УДК 681.322.012

**Л.Д. ЕЛФИМОВА**

# БЫСТРЫЙ КЛЕТОЧНЫЙ МЕТОД УМНОЖЕНИЯ МАТРИЦ

**Ключевые слова:** *клеточные методы линейной алгебры, алгоритмы матричного умножения, параллельные вычисления*.

Решение проблемы погружения алгоритмов произвольных размеров в архитектуру конкретных вычислительных систем основано на декомпозиции вычислительного процесса больших размеров на множество взаимосвязанных подпроцессов меньших размеров на различных уровнях представления вычислений. Потенциально наиболее перспективна декомпозиция вычислительного процесса на уровне исходных алгоритмов, в результате применения которой образуются клеточные алгоритмы, где базовыми операциями являются операции над клетками матриц.

Известны клеточные алгоритмы [1, 2], которые требуют выполнения только двух типов операций: базовой операции матричного умножения со сложением $AB+D=C$ (где $A$, $B$, $C$, $D$ — квадратные матрицы размера клетки) и нестандартных операций, составляющих малый процент общего числа операций алгоритма. Такие клеточные алгоритмы для решения практически всех задач линейной алгебры и многих задач вычислительной математики реализуются на различных вычислительных системах, а также в спецпроцессорах, включающих быстрый вычислитель для выполнения клеточной операции $AB+D=C$, которая может быть реализована с использованием известных быстрых алгоритмов матричного умножения [3–5].

Наиболее эффективна реализация базовой операции на СБИС-вычислителях, в частности процессорных массивах с систолической организацией вычислений. В настоящее время на основе интегрированного подхода к проектированию указанных массивов [6] разработаны СБИС-ориентированные версии быстрых алгоритмов Винограда [3], Штрассена [4], нового быстрого алгоритма [5], а также синтезированы быстрые систолические массивы различной архитектуры [6–8], эффективно реализующие перечисленные СБИС-алгоритмы матричного умножения.

Известны также клеточные алгоритмы [9–12], полученные заменой скалярных операций над числами операциями над клетками матриц, либо таким переупорядочиванием множества скалярных операций, которое позволяет выделить макрооперации над клетками матриц. Если алгоритм допускает свободу в выборе масштаба параллелизма, то переход к крупномасштабному параллелизму на уровне клеточных операций позволяет достичь более высокой степени распараллеливания вычислений, загруженности оборудования, сокращения издержек на обмены. Так, например, для реализации на параллельных вычислительных системах и преодоления рекурсивной природы быстрого алгоритма Штрассена [4] используются восходящая и нисходящая схемы вычислений, где исходные матрицы разбиваются на клетки [12]. В восходящей схеме вычислений в качестве внешнего алгоритма, оперирующего клетками, используется традиционный алгоритм матричного умножения [13], а в качестве внутреннего алгоритма, оперирующего числами, — алгоритм Штрассена. Нисходящая схема вычислений выполняется в обратном порядке: она использует алгоритм Штрассена как внешний алгоритм, а в качестве внутреннего применяется традиционный алгоритм.

В настоящей статье рассматривается клеточный метод умножения матриц, на базе которого получены клеточные аналоги известных алгоритмов матричного умножения с минимизированной на 12,5 % мультипликативной и аддитивной сложностью.

Исходные матрицы $A$ и $B$ порядка $n=2^{\gamma}$ $(\gamma>2)$, которые декомпозируются на клетки порядка $2^q$ $(q\le\gamma-2)$, представлены на рис. 1, где $m=n/2^q=2^{\gamma-q}$.

Внешний (клеточный) алгоритм оперирует матрицами порядка $2^q$. В качестве внутренних алгоритмов, оперирующих числами, могут быть использованы известные алгоритмы матричного умножения и сложения [3–5, 13].

$$\overset{A}{\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \iddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mm} \end{bmatrix}} \times \overset{B}{\begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1m} \\ B_{21} & B_{22} & \cdots & B_{2m} \\ \vdots & \vdots & \iddots & \vdots \\ B_{m1} & B_{m2} & \cdots & B_{mm} \end{bmatrix}} = \overset{C}{\begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1m} \\ C_{21} & C_{22} & \cdots & C_{2m} \\ \vdots & \vdots & \iddots & \vdots \\ C_{m1} & C_{m2} & \cdots & C_{mm} \end{bmatrix}}$$

*Рис. 1*

На первом этапе выполнения клеточного метода вычисляются матричные коэффициенты:

$$\begin{gathered} R_{ik}^{1}=(A_{2i-1,\,2k-1}+A_{2i,\,2k}),\ F_{kj}^{1}=(B_{2k-1,\,2j-1}+B_{2k,\,2j}), \\ R_{ik}^{2}=(A_{2i,\,2k-1}+A_{2i,\,2k}),\ F_{kj}^{2}=(B_{2k-1,\,2j}-B_{2k,\,2j}), \\ R_{ik}^{3}=(A_{2i-1,\,2k-1}+A_{2i-1,\,2k}),\ F_{kj}^{3}=(B_{2k,\,2j-1}-B_{2k-1,\,2j-1}), \\ R_{ik}^{4}=(A_{2i,\,2k-1}-A_{2i-1,\,2k-1}),\ F_{kj}^{4}=(B_{2k-1,\,2j-1}+B_{2k-1,\,2j}), \\ R_{ik}^{5}=(A_{2i-1,\,2k}-A_{2i,\,2k}),\ F_{kj}^{5}=(B_{2k,\,2j-1}+B_{2k,\,2j}), \end{gathered} \tag{1}$$

где $i, j, k=1, 2,\ldots, p;\ p=n/2^{q+1}=2^{\gamma-q-1}$.

На втором этапе выполняются регулярные вычисления матриц $Q_{ij}$:

$$\begin{gathered} Q_{ij}^{1}=\sum_{k=1}^{p}R_{ik}^{1}F_{kj}^{1},\quad Q_{ij}^{2}=\sum_{k=1}^{p}R_{ik}^{2}B_{2k-1,\,2j-1}, \\ Q_{ij}^{3}=\sum_{k=1}^{p}A_{2i-1,\,2k-1}F_{kj}^{2},\quad Q_{ij}^{4}=\sum_{k=1}^{p}A_{2i,\,2k}F_{kj}^{3},\ Q_{ij}^{5}=\sum_{k=1}^{p}R_{ik}^{3}B_{2k,\,2j}, \\ Q_{ij}^{6}=\sum_{k=1}^{p}R_{ik}^{4}F_{kj}^{4},\ Q_{ij}^{7}=\sum_{k=1}^{p}R_{ik}^{5}F_{kj}^{5},\ \text{где}\ i, j, k=1, 2,\ldots, p. \end{gathered} \tag{2}$$

На третьем (последнем) этапе вычисляются результирующие матрицы $C_{ij}$:

$$\begin{gathered} C_{2i-1,\,2j-1}=Q_{ij}^{1}+Q_{ij}^{4}-Q_{ij}^{5}+Q_{ij}^{7}, \\ C_{2i-1,\,2j}=Q_{ij}^{3}+Q_{ij}^{5}, \\ C_{2i,\,2j-1}=Q_{ij}^{2}+Q_{ij}^{4}, \\ C_{2i,\,2j}=Q_{ij}^{1}-Q_{ij}^{2}+Q_{ij}^{3}+Q_{ij}^{6},\ \text{где}\ i, j=1, 2,\ldots, p. \end{gathered} \tag{3}$$

Особенность предлагаемого клеточного метода заключается в том, что матричные коэффициенты $R_{ik}$ (соответственно $F_{kj}$) вычисляются только один раз и используются для всей матричной строки $i$ (соответственно матричного столбца $j$), состоящей из клеток порядка $2^{q+1}$. Поэтому при вычислении матричных коэффициентов (1) внешний алгоритм требует $10p^2$ матричных операций сложения. Для каждой матричной операции сложения внутренний алгоритм требует $2^{2q}$ скалярных операций сложения, так как порядок оперируемых матриц равен $2^q$. Следовательно, аддитивная сложность вычислений (1), определяемая количеством скалярных операций сложения, составляет

$$W_{\text{а}}^{(1)}=10p^2\cdot 2^{2q}=10\left(\frac{n}{2^{q+1}}\right)^2\cdot 2^{2q}=2{,}5n^2\ \ \text{(операций сложения).}$$

При вычислении матричных выражений (2) внешний алгоритм требует $7p^3$ матричных операций умножения и $7(p^3 - p^2)$ матричных операций сложения. Как было отмечено выше, каждое частичное произведение $2^q \times 2^q$ матриц (внутренний алгоритм) может вычисляться с использованием известных алгоритмов матричного умножения [3–5, 13]. Используя традиционный алгоритм матричного умножения [13], который требует $2^{3q}$ скалярных операций умножения и $(2^{3q} - 2^{2q})$ скалярных операций сложения, получим для мультипликативной сложности, определяемой количеством скалярных операций умножения, и аддитивной сложности вычислений (2) следующие выражения:

$$W_{\text{м}}^{(2)} = 7p^3 \cdot 2^{3q} = 7\left(\frac{n}{2^{q+1}}\right)^3 \cdot 2^{3q} = 0{,}875n^3 \text{ (операций умножения)},$$

$$W_{\text{а}}^{(2)} = 7[(p^3 - p^2)\cdot 2^{2q} + p^3(2^{3q} - 2^{2q})] =$$

$$= 7(p^3 \cdot 2^{3q} - p^2 \cdot 2^{2q}) = 0{,}875n^3 - 1{,}75n^2 \text{ (операций сложения)}.$$

При вычислении результирующей матрицы (3) внешний алгоритм требует $8p^2$ матричных операций сложения, а внутренний — $2^{2q}$ скалярных операций сложения. Следовательно, аддитивная сложность вычислений (3) составит

$$W_{\text{а}}^{(3)} = 8p^2 \cdot 2^{2q} = 8\left(\frac{n}{2^{q+1}}\right)^2 \cdot 2^{2q} = 2n^2 \text{ (операций сложения)}.$$

Таким образом, на основе предложенного метода (1)–(3) получаем клеточный аналог традиционного алгоритма со следующими значениями мультипликативной и аддитивной сложности:

$$W_{\text{м}} = 0{,}875n^3 \text{ (операций умножения)},$$

$$W_{\text{а}} = W_{\text{а}}^{(1)} + W_{\text{а}}^{(2)} + W_{\text{а}}^{(3)} = 0{,}875n^3 + 2{,}75n^2 \text{ (операций сложения)}.$$

Оценим операционную сложность вычислений (2) при условии, что в качестве внутреннего алгоритма будет использован самый быстрый алгоритм [5], требующий $\approx 0{,}437n^3$ операций умножения и $\approx 1{,}311n^3$ операций сложения. В этом случае мультипликативная и аддитивная сложности вычислений (2) составят

$$W_{\text{м}}^{(2)} \approx 7\left(\frac{n}{2^{q+1}}\right)^3 \cdot 0{,}437 \cdot 2^{3q} \approx 0{,}382\,n^3 \text{ (операций умножения)},$$

$$W_{\text{а}}^{(2)} \approx 7\left(\frac{n}{2^{q+1}}\right)^3 \cdot 1{,}311 \cdot 2^{3q} - 7\left(\frac{n}{2^{q+1}}\right)^2 \cdot 2^{2q} \approx$$

$$\approx 1{,}147n^3 - 1{,}75n^2 \text{ (операций сложения)}.$$

Следовательно, предложенный метод (1)–(3) минимизирует вычислительную сложность быстрого алгоритма [5] до следующих значений:

$$W_{\text{м}} \approx 0{,}382n^3 \text{ (операций умножения)},$$

$$W_{\text{а}} = W_{\text{а}}^{(1)} + W_{\text{а}}^{(2)} + W_{\text{а}}^{(3)} \approx 1{,}147n^3 + 2{,}75n^2 \text{ (операций сложения)}.$$

Подобным образом получаем клеточный аналог алгоритма Винограда [3] со следующими значениями вычислительной сложности:

$$W_{\text{м}} \approx 0{,}437n^3 \text{ (операций умножения)},$$

$$W_{\text{а}} \approx 1{,}311n^3 + 2{,}75n^2 \text{ (операций сложения)}.$$

При использовании алгоритма Штрассена [4] в качестве внутреннего алгоритма требуется $7^q$ скалярных операций умножения и $6(7^q - 2^{2q})$ скалярных операций сложения для каждой матричной операции умножения. Тогда клеточный аналог алгоритма Штрассена имеет следующие значения вычислительной сложности:

$$W_{\text{м}} = 7p^3 \cdot 7^q = 7\left(\frac{n}{2^{q+1}}\right)^3 \cdot 7^q = 0{,}875\left(\frac{7^q}{2^{3q}}\right)n^3 \quad \text{(операций умножения)},$$

$$W_{\text{а}} = 7p^3 \cdot 6(7^q - 2^{2q}) + 7(p^3 - p^2)\cdot 2^{2q} + 18p^2 \cdot 2^{2q} =$$

$$= 0{,}875\left(\frac{6\cdot 7^q - 5\cdot 2^{2q}}{2^{3q}}\right)n^3 + 2{,}75n^2 \quad \text{(операций сложения)},$$

$$W_{\text{общ.}} = 0{,}875\left(\frac{7^{q+1} - 5\cdot 2^{2q}}{2^{3q}}\right)n^3 + 2{,}75n^2 .$$

**Таблица 1**

| $q$ | 1 | 2 | 3 | 4 | 6 | 7 | 10 |
|---|---|---|---|---|---|---|---|
| $a(q)$ | 0,77 | 0,67 | 0,59 | 0,51 | 0,39 | 0,34 | 0,23 |

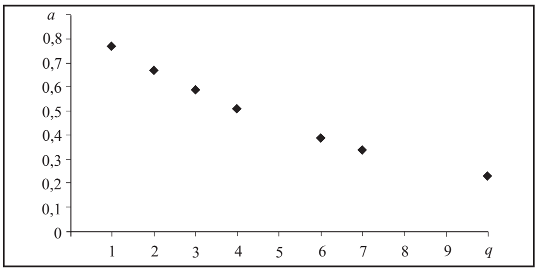


*Рис. 2*

**Таблица 2**

| $q$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $a(q)$ | 3,06 | 3,60 | 3,56 | 3,32 | 3,00 | 2,56 | 2,37 |

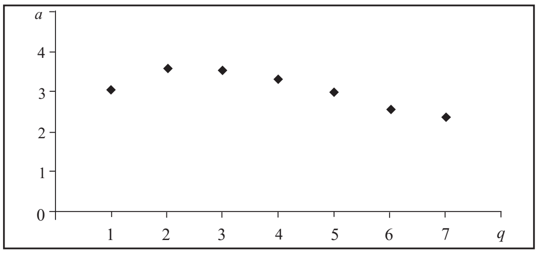


*Рис. 3*

На рис. 2 представлен график функции $a(q) = 0{,}875\left(\frac{7^q}{2^{3q}}\right)$ (построенный на основании табл. 1), для мультипликативной сложности полученного клеточного аналога при $q = 1, 2, \ldots, 10$. Из графика и табл. 1 следует, что число операций умножения нелинейно снижается с ростом $q$.

На рис. 3 представлен график функции $a(q) = 0{,}875\left(\frac{7^{q+1} - 5\cdot 2^{2q}}{2^{3q}}\right)$ для общих вычислительных затрат рассматриваемого клеточного аналога алгоритма Штрассена при $q = 1, 2, \ldots, 7$, построенный на основании табл. 2. Из графика следует, что предложенный клеточный метод умножения матриц уменьшает значение функции $a(q)$ во всем диапазоне изменения $q$ в 1,143 раза по сравнению с восходящей схемой вычислений [12], где в качестве внешнего алгоритма используется традиционный алгоритм матричного умножения, а в качестве внутреннего — алгоритм Штрассена.

Выбор оптимального значения величины $q$ определяется архитектурой конкретной параллельной вычислительной системы, реализующей алгоритм, и зависит от объема используемой памяти, а также требуемого времени реализации. Таким образом, предложенный в статье метод умножения матриц (1)–(3) приводит к образованию клеточных аналогов рассмотренных выше алгоритмов с минимизированной на 12,5 % вычислительной сложностью.

В табл. 3 приведена вычислительная сложность известных быстрых алгоритмов матричного умножения и полученного на основе предложенного метода кле-

**Т а б л и ц а 3**

| Быстрые алгоритмы для умножения матриц | Вычислительная сложность (число операций) | |
|---|---|---|
| | Мультипликативная $W_м$ | Аддитивная $W_а$ |
| Алгоритм Винограда [3] | $\approx 0{,}5n^3$ | $\approx 1{,}5n^3$ |
| Алгоритм Штрассена [4] | $n^{2,807}$ | $6(7^{\log_2 n} - 2^{2\log_2 n})$ |
| Алгоритм Елфимовой–Капитоновой [5] | $\approx 0{,}437n^3$ | $\approx 1{,}311n^3$ |
| Клеточный аналог алгоритма [5] (настоящая статья) | $\approx 0{,}382n^3$ | $\approx 1{,}147n^3$ |

точного аналога быстрого алгоритма, описанного в [5]. Представленный в таблице клеточный аналог характеризуется наименьшей операционной сложностью по сравнению с упомянутыми алгоритмами.

Клеточные аналоги алгоритмов матричного умножения могут быть эффективно реализованы в матричных машинах, параллельных вычислительных системах и СБИС-ориентированных процессорных массивах размера клетки.

СПИСОК ЛИТЕРАТУРЫ


1. В о е в о д и н А . В . О классе клеточных алгоритмов и его свойствах // Вопр. кибернетики. — 1988. — № 135. — С. 50–64.
2. Л ы с а н о в С . Ю . Клеточные методы решения задач линейной алгебры // Там же. — С. 64–73.
3. W i n o g r a d S . A . A new algorithm for inner product // IEEE Trans. Comp. — 1968. — **C-18.** — P. 693–694.
4. S t r a s s e n V . Gaussian elimination is not optimal // Numer. Math. — 1969. — **13.** — P. 354–356.
5. Е л ф и м о в а Л . Д . , К а п и т о н о в а Ю . В . Быстрый алгоритм для умножения матриц и его эффективная реализация на систолических массивах // Кибернетика и системный анализ. — 2001. — № 1. — С. 135–150.
6. Е л ф и м о в а Л . Д . , К а п и т о н о в а Ю . В . Интегрированный подход к проектированию процессорных массивов с систолической организацией вычислений // Там же. — 2002. — № 6. — С. 3–15.
7. J e l f i m o v a L . A new fast systolic array for modified Winograd algorithm // Proc. Sevens Int. Workshop on Parallel Processing by Cellular Automata and Array, PARCELLA-96 (Berlin, Germany, Sept. 1996). — Berlin: Akad. Verlag. — 1996. — **96.** — P. 157–164.
8. Д е к л а р а ц і й н и й патент на корисну модель № 14282, Україна, МПК G06F 15/00. Пристрій для множення матриць / Єлфімова Л.Д., Капітонова Ю.В. (Україна); Заявл. 24.10.2005; Опубл. 15.05.2006. Бюл. № 5.
9. В ы ш и н с к и й В . А . , Р а б и н о в и ч З . Л . О создании высокопроизводительных ЭВМ, работающих в алгебре матриц // Автоматика. — 1983. — № 5. — С. 80–84.
10. S c h r e i b e r R . Block algorithms for parallel machines // SIAM J. Sci. Stat. Comput. — 1987. — **8,** N 2. — P. 197–207.
11. M o d i J . J . Parallel algorithms and matrix computation. — Oxford: Clarendon Press, 1988. — 260 p.
12. F r a n c o m a n o E . , T o r t o r i c i - M a c a l u s o A . , V a j t e r s i c M . Implementation analysis of fast matrix multiplication algorithms on shared memory computers // Comp. & Artif. Intel. — 1995. — **14,** N 3. — P. 299–313.
13. Ф а д д е е в Д . К . , Ф а д д е е в а В . Н . Вычислительные методы линейной алгебры. — М.; Л.: Физматгиз, 1963. — 734 с.


*Поступила 05.10.2007*