3} \ * \ k) \ = \ 1. \tag{7}$$

Следовательно, $f(n) \ = \ O(g(n))$, при $n \to \infty$, поэтому асимптотическая сложность алгоритма

$$O\big(f(n)\big) = \ O(N^7 \ ) \approx \ O(N^{2{,}8074}), \ n \to \infty. \tag{8}$$

Для стандартного алгоритма асимптотическая сложность равна

$$O(N^8 \ ) = \ O(N^3), \ N \to \infty. \tag{9}$$

Заметим, что при уменьшении размера матрицы до достаточно малых наиболее эффективным методом становится использование стандартного метода умножения для получения вспомогательных матриц $M_1, M_2, \ldots, M_7$.

Асимптотическая сложность позволяет оценить время работы алгоритмов и сравнить их между собой. При увеличении размерности матрицы относительная эффективность работы метода Штрассена относительно стандартного алгоритма будет расти так же, как будет увеличиваться разность абсолютных значений времени, затрачиваемого на алгоритм. При возведении матрицы в некоторую натуральную степень n изменение показателя степени матрицы не изменит относительную эффективность метода Штрассена относительно стандартного алгоритма, так как время выполнения обеих операций увеличится в n раз. Разница же между абсолютными значениями будет существенно увеличиваться.

Реализация алгоритма Штрассена

Для исследования асимптотической эффективности различных методов нахождения произведения матриц были написаны программы на языке программирования C++, осуществляющие стандартный алгоритм, модифицированный стандартный алгоритм и метод Штрассена. Тестирование каждого метода проводилось на 9 различных значениях порядка матриц. Каждый тест повторялся по 10 раз для уменьшения влияния фактора случайности и погрешности и выведения среднего значения времени, затрачиваемого на работу функции. На основании полученных данных были построены таблицы.

Полученные данные представлены в таблице 1.

Таблица 1 – сравнительная таблица времени работы алгоритмов перемежения матриц

Алгоритм/Размерность матрицы	64	128	256	512	1024	2048	4096	8192	16384
Стандартный алгоритм	0,8	6,3	51,9	467,4	5008,5	42624,6	379566,4	2704519	23323465
Метод Штрассена	1	6,2	39,4	281	1949,8	13718,8	97021,1	677313,7	4746309
Модифицированный стандартный алгоритм	1	6,1	48,8	376,1	3045,2	24291,4	198213,7	1551148	12561194

Для лучшего восприятия эти данные были изображены на графике 1.

График 1 – сравнительный график времени работы алгоритмов

На графике 2 представлена зависимость относительной эффективности метода Штрассена и стандартного алгоритма от размерности матрицы.

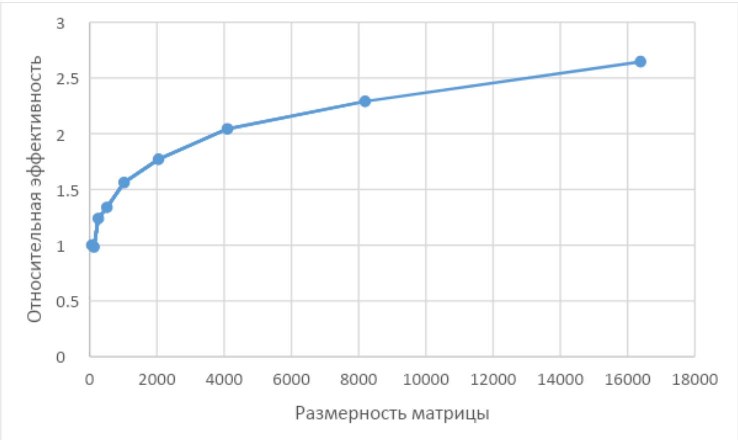

График 2 – график относительной эффективности метода Штрассена к стандартному алгоритму

На графике 3 представлена зависимость относительной эффективности модифицированного стандартного алгоритма и стандартного алгоритма без модификаций от размерности матриц.

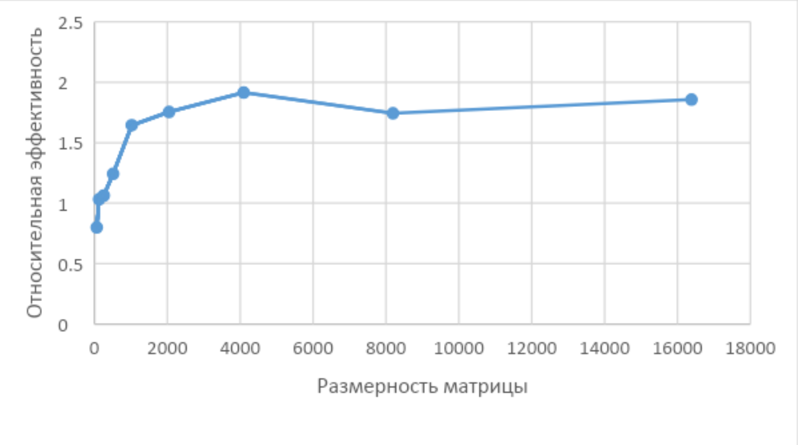

График 3 – график относительной эффективности модифицированного стандартного алгоритма к стандартному алгоритму

Заключение

В данной работе были исследованы способы ускорения нахождения произведения матриц. Наиболее эффективным методом для умножения матриц больших размеров является метод Штрассена. Однако стоит учитывать, что этот метод ограничен входными данными, поскольку он применим только для квадратных матриц, порядок которых равен степени числа 2. Для матриц, порядок которых меньше 1024, использование метода Штрассена нецелесообразно, так как затраты на хранение в памяти семи дополнительных матриц в каждом рекурсивном вызове превышают выигрыш во времени работы алгоритма.

Таким образом, получены следующие результаты:

1. Теоретически выведенная асимптотическая сложность алгоритма соответствует данным, полученным в ходе тестов.
2. Использование метода Штрассена становится более эффективным относительно стандартного алгоритма для матриц, размеры которых превышают 128.
3. Использование оптимизированных версий становится целесообразным для матриц, размерность, размеры которых превышают 256, но меньше 1024.
4. При увеличении размерности матрицы в два раза эффективность метода Штрассена относительно стандартного алгоритма возрастает в 1,14 раза относительно предыдущего значения.
5. При возведении матрицы в некоторую натуральную степень n применение метода Штрассена для умножения матриц кратно уменьшает затрачиваемое на операцию время.

Список использованных источников:

Левитин А. В. Алгоритмы: введение в разработку и анализ / А.В. Левитин. – М.: «Вильямс», 2006. – 576 с.
Алгоритмы: построение и анализ / Т. Кормен [и др.]; – 2-е издание. — М.: «Вильямс», 2005. – 1296 с.
Wolfram MathWorld [Electronic resource]: Strassen Formulas. – Mode of access: https://mathworld.wolfram.com/StrassenFormulas.html. – Date of access: 04.03.2024.

UDC 004.422.632:004.421–048:004

STUDY OF LINEAR ALGEBRA ALGORITHMS IN INFORMATION TECHNOLOGIES ON THE EXAMPLE OF MATRIX MULTIPLICATION ALGORITHMS.

Kamyshev S.V. [1], Petrochenko M.M. [1]

Belarusian State University of Informatics and Radioelectronics[1], Minsk, Republic of Belarus

Primicheva Z.N. – PhD in Physics and Mathematics

Annotation. The paper is dedicated to the study of different matrix multiplication methods, analyzing their efficiency, and the feasibility of applying different methods to matrices of different dimensions.

Keywords. Information technologies, matrices, matrix multiplication, algorithms, Strassen algorithm.