

МАТЭМАТЫКА, ФІЗІКА, БІЯЛОГІЯ

УДК 517.4

БЫСТРОЕ ПРЕОБРАЗОВАНИЕ ФУРЬЕ В ПОДСЧЕТЕ КОЛИЧЕСТВА S_n^2 -ОРБИТ КЭМЕРОНОВСКИХ МАТРИЦ

В. А. Липницкий

доктор технических наук, профессор
Военная академия Республики Беларусь

А. И. СЕРГЕЙ

аспирант

Гродненский государственный университет имени Я. Купалы

Н. В. СПИЧЕКОВА

кандидат физико-математических наук, доцент

Белорусский государственный университет информатики и радиоэлектроники

В данной работе предлагается алгоритм, основанный на быстром умножении многочленов, для подсчета количества орбит, на которые разбивается множество P_n квадратных $(0,1)$ -матриц под действием квадрата S_n^2 симметрической группы S_n . Рассматриваемый алгоритм имеет вычислительную сложность $O(p(n)n^{2.5} \log n)$, где $p(n)$ – количество неупорядоченных разбиений числа n . Наилучший алгоритм, известный до публикации данной работы, имеет вычислительную сложность $O(p(n)n^4)$.

Ключевые слова: бинарная матрица, симметрическая группа, орбита, мощность орбиты, третья проблема Питера Кэмерона, лемма Бёрнсайда, цикленный тип подстановки, дискретное преобразование Фурье, быстрое умножение многочленов.

Введение

В данной работе рассматривается третья проблема Питера Кэмерона [1], которая формулируется следующим образом: *найти общую формулу или алгоритм вычисления количества орбит α_n , на которые разбивается множество P_n всех бинарных $(0,1)$ -матриц, содержащих в точности n единиц, под действием квадрата $G = S_n^2 = S_n \times S_n$ симметрической группы S_n .*

Группа G задает перестановки строк и столбцов матриц из множества P_n .

Данная работа является продолжением работ авторов [2] и [3]. Подробный обзор литературы по изучаемой теме приводится в [2] и [3]. В [2] также описывается алгоритм вычисления α_n , имеющий сложность $O(p^2(n)n^2 \log(n))$, где $p(n)$ – количество неупорядоченных разбиений числа n . Предлагаемый алгоритм основан на лемме Бёрнсайда и линейной развертке бинарной матрицы и реализует идеи метода динамического программирования. В [3] приводится усовершенствованный алгоритм вычисления α_n , имеющий сложность $O(p(n)n^4)$. Целью данной работы является применение быстрого умножения многочленов при помощи преобразования Фурье для построения алгоритма вычисления α_n , который имеет меньшую вычислительную сложность по сравнению с алгоритмами в [2] и [3].

© Липницкий В. А., 2019

© Сергей А. И., 2019

© Спичекова Н. В., 2019

Основная часть

Быстрое преобразование Фурье. Пусть $\omega_n^0, \omega_n^1, \dots, \omega_n^{n-1}$ – комплексные корни n -ой степени из единицы, т. е. $\omega_n^k = e^{2\pi k/n}$, $k = 0, 1, \dots, n-1$.

Пусть имеется многочлен $A(x) = \sum_{j=0}^{n-1} a_j x^j$. Будем предполагать, что n является степенью 2. Это требование всегда можно удовлетворить, добавив равные нулю старшие коэффициенты. Обозначим

$$y_k = A(\omega_n^k) = \sum_{j=0}^{n-1} a_j \omega_n^{kj}. \quad (1)$$

Вектор $y = (y_0, y_1, \dots, y_{n-1})$ представляет собой дискретное преобразование Фурье (ДПФ) вектора коэффициентов $a = (a_0, a_1, \dots, a_{n-1})$. Будем также писать $y = \text{ДПФ}_n(a)$. ДПФ переходит от коэффициентов многочлена к значениям многочлена в комплексных корнях n -ой степени из 1.

Быстрое преобразование Фурье (БПФ) – это метод, основанный на использовании специальных свойств комплексных корней из единицы, позволяющий находить ДПФ a и имеющий сложность $O(n \log n)$. Суть этого метода кратко заключается в следующем.

На основании многочлена $A(x)$ построим два новых полинома степени $\frac{n}{2} - 1$:

$$A^{[0]}(x) = a_0 + a_2 x + a_4 x^2 + \dots + a_{n-2} x^{n/2-1},$$

$$A^{[1]}(x) = a_1 + a_3 x + a_5 x^2 + \dots + a_{n-1} x^{n/2-1}.$$

$A^{[0]}(x)$ содержит все коэффициенты полинома $A(x)$ с четными индексами, $A^{[1]}(x)$ – все коэффициенты с нечетными индексами. Очевидно, что выполняется равенство

$$A(x) = A^{[0]}(x^2) + x A^{[1]}(x^2). \quad (2)$$

Задача вычисления $A(x)$ в точках $\omega_n^0, \omega_n^1, \dots, \omega_n^{n-1}$ сводится к следующим двум задачам:

1) вычислить полиномы $A^{[0]}(x)$ и $A^{[1]}(x)$ степеней $\frac{n}{2} - 1$ в точках $(\omega_n^0)^2, (\omega_n^1)^2, \dots, (\omega_n^{n-1})^2$; (3)

2) объединить полученные результаты в соответствии с формулой (2).

Список (3) содержит не n различных значений, а только $\frac{n}{2}$ комплексных корней степени $\frac{n}{2}$ из единицы, причем каждый корень встречается в списке ровно дважды. Поэтому полиномы $A^{[0]}(x)$ и $A^{[1]}(x)$ рекурсивно вычисляются в $\frac{n}{2}$ комплексных корнях $\frac{n}{2}$ -ой степени из единицы. Эти подзадачи имеют точно такой вид, как и исходная задача, но их размерность вдвое меньше, т. е. вычисление n -элементного ДПФ сводится к вычислению двух $\frac{n}{2}$ -элементных ДПФ. Такая декомпозиция позволяет реализовать [4, с. 953] рекурсивный алгоритм БПФ, который вычисляет ДПФ a и имеет сложность $O(n \log n)$.

Обратным дискретным преобразованием Фурье (ОДПФ) для вектора $y = (y_0, y_1, \dots, y_{n-1})$ значений многочлена $A(x)$ в точках $\omega_n^0, \omega_n^1, \dots, \omega_n^{n-1}$ называется вектор $a = (a_0, a_1, \dots, a_{n-1})$ коэффициентов этого многочлена, обозначается $a = \text{ОДПФ}_n(y)$.

ОДПФ восстанавливает коэффициенты многочлена по значениям этого многочлена в точках – комплексных корнях n -ой степени из 1.

ОДФП $_n(y)$ вычисляется [4, с. 956] по формуле

$$a_j = \frac{1}{n} \sum_{k=0}^{n-1} y_k \omega_n^{-kj}, \quad j = 0, 1, \dots, n-1. \quad (4)$$

Так как формулы (1) и (4) похожи, то коэффициенты a_j могут быть найдены таким же алгоритмом, как и описанный выше алгоритм вычисления ДФП $_n(a)$, только вместо ω_n^k нужно использовать ω_n^{-k} и каждый элемент результата разделить на n . Следовательно, алгоритм вычисления ОДФП $_n(y)$ имеет сложность $O(n \log n)$.

Пусть требуется вычислить произведение полиномов $A(x) = \sum_{j=0}^{n-1} a_j x^j$ и $B(x) = \sum_{j=0}^{k-1} b_j x^j$.

Можно полагать, что $k = n$, так как в противном случае в многочлен меньшей степени можно добавить равные нулю старшие коэффициенты. Так как в каждой точке x значение полинома AB равно произведению величин $A(x)$ и $B(x)$, то ДФП $_n(c)$, где $c = (c_0, c_1, \dots, c_{2n-2})$ – вектор, составленный из коэффициентов многочлена $A(x)B(x)$, равно покомпонентному произведению векторов ДФП $_n(a)$ и ДФП $_n(b)$, где $a = (a_0, a_1, \dots, a_{n-1}, 0, \dots, 0)$ и $b = (b_0, b_1, \dots, b_{n-1}, 0, \dots, 0)$ – вектора длины $2n-1$. Для нахождения вектора c достаточно вычислить ОДФП(ДФП $_n(c)$). Из вышесказанного следует, что предлагаемый алгоритм вычисления вектора c будет иметь сложность $O(n \log n)$.

Краткое изложение полученных ранее результатов. В данном пункте приведем основные обозначения и результаты из [3], используемые в дальнейшем.

Через $P_{i,j,k}$ будем обозначать множество бинарных матриц размера $i \times j$, которые содержат в точности k единиц.

Пусть $h = (h_1, h_2) \in S_i \times S_j$, $h_1 = C_1^1 C_2^1 \dots C_s^1$, $h_2 = C_1^2$ – разложения h_1 и h_2 в произведение независимых циклов, содержащие в том числе и циклы длины 1, причем порядок следования циклов в разложении подстановки h_1 фиксирован. Через $t_{h,i,j,k}$ будем обозначать количество матриц из множества $P_{i,j,k}$ являющихся неподвижными точками для подстановки h , т. е. $t_{h,i,j,k} = |\text{Inv}(h)|$. Полагаем, что $t_{h,0,j,0} = 1$. Значения $t_{h,i,j,k}$ могут быть найдены по следующей рекуррентной формуле:

$$t_{h,i,j,k} = \sum_{\substack{l \\ k \geq l \cdot \text{НОК}(|C_2^1|, |C_1^2|)}} C_l^1 \text{НОД}(|C_2^1|, |C_1^2|) t_{\tilde{h}, i-|C_2^1|, j, k-l \cdot \text{НОК}(|C_2^1|, |C_1^2|)}, \quad (5)$$

где $|C_u^v|$ – длина цикла C_u^v , $\tilde{h} = (\tilde{h}_1, h_2) \in S_{i-|C_2^1|} \times S_j$, $\tilde{h}_1 = C_1^1 C_2^1 \dots C_{s-1}^1$.

Через $P_n(i, j)$ будем обозначать множество бинарных матриц размера $n \times i$, которые содержат в точности j единиц.

Зафиксируем натуральное число $k \leq i$, подстановку $g \in S_n$ и порядок следования множителей в ее разложении $g = C_1^g C_2^g \dots C_v^g$ в произведение независимых циклов.

Рассмотрим множество $H_{g,i,k} = \{(g, h_k) | h_k \in S_i\} \subset G_i = S_n \times S_i$, где h_k из S_i удовлетворяет следующему условию: в разложении h_k в произведение независимых циклов число i входит в цикл длины k .

Пусть $h_{g,i,k} = (g, h_k) \in H_{g,i,k}$ и $h_k = C_1^{h_k} C_2^{h_k} \dots C_{\mu_{h_k}}^{h_k}$ (6)

– это разложения подстановки $h_k \in S_i$ в произведение независимых циклов, содержащее в том числе и все циклы длины 1. Далее будем считать, что в разложении (6) подстановки h_k множители упорядочены так, что число i входит в цикл $C_1^{h_k}$ длины k , т. е. $C_1^{h_k} = (h_1^{h_k}, h_2^{h_k}, \dots, h_{k-1}^{h_k}, i)$, где $h_1^{h_k}, h_2^{h_k}, \dots, h_{k-1}^{h_k}, i$ – некоторые натуральные числа, не превосходящие i . В дальнейшем элемент $h_{g,i,k}$ также будем записывать в виде $(C_1^g C_2^g \dots C_v^g, C_1^{h_k} C_2^{h_k} \dots C_{\mu_{h_k}}^{h_k})$.

Пусть $f_{g,i,j} = \sum_{h_{g,i,k} \in \bigcup_{k=1}^i H_{g,i,k}} |\text{Inv}(h_{g,i,k})|$. Полагаем, что $f_{g,0,0} = 1$. $f_{g,i,j}$ равно числу матриц

из множества $P_n(i, j)$, которые являются неподвижными точками для подстановок из множества

$$H_{g,i} = \bigcup_{k=1}^i H_{g,i,k}. \tag{7}$$

$f_{g,i,j}$ может быть найдено по следующей рекуррентной формуле:

$$f_{g,i,j} = \sum_{k=1}^i \sum_{l=0}^j A_{i-1}^{k-1} f_{g,i-k,j-l} t_{\tilde{g},n,k,l}, \tag{8}$$

где $t_{\tilde{g},n,k,l}$ равно количеству матриц из множества $P_{n,k,l}$, являющихся неподвижными точками для подстановки $\tilde{g} = (g, C^{h_k}) = (C_1^g C_2^g \dots C_v^g, C^{h_k}) \in S_n \times S_k$, где $C^{h_k} = (1, 2, \dots, k-1, k)$ – цикл длины k , и может быть найдено по формуле (5).

Пусть p_i – это одно из разбиений числа n . p_i задает цикленный тип подстановки $g_i \in S_n$. Зная цикленный тип подстановки g_i по формуле (8) можно вычислить $f_{g_i,n,n}$. Пусть k_{p_i} – это количество подстановок множества S_n , имеющих такой же цикленный тип, как и подстановка g_i . В соответствии с предложением 7 из [2] если в подстановке g_i имеется c_i циклов длины $l_i, i = \overline{1, k}$, то $k_{p_i} = n! \prod_{i=1}^k (c_i! l_i^{c_i})^{-1}$. Тогда формулу для числа α_n орбит множества P_n можно записать в следующем виде:

$$\alpha_n = \frac{1}{(n!)^2} \sum_{i=1}^{p(n)} f_{g_i,n,n} k_{p_i}. \tag{9}$$

Вычисление $t_{g,i,j,k}$. Выведем еще одну формулу для вычисления $t_{g,i,j,k}$.

Пусть $g = (g_1, g_2) \in S_i \times S_j$, $g_1 = C_{11}^1 C_{21}^1 \dots C_{p_1 1}^1 C_{12}^1 C_{22}^1 \dots C_{p_2 2}^1 \dots C_{1s}^1 C_{2s}^1 \dots C_{p_s s}^1$, $g_2 = C_1^2$ – разложения g_1 и g_2 в произведение независимых циклов, содержащие в том числе и циклы длиной 1. Будем считать, что все циклы $C_{1u}^1, C_{2u}^1, \dots, C_{p_u u}^1$ из разложения g_1 имеют одну и ту же длину, а длины циклов $C_{p_u u}^1$ и $C_{p_v v}^1$ различны для $u \neq v$. Зафиксируем порядок следования циклов в разложении подстановки g_1 .

Предложение 1. Справедлива следующая рекуррентная формула:

$$t_{g,i,j,k} = \sum_{l=0}^L C_{p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|)}^l t_{\tilde{g}, i-p_l, |C_{p_s s}^1|, j, k-l, \text{НОК}(|C_{p_s s}^1|, |C_1^2|)}, \tag{10}$$

где $L = \min \left\{ p_s \text{НОД}(|C_{p_s s}^1|, |C_{p_s s}^1|), \left\lceil \frac{k}{\text{НОК}(|C_{p_s s}^1|, |C_1^2|)} \right\rceil \right\}$ (здесь и далее $[]$ означают целую

часть), $|C_u^v|$ – длина цикла C_u^v , $\tilde{g} = (\tilde{g}_1, g_2) \in S_{i-p_s|C_{p_s^s}^1|} \times S_j$,

$$\tilde{g}_1 = C_{11}^1 C_{21}^1 \dots C_{p_1 1}^1 C_{12}^1 C_{22}^1 \dots C_{p_2 2}^1 \dots C_{1(s-1)}^1 C_{2(s-1)}^1 \dots C_{p_{s-1}(s-1)}^1.$$

Доказательство во многом повторяет доказательство предложения 3 из [3].

Пусть

$$h(g) = C_1 C_2 \dots C_\alpha \quad (11)$$

– разложение в произведение циклов матричной подстановки, построенной по элементу g , содержащее в том числе и все циклы длиной 1. Пусть матрица $A \in \text{Inv}(g)$. Из предложения 1 из [3] следует, что элементы матрицы A , соответствующие любому из циклов разложения (11), равны 0 или 1. Индексы элементов матрицы A , которые стоят на пересечении строк и столбцов, входящих в циклы $C_{1s}^1, C_{2s}^1, \dots, C_{p_s s}^1$ и C_1^2 , в соответствии с предложением 2 из [3], образуют p_s НОД $(|C_{p_s s}^1|, |C_1^2|)$ циклов разложения (11), каждый цикл имеет длину $\text{НОК}(|C_{p_s s}^1|, |C_1^2|)$. Пусть элементы матрицы A , соответствующие l из этих циклов, $l \leq p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|)$, равны 1. Существует $C_{p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|)}^l$ способов выбрать эти l циклов.

Из циклов разложения (11) удалим циклы, которые соответствуют элементам матрицы A , расположенным на пересечении строк и столбцов из $C_{1s}^1, C_{2s}^1, \dots, C_{p_s s}^1$ и C_1^2 .

Оставшиеся циклы $C_{\alpha_1}, C_{\alpha_2}, \dots, C_{\alpha_k}$ образуют подстановку $\tilde{g} = (\tilde{g}_1, g_2) \in S_{i-p_s|C_{p_s^s}^1|} \times S_j$, $\tilde{g}_1 = C_{11}^1 C_{21}^1 \dots C_{p_1 1}^1 \dots C_{1(s-1)}^1 C_{2(s-1)}^1 \dots C_{p_{s-1}(s-1)}^1$. Среди элементов матрицы A , соответствующих этим циклам, имеется $k-l \cdot \text{НОК}(|C_{p_s s}^1|, |C_1^2|)$ единиц. Понятно, что должно выполняться неравенство $k-l \cdot \text{НОК}(|C_{p_s s}^1|, |C_1^2|)$. Существует $t_{\tilde{g}, i-p_s|C_{p_s^s}^1|, j, k-l \cdot \text{НОК}(|C_{p_s s}^1|, |C_1^2|)}$ способов выбрать из $C_{\alpha_1}, C_{\alpha_2}, \dots, C_{\alpha_k}$ циклы $C_{\beta_1}, C_{\beta_2}, \dots, C_{\beta_k}$ так, чтобы их суммарная длина была равна $k-l \cdot \text{НОК}(|C_{p_s s}^1|, |C_1^2|)$.

Так как выбор l циклов, соответствующих элементам матрицы A , стоящим на пересечении строк и столбцов, входящих в $C_{1s}^1, C_{2s}^1, \dots, C_{p_s s}^1$ и C_1^2 , и циклов $C_{\beta_1}, C_{\beta_2}, \dots, C_{\beta_k}$ не зависит друг от друга, то существует $A_l = C_{p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|)}^l t_{\tilde{g}, i-p_s|C_{p_s^s}^1|, j, k-l \cdot \text{НОК}(|C_{p_s s}^1|, |C_1^2|)}$ вариантов такого выбора. Для нахождения $t_{g, i, j, k}$ необходимо просуммировать A_l по всем l таким, что $0 \leq l \leq L$, $L = \min \left\{ p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|), \left\lceil \frac{k}{\text{НОК}(|C_{p_s s}^1|, |C_1^2|)} \right\rceil \right\}$. Доказательство завершено.

Предложение 2. Все коэффициенты $t_{g, n, i, j}$ для $i \leq n, j \leq n$ могут быть найдены за $O(n^{2.5} \log n)$ операций.

Доказательство. Зафиксируем величину i и введем обозначение $\Phi_{g, k, j} = t_{g, k, i, j}$. Из формулы (10) следует, что

$$\Phi_{g, k, j} = \sum_{l=0}^L C_{p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|)}^l \Phi_{\tilde{g}, k-p_s|C_{p_s^s}^1|, j-l \cdot \text{НОК}(|C_{p_s s}^1|, |C_1^2|)}, \quad (12)$$

где $L = \min \left\{ p_s \text{НОД}(|C_{p_s s}^1|, |C_1^2|), \left\lceil \frac{j}{\text{НОК}(|C_{p_s s}^1|, |C_1^2|)} \right\rceil \right\}$, $|C_u^v|$ – длина цикла C_u^v ,

$$\tilde{g} = (\tilde{g}_1, g_2) \in S_{k-p_s|C_{p_s^s}^1|} \times S_j, \quad \tilde{g}_1 = C_{11}^1 C_{21}^1 \dots C_{p_1 1}^1 \dots C_{1(s-1)}^1 C_{2(s-1)}^1 \dots C_{p_{s-1}(s-1)}^1.$$

Пусть $\Phi_{g,k}(x) = \sum_{j=0}^n \varphi_{g,k,j} x^j$. Из (12) следует, что

$$\Phi_{g,k}(x) = \sum_{j=0}^n \left(\sum_{l=0}^L C_{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)}^l \Phi_{\tilde{g}, k-p_s |C_{p_s^1}|, j-l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right) x^j.$$

Пусть запись $\text{trunc}_n(f(x))$ означает отбрасывание всех слагаемых многочлена $f(x)$ степени, большей n . С учетом формулы (12) получаем, что

$$\begin{aligned} & \text{trunc}_n \left(\Phi_{\tilde{g}, k-p_s |C_{p_s^1}|}(x) \sum_{l=0}^{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)} C_{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)}^l x^{l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right) = \\ & = \text{trunc}_n \left(\left(\sum_{t=0}^n \varphi_{\tilde{g}, k-p_s |C_{p_s^1}|, t} x^t \right) \left(\sum_{l=0}^{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)} C_{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)}^l x^{l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right) \right) = \\ & = \text{trunc}_n \left(\sum_{t=0}^n \left(\sum_{l=0}^{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)} C_{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)}^l \varphi_{\tilde{g}, k-p_s |C_{p_s^1}|, t+l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right) x^{t+l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right) = \\ & = \sum_{j=0}^n \left(\sum_{l=0}^L C_{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)}^l \varphi_{\tilde{g}, k-p_s |C_{p_s^1}|, j-l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right) x^j. \end{aligned}$$

Последнее равенство получено после выполнения замены $t+l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|) = j$. Следовательно,

$$\Phi_{g,k}(x) = \text{trunc}_n \left(\Phi_{\tilde{g}, k-p_s |C_{p_s^1}|}(x) \sum_{l=0}^{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)} C_{p_s \text{НОД}(|C_{p_s^1}|, |C_1^2|)}^l x^{l \cdot \text{НОД}(|C_{p_s^1}|, |C_1^2|)} \right). \quad (13)$$

Для вычисления $\Phi_{g,n}(x)$ при фиксированном i необходимо s раз применить формулу (13). В соответствии с введенными обозначениями s – это количество различных чисел в разбиении числа n . Максимум количества различных элементов разбиения будет достигаться на разбиении вида $n = 1 + 2 + 3 + \dots + (t-1) + t + p$. Из последней формулы следует, что $n \geq 0.5t(t+1)$ и, соответственно, $t = O(\sqrt{n})$. Значит, для вычисления $\Phi_{g,n}(x)$ при фиксированном i требуется выполнить $O(\sqrt{n})$ умножений многочленов степени n , стоящих в правой части формулы (13), каждое из которых может быть выполнено с помощью быстрого преобразования Фурье за $O(n \log n)$ операций. Так как i принимает значения от 1 до n , то $t_{g,n,i,j}$ для $i \leq n, j \leq n$ как коэффициенты $\Phi_{g,n}(x)$ могут быть найдены за $O(n^{2.5} \log n)$ операций. Доказательство завершено.

Быстрое вычисление элементов матриц специального вида. Пусть имеются квадратные матрицы A и B с элементами $a_{i,j}$ и $b_{i,j}$, $0 \leq i, j < n$ соответственно. Рассмотрим матрицу $C = C(c_{ij})$.

$$c_{i,j} = \sum_{k=0}^i \sum_{l=0}^j a_{k,l} b_{i-k, j-l}. \quad (14)$$

Обозначим операцию вычисления матрицы C знаком \bullet , т. е. $C = A \bullet B$. Порядок матрицы C равен $n + 1$.

Предложение 3. Матрица C может быть найдена за $O(n^2 \log n)$ операций.

Доказательство. Элемент c_{ij} равен коэффициенту многочлена $A(x)B(x)$, где

$$A(x) = \sum_{k=0}^{n-1} \sum_{l=0}^{n-1} a_{k,l} x^{2n(k+n)+l+n} \text{ и } B(x) = \sum_{k=0}^{n-1} \sum_{l=0}^{n-1} b_{k,l} x^{2nk+l}, \quad \text{при } x^{2n(i+n)+j+n}.$$

С помощью быстрого преобразования Фурье произведение $A(x)B(x)$ может быть найдено за $O(n^2 \log n)$ операций. Доказательство завершено.

Алгоритм вычисления количества орбит, основанный на быстром преобразовании Фурье. Пусть

$$r_{g,i,j} = \frac{f_{g,i,j}}{i!}. \quad (15)$$

Из формул (8) и (15) следует, что $r_{g,i,j} = \frac{1}{i} \sum_{k=1}^i \sum_{l=0}^j r_{g,i-k,j-l} t_{\tilde{g},n,k,l}$. Переобозначив индексы суммирования, из последнего равенства получим

$$r_{g,i,j} = \frac{1}{i} \sum_{k=0}^{i-1} \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i-k,j-l}. \quad (16)$$

Подстановки g и \tilde{g} в формуле (16) связаны между собой так же, как подстановки g и \tilde{g} в формуле (8):

$$\tilde{g} = (g, C^{h_k}) = (C_1^g C_2^g \dots C_v^g, C^{h_k}) \in S_n \times S_{i-k}, \quad (17)$$

где $C^{h_k} = (1, 2, \dots, i-k)$ – цикл длины $i-k$. В дальнейшем для фиксированного k под \tilde{g} будем понимать подстановку из $S_n \times S_{i-k}$, задаваемую формулой (17). Используя запись $t_{\tilde{g},n,k,l}$, будем считать, что \tilde{g} задается формулой (17), при этом соответствующее значение k равно третьему индексу в $t_{\tilde{g},n,k,l}$.

Предложение 4. Величина $r_{g,i,j}$ является целым числом.

Доказательство. Согласно определению $f_{g,i,j}$ – это количество матриц из множества $P_n(i, j)$, которые являются неподвижными точками для подстановок из множества (7). На множестве $P_n(i, j)$ действует группа перестановок столбцов мощностью $i!$.

В соответствии с леммой Бёрнсайда $\frac{f_{g,i,j}}{i!}$ равно числу орбит, на которые разбивается $P_n(i, j)$ под воздействием группы перестановки столбцов. Так как число орбит – целое число, то $r_{g,i,j}$ также целое. Доказательство завершено.

Зафиксируем целое i , $0 \leq i \leq n$ и подстановку $g \in S_n$. Пусть $\overline{R_{g,i}} = (\overline{r_{g,j,k}})$, $j = \overline{0, n}$, $k = \overline{0, n}$ – это $(n+1) \times (n+1)$ матрица с элементами

$$\overline{r_{g,j,k}} = \begin{cases} r_{g,j,k} & \text{при } j \leq i, \\ 0 & \text{при } j \geq i+1. \end{cases} \quad (18)$$

Пусть $\overline{T_{\tilde{g}}} = (\overline{t_{\tilde{g},j,k}})$, $j = \overline{0, n}$, $(n+1) \times (n+1)$ – это $(n+1) \times (n+1)$ матрица с элементами $\overline{t_{\tilde{g},j,k}} = \begin{cases} t_{\tilde{g},n,j,k} & \text{при } j > 0, \\ 0 & \text{при } j = 0. \end{cases}$

Предложение 5. Сложность вычисления $\overline{R_{g,n}}$ составляет $O(n^{2.5} \log n)$.

Доказательство. Рассмотрим матрицу $\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}}$. Через $(\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}})_{j,i}$ условимся обозначать элемент матрицы $\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}}$, стоящий на пересечении i -ой строки и j -го столбца. Применяя формулы (14), (18) и (16), получим

$$\begin{aligned} \frac{(\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}})_{i+t,j}}{i+t} &= \frac{1}{i+t} \sum_{k=0}^{i+t} \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},i+t-k,j-l} = \frac{1}{i+t} \sum_{k=0}^i \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i+t-k,j-l} = \\ &= \frac{1}{i+t} \sum_{k=0}^{i+t-1} \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i+t-k,j-l} - \frac{1}{i+t} \sum_{k=i+1}^{i+t} \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i+t-k,j-l} = \\ &= r_{g,i+t,j} - \frac{1}{i+t} \sum_{k=i+1}^{i+t-1} \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i+t-k,j-l}. \end{aligned} \quad (19)$$

Из (19) следует, что

$$r_{g,i+t,j} = \frac{(\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}})_{i+t,j}}{i+t} + \frac{1}{i+t} \sum_{k=i+1}^{i+t-1} \sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i+t-k,j-l}, \quad (20)$$

т. е., зная $\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}}$, можно найти матрицу $\overline{R_{g,i+t}}$ с элементами $r_{g,s,j} = \begin{cases} r_{g,s,j} & \text{при } s \leq i+t, \\ 0 & \text{при } s \geq i+t+1. \end{cases}$ Значения, найденные при вычислении элементов матрицы $\overline{R_{g,i+t}}$, также позволяют выписать матрицы $\overline{R_{g,i+1}}, \overline{R_{g,i+2}}, \dots, \overline{R_{g,i+t-1}}$.

Пусть $R_{g,i}(x) = \sum_{j=0}^n r_{g,i,j} x^j$, $T_{\tilde{g},i}(x) = \sum_{j=0}^n t_{\tilde{g},n,i,j} x^j$. Тогда при каждом фиксированном k величина $\sum_{l=0}^j r_{g,k,l} t_{\tilde{g},n,i+t-k,j-l}$, стоящая в правой части формулы (20), равна коэффициенту при x^j в произведении многочленов $R_{g,k}(x)T_{\tilde{g},i+t-k}(x)$ степени n . При этом вычисление $r_{g,i+1,j}$ потребует 0 умножений многочленов, $r_{g,i+2,j}$ – 1 умножение, ..., $r_{g,i+t,j}$ – $t-1$ умножение. Всего такой подход потребует выполнения $1+2+\dots+(t-1) = \frac{t(t-1)}{2}$ операций умножения многочленов и с использованием быстрого преобразования Фурье может быть реализован за $O\left(\frac{t(t-1)}{2} n \log n\right)$ операций.

Будем действовать следующим образом: выберем некоторое U и для i , отстоящих друг от друга на величину U , будем вычислять $\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}}$ для нахождения $r_{g,j,k}$. Для всех остальных i величины $r_{g,j,k}$ будем вычислять, используя формулу (20) так, как описано выше. С использованием такого подхода вычисление $\overline{R_{g,n}}$ требует выполнения двух типов операций: вычисления произведения $\overline{R_{g,i}} \bullet \overline{T_{\tilde{g}}}$ и перехода на U строк вперед. Каждая из этих операций будет выполняться $O\left(\frac{n}{U}\right)$ раз. Учитывая предложение 3, получим, что сложность вычисления $\overline{R_{g,n}}$ составит $O\left(\frac{n^3 \log n}{U} + \frac{U-1}{2} n^2 \log n\right)$. Из полученной формулы следует, что оптимально выбирать U порядка \sqrt{n} . В этом случае сложность вычисления $\overline{R_{g,n}}$ составит $O(n^{2.5} \log n)$. Доказательство завершено.

Предложение 6. Количество орбит множества P_n может быть найдено за $O(p(n)n^{2.5} \log n)$ операций.

Доказательство. Количество α_n орбит множества P_n будем искать по формуле (9), правая часть которой содержит $p(n)$ слагаемых. Каждое из значений $f_{g_i, n, n}$, входящих в (9), можно найти, выразив его из формулы (15) через $r_{g_i, n, n}$.

Как элемент матрицы $\overline{R_{g_i, n}}$, $r_{g_i, n, n}$, в соответствии с предложением 5, может быть найдено за $O(n^{2.5} \log n)$ операций. Для вычисления $\overline{R_{g_i, n}}$ необходимо знать матрицу $\overline{T_{g_i}}$, элементы которой, в соответствии с предложением 2, также могут быть найдены за $O(n^{2.5} \log n)$ операций. Следовательно, нахождение количества орбит α_n множества P_n потребует $O(n^{2.5} \log n)$ операций. Доказательство завершено.

Заключение

В данной работе, являющейся продолжением работ авторов [2] и [3], предложено использовать быстрое умножение многочленов, основанное на преобразовании Фурье, для уменьшения вычислительной сложности алгоритма для подсчета количества орбит α_n , на которые разбивается множество P_n квадратных $(0,1)$ -матриц под действием квадрата S_n^2 симметрической группы S_n . Предлагаемая модификация алгоритма имеет вычислительную сложность $O(p(n)n^{2.5} \log n)$.

Методы и алгоритмы, развиваемые на протяжении трех статей, позволили предложить программный метод решения проблемы Кэмерона, расширить на практике ее решение с $n = 28$ до $n = 102$. Если при $n = 28$ количество орбит было равно 20141650236664, то при $n = 102$ число орбит достигло величины 10573638098259743734406815287874396796082663132302516285076455894397796731 – целое число из 74 десятичных знаков. Если такая величина для кого-то все еще имеет смысл, то разработанная программа позволяет продолжить вычисления.

Репозиторий с исходным кодом доступен по адресу <https://github.com/NuM314/thesis-codes/tree/master/solution-pn-n2.5-logn>.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. **Cameron, P. J.** Problems on permutation groups / P. J. Cameron. – [Электронный ресурс]. – Режим доступа: <http://www.maths.qmul.ac.uk/~pjc/pgprob.html>. – Дата доступа: 21.10.2018.
2. **Липницкий, В. А.** Алгоритм развертки при подсчете количества S_n^2 -орбит кэмеронских матриц / В. А. Липницкий, А. И. Сергей, Н. В. Спичекова // Веснік Магілёўскага дзяржаўнага ўніверсітэта імя А. А. Куляшова. Серыя В, Прыродазнаўчыя навукі: матэматыка, фізіка, біялогія. – 2017. – № 2(50). – С. 23–37.
3. **Липницкий, В. А.** Динамическое программирование в методе развертки решения третьей проблемы Кэмерона / В. А. Липницкий, А. И. Сергей, Н. В. Спичекова // Веснік Магілёўскага дзяржаўнага ўніверсітэта імя А. А. Куляшова. Серыя В, Прыродазнаўчыя навукі: матэматыка, фізіка, біялогія. – 2018. – № 1(51). – С. 11–21.
4. **Кормен, Т. Х.** Алгоритмы: построение и анализ / Т. Х. Кормен, Ч. И. Лейзерсон, Р. Л. Риверст, К. Штайн. – Москва : И. Д. Вильямс, 2013. – 1328 с.

Поступила в редакцию 17.11.2018 г.

Контакты: valipnitski@yandex.ru (Липницкий Валерий Антонович)

sergej.a.i@mail.ru (Сергей Александр Иванович)

n.spichekova@gmail.com (Спичекова Наталья Викторовна)

Lipnitsky V., Sergey A., Spichekova N. FAST FOURIER TRANSFORM TO CALCULATE

THE NUMBER OF S_n^2 – ORBITS FOR CAMERON MATRICES.

In the article an algorithm to calculate the number of orbits in the set of binary square matrices of order n , $n \geq 2$, with n ones that are formed under the action of square S_n^2 of the symmetric group

S_n is considered. The algorithm is based on the fast multiplication of polynomials and has the computational complexity of $O(p(n)n^{2.5} \log n)$, where $p(n)$ equals to the number of unordered partitions of n . Up to this paper, the best known algorithm to solve the problem under consideration has the complexity of $O(p(n)n^4)$.

Keywords: binary matrix, symmetric group, orbit, orbit cardinality, the third Peter Cameron's problem, Burnside's lemma, orbital type of a substitution, discrete Fourier transform, fast multiplication of polynomials.