

ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ

УДК 512.54

РЕСУРСНО-ЭФФЕКТИВНЫЙ АЛГОРИТМ ДЛЯ ИССЛЕДОВАНИЯ РОСТА В КОНЕЧНЫХ ДВУПОРОЖДЁННЫХ ГРУППАХ ПЕРИОДА 5¹

А. А. Кузнецов*, А. С. Кузнецова**

* *Сибирский государственный университет науки и технологий имени академика
М.Ф. Решетнева, г. Красноярск, Россия*

** *Красноярский государственный аграрный университет, г. Красноярск, Россия*

Пусть $B_0(2, 5) = \langle a_1, a_2 \rangle$ — наибольшая конечная двупорождённая бернсайдова группа периода 5, порядок которой равен 5^{34} . Для каждого элемента данной группы существует уникальное коммутаторное представление вида $a_1^{\alpha_1} \cdot a_2^{\alpha_2} \cdot \dots \cdot a_{34}^{\alpha_{34}}$, где $\alpha_i \in \mathbb{Z}_5$, $i = 1, 2, \dots, 34$. Здесь a_1 и a_2 — порождающие элементы $B_0(2, 5)$; $a_3, \dots, a_{34}$ — коммутаторы, которые вычисляются рекурсивно через a_1 и a_2 . Определим фактор-группу группы $B_0(2, 5)$ следующего вида: $B_k = B_0(2, 5) / \langle a_{k+1}, \dots, a_{34} \rangle$. Очевидно, что $|B_k| = 5^k$. В работе представлен ресурсно-эффективный алгоритм для исследования роста в конечных группах. Цель — минимизировать пространственную сложность алгоритма, сохранив при этом вычислительную сложность на приемлемом уровне. При помощи нового алгоритма вычислены функции роста группы B_{18} для минимального $A_2 = \{a_1, a_2\}$ и симметричного $A_4 = \{a_1, a_1^{-1}, a_2, a_2^{-1}\}$ порождающих множеств, а для группы B_{19} — только для A_4 . На основе полученных данных сформулирована гипотеза о значениях диаметров графов Кэли группы $B_0(2, 5)$ для указанных порождающих множеств.

Ключевые слова: функция роста, группа Бернсайда, граф Кэли.

DOI 10.17223/20710410/42/7

A RESOURCE-EFFICIENT ALGORITHM FOR STUDY THE GROWTH IN FINITE TWO-GENERATOR GROUPS OF EXPONENT 5

А. А. Kuznetsov*, А. С. Kuznetsova**

* *Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, Russia*

** *Krasnoyarsk State Agrarian University, Krasnoyarsk, Russia*

E-mail: alex_kuznetsov80@mail.ru

For studying the growth in finite groups, we present a resource-efficient algorithm which is a modification of our early algorithm. The purpose of the modification is to

¹Исследование выполнено при финансовой поддержке РФФИ, Правительства Красноярского края, Красноярского краевого фонда поддержки научной и научно-технической деятельности в рамках научного проекта № 17-47-240318.

minimize the space complexity of the algorithm and to save its time complexity at an acceptable level. The main idea of the modified algorithm is to take in the given group G a suitable subgroup N such that $|N| \ll |G|$, to calculate growth functions for all cosets gN independently of each other, to summarize these functions and to obtain the growth function for the group G . By using this algorithm, we calculate the growth functions for the group B_{18} with two generators a_1 and a_2 and for the groups B_{18} , B_{19} with four generators a_1, a_1^{-1}, a_2 and a_2^{-1} , where $B_k = B_0(2, 5)/\langle a_{k+1}, \dots, a_{34} \rangle$ is a quotient of the group $B_0(2, 5) = \langle a_1, a_2 \rangle$ which is the largest two-generator Burnside group of exponent 5 (its order is 5^{34}), a_1 and a_2 are generators of $B_0(2, 5)$ and $a_3, \dots, a_{34}$ are the commutators of $B_0(2, 5)$, so each element in $B_0(2, 5)$ can be represented as $a_1^{\alpha_1} \cdot a_2^{\alpha_2} \cdot \dots \cdot a_{34}^{\alpha_{34}}$, $\alpha_i \in \mathbb{Z}_5$, $i = 1, 2, \dots, 34$. Based on these data, we formulate a hypothesis about the diameters of Cayley graphs of the group $B_0(2, 5)$ with generating sets $A_2 = \{a_1, a_2\}$ and $A_4 = \{a_1, a_1^{-1}, a_2, a_2^{-1}\}$, namely, $D_{A_2}(B_0(2, 5)) \approx 105$ and $D_{A_4}(B_0(2, 5)) \approx 69$.

Keywords: *Burnside group, the Cayley graph, the growth function.*

Введение

Настоящая работа продолжает исследования, начатые в [1], и посвящена разработке ресурсно-эффективного алгоритма для исследования роста в конечных группах, в частности в двупорождённых группах периода 5. В [1] упор сделан на создании алгоритмов минимальной вычислительной сложности, что позволило получить ряд новых результатов о росте в указанных группах. Однако при работе с группой, состоящей из $5^{18} \approx 4 \cdot 10^{12}$ элементов, возникла принципиальная проблема — нехватка объёма памяти, несмотря на то, что для вычислений было задействовано значительное количество ресурсов: 1,2 Тбайт оперативной и 10 Тбайт дисковой памяти. По этой причине была поставлена цель — существенно снизить пространственную сложность алгоритма, сохранив при этом вычислительную сложность на приемлемом уровне.

Напомним основные определения, используемые в [1]. Пусть $G = \langle X \rangle$. Шаром K_s радиуса s группы G будем называть множество всех её элементов, которые могут быть представлены в алфавите X в виде несократимых групповых слов длины не больше s . Соответственно все элементы одинаковой длины i образуют сферу P_i радиуса i . Единица группы e является пустым словом, длина которого равна нулю. Согласно данным определениям,

$$K_s(G, X) = \bigcup_{i=0}^s P_i(G, X).$$

Для каждого целого неотрицательного i можно определить (сферическую) функцию роста группы F_i , которая равна числу элементов в сфере P_i :

$$F_i(G, X) = |P_i(G, X)|.$$

Если из контекста ясно, о какой группе $G = \langle X \rangle$ идёт речь, то для краткости вместо $K_i(G, X)$, $P_i(G, X)$ и $F_i(G, X)$ будем писать K_i , P_i и F_i , соответственно.

Обратим внимание, что при вычислении функции роста группы мы одновременно выясняем характеристики ассоциированного с группой графа Кэли, например, такие, как диаметр и средний диаметр [2]. Пусть $F_{s_0} > 0$, но $F_{s_0+1} = 0$, тогда s_0 является диаметром графа Кэли группы G в алфавите порождающих X , который будем обозначать $D_X(G)$. Соответственно средний диаметр $\overline{D}_X(G)$ равен $\frac{1}{|G|} \sum_{s=0}^{s_0} s \cdot F_s$.

К сожалению, вычисление функции роста большой конечной группы является хотя и разрешимой, но весьма сложной проблемой. Это связано с тем, что в общем случае задача по определению минимального слова элемента группы, как показали С. Ивен и О. Голдрейх [3], является NP-трудной. Поэтому для эффективного решения указанной задачи необходимо создание параллельных алгоритмов, адаптированных к использованию на многопроцессорных вычислительных системах.

Далее представлены алгоритмы, на основе которых при помощи компьютерных вычислений получены новые результаты о росте в конечных бернсайдовых двупорождённых группах периода 5.

1. Алгоритм 1

Пусть X — конечное порождающее множество произвольной конечной группы G . Базовый алгоритм 1, который вычисляет шар $K_s(G, X)$ фиксированного радиуса s , представляет собой ограниченную версию алгоритма A-I из [1].

Алгоритм 1. $K_s = \text{Ball}(G, X, s)$

Вход: X — порождающее множество группы G , радиус s

Выход: шар K_s группы G радиуса s

```

1:  $K_s := \bigcup_{i=0}^s P_i$ , где  $P_i := \emptyset$  — сферы радиуса  $i$ 
2:  $P_0 := \{e\}$ 
3: Для всех  $i = 1, 2, \dots, s$ 
4:   Для всех  $x \in X$  и  $p \in P_{i-1}$ 
5:      $g := x \cdot p$ 
6:     Если  $g \notin K_s$ , то
7:       добавить  $g$  в  $P_i \subset K_s$ 
8:   Если  $|P_i| = 0$ , то
9:     переход в п. 10
10: Вернуть  $K_s$ 
```

Лемма 1. Алгоритм 1 корректен, т.е. он за конечное число шагов вычисляет шар K_s фиксированного радиуса s произвольной конечной группы G , заданной порождающим множеством X .

Доказательство. По построению алгоритм 1 выражает каждый элемент группы G в виде группового слова наименьшей длины в алфавите X . После каждого прохода от п. 3 до п. 7 множество K_s представляет собой шар радиуса i группы G относительно X . Конечность G гарантирует остановку при некотором $i \leq s$. ■

Для оценки пространственной и вычислительной сложности алгоритмов воспользуемся асимптотическим анализом [4]. Введём следующие обозначения:

- $T_i(G, X, s)$ — вычислительная сложность (i — номер алгоритма);
- $M_i(G, X, s)$ — пространственная сложность;
- $O(f(G, X, s))$ — верхняя асимптотическая оценка сложности;
- $\Theta(f(G, X, s))$ — одновременно верхняя и нижняя оценка сложности.

Здесь и далее нас будет интересовать случай $|X| \ll |G|$.

Лемма 2. $T_1 \in \Theta(|K_s|^2)$ и $M_1 \in \Theta(|K_s|)$.

Доказательство. Алгоритм 1 является ограниченной по радиусу s версией алгоритма А-І из [1], поэтому, согласно [1], $T_1 \in \Theta(|K_s|^2)$.

Для того чтобы получить асимптотическую оценку пространственной сложности, следует брать во внимание только множество K_s . Поскольку $|K_s| < M_1 < 2|K_s|$, то $M_1 \in \Theta(|K_s|)$. ■

2. Алгоритм 2

В случае достаточно большой конечной группы G в базовом алгоритме 1 для хранения одного элемента группы необходимо по крайней мере несколько байт. Это означает, что при наличии одного Тбайта памяти мы сможем построить шар группы, состоящий не более чем из 10^{12} элементов. Однако если удастся найти подходящую фактор-группу Q группы G , то можно значительно увеличить первоначально допустимый предел вычислений.

Пусть φ — гомоморфизм G на группу Q и N — ядро φ , т.е. $Q = G/N$. По аналогии с группой, для каждого смежного класса qN определим сферу $P_i(q)$, шар $K_s(q)$ и функцию роста $F_i(q)$:

$$P_i(q) = \{g : g \in P_i \text{ и } \varphi(g) = q\}, \quad K_s(q) = \bigcup_{i=0}^s P_i(q), \quad F_i(q) = |P_i(q)|.$$

Если Q — сравнительно большая группа, то множество $K_{2s}(q)$ будет значительно меньше, чем $K_{2s}(G)$. Данный факт взят за основу построения алгоритма 2, который, получив на входе шар K_s группы G радиуса s , фактор-группу $Q = G/N$ и некоторый элемент $q \in Q$, возвращает функцию роста $F(q)$ для шара $K_{2s}(q)$ смежного класса qN радиуса $2s$.

Алгоритм 2. $F(q) = \text{QuotientGrowthFunction}(K_s, Q, q)$

Вход: Шар $K_s = \bigcup P_i$ группы G радиуса s , фактор-группа $Q = G/N$, $q \in Q$

Выход: Функция роста $F(q)$ шара смежного класса $qN \subset G$ радиуса $2s$

- 1: $F(q) := (0, \dots, 0)$ — нулевой вектор размерности $2s + 1$
 - 2: **Для всех** $w \in Q$ и $i = 0, 1, \dots, s$
 - 3: $P_i(w) := \{g \mid g \in P_i \text{ и } \varphi(g) = w\}$
 - 4: $F_i(q) := |P_i(q)|$
 - 5: $K_s(q) := \bigcup_{i=0}^s P_i(q)$
 - 6: **Для всех** $i = s + 1, s + 2, \dots, 2s$
 - 7: $K_i(q) := K_{i-1}(q)$
 - 8: **Для всех** $v \in Q$
 - 9: $u := q \circ v^{-1}$
 - 10: **Для всех** $g_1 \in P_{i-s}(u)$ и $g_2 \in P_s(v)$
 - 11: $g := g_1 \cdot g_2$
 - 12: **Если** $g \notin K_i(q)$, **то**
 - 13: добавить g в $K_i(q)$
 - 14: $F_i(q) := F_i(q) + 1$
 - 15: **Если** $|K_i(q)| = |N|$, **то**
 - 16: переход в п. 17
 - 17: **Вернуть** $F(q)$
-

Лемма 3. Алгоритм 2 корректен, т.е. он, получив на входе шар K_s конечной группы G радиуса s , фактор-группу $Q = G/N$ и некоторый элемент $q \in Q$, за конечное число шагов вычислит функцию роста $F(q)$ для шара $K_{2s}(q)$ смежного класса qN радиуса $2s$.

Доказательство. Алгоритм 2 получает на входе шар $K_s = \bigcup_{i=0}^s P_i$ и в результате шагов 2–5 преобразует его к виду $K_s = \bigcup_{q \in Q} \bigcup_{i=0}^s P_i(q)$, а также вычисляет $K_s(q) = \bigcup_{i=0}^s P_i(q)$ и $F_i(q)$ для $0 \leq i \leq s$.

Отличительной особенностью алгоритма 2 является то, что при $i > s$ не строятся шары K_i всей группы. Вместо этого вычисляются шары $K_i(q)$ смежного класса qN (пп. 6–17 алгоритма).

Заметим, что при $i > s$ мы не сможем по аналогии с алгоритмом 1 вычислить $P_i(q)$ путём умножения элементов порождающего множества на $P_{i-1}(q)$. Вместо этого будем получать элементы $g \in P_i(q)$ путём выбора таких $u, v \in Q$, что $u \circ v = q$, на основе которых вычислим $g = g_1 \cdot g_2$, где $g_1 \in P_{i-s}(u)$ и $g_2 \in P_s(v)$.

По построению алгоритм 2 представляет каждый элемент $K_i(q)$ в виде слова минимальной длины. Конечность K_s гарантирует построение $K_{2s}(q)$ через конечное число шагов. Кроме того, если возникает ситуация, при которой $|K_i(q)| = |N|$, то это означает, что шар смежного класса qN вычислен. В результате построения шара $K_{2s}(q)$ получим искомую функцию роста $F(q)$. ■

Лемма 4. $T_2 \in O(|N| \cdot |K_s|^2)$ и $M_2 \in \Theta(|K_s| + |Q| + |N|)$.

Доказательство. Вычислительная сложность пп. 1–5 алгоритма 2 линейно зависит от $|K_s|$. Наиболее трудоёмкий участок — пп. 6–16. Очевидно, что число элементов g , которые необходимо вычислить, не превышает $|K_s|^2$. Для проверки $g \notin K_{2s}(q)$ необходимо не более $|N|$ элементарных операций, поэтому $T_2 \in O(|N| \cdot |K_s|^2)$.

Для анализа пространственной сложности необходимо принять во внимание только множества K_s , Q и $K_{2s}(q)$, откуда получим $M_2 \in \Theta(|K_s| + |Q| + |N|)$. ■

При реализации алгоритма 2 возникает естественный вопрос: какой порядок $|Q|_{\text{opt}}$ должен быть у группы Q , чтобы M_2 приняла наименьшее значение? Ответ на поставленный вопрос дает следующая

Лемма 5. $\min_{|Q|} (M_2) \in \Theta(\max(|K_s|, |G|^{1/2}))$, где $|Q|_{\text{opt}} = |G|^{1/2}$.

Доказательство. Пусть $f(Q) = |K_s| + |Q| + |N|$. Так как $|N| = |G|/|Q|$, то $f(Q) = |K_s| + |Q| + |G|/|Q|$. Нетрудно показать, что $f(Q)$ принимает минимальное значение при $|Q| = |G|^{1/2}$. Согласно лемме 4, $\min_{|Q|} (M_2) \in \Theta(|K_s| + 2|G|^{1/2}) = \Theta(\max(|K_s|, |G|^{1/2}))$.

Лемма доказана. ■

Замечание 1. Если мы располагаем быстрым способом нумерации элементов $K_{2s}(q)$, например как в [1], то $K_{2s}(q)$ можно представить в виде булева вектора размерности $|N|$, в котором на i -м месте стоит единица, если элемент с указанным номером лежит в $K_{2s}(q)$, и ноль в противном случае. Теперь для хранения элемента достаточно одного бита. Кроме того, по номеру элемента легко осуществить проверку: встречался ли ранее данный элемент группы? Сложность этой операции $O(1)$. В этом случае, взяв во внимание лемму 4, получим $T'_2 \in O(|K_s|^2)$.

3. Алгоритм 3

Объединив алгоритмы 1 и 2, получим алгоритм 3, который вычисляет функцию роста $F(G)$ шара K_{2s} фиксированного радиуса $2s$ произвольной конечной группы G , заданной порождающим множеством X .

Алгоритм 3. $F(G) = \text{GrowthFunction}(G, X, Q, s)$

Вход: X – порождающее множество группы G , фактор-группа $Q = G/N$, радиус s

Выход: Функция роста $F(G)$ шара K_{2s} радиуса $2s$

- 1: $F(G) := (0, \dots, 0)$ – нулевой вектор размерности $2s + 1$
 - 2: $K_s := \text{Ball}(G, X, s)$
 - 3: Для всех $q \in Q$
 - 4: $F(q) := \text{QuotientGrowthFunction}(K_s, Q, q)$
 - 5: $F(G) := F(G) + F(q)$
 - 6: Вернуть $F(G)$
-

Теорема 1. Алгоритм 3 корректен, т.е. он за конечное число шагов вычисляет функцию роста $F(G)$ шара K_{2s} фиксированного радиуса $2s$ произвольной конечной группы G , заданной порождающим множеством X . Кроме того, верны следующие оценки вычислительной и пространственной сложности данного алгоритма:

- 1) $T_3 \in O(|G| \cdot |K_s|^2)$;
- 2) $M_3 \in \Theta(|K_s| + |Q| + |N|)$;
- 3) $\min_{|Q|} (M_3) \in \Theta(\max(|K_s|, |G|^{1/2}))$, где $|Q|_{\text{opt}} = |G|^{1/2}$.

Доказательство. Следует из лемм 1–5 и формулы $K_{2s} = \bigcup_{q \in Q} K_{2s}(q)$. ■

Замечание 2. Эксперименты в различных группах по алгоритму 3 показали, что для вычисления функции роста всей группы G достаточно подобрать радиус s таким, чтобы выполнялось $|K_s| \sim |G|^{2/3}$. При этом $|Q| \sim |G|^{1/2}$ и, следовательно, $T_3 \in O(|G|^{7/3})$ и $M_3 \in \Theta(|G|^{2/3})$. Для сравнения: $T_0 \in \Theta(|G|^2)$ и $M_0 \in \Theta(|G|)$ – оценки сложности алгоритма А-1 из [1]. Для больших групп получим $M_3 \ll M_0$.

Если выполнено условие замечания 1, то $T'_3 \in O(|Q| \cdot |K_s|^2) = O(|G|^{11/6})$. Для модифицированного алгоритма А-1 показано [1], что $T'_0 \in \Theta(|G|)$.

Замечание 3. Для снижения вычислительной сложности алгоритма 3 воспользуемся остроумным трюком, предложенным Ч. Симсом [5].

Пусть A – группа автоморфизмов $G = \langle X \rangle$, элементы которой оставляют неизменной длину любого группового слова G . В частности, это автоморфизмы, которые фиксируют порождающее множество X . Кроме того, если $x^{-1} \in X$ для любого $x \in X$, то автоморфизм, преобразующий элементы $g \in G$ в обратные, сохраняет длину g . Предположим, что ядро N инвариантно под действием A . Это означает, что A также действует на Q . Выбрав в каждой A -орбите группы Q по одному представителю, получим множество Q_0 . Для всех $q \in Q_0$ определим размер орбиты $l(q)$. Теперь достаточно вычислить шары $K_{2s}(q)$ только для $q \in Q_0$. В результате пп. 3–5 алгоритма 3 примут следующий вид:

- 3: Для всех $q \in Q_0$ // параллельно
- 4: $F(q) := \text{QuotientGrowthFunction}(K_s, Q, q)$
- 5: $F(G) := F(G) + l(q) \cdot F(q)$

Отметим, что цикл по переменной q легко распараллеливается, что позволяет значительно ускорить вычисления.

4. Компьютерные вычисления в двупорождённых группах периода 5

Пусть $B_0(2, 5) = \langle a_1, a_2 \rangle$ — максимальная конечная двупорождённая бернсайдова группа периода 5, порядок которой равен 5^{34} [6]. Используя систему компьютерной алгебры GAP, несложно получить *pc*-представление (*Power Commutator presentation*) данной группы [8, 7]. В этом случае каждый элемент $g \in B_0(2, 5)$ может быть однозначно записан в следующем виде:

$$g = a_1^{\alpha_1} \cdot a_2^{\alpha_2} \cdot \dots \cdot a_{34}^{\alpha_{34}}, \quad \alpha_i \in \mathbb{Z}_5, \quad i = 1, 2, \dots, 34.$$

Здесь a_1 и a_2 — порождающие элементы $B_0(2, 5)$; $a_3, \dots, a_{34}$ — коммутаторы, которые вычисляются рекурсивно через a_1 и a_2 [6].

Обозначим через B_k фактор-группу $B_0(2, 5)$ следующего вида:

$$B_k = B_0(2, 5) / \langle a_{k+1}, \dots, a_{34} \rangle.$$

Очевидно, что $|B_k| = 5^k$ и для всех $g \in B_k$

$$g = a_1^{\alpha_1} \cdot a_2^{\alpha_2} \cdot \dots \cdot a_k^{\alpha_k}.$$

Пусть $A_2 = \{a_1, a_2\}$ и $A_4 = \{a_1, a_1^{-1}, a_2, a_2^{-1}\}$ — минимальное и симметричное порождающие множества групп B_k соответственно.

Вычислить функцию роста $B_0(2, 5)$ относительно порождающего множества A_2 или A_4 в настоящее время едва ли возможно, поскольку количество её элементов очень велико:

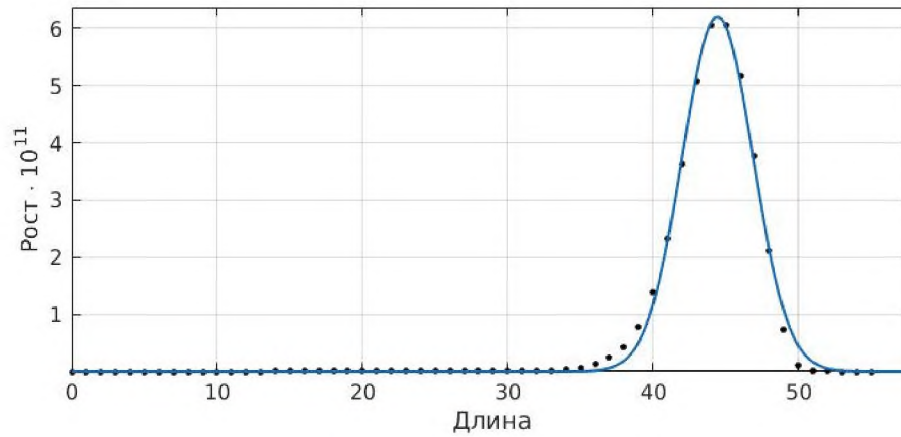
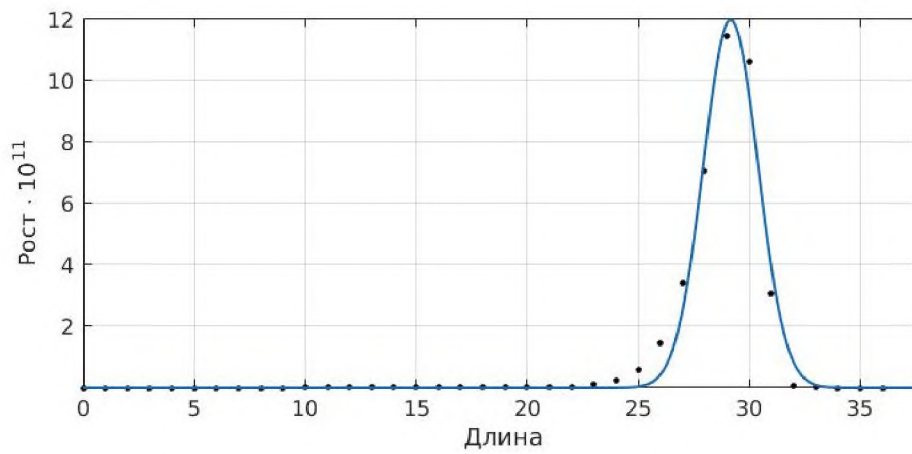
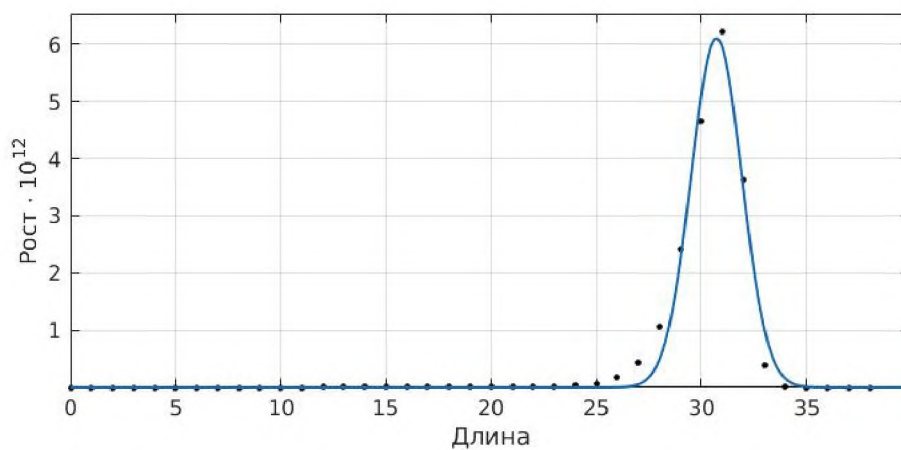
$$5^{34} = 582076609134674072265625 \approx 5 \cdot 10^{23}.$$

Отметим, что на данный момент при помощи компьютерных вычислений удалось получить функции роста групп B_k , порядок которых не превышает 5^{17} [1].

Как уже было сказано, попытка применения алгоритма А-І из [1] для исследования роста группы B_{18} , состоящей из $5^{18} \approx 4 \cdot 10^{12}$ элементов, потерпела неудачу: возникла принципиальная проблема — нехватка объёма памяти. В связи с этим была осуществлена попытка применить алгоритм 3, который, согласно замечанию 2, имеет значительно меньшую пространственную сложность.

Алгоритм 3 был реализован на языке C++. Для снижения вычислительной сложности (см. замечание 1) элементы смежных классов qN нумеровались аналогичным с [1] способом. Учитывая замечание 3, при помощи GAP для каждого случая найдены A , Q , Q_0 и $l(q)$. Для эффективного умножения элементов применялись полиномы Холла [9]. В качестве инструмента распараллеливания использована библиотека OpenMP. Для вычислений был задействован компьютер, имеющий два 16-ядерных процессора и 64 Гбайта оперативной памяти, на котором установлена операционная система Linux. Трансляция программ осуществлялась встроенным в систему компилятором gcc.

На рис. 1–3 представлены графики функций роста групп $B_{18} = \langle A_2 \rangle$, $B_{18} = \langle A_4 \rangle$ и $B_{19} = \langle A_4 \rangle$. Для наглядности на каждом рисунке проведена аппроксимирующая гауссова кривая. В таблице указаны диаметры D и средние диаметры $\bar{D}$ соответствующих графов Кэли, параметры алгоритма, время вычислений, а также объём используемой оперативной памяти.

Рис. 1. График функции роста группы $B_{18} = \langle A_2 \rangle$ Рис. 2. График функции роста группы $B_{18} = \langle A_4 \rangle$ Рис. 3. График функции роста группы $B_{19} = \langle A_4 \rangle$

Группа	D	$\overline{D}$	Параметры алгоритма	Время	Память
$B_{18} = \langle A_4 \rangle$	36	29	$ A = 16, Q = 5^8, Q_0 = 25311$	62 ч	8 Гбайт
$B_{18} = \langle A_2 \rangle$	55	44	$ A = 2, Q = 5^8, Q_0 = 195375$	20 сут	11 Гбайт
$B_{19} = \langle A_4 \rangle$	38	31	$ A = 8, Q = 5^{10}, Q_0 = 1226797$	44 сут	25 Гбайт

На рис. 4 и 5 приведены графики известных диаметров графов Кэли $D_{A_2}(B_k)$ и $D_{A_4}(B_k)$, а также их линейные аппроксимирующие функции, на основе которых можно сделать следующее предположение.

Гипотеза 1. $D_{A_2}(B_0(2, 5)) \approx 105$ и $D_{A_4}(B_0(2, 5)) \approx 69$.

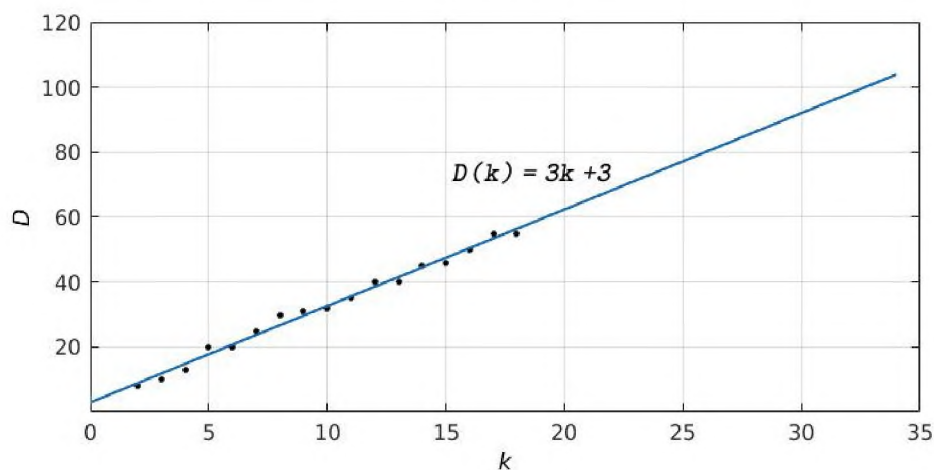


Рис. 4. График $D_{A_2}(B_k)$

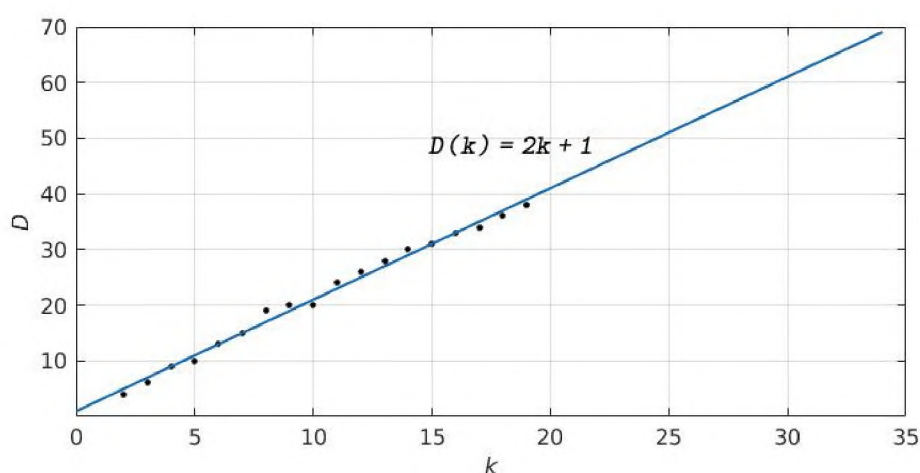


Рис. 5. График $D_{A_4}(B_k)$

ЛИТЕРАТУРА

1. Кузнецов А. А. Об одном алгоритме вычисления функций роста в конечных двупорождённых группах периода 5 // Прикладная дискретная математика. 2016. № 3(33). С. 116–125.
2. Кузнецов А. А., Кузнецова А. С. Параллельный алгоритм для исследования графов Кэли групп подстановок // Вестник СибГАУ. 2014. № 1. С. 34–39.
3. Even S. and Goldreich O. The Minimum Length Generator Sequence is NP-Hard // J. Algorithms. 1981. No. 2. P. 311–313.
4. Skiena S. The Algorithm Design Manual. London: Springer Science+Business Media, 2008. 730 p.
5. Sims C. Fast multiplication and growth in groups // Proc. 1998 Intern. Symp. Symbolic Algebraic Computation. 1998. P. 165–170.

6. *Havas G., Wall G., and Wamsley J.* The two generator restricted Burnside group of exponent five // Bull. Austral. Math. Soc. 1974. No. 10. P. 459–470.
7. *Sims C.* Computation with Finitely Presented Groups. Cambridge: Cambridge University Press, 1994. 628 p.
8. *Holt D., Eick B., and O'Brien E.* Handbook of Computational Group Theory. Boca Raton: Chapman & Hall/CRC Press, 2005. 514 p.
9. *Кузнецов А. А. Кузнецова А. С.* Быстрое умножение элементов в конечных двупорождённых группах периода пять // Прикладная дискретная математика. 2013. № 1. С. 110–116.

REFERENCES

1. *Kuznetsov A. A.* Ob odnom algoritme vychisleniya funkciy rosta v konechnykh dvuporozhdyonnykh gruppakh perioda 5 [An algorithm for computation of the growth functions in finite two-generated groups of exponent 5]. Prikladnaya Diskretnaya Matematika, 2016, no. 3(33), pp. 116–125. (in Russian)
2. *Kuznetsov A. A. and Kuznetsova A. S.* Parallelnyy algoritm dlya issledovaniya grafov Keli grupp podstanovok [A parallel algorithm for study of the Cayley graphs of permutation groups]. Vestnik SibSAU, 2014, no. 1, pp. 34–39. (in Russian)
3. *Even S. and Goldreich O.* The Minimum Length Generator Sequence is NP-Hard. J. Algorithms, 1981, no. 2, pp. 311–313.
4. *Skiena S.* The Algorithm Design Manual. London, Springer Science+Business Media, 2008. 730 p.
5. *Sims C.* Fast multiplication and growth in groups. Proc. 1998 Intern. Symp. Symbolic Algebraic Computation, 1998, pp. 165–170.
6. *Havas G., Wall G., and Wamsley J.* The two generator restricted Burnside group of exponent five. Bull. Austral. Math. Soc., 1974, no. 10, pp. 459–470.
7. *Sims C.* Computation with Finitely Presented Groups. Cambridge, Cambridge University Press, 1994. 628 p.
8. *Holt D., Eick B., and O'Brien E.* Handbook of Computational Group Theory. Boca Raton, Chapman & Hall/CRC Press, 2005. 514 p.
9. *Kuznetsov A. A. and Kuznetsova A. S.* Bystroe umnozhenie elementov v konechnykh dvuporozhdennykh gruppakh perioda pyat' [Fast multiplication in finite two-generated groups of exponent five]. Prikladnaya Diskretnaya Matematika, 2013, no. 1, pp. 110–116. (in Russian)