УДК 681.5

ПАРАЛЛЕЛЬНЫЙ РЕКУРСИВНЫЙ АЛГОРИТМ БЫСТРОГО МАТРИЧНОГО УМНОЖЕНИЯ

Л.П.Фельдман, И.А. Назарова, А.Е. Шматько
Донецкий национальный технический университет

В статті розглянуто паралельний алгоритм на основі рекурсивного алгоритму швидкого матричного множення. Розроблені обчислювальні схеми відображення методу на паралельні структури різної топології. Отримані порівняльні характеристики з традиційними алгоритмами, проведено чисельний експеримент.

Матричное умножение – это базовая операция линейной алгебры и доминирующая вычислительная часть многих научных приложений. Существует множество традиционных параллельных алгоритмов вычисления матричного произведения для плотнозаполненных матриц [1-2]. В этой статье анализируется эффективность параллельного алгоритма быстрого матричного умножения с использованием модификации рекурсивного алгоритма Штрассена-Винограда.

В оригинале алгоритм Штрассена-Винограда – это алгоритм умножения блочных матриц половинного размера, где каждый блок квадратный, т.е. размерности матриц должны быть четными числами. Метод Штрассена-Винограда состоит из 7 блочных умножений матриц и 15 блочных сложений\вычитаний матриц (рис. 1).

Идея Штрассена может быть применена рекурсивно для нахождения произведений блоков матриц $M_i, i = \overline{1,7}$. Если исходные матрицы A и B имели размерность $m \times m$, то алгоритм быстрого умножения можно использовать многократно, получая на самом нижнем уровне получим блоки 1×1. Однако нет никакой необходимости опускаться вниз до уровня блоков единичного порядка. При достаточно малых размерах блока ($k < k_{min}$) может оказаться выгодным вычислять блоки, используя стандартный алгоритм.

Вычислительная сложность предложенной схемы алгоритма быстрого умножения определяется функцией порядка исходных матриц и минимального порядка умножаемых блоков: m, m_{min}. Пусть при вычислении матричного умножения алгоритм Штрассена рекурсивно вызывался d раз, тогда порядок умножаемых матриц равен $m_{min} = m / 2^d$. На первом шаге алгоритм предусматривает 7 обращений

к самому себе с матрицами порядка $m/2$ и 15 операций типа сложение для матриц того же порядка. Далее идет развертка рекурсии до достижения минимального размера блока и умножение блоков по традиционному алгоритму МУ.

$$
\begin{array}{lll}
S_1 = A_{21} + A_{22} & M_1 = S_2 S_6 & T_1 = M_1 + M_3 \\
S_2 = S_1 - A_{11} & M_2 = A_{11} B_{11} & T_2 = T_1 + M_4 \\
S_3 = A_{21} - A_{12} & M_3 = A_{12} B_{21} & T_3 = M_5 + M_6 \\
S_4 = A_{12} - S_2 & M_4 = S_3 S_7 & C_{11} = M_2 + M_3 \\
S_5 = B_{12} - B_{11} & M_5 = S_1 S_5 & C_{12} = T_1 + T_3 \\
S_6 = B_{22} - S_5 & M_6 = S_4 B_{22} & C_{21} = T_2 - M_7 \\
S_7 = B_{22} - B_{12} & M_7 = A_{22} S_8 & C_{22} = T_2 + M_5 \\
S_8 = S_6 - B_{12} & &
\end{array}
$$

$$
A = \left\langle \frac{A_{11}}{A_{21}} \middle| \frac{A_{12}}{A_{22}} \right\rangle, B = \left\langle \frac{B_{11}}{B_{21}} \middle| \frac{B_{12}}{B_{22}} \right\rangle, C = \left\langle \frac{C_{11}}{C_{21}} \middle| \frac{C_{12}}{C_{22}} \right\rangle
$$

Рис. 1. Метод быстрого умножения матриц Штрассена-Винограда

Алгоритм быстрого умножения рекурсивен, поэтому имеется возможность построить полиалгоритм из некоторого традиционного алгоритма умножения матриц на верхнем уровне рекурсии и метода Штрассена на нижнем уровне, и наоборот. Алгоритм Штрассена является блочным, поэтому естественно комбинировать его со стандартным алгоритмом, также использующим блочное разбиение данных.

Пусть имеется мультикомпьютер из p^2 процессоров, объединенных коммуникационной сетью топологии тор. Исходные матрицы распределены по сетке процессоров на блоки $k \times k$, $k = m/p$, количество блоков равно $q^2 = p^2$. Построим полиалгоритм из блочного систолического алгоритма УМ между процессорами и серии применения метода Штрассена на каждом процессоре. Блоки исходных матриц и результата с координатами $< i, j >$ хранятся в соответствующем процессоре с теми же координатами. Предварительно по вычислительной схеме блочного систолического умножения выполняются:

1) $i \leftarrow A -$ косой сдвиг влево по строкам для блоков матрицы A;

2) $B \uparrow^j -$ косой сдвиг вверх по столбцам для блоков матрицы B.

На каждом из p шагов алгоритма производится умножение блоков матриц A и B, хранимых в процессоре с номером $<i,j>$ и сложение с уже вычисленным значением блока матрицы результата, расположенным на этом же процессоре C_{ij}. Для первого шага это значение равно нулевому блоку. Затем производится одиночный сдвиг влево по строкам параллельно для блоков матрицы $A: A_{ij} \leftarrow A_{i,j+1}$ и одиночный сдвиг вверх по столбцам для блоков матрицы $B: B_{ij} \leftarrow B_{i+1,j}$.Умножение блоков матриц выполняется внутри одного процессорного элемента, что позволяет избежать дополнительных пересылок данных, по рекурсивному алгоритму Штрассена. На нижнем уровне рекурсии применяется стандартный алгоритм умножения матриц, глубина рекурсии равна d. Разработанный алгоритм является масштабируемым для любого числа процессорных элементов и порядка исходных матриц.

Общее время выполнения арифметических операций для полиалгоритма равно:

$$T_{p,comp} = \left[2\left(\frac{7}{8}\right)^d \frac{m^3}{p^2} + \left(5\left(\frac{7}{4}\right)^d - 4 \right) \frac{m^2}{p} \right] t_{op}.$$

Время обменных операций для описанной схемы определяется, как и для блочного систолического алгоритма, и равно:

$$T_{p,comm} = 4(p-1) \cdot \left(t_s + \frac{m^2}{p^2} \cdot t_w \right).$$

Определение потенциальных и реальных характеристик параллелизма осуществлялось с помощью пакета *Mathematica®* (Wolfram Research Inc.)*:*

$$E = \frac{T_1}{T_p \cdot p^2}, T_1(m) = \left[2\left(\frac{7}{8}\right)^d m^3 + \frac{15}{4} m^2 \left(1 + \frac{7}{4} + \frac{7^2}{4^2} + \ldots + \frac{7^{d-1}}{4^{d-1}} \right) \right] t_{op}.$$

Очевидно, что динамические характеристики параллельных вычислительных схем МУ зависят от соотношения между числом процессоров и размерностью матриц. Для алгоритма Штрассена и полиалгоритмов на его основе существенным параметром является величина глубины рекурсии, d.

Анализ аналитических выражений, характеризующих выполнение параллельных алгоритмов, а также проведенный численный эксперимент, позволяют сделать следующие выводы:

1) предложенная параллельная вычислительная схема полиалгоритма на основе быстрого умножения Штрассена имеет лучшее время выполнения по сравнению с блочным систолическим алгоритмом в $(8/7)^d$ раз для матриц больших размерностей и невысокой глубине рекурсии;

2) высокий уровень рекурсии отрицательно сказывается, как на времени выполнения, так и на объеме используемой памяти.

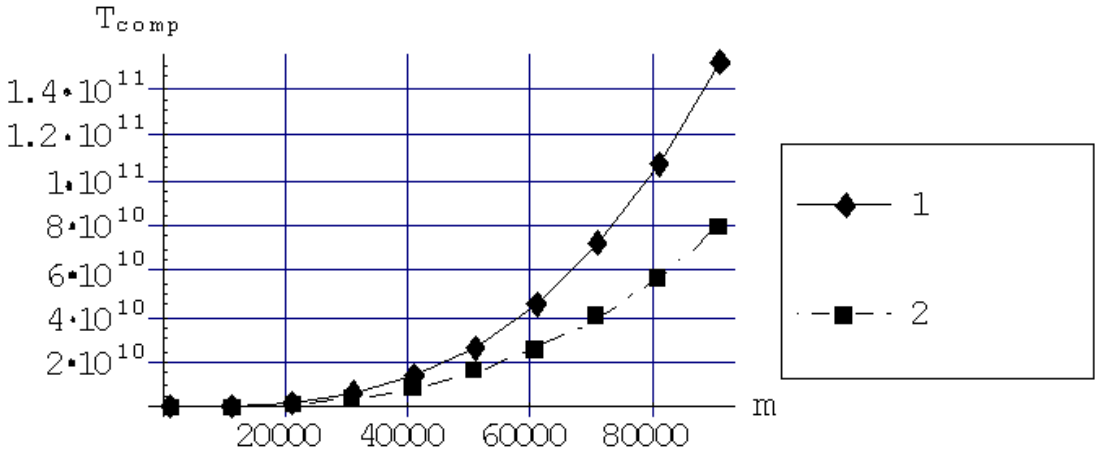

Рис. 2. График зависимости времени реализации параллельных алгоритмов от размерности матриц, $d = 4$
1)блочного систолического, 2) Штрассена+блочный систолический

Аналитические исследования и численный эксперимент, показали, что применение комбинации алгоритма Штрассена и блочного систолического умножения матриц эффективно для матриц больших размерностей и малой глубины рекурсии. Недостатками этого подхода являются большие затраты памяти и несколько худшая численная устойчивость, хотя и достаточная для большинства практических задач. Перспективным направлением дальнейших исследований является разработка, исследование эффективности, определение области применения полиалгоритмов метода Штрассена на основе других традиционных алгоритмов умножения матриц.

Литература

1.Деммель Дж. Вычислительная линейная алгебра. Теория и приложения. Пер. с англ. – М.: Мир, 2001. – 430с.

2. Голуб Д., Ван Лоун Ч. Матричные вычисления: Пер. с англ. – М.: Мир, 1999. – 548с.

Получено 20.05.07