«Автоматизация проектирования дискретных устройств».
Минск, 2004. Т. 2. С. 71-79.

УДК 519.712

ПОЛИНОМИАЛЬНАЯ ФАКТОРИЗАЦИЯ СПЕКТРАЛЬНЫХ БАЗИСОВ

В. С. Выхованец
Россия, Москва, Институт проблем управления РАН

Рассматривается проблема полиномиальной факторизации спектральных базисов. Под полиномиальной факторизацией понимается представление системы спектральных функций, заданных целочисленной матрицей дискретного преобразования, в виде кронекеровского произведения матриц меньшей размерности. Такое представление позволяет выразить систему спектральных функций в виде единой формулы в базисе бинарных операций. Дается теоретическое обоснование алгоритма полиномиальной факторизации матрицы, приводятся результаты его экспериментального исследования.

1. Введение

Традиционно под факторизацией спектрального базиса понимается представление системы спектральных функций, заданных матрицей прямого (обратного) преобразования в виде произведения слабозаполненных матриц меньшего порядка [1]. В результате факторизации получают процедуры быстрых преобразований, основанные на исключении операций умножения и сложения в тривиальных случаях, когда не выполняется умножение на ноль и единицу, сложение с нулем или возможно приведение подобных слагаемых на основе дистрибутивности умножения относительно сложения.

Уменьшение количества операций является первым и необходимым условием успешной факторизации. Второе условие связано с алгоритмом преобразования. В этом случае дополнительно требуется получение такого представления матрицы, при котором выполняемые операции имеют регулярное расположение в структуре алгоритма. Последнее позволяет не только сократить время вычислений, но и упростить сам алгоритм, повысить его комплексную эффективность.

Таким образом, при традиционном подходе к факторизации ставится задача получения эффективных процедур спектральных преобразований. При этом эффективность вычисления самих спектральных функций не принимается во внимание и предполагается, что функции уже вычислены и заданы в виде исходной матрицы дискретного преобразования.

2. Постановка задачи факторизации

Пусть задана матрица $\boldsymbol{D}$ – целочисленная матрица дискретного преобразования размерности $\boldsymbol{m}\times\boldsymbol{m}$. Известно, что столбцы матрицы $\boldsymbol{D}$ есть характеристические вектора спектральных функций θ_i. Выбор целочисленной матрицы в качестве исходного представления для системы спектральных функций определяется использованием вычислительных средств дискретного действия. Даже если элементы матрицы являются числами с плавающей запятой, то количество таких чисел ограничено, и их можно пронумеровать.

Для факторизации $\boldsymbol{D}$ найдем такие матрицы $\boldsymbol{P_j}$ и $\boldsymbol{O_t}$ ($\boldsymbol{j}=\overline{\boldsymbol{0},\boldsymbol{n}-\boldsymbol{1}}$, $\boldsymbol{t}=\overline{\boldsymbol{0},\boldsymbol{n}-\boldsymbol{2}}$), для которых с точностью до расстановки скобок выполняется равенство

$$\boldsymbol{D}=\boldsymbol{P_0}\otimes_{\boldsymbol{0}}\boldsymbol{P_1}\otimes_{\boldsymbol{1}}\ldots\otimes_{\boldsymbol{n-2}}\boldsymbol{P_{n-1}}=\bigotimes_{\boldsymbol{j=0}}^{\boldsymbol{n-1}}\boldsymbol{P_j}, \qquad (1)$$

где элементы и размерность $\boldsymbol{P_j}$ не превосходят некоторого числа $\boldsymbol{k_p}$ - значности факторизации, а кронекеровские произведения $\otimes_{\boldsymbol{t}}$ выполняются относительно бинарных операций, заданных матрицами $\boldsymbol{O_t}$, которые, в свою очередь, также имеют элементы и размерность, не превосходящие $\boldsymbol{k_p}$.

В результате представления (1) исходная матрица размерности $\boldsymbol{m}\times\boldsymbol{m}$ заменяется на $\boldsymbol{2n-1}$ матриц размерности $\boldsymbol{k_p}\times\boldsymbol{k_p}$, где $\boldsymbol{k_p}\approx\sqrt[\boldsymbol{n}]{\boldsymbol{m}}$. В итоге, объем памяти, необходимый для хранения матрицы дискретного преобразования $\boldsymbol{D}$ сокращается примерно в $\boldsymbol{2n}$ раз.

Известно [2], что матрица $\boldsymbol{D}$ представляется в виде (1), если спектральные функции, ее составляющие, являются полиномиальными. Аналитическая конструкция полиномиальных функций, задаваемая с точностью до нумерации переменных и расстановки скобок, имеет вид:

$$\boldsymbol{\theta_i(X)=x_0^{i_0}\circ_0 x_1^{i_1}\circ_1\ldots\circ_{n-2}x_{n-1}^{i_{n-1}}}\quad(\boldsymbol{i}=\overline{0,\boldsymbol{m}-1}) \qquad (2)$$

или, после подстановки $\boldsymbol{x_j^{i_j}=x_j\rhd_j i_j}$,

$$\boldsymbol{\theta_i(X)=(x_0\rhd_0 i_0)\circ_0(x_1\rhd_1 i_1)\circ_1\ldots\circ_{n-2}(x_{n-1}\rhd_{n-1}i_{n-1})} \qquad (3)$$

где $\boldsymbol{X}=\{\boldsymbol{x}_0,\boldsymbol{x}_1,\ldots,\boldsymbol{x}_{\boldsymbol{n}-1}\}$ – множество переменных со значностями $\boldsymbol{k}_0$, $\boldsymbol{k}_1$, …, $\boldsymbol{k}_{\boldsymbol{n}-1}$ соответственно, $\boldsymbol{m}=\boldsymbol{k}_0\boldsymbol{k}_1\ldots\boldsymbol{k}_{\boldsymbol{n}-1}$ – произведение значностей переменных или длина вектора полиномиальной функции, $\boldsymbol{i_j}$ – цифры индекса функции $\boldsymbol{i}$ в позиционной системе счисления с основаниями, задаваемыми значностями переменных, $\triangleright_{\boldsymbol{j}}$ – бинарные степенные операции (определяются матрицами $\boldsymbol{P_j}$), $\circ_{\boldsymbol{j}}$ – бинарные соединительные операции (определяются матрицами $\boldsymbol{Q_t}$).

Таким образом, представление (1) позволяет также получить и единую формулу для спектральных функций, которую легко отразить в структуру алгоритма вычисления матрицы $\boldsymbol{D}$ или отдельных ее элементов.

Поставим задачу поиска эффективных формульных представлений для системы спектральных функций, заданной в виде целочисленной матрицы. Если найдены операции полиномиальной аналитической конструкции, которые имеют значность не более чем $\boldsymbol{k_p}$, то факторизацию будем называть существующей (успешной), в противном случае – несуществующей. Число $\boldsymbol{k_p}$ назовем значностью полиномиальной факторизации матрицы.

3. Полиномиальная факторизация

Из (1) непосредственно следует рекуррентное правило для полиномиальной факторизации,

$$\boldsymbol{T}_0^* = \boldsymbol{D}, \quad \boldsymbol{T}_{\boldsymbol{t}+1}^* = \boldsymbol{T}_{\boldsymbol{t}}^{\boldsymbol{L}} \otimes_{\boldsymbol{t}} \boldsymbol{T}_{\boldsymbol{t}}^{\boldsymbol{R}} \quad (\boldsymbol{t} = 0, 1, \ldots), \quad \boldsymbol{P_j} = \boldsymbol{T}_{\boldsymbol{k}}^* \tag{4}$$

где $\boldsymbol{T}_{\boldsymbol{t}+1}^*$ – левая $\boldsymbol{T}_{\boldsymbol{t}+1}^{\boldsymbol{L}}$ или правая $\boldsymbol{T}_{\boldsymbol{t}+1}^{\boldsymbol{R}}$ промежуточная матрица, полученная на предыдущем шаге. Если размерность $\boldsymbol{T}_{\boldsymbol{k}}^*$ не превосходит $\boldsymbol{k_p}$, то матрица отождествляется с некоторой матрицей степенной операции $\boldsymbol{P_j}$.

Вычисления завершаются, когда отождествляются все промежуточные матрицы. Если $\boldsymbol{T}_{\boldsymbol{t}+1}^{\boldsymbol{L}}$ или $\boldsymbol{T}_{\boldsymbol{t}+1}^{\boldsymbol{R}}$ не отождествлена, то результат последующей факторизации этих матриц заключается в скобки. Заметим, что индексы степеней переменных $\boldsymbol{i_j}$ последовательно возрастают в порядке и направлении вычислений.

3.1. Полиномиальная декомпозиция матрицы

Базовой операцией в (4) является представление матрицы $\boldsymbol{T}$ размерности $\boldsymbol{m'} \times \boldsymbol{m''}$ в виде кронекеровского произведения матриц $\boldsymbol{L}$ и $\boldsymbol{R}$ относительно неизвестной операции $\circ$,

$$\boldsymbol{T} = \boldsymbol{L} \otimes_{\circ} \boldsymbol{R}. \tag{5}$$

Найдем для заданной матрицы $\boldsymbol{T}$ такие матрицы $\boldsymbol{L}$ и $\boldsymbol{R}$, а также операцию $\circ$, определяемую матрицей $\boldsymbol{O}$, которые обращают выражение (5) в равенство. Для начала определим $\boldsymbol{L}$, $\boldsymbol{R}$ и $\boldsymbol{O}$ следующим образом,

$$\begin{cases} \boldsymbol{L} = [\boldsymbol{l}_{uv}], & \boldsymbol{l}_{uv} = \boldsymbol{v} + \boldsymbol{u}\boldsymbol{k}_l'' \quad (\boldsymbol{u} = \overline{0, \boldsymbol{k}_l' - 1}, \quad \boldsymbol{v} = \overline{0, \boldsymbol{k}_l'' - 1}), \\ \boldsymbol{R} = [\boldsymbol{r}_{sw}], & \boldsymbol{r}_{sw} = \boldsymbol{w} + \boldsymbol{s}\boldsymbol{k}_r'' \quad (\boldsymbol{s} = \overline{0, \boldsymbol{k}_r' - 1}, \quad \boldsymbol{w} = \overline{0, \boldsymbol{k}_r'' - 1}), \\ \boldsymbol{O} = [\boldsymbol{o}_{pq}], & \boldsymbol{o}_{pq} = \boldsymbol{p} \circ \boldsymbol{q} \quad (\boldsymbol{p} = \overline{0, \boldsymbol{k}_l'\boldsymbol{k}_l'' - 1} \quad \boldsymbol{q} = \overline{0, \boldsymbol{k}_r'\boldsymbol{k}_r'' - 1}), \end{cases} \tag{6}$$

где $\boldsymbol{k}_l'$ и $\boldsymbol{k}_r'$ – делители $\boldsymbol{m}'$, а $\boldsymbol{k}_l''$ и $\boldsymbol{k}_r''$ – делители $\boldsymbol{m}''$, такие, что $\boldsymbol{m}' = \boldsymbol{k}_l'\boldsymbol{k}_r'$ и $\boldsymbol{m}'' = \boldsymbol{k}_l''\boldsymbol{k}_r''$.

Так как матрицы $\boldsymbol{L}$ и $\boldsymbol{R}$ заданы, нам остается найти только матрицу $\boldsymbol{O}$, такую, что выражение (5) обращается в равенство. Из определения кронекеровского произведения матриц и выражений (6) следует

$$\boldsymbol{t}_{ij} = \boldsymbol{l}_{i_l j_l} \circ \boldsymbol{r}_{i_r j_r} = (\boldsymbol{j}_l + \boldsymbol{i}_l \boldsymbol{k}_l'') \circ (\boldsymbol{j}_r + \boldsymbol{i}_r \boldsymbol{k}_r'') = \boldsymbol{o}_{pq}$$

где $\boldsymbol{t}_{ij}$ – элемент $\boldsymbol{T}$, $\boldsymbol{i} = (\boldsymbol{i}_l, \boldsymbol{i}_r)_{\boldsymbol{k}_l'\boldsymbol{k}_r'}$ и $\boldsymbol{j} = (\boldsymbol{j}_l, \boldsymbol{j}_r)_{\boldsymbol{k}_l''\boldsymbol{k}_r''}$ – представления индексов $\boldsymbol{i}$ и $\boldsymbol{j}$ в позиционной системе счисления с основаниями, задаваемыми делителями размерностей матрицы $\boldsymbol{T}$. Отсюда получаем выражение для вычисления соединительной операции $\circ$,

$$\boldsymbol{p} \circ \boldsymbol{q} = \boldsymbol{T}[(\boldsymbol{p}'', \boldsymbol{q}'')_{\boldsymbol{k}_l''\boldsymbol{k}_r''} \ (\boldsymbol{p}', \boldsymbol{q}')_{\boldsymbol{k}_l'\boldsymbol{k}_r'}] \tag{7}$$

где $\boldsymbol{p} = (\boldsymbol{p}'', \boldsymbol{p}')_{\boldsymbol{k}_l''\boldsymbol{k}_l'}$, $\boldsymbol{q} = (\boldsymbol{q}'', \boldsymbol{q}')_{\boldsymbol{k}_r''\boldsymbol{k}_r'}$, $\boldsymbol{T}[\boldsymbol{j} \ \boldsymbol{i}]$ – элемент $\boldsymbol{t}_{ij}$ матрицы $\boldsymbol{T}$.

Заметим, что формула (7) позволяет вычислить ранее неизвестную матрицу $\boldsymbol{O}$. Для этого операнды $\boldsymbol{p}$ и $\boldsymbol{q}$ операции $\circ$ представляются в позиционной системе счисления с основаниями, являющимися делителями чисел $\boldsymbol{m}' = \boldsymbol{k}_l'\boldsymbol{k}_r'$ и $\boldsymbol{m}'' = \boldsymbol{k}_l''\boldsymbol{k}_r''$ соответственно. Затем цифры $\boldsymbol{p}'$, $\boldsymbol{p}''$, $\boldsymbol{q}'$ и $\boldsymbol{q}''$ используются для вычисления в соответствии с (7) индексов элемента матрицы $\boldsymbol{T}$ – результата искомой операции.

В итоге для произвольной матрицы $\boldsymbol{T}$ получаем полностью определенные матрицы $\boldsymbol{L}$, $\boldsymbol{R}$ и $\boldsymbol{O}$, связанные уравнением (5). Очевидно, сделать это можно различными способами. Число решений уравнения (5) равно количеству нетривиальных разбиений текущего множества переменных на два непересекающихся подмножества (произведение значностей которых $\boldsymbol{k}_l'$ и $\boldsymbol{k}_r'$ равно $\boldsymbol{m}'$), умноженное на количество представлений числа $\boldsymbol{m}''$ в виде двух нетривиальных сомножителей ($\boldsymbol{k}_l''$ и $\boldsymbol{k}_r''$). Из всех решений уравнения (5) выбирается такое, у которого матрица соединительной операции $\boldsymbol{O}$ может быть редуцирована до размеров, не превосходящих значность факторизации $\boldsymbol{k}_p$.

«Автоматизация проектирования дискретных устройств».
Минск, 2004. Т. 2. С. 71-79.

3.2. Полиномиальная редукция

Следующим шагом полиномиальной факторизации является редукция матрицы $\boldsymbol{O}$ и соответствующая модификация матриц $\boldsymbol{L}$ и $\boldsymbol{R}$. Для этого строки и столбцы матрицы $\boldsymbol{O}$ разбиваются на классы эквивалентности. Сама матрица $\boldsymbol{O}$ преобразуется в вид, содержащий по одной строке (столбцу) их каждого класса.

Пусть $\{\boldsymbol{R}_0, \boldsymbol{R}_1, \ldots, \boldsymbol{R}_{\boldsymbol{n}_r - 1}\}$ и $\{\boldsymbol{C}_0, \boldsymbol{C}_1, \ldots, \boldsymbol{C}_{\boldsymbol{n}_c - 1}\}$ – классы эквивалентности строк и столбцов матрицы $\boldsymbol{O}$, содержащие их индексы в матрице $\boldsymbol{O}$. Редуцированную матрицу строим следующим образом. В качестве $\boldsymbol{i}$-ой строки новой матрицы $\boldsymbol{O}$ используем строку из класса $\boldsymbol{R}_{\boldsymbol{i}}$, а в качестве $\boldsymbol{j}$-го столбца – столбец из класса $\boldsymbol{C}_{\boldsymbol{j}}$.

Для сохранения равенства (5), все элементы $\boldsymbol{r} \in \boldsymbol{R}_{\boldsymbol{i}}$ левой матрицы $\boldsymbol{L}$ должны быть заменены $\boldsymbol{i}$ – индексом класса эквивалентности, которому принадлежит элемент $\boldsymbol{r}$. Аналогичным образом заменяются элементы правой матрицы. Все элементы $\boldsymbol{c} \in \boldsymbol{C}_{\boldsymbol{j}}$ матрицы $\boldsymbol{R}$ заменяются $\boldsymbol{j}$ – индексом класса эквивалентности, в который входит элемент $\boldsymbol{c}$.

В результате получаем редуцированную матрицу $\boldsymbol{O}$ и модифицированные матрицы $\boldsymbol{L}$ и $\boldsymbol{R}$. Если размерность матрицы $\boldsymbol{O}$ не превосходит $\boldsymbol{k}_{\boldsymbol{p}}$ по любому из измерений, элементы матриц $\boldsymbol{L}$ и $\boldsymbol{R}$ не превосходят $\boldsymbol{k}_{\boldsymbol{p}}$, то редукция считается успешной. В противном случае выбирается другое разбиение переменных на два непересекающиеся подмножества и процесс полиномиальной декомпозиция и редукции повторяется.

3.3. Пример полиномиальной факторизации

Произведем полиномиальную факторизацию упорядоченной системы спектральных функций, заданных матрицей

$$\boldsymbol{T} = \begin{bmatrix} 0 & 2 & 0 & 2 & 1 & 0 \\ 0 & 3 & 0 & 3 & 1 & 0 \\ 0 & 2 & 0 & 2 & 1 & 0 \\ 2 & 3 & 1 & 0 & 2 & 3 \\ 2 & 2 & 1 & 0 & 2 & 2 \\ 2 & 3 & 1 & 0 & 2 & 3 \end{bmatrix}$$

размерности 6×6 при $\boldsymbol{k}_{\boldsymbol{p}} = 3$.

Пусть $\boldsymbol{k}'_{\boldsymbol{l}} = 3$, $\boldsymbol{k}'_{\boldsymbol{r}} = 2$, $\boldsymbol{k}''_{\boldsymbol{l}} = 2$ и $\boldsymbol{k}''_{\boldsymbol{r}} = 3$. Тогда из (6) и (7) получим:

«Автоматизация проектирования дискретных устройств».
Минск, 2004. Т. 2. С. 71-79.

$$\boldsymbol{L}=\begin{bmatrix}0&1\\2&3\\4&5\end{bmatrix},\ \boldsymbol{O}=\begin{bmatrix}0&0&1&2&1&2\\2&2&0&3&0&3\\0&0&1&2&1&2\\3&3&0&2&0&2\\0&0&1&2&1&2\\2&2&0&3&0&3\end{bmatrix},\ \boldsymbol{R}=\begin{bmatrix}0&1&2\\3&4&5\end{bmatrix}.$$

В частности для элемента $\boldsymbol{o}_{53}$ матрицы $\boldsymbol{O}$ имеем

$$5\circ 3=(1,2)_{23}\circ(0,1)_{32}=\boldsymbol{T}[(1,0)_{23}\ (2,1)_{32}]=\boldsymbol{T}[1\ 5]=\boldsymbol{t}_{51}=3.$$

Далее находим классы эквивалентности строк и столбцов $\boldsymbol{O}$:

$$\boldsymbol{R}_0=\{0,2,4\},\ \boldsymbol{R}_1=\{1,5\},\ \boldsymbol{R}_2=\{3\},$$
$$\boldsymbol{C}_0=\{0,1\},\ \boldsymbol{C}_1=\{2,4\},\ \boldsymbol{C}_2=\{3,5\}.$$

После построения редуцированной матрицы соединительной операции и замены элементов левой и правой матриц на индексы классов, в которые они входят, получим

$$\boldsymbol{L}=\begin{bmatrix}0&1\\0&2\\0&1\end{bmatrix},\ \boldsymbol{O}=\begin{bmatrix}0&1&2\\2&0&3\\3&0&2\end{bmatrix},\ \boldsymbol{R}=\begin{bmatrix}0&0&1\\2&1&2\end{bmatrix}.$$

Таким образом $\boldsymbol{T}=\boldsymbol{L}\otimes_{\circ}\boldsymbol{R}$, где размерности матриц $\boldsymbol{L}$, $\boldsymbol{R}$ и $\boldsymbol{O}$ не больше чем $\boldsymbol{k}_{\boldsymbol{p}}$. Следовательно, система спектральных функций, заданная матрицей $\boldsymbol{T}$, может быть представлена формулой

$$\theta_{\boldsymbol{i}}(\boldsymbol{x}_0,\boldsymbol{x}_1)=\boldsymbol{x}_0^{\boldsymbol{i}_0}\circ\boldsymbol{x}_1^{\boldsymbol{i}_1}=(\boldsymbol{x}_0\rhd_{\boldsymbol{L}}\boldsymbol{i}_0)\circ(\boldsymbol{x}_1\rhd_{\boldsymbol{R}}\boldsymbol{i}_1)$$

где значность переменной $\boldsymbol{x}_0$ равна 3, $\boldsymbol{x}_1$ – 2, индексы спектральных функций представляются в системе счисления с основаниями 3 и 2, а степенные и соединительная операции задаются редуцированными матрицами $\boldsymbol{L}$, $\boldsymbol{R}$ и $\boldsymbol{O}$.

Из примера видно, что факторизация матриц может быть использована для минимизации или модификации операций полиномиальной формы, изменения значностей переменных, их порядка в выражении. В этом случае по известной формуле для полиномиальных функций вычисляется, а затем факторизуется матрица дискретного преобразования $\boldsymbol{D}$. В результате может быть получено другое формульное представление для той же системы спектральных функций, если, конечно, оно существует.

4. Алгоритм полиномиальной факторизации

Ниже приведен алгоритм полиномиальной факторизации системы спектральных функций, зависящих от переменных $\boldsymbol{X}$ и заданных квадратной матрицей $\boldsymbol{T}$. Алгоритм реализует частный случай факторизации, когда степенные операции ищутся в виде квадратных матриц.

```
scalar kp
stack powers, connectives
function pFormula(matrix T, set X) string
    scalar ind, var, kl, kr
    string Fl, Fr
    matrix Tl, To, Tr
    vector R, C
    if X.count = 1 then
        ind := powers.push(T)
        var := X(1)
        pFormula := "x_var ▷_ind i_ind"
    else
        X.firstSubset
        pFormula := empty
        do
            kl := X.implement.digits
            kr := X.complement.digits
            To := pMiddle(T, kl, kr)
            R := pRowClasses(To)
            C := pColumnClasses(To)
            if R.classes <= kp and C.classes <= kp
                Tl := pLeft(R)
                To := pOperation(To)
                Tr := pRight(C)
                Fl := pFormula(Tl, X.implement)
                Fr := pFormula(Tr, X.complement)
                if Fl <> empty and Fr <> empty
                    ind := connectives.push(To)
                    pFormula := "("+Fl+"∘_ind"+Fr+")"
                end if
                exit do
            end if
        while X.nextSubset
    end if
end function
```

Функция `pFormula` является рекурсивной и возвращает формулу для системы полиномиальных функций `T`. Полиномиальная декомпозиция матрицы `T` осуществляется функцией `pMiddle`. Редукция матрицы `To` реализуется функцией `pOperation`.

Алгоритм использует единственность полиномиальной декомпозиции матрицы. Оператор **`exit do`** выполняет выход из цикла по подмножествам переменных `X`, если матрица соединительной операции `To` имеет успешную редукцию, причем независимо от того, факторизуются ли матрицы `Tl` и `Tr` или нет. Приведем без доказательства теорему о единственности полиномиального представления, лежащую в основе алгоритма.

Теорема. Пусть $\boldsymbol{T} = \boldsymbol{L} \otimes_{\circ} \boldsymbol{R}$ и известно, что строки (столбцы) $\boldsymbol{T}$ попарно различны. Тогда не существует операции $\bullet$ и матриц $\widetilde{\boldsymbol{L}}$ и $\widetilde{\boldsymbol{R}}$ с размерностями, отличными от размерностей $\boldsymbol{L}$ и $\boldsymbol{R}$, таких, что $\boldsymbol{T} = \widetilde{\boldsymbol{L}} \otimes_{\bullet} \widetilde{\boldsymbol{R}}$, где значности операндов $\bullet$ не превосходят значности операндов $\circ$.

Экспериментальные результаты исследования алгоритма полиномиальной факторизации представлены на рисунке, где показана зависимость времени факторизации $\boldsymbol{t}$ от размерности матриц дискретного преобразования $\boldsymbol{m}$.

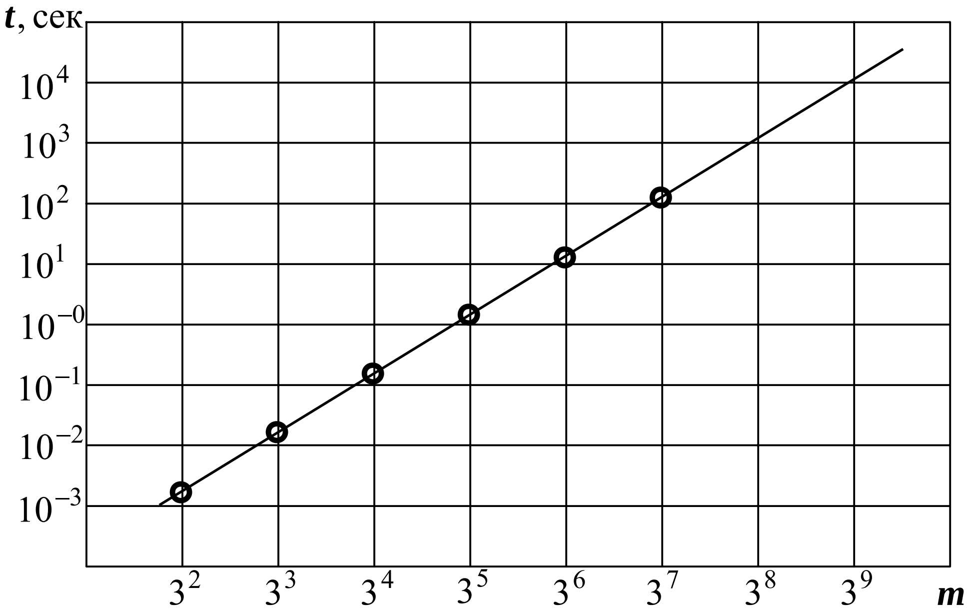

Рис. Зависимость времени факторизации от размерности матрицы

Литература

1. Ахмед Н., Рао К. Р. Ортогональные преобразования при обработке цифровых сигналов. - М.: Связь, 1980.

2. Выхованец В.С. Спектральные методы в логической обработке данных // Автоматика и телемеханика. - 2001. - № 10. - С. 28-53.