ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

DOI 10.17223/20710410/16/1

УДК 519.6

О ПРИМИТИВНЫХ НАБОРАХ НАТУРАЛЬНЫХ ЧИСЕЛ

С. Н. Кяжин, В. М. Фомичев

Национальный исследовательский ядерный университет МИФИ, г. Москва, Россия

E-mail: litota975@mephist.ru, fomichev@nm.ru

Описано строение множества примитивных наборов натуральных чисел и установлены их основные свойства. С использованием понятий тупиковости и k-минимальности построен алгоритм перечисления примитивных наборов чисел, не превышающих заданного числа m. Предложены алгоритмы определения показателя примитивности ориентированного конечного графа с помощью поиска в глубину на графе и возведения в степень матрицы смежности вершин и оценена их вычислительная сложность.

Ключевые слова: *примитивный набор натуральных чисел, примитивный граф, примитивная матрица, экспонент, субэкспонент.*

Введение

Матрица M называется *положительной* (*неотрицательной*), если положительны (неотрицательны) все её элементы; этот факт обозначается $M > 0$ ($M \geqslant 0$). Неотрицательная квадратная *матрица* называется *примитивной*, если $M^t > 0$ при некотором натуральном t, а наименьшее натуральное γ, при котором $M^\gamma > 0$, называется *экспонентом* (*показателем примитивности*) *матрицы* M и обозначается $\exp M$. Если такого t не существует, то $\exp M = \infty$. Если для неотрицательной матрицы M выполнено соотношение $M + M^2 + \ldots + M^t > 0$, то наименьшее такое t называется *субэкспонентом матрицы* M.

Сильносвязный *орграф* Γ называется *примитивным*, если при некотором натуральном m для любой пары вершин (i, j) в Γ существует путь из i в j длины m. Наименьшее такое m называется *экспонентом графа* Γ. Под *субэкспонентом орграфа* Γ понимается субэкспонент матрицы смежности его вершин. Субэкспонент сильносвязного n-вершинного графа не превышает n (он равен диаметру графа).

Одной из важных задач при изучении перемешивающих свойств преобразований является определение экспонентов перемешивающих матриц и соответственно перемешивающих графов [1, 2]. При исследовании экспонентов матриц и графов часто используется эпиморфизм мультипликативного моноида неотрицательных матриц порядка n на моноид n-вершинных орграфов, где умножение орграфов определено как умножение бинарных отношений [3]. Матрица положительна, если и только если соответствующий граф полный. Отсюда следует, что орграф и его матрица смежности M одновременно примитивны или не примитивны, в случае примитивности экспоненты их равны.

Критерий примитивности орграфа определяется длинами его простых контуров [2] (контур простой, если проходит через любую вершину не более одного раза). Если

$C_1, \ldots, C_k$ суть все простые контуры орграфа длин $l_1, \ldots, l_k$ соответственно, то орграф примитивный, если и только если числа $l_1, \ldots, l_k$ взаимно просты.

Набор взаимно простых в совокупности натуральных чисел $A = (a_1, \ldots, a_k)$, где $1 < a_1 < \ldots < a_k$, называется *примитивным*. Заметим, что множество всех примитивных наборов натуральных чисел совпадает с областью определения функции Фробениуса $f(a_1, \ldots, a_k)$ [4].

Распознавание примитивности орграфа может быть выполнено как определение длин всех его простых контуров и распознавание примитивности набора всех различных длин этих контуров. Другой подход заключается в определении показателя примитивности графа с помощью возведения в степень матрицы смежности его вершин.

Распознавание примитивности набора чисел $(a_1, \ldots, a_k)$ можно выполнить, применив $k - 1$ раз алгоритм Евклида к элементам набора. Можно также воспользоваться заранее составленными таблицами примитивных наборов.

В п. 1 работы описана связь экспонента и субэкспонента матрицы и ее полугрупповых характеристик, в п. 2 представлены основные свойства примитивных наборов натуральных чисел, в п. 3 сформулированы и доказаны критерии тупиковости и k-минимальности наборов, в п. 4 описан алгоритм перечисления k-минимальных тупиковых примитивных наборов, в п. 5 оценена вычислительная сложность определения длин простых циклов графа с помощью поиска в глубину, а в п. 6 — вычислительная сложность определения экспонента матрицы (и соответствующего графа) с помощью возведения матрицы в степень.

1. О связи экспонента и субэкспонента матрицы с её полугрупповыми характеристиками

Обозначим через $A = (a_{ij})$ матрицу смежности вершин орграфа Γ и через $A^t = \left(a_{ij}^{(t)}\right)$ — степень матрицы A. *Периодом вершины орграфа* называется наибольший общий делитель (НОД) всех длин контуров, содержащих данную вершину. Заметим, что при подсчёте достаточно учитывать только простые контуры, так как любой контур представляет собой соединение нескольких простых контуров и его длина кратна НОД длин всех составляющих его простых контуров. Если орграф сильно связен (матрица A является неразложимой [2, гл. VI, § 2]), то все его вершины имеют одинаковый период, называемый также *к-периодом матрицы* A (орграфа Γ).

Носителем неотрицательной матрицы $B = (b_{ij})$ называется 0,1-матрица $\nu(B) = (\nu b_{ij})$, где

$$\nu b_{ij} = \begin{cases} 1, \text{ если } b_{ij} > 0, \\ 0, \text{ если } b_{ij} = 0. \end{cases}$$

Множество 0,1-матриц размера $n \times n$ образует полугруппу G_n относительно операции $*$, где $A * B = \nu(AB)$.

Утверждение 1 [2, гл. VI, теорема 2.2]. Пусть к-период сильносвязного орграфа Γ равен $q > 0$. Тогда для любых вершин i, j существует единственное число $r(i, j)$, $0 \leqslant r(i, j) < q$, такое, что

1) если $a_{ij}^{(t)} > 0$, то $t \equiv r(i, j) (\mathrm{mod}\ q)$;

2) существует число $k(i, j) > 0$, такое, что $a_{ij}^{kq+r} > 0$ для любого $k \geqslant k(i, j)$.

Следовательно, множество вершин V графа Γ разбивается на q блоков:

$$V = C_0 \cup \ldots \cup C_{q-1},$$

при этом имеется единственная упорядоченная последовательность этих блоков со свойством: если (i,j) — дуга графа Γ и $i \in C_s$, где $s \in \{0,\dots,q-1\}$, то $j \in C_{(s+1)\bmod q}$. Отсюда $A^{kq+r} + A^{kq+r+1} + \dots + A^{kq+r+q-1} > 0$ для любого $k \geqslant \max\limits_{i,j}\{k(i,j)\}$.

Утверждение 2. Пусть неразложимая матрица $A = (a_{ij})$ с к-периодом q имеет тип (d,τ) в полугруппе G_n, где d — циклическая глубина и τ — период матрицы A. Тогда

1) $\tau = q$;
2) если матрица A примитивная, то $d = \exp A$ и $\tau = 1$;
3) если A не примитивная, то $\tau > 1$ и субэкспонент матрицы A не превышает $d + \tau - 1$.

Доказательство. Из определения типа матрицы следует

$$A^d = A^{d+\tau}. \tag{1}$$

Если матрица A примитивная, то $q = 1$ в силу критерия примитивности сильносвязного орграфа. Пусть $\gamma = \exp A$, то есть $A^\gamma > 0$, тогда в полугруппе G_n выполнены условия $A^\gamma = A^{\gamma+1}$ и $A^\gamma \neq A^{\gamma-1}$. Следовательно, (1) выполняется при $d = \gamma, \tau = 1$. Субэкспонент матрицы не превышает γ.

Если A не примитивная, то $q > 1$. Из утверждения 1 следует, что в полугруппе G_n имеет место $A^{kq+r} = A^{kq+r+q}$ для любого $k \geqslant \max\limits_{i,j}\{k(i,j)\}$ и $A^{kq+r} \neq A^{kq+r-q}$ при $k < \max\limits_{i,j}\{k(i,j)\}$. Следовательно, (1) выполняется при $\tau = q$ и $d \geqslant k(i,j)q + r$. Из утверждения 1 также следует, что $A^{kq+r} + A^{kq+r+1} + \dots + A^{kq+r+q-1} > 0$ для любого $k \geqslant \max\limits_{i,j}\{k(i,j)\}$. Значит, субэкспонент матрицы A не превышает $d + \tau - 1$. ■

2. Определяющие свойства примитивных наборов натуральных чисел

Исследуем свойства примитивных наборов.

Утверждение 3. Если A — примитивный набор чисел, то примитивен любой набор, полученный из A добавлением любого натурального числа или (при $|A| > 1$) удалением числа a, кратного одному из остальных чисел набора.

Следствие 1. Набор примитивен, если содержит пару взаимно простых чисел.

Определение 1. *Набор* из N^k назовем *приведённым*, если в нём любое число не кратно любому другому числу.

Иначе говоря, приведённый набор $(a_1,\dots,a_k)$ является антицепью в решётке натуральных делителей числа $k_A = \operatorname{lcm}(a_1,\dots,a_k)$, обозначаемой $D(k_A)$, где $\operatorname{lcm}(a_1, \dots, a_k)$ — наименьшее общее кратное чисел $a_1,\dots,a_k$.

Любому примитивному набору A размера $k \geqslant 1$ однозначно соответствует приведённый набор $\pi(A)$ размера l, где $1 \leqslant l \leqslant k$: если в наборе A число a_j кратно a_i, то набор $\pi(A)$ не содержит a_j.

Определение 2. *Примитивный набор* A размера $k \geqslant 1$ назовем *тупиковым*, если $A = (1)$ или при $k > 1$ удаление из набора любого элемента нарушает его примитивность.

Все примитивные наборы длины 2, не содержащие 1, являются тупиковыми, так как при удалении одного из элементов набора остается число, отличное от 1. В соответствии с утверждением 3 примитивный тупиковый набор является приведённым набором.

Определение 3. *Примитивный набор* A размера $k > 1$ назовем *r-примитивным*, где $0 \leqslant r \leqslant k-1$, если после удаления из A любого подмножества порядка r примитивность получившегося набора сохраняется.

Тупиковый набор 0-примитивен, но не 1-примитивен.

Далее полагаем, что $A = (a_1, \ldots, a_k) \in N^k$ — приведённый набор, элементы которого не превышают числа m. Пусть 2^A — булеан множества $\{a_1, \ldots, a_k\}$; $P(A, r)$ — множество всех примитивных наборов (упорядоченных по возрастанию) размера (порядка) r, $P(A) = \bigcup\limits_{r \leqslant k} P(A, r)$. Заметим, что если A — примитивный приведённый набор, то все наборы из $P(A)$ также приведённые.

На множестве $P(A)$ определим отношение частичного порядка: $(b_1, \ldots, b_l) \leqslant (a_1, \ldots, a_k)$, если и только если $l \leqslant k$ и найдется бесповторная упорядоченная выборка $(i_1, \ldots, i_l)$ из $(1, \ldots, k)$, такая, что $i_1 < \ldots < i_l$ и b_j делит a_{i_j}, $j = 1, \ldots, l$.

Определение 4. *Тупиковый набор* $B \in P(A)$ назовем *минимальным* в $P(A)$, если не существует другого набора $B' \in P(A)$, такого, что $B' \leqslant B$.

Для любого набора B из $P(A)$ имеется хотя бы один минимальный тупиковый набор $\Theta(B)$, такой, что $\Theta(B) \leqslant B$.

Определение 5. *Тупиковый набор* $B \in P(A)$ назовем *r-минимальным* в $P(A)$, если не существует другого набора $B' \in P(A)$ размера r, такого, что $B' \leqslant B$.

Утверждение 4. Если A — примитивный приведённый набор, то $\langle P(A), \leqslant \rangle$ — верхняя полурешётка, в которой максимальный элемент есть A и любой минимальный элемент — это тупиковый минимальный набор.

Доказательство. Если наборы $A_1, A_2 \in P(A)$ имеют размеры соответственно l и r, то их верхняя грань $\sup\{A_1, A_2\}$ определена как набор размера t упорядоченных по возрастанию элементов множества $A_1 \cup A_2$, где $\max\{l, r\} \leqslant t \leqslant r + l$. В соответствии с утверждением 3, $\sup\{A_1, A_2\}$ также есть примитивный набор из $P(A)$, то есть $\langle P(A), \leqslant \rangle$ — верхняя полурешётка.

Утверждения о максимальном и минимальных элементах полурешётки вытекают из определений множества $P(A)$ и тупикового минимального набора. ■

Для набора A рассмотрим наибольший общий делитель как функцию, определённую на 2^A. При $B = \{a_{i_1}, \ldots, a_{i_l}\} \in 2^A$ обозначим $\gcd(B) = \gcd(a_{i_1}, \ldots, a_{i_l})$, если $B \neq \varnothing$, и $\gcd(\varnothing) = \operatorname{lcm}(a_1, \ldots, a_k)$, где $\gcd(a_{i_1}, \ldots, a_{i_l})$ — наибольший общий делитель чисел a_{i_1}, …, a_{i_l}; $D(A) = \{\gcd(B) : B \in 2^A\}$. Множество $D(A)$ частично упорядочено по отношению делимости: $\gcd(B) \leqslant \gcd(B')$ для $B, B' \in 2^A$, если и только если $\gcd(B)$ делит $\gcd(B')$.

Утверждение 5. Если A — примитивный тупиковый набор, то $D(A)$ — решётка, антиизоморфная решетке 2^A.

Доказательство. Установим биективность функции gcd: $2^A \to D(A)$, для этого достаточно убедиться в её инъективности.

Предположим, что функция gcd не инъективна, то есть найдутся множества $B_1, B_2 \in 2^A$, $B_1 \neq B_2$, такие, что $\gcd(B_1) = \gcd(B_2) = d$. Заметим, что $\gcd(B_1 \cup B_2)$ делит d в соответствии с определением функции gcd. Вместе с тем d делит каждое из чисел множества $B_1 \cup B_2$. Значит, $\gcd(B_1 \cup B_2) = d$.

Так как $B_1 \neq B_2$, то одно из этих множеств не включено в другое, пусть для определённости $B_2 \setminus B_1 \neq \varnothing$. Обозначим $B_3 = A \setminus (B_1 \cup B_2)$. В соответствии с определением

функции gcd имеем цепочку равенств

$$1 = \gcd(A) = \gcd(B_1 \cup B_2 \cup B_3) = \gcd(\gcd(B_1 \cup B_2), B_3) = \\ = \gcd(d, B_3) = \gcd(\gcd(B_1), B_3) = \gcd(B_1 \cup B_3).$$

Следовательно, множество $B_1 \cup B_3$ примитивное, где $B_1 \cup B_3 \subset A$, так как $B_2 \setminus B_1 \neq \varnothing$. Отсюда получаем противоречие с тупиковостью набора A.

В соответствии с определением функции gcd если $B \subset B'$, то $\gcd(B')$ делит $\gcd(B)$, значит, биекция $2^A \leftrightarrow D(A)$ антиизотонна. ■

3. Критерии тупиковости и k-минимальности наборов натуральных чисел

Обозначим через A_i коатомы решётки 2^A и через μ_i — атомы решётки $D(A)$: $A_i = \{a_1, \ldots, a_k\} \setminus \{a_i\}$, $\mu_i = \gcd(A_i)$, $i = 1, \ldots, k$.

Теорема 1. Набор A примитивный тупиковый, если и только если $(\mu_1, \ldots, \mu_k)$ — набор попарно взаимно простых чисел, отличных от 1. При этом

$$a_i = \frac{c_i \mu_1 \cdot \ldots \cdot \mu_k}{\mu_i}, \tag{2}$$

где $(c_1, \ldots, c_k)$ есть 1-примитивный набор натуральных чисел и $\gcd(c_i, \mu_i) = 1$ для $i = 1, \ldots, k$.

Доказательство. Пусть набор A примитивный тупиковый. Если $\mu_i = 1$ при некотором $i \in \{1, \ldots, k\}$, то множество A_i примитивное, что противоречит тупиковости набора A. Если $\gcd(\mu_i, \mu_j) = d > 1$ при $i \neq j$, то d делит все числа множеств A_i и A_j, значит, d делит все числа набора A, что противоречит его примитивности.

Докажем достаточность. Если набор A не примитивный, то $\gcd(A) = d > 1$. Отсюда d делит μ_i при любом $i = 1, \ldots, k$, значит, числа $\mu_1, \ldots, \mu_k$ не являются попарно взаимно простыми, то есть имеем противоречие. Если набор A не тупиковый, то $\mu_i = 1$ при некотором $i \in \{1, \ldots, k\}$, что противоречит условию.

По определению чисел $\mu_1, \ldots, \mu_k$ число a_i делится на каждое из чисел множества $\{\mu_1, \ldots, \mu_k\} \setminus \{\mu_i\}$, следовательно, для $i = 1, \ldots, k$ верно (2), где $(c_1, \ldots, c_k) \in N^k$.

Заметим, что набор $(c_1, \ldots, c_k)$ примитивный, так как иначе набор A не примитивный в силу (2). Если $\gcd(c_i, \mu_i) = d > 1$ при некотором $i \in \{1, \ldots, k\}$, то d делит все числа набора A в соответствии с (2) и с определением чисел $\mu_1, \ldots, \mu_k$, что противоречит примитивности набора A. Следовательно, $\gcd(c_i, \mu_i) = 1$ при любом $i = 1, \ldots, k$. Из (2) и определения чисел $\mu_1, \ldots, \mu_k$ следует также, что

$$\mu_i = \gcd\left(\{c_1\mu_2 \cdot \ldots \cdot \mu_k, \ldots, c_k\mu_1 \cdot \ldots \cdot \mu_{k-1}\} \setminus \{(c_i\mu_1 \cdot \ldots \cdot \mu_k)/\mu_i\}\right).$$

Значит, $\gcd(\{c_1, \ldots, c_k\} \setminus \{c_i\}) = 1$ для $i = 1, \ldots, k$, и $(c_1, \ldots, c_k)$ есть 1-примитивный набор. ■

Следствие 2. Пусть $B = \{a_{i_1}, \ldots, a_{i_l}\} \in 2^A$ и $\overline{B} = A \setminus B$, тогда

$$\gcd(B) = \gcd(\{c_{i_1}, \ldots, c_{i_l}\}) \prod_{j \in \overline{B}} \mu_j.$$

Доказательство. Если $c_1, \ldots, c_k, x_1, \ldots, x_k \in N$, то $\gcd(c_1x_1, \ldots, c_kx_k)$ делится на $\gcd(\{c_1, \ldots, c_k\})\gcd(x_1, \ldots, x_k)$. Отсюда, положив $x_i = (\mu_1 \cdot \ldots \cdot \mu_k)/\mu_i$, $i = 1, \ldots, k$, в соответствии с (2) получаем, что $\gcd(B)$ делится на $\gcd(\{c_{i_1}, \ldots, c_{i_l}\}) \times$ $\times\gcd(\{x_{i_1}, \ldots, x_{i_l}\})$, где из теоремы 1 следует, что $\gcd(\{x_{i_1}, \ldots, x_{i_l}\}) = \prod_{j \in \overline{B}} \mu_j$.

Без ущерба для общности положим $B = \{a_1, \ldots, a_l\}$, где $1 \leqslant l \leqslant k$, и обозначим $c' = \gcd(c_1, \ldots, c_l)$, $x' = \gcd(x_1, \ldots, x_l)$. В этих условиях множества $C' = \{c_1/c', \ldots, c_l/c'\}$ и $X' = \{x_1/x', \ldots, x_l/x'\}$ примитивные.

Пусть $\gcd(B) = d \cdot c' \cdot x'$ при натуральном $d > 1$, тогда $\gcd(B') = d$, где $B' = \{a_1/c' \cdot x', \ldots, a_l/c' \cdot x'\}$. Следовательно, d делит $c_r x_r / c' x'$ при $r = 1, \ldots, l$, отсюда

$$d = d(c_r)d(x_r), \tag{3}$$

где $d(c_r)$ делит c_r/c' и $d(x_r)$ делит x_r/x'. В силу примитивности множества C' найдется номер $r \in \{1, \ldots, l\}$, такой, что $d(x_r) > 1$. Тогда, учитывая, что $x_i/x' = (\mu_1 \cdot \ldots \cdot \mu_l)/\mu_i$, $i = 1, \ldots, l$, и числа $\mu_1, \ldots, \mu_l$ попарно взаимно простые, найдется номер $j \in \{1, \ldots, l\}$, такой, что $\gcd(d(x_r), \mu_j) = d_{rj} > 1$, значит, d_{rj} делит d. Из того, что $\gcd(x_j/x', \mu_j) = 1$, следует $\gcd(x_j/x', d_{rj}) = 1$, отсюда в силу (3) d_{rj} делит $d(c_j)$, поэтому d_{rj} делит cc_j/c'.

Вместе с тем в соответствии с теоремой 1 $\gcd(c_j, \mu_j) = 1$, тогда $\gcd(d(c_j), d_{rj}) = 1$. Имеем противоречие. Следовательно, $d = 1$. ■

Представление целого числа n произведением степеней простых чисел $n = \varepsilon \cdot p_1^{k_1} \cdot \ldots \cdot p_s^{k_s}$, где $\varepsilon = \pm 1$, $k_i > 0$ — кратность числа p_i, называется *каноническим разложением числа* n, при этом множество чисел $\{p_1, \ldots, p_s\}$ называется *факторной базой числа* n и обозначается $F(n)$.

Определение 6. *Факторной базой набора* $A = (a_1, \ldots, a_k)$ назовем множество чисел $F(A) = F(a_1) \cup \ldots \cup F(a_k)$.

Докажем критерий k-минимальности тупикового примитивного набора.

Следствие 3. Примитивный тупиковый набор A является k-минимальным, если и только если μ_1, …, μ_k — простые числа и $c_i = 1$, $i = 1, \ldots, k$.

Доказательство. Пусть A есть k-минимальный набор и каноническое разложение μ_i при некотором $i \in \{1, \ldots, k\}$ имеет вид $\mu_i = p_1^{k_1} \cdot \ldots \cdot p_s^{k_s}$. Рассмотрим набор B, состоящий из чисел $\mu_1, \ldots, \mu_{i-1}, \mu_{i+1}, \ldots, \mu_k, \mu_i' = p_1^{k_1} \cdot \ldots \cdot p_s^{k_s - 1}$. Согласно (2), $B \leqslant A$, причём его размер равен k. Тогда A не является k-минимальным. Если $c_i > 1$ при некотором $i \in \{1, \ldots, k\}$, то существует набор B', которому соответствуют $c_1, \ldots, c_{i-1}, c_{i+1}, \ldots, c_k, c_i' = 1$. Согласно (2), $B' \leqslant A$, что противоречит k-минимальности A. Следовательно, μ_1, …, μ_k — простые числа и $c_i = 1$, $i = 1, \ldots, k$.

Если набор A не является k-минимальным, то существует набор $A' = (a_1', \ldots, a_k')$, такой, что $A' \leqslant A$. Согласно (2), $a_i' = (c_i' \mu_1' \cdot \ldots \cdot \mu_k')/\mu_i'$, $i = 1, \ldots, k$, причём c_i' делит c_i или μ_j' делит μ_j при некоторых $i, j \in \{1, \ldots, k\}$. Тогда $c_i > 1$ или μ_j составное соответственно. ■

Следствие 4. Факторной базой k-минимального тупикового набора является множество $\{\mu_1, \ldots, \mu_k\}$.

Примеры k-минимальных тупиковых примитивных наборов:

1) 3-минимальные наборы $A = (6, 10, 15)$, $F(A) = \{2, 3, 5\}$; $B = (10, 14, 35)$, $F(B) = \{2, 5, 7\}$;
2) 4-минимальный набор $C = (30, 42, 70, 105)$, $F(C) = \{2, 3, 5, 7\}$.

4. Перечисление k-минимальных примитивных наборов натуральных чисел

По утверждению 3 любой примитивный набор A можно получить из соответствующего тупикового набора A' добавлением любого числа. По следствию 3 любой ту-

пиковый набор A' можно получить из соответствующего k-минимального набора A'' умножением элемента набора a_i на число, взаимно простое с μ_i, $i \in \{1, \ldots, k\}$.

Построим алгоритм перечисления множества всех k-минимальных примитивных наборов, состоящих из чисел, не превышающих m (обозначим его PR_m).

Пусть $P(x)$ — множество простых чисел, не больших x. Известно [5], что $\pi(x) = |P(x)| \approx x/\ln x$. Набор размера 2 является 2-минимальным, если и только если он представляет собой пару различных простых чисел. Число таких наборов равно $\pi(m)(\pi(m)-1)/2$. Задача перечисления 2-минимальных примитивных наборов решается, в частности, с использованием «решета Эратосфена».

При $k > 2$ в соответствии с теоремой 1 и следствием 3 k-минимальный тупиковый примитивный набор $A = (a_1, \ldots, a_k)$ состоит из чисел $a_i = (\mu_1 \cdot \ldots \cdot \mu_k)/\mu_i$, где $(\mu_1, \ldots, \mu_k)$ — набор различных простых чисел. Тогда если $\mu_1 < \ldots < \mu_k$, то достаточно перечислить все наборы $(\mu_1, \ldots, \mu_k)$ со свойством $\mu_2 \cdot \ldots \cdot \mu_k \leqslant m$. Заметим, что при данных ограничениях $\mu_2 < m^{\frac{1}{k-1}}$.

Пусть n-е простое число есть p_n. Тогда $p_1 = 2 \leqslant \mu_1$, $p_2 = 3 \leqslant \mu_2$. Для $k > 2$ обозначим $\Psi_k = p_3 \cdot \ldots \cdot p_k$ и положим $\Psi_2 = 1$. Значения Ψ_k для $k = 3, \ldots, 8$ приведены в табл. 1.

Таблица 1

Значения функции Ψ_k

k	3	4	5	6	7	8
Ψ_k	5	35	385	5005	85085	1616615

Для любого подходящего набора $(\mu_1, \ldots, \mu_k)$ из неравенства $\mu_2 \cdot \ldots \cdot \mu_k \leqslant m$ при $k > 2$ следует, что $p_2 \leqslant \mu_2 \leqslant m^{\frac{1}{k-1}}$ и для $s = 3, \ldots, k$

$$p_s \leqslant \mu_s \leqslant \left(\frac{m}{3\Psi_{s-1}}\right)^{\frac{1}{k-s+1}}, \tag{4}$$

где при $s < k$ неравенство строгое.

Алгоритм перечисления при $k > 2$ состоит в следующем. В качестве μ_k перебираем все простые числа в пределах, указанных двусторонним неравенством (4). При $3 \leqslant s < k$ и каждом фиксированном наборе чисел $(\mu_{s+1}, \ldots, \mu_k)$ в качестве μ_s перебираем все простые числа в пределах, указанных в (4). При каждом фиксированном наборе чисел $(\mu_3, \ldots, \mu_k)$ перебираем все простые числа μ_1 и μ_2, где $2 \leqslant \mu_1 < \mu_2 < m^{\frac{1}{k-1}}$.

Оценим вычислительную сложность алгоритма, измеренную числом построенных наборов различных простых чисел $(\mu_1, \ldots, \mu_k)$. Из алгоритма следует, что при $s = 3, \ldots, k$ число различных значений μ_s не больше $\pi\left(\left(m/\left(3\Psi_{s-1}\right)\right)^{\frac{1}{k-s+1}}\right)$ (при $s < k$ строго меньше). Число различных пар (μ_1, μ_2) не больше $\pi(x)(\pi(x)-1)/2$ при $x = m^{\frac{1}{k-1}}$. Тогда общее количество искомых наборов оценивается величиной

$$\frac{\pi\left(m^{\frac{1}{k-1}}\right)\left(\pi\left(m^{\frac{1}{k-1}}\right)-1\right)}{2} \prod_{s=3}^{k} \pi\left(\left(\frac{m}{3\Psi_{s-1}}\right)^{\frac{1}{k-s+1}}\right).$$

При больших m и k эта величина имеет порядок не более

$$\mathrm{O}\left(\left(km^{\frac{2}{k-1}}(m/3)^{H(k-1)}\right) / \left(\ln^2 m \prod_{j=2}^{k-1} \Psi_j\right)\right),$$

где $H(k-1) = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{k-2}$ — сумма первых $(k-1)$ членов гармонического ряда. Порядок последней величины не превышает $\mathrm{O}\left(m^{\ln k} / \left(\ln^2 m \prod_{j=2}^{k-1} \Psi_j\right)\right)$. При $k > 2$ для оценки величин Ψ_k можно использовать оценку [6]: $p_k > k \ln k$. Тогда $\Psi_k > \frac{k!}{2} \prod_{j=3}^{k} \ln j$.

Пусть теперь числа $a_1, \ldots, a_k$ не ограничены. Тогда наибольшее число в k-минимальном тупиковом примитивном наборе равно $\max(a_1, \ldots, a_k) = \mu_2 \cdot \ldots \cdot \mu_k$. С использованием точных значений простых чисел вычислены достижимые нижние оценки для наибольших чисел в k-минимальных тупиковых примитивных наборах при $k = 3, \ldots, 8$ (табл. 2).

Таблица 2

Числовые границы для наборов длины k

Размер набора k	3	4	5	6	7	8
Нижняя граница $\max(a_1, \ldots, a_k)$	15	105	1155	15015	255255	4849845

5. Определение длин простых циклов с помощью поиска в глубину

Для определения длин простых циклов используем известный алгоритм поиска в глубину [7].

Пусть дан граф $G = (V, E)$, где V — множество вершин графа, а E — множество его неориентированных рёбер либо ориентированных дуг. Опишем алгоритм обхода всех рёбер графа. В качестве начальной выбираем произвольную вершину и двигаемся по рёбрам, пока не встретится тупик (вершина, не имеющая исходящих рёбер, ведущих в непосещённые вершины). После попадания в тупик возвращаемся назад по пройденному пути, пока не встретится вершина, у которой есть исходящие ребра, ведущие в непосещённые вершины, и из неё двигаемся дальше по одному из таких рёбер. Алгоритм обхода рёбер завершает работу, когда встречается начальная вершина и все её соседние вершины уже посещены. Если после этого остаются непосещённые вершины, то повторяется поиск из одной из них в соответствии с вышеописанным алгоритмом. Алгоритм завершается, когда обнаружены все вершины графа.

Для наглядности считаем, что в ходе работы алгоритма вершины графа могут быть белыми, серыми и чёрными. Изначально все вершины помечены белым цветом. Впервые обнаружив вершину v, красим её серым цветом. По окончании обработки всех исходящих рёбер красим вершину v в чёрный цвет. Таким образом, белый цвет соответствует тому, что вершина ещё не обнаружена, серый — что вершина обнаружена, но просмотрены ещё не все исходящие из неё ребра, чёрный — что вершина обнаружена и все исходящие из неё рёбра просмотрены.

Оценим вычислительную сложность реализации алгоритма на однопроцессорной системе. В качестве элементарных операций выберем прохождение ребра графа и сравнение двух чисел. Так как каждое ребро проходится не более одного раза в прямом и обратном направлении, то время работы алгоритма поиска в глубину оценивается величиной $\mathrm{O}(n^2)$.

Модифицируем данный алгоритм поиска в глубину с целью определения длин всех простых циклов. В частности, переходя в вершину u из вершины v по ребру (v, u),

будем запоминать предшественника u, записывая $p[u] = v$. Для вершин, у которых предшественников нет, положим $p[u] = -1$.

Рёбра ориентированного графа можно разделить на несколько категорий в зависимости от их роли при поиске в глубину. *Рёбра деревьев* — это ребра, входящие в лес поиска в глубину. *Обратные рёбра* — это рёбра, соединяющие вершину с её предком в дереве поиска в глубину. *Прямые ребра* — это рёбра, соединяющие вершину с её потомком, но не входящие в лес поиска в глубину. *Перекрёстные ребра* — все остальные. Они могут соединять две вершины из одного дерева поиска в глубину, если ни одна их этих вершин не является предком другой, или же вершины из разных деревьев.

Тип ребра (v, u) можно определить по цвету вершины u в момент, когда ребро проходится в первый раз: белый цвет означает ребро дерева ((v, u) войдёт в лес поиска в глубину); серый (u является предком v) — обратное ребро; чёрный (ни одна из вершин не является предком другой) — прямое или перекрёстное ребро.

Для каждой вершины v в процессе поиска в глубину запомним ещё два параметра: в $d[v]$ запишем «время» i-го попадания в вершину, а в $f[v]$ — «время» $(i+1)$-го попадания. Здесь под «временем» понимается номер шага алгоритма. Если вершина u серая, то это означает, что последнее ребро — обратное и получен цикл, содержащий u. Если $d[u] \neq 0$, то данный цикл является простым, так как вершина u встретилась второй раз. Длина цикла равна разности $f[u] - d[u]$.

Вычисление длин простых циклов реализуется в ходе алгоритма поиска в глубину и имеет порядок временной сложности $\mathrm{O}(n)$.

Из полученного набора длин циклов необходимо выделить множество всех различных длин с помощью упорядочивания чисел [8]. Сложность увеличится не более чем в $\mathrm{O}(\log n)$ раз, то есть вычислительная сложность алгоритма определения длин всех простых циклов графа оценивается величиной $\mathrm{O}(n^2 \log n)$.

Ёмкостная сложность алгоритма определяется размером памяти, необходимым для хранения матрицы смежности вершин графа, то есть составляет $\mathrm{O}(n^2)$ битов.

6. Определение экспонента графа с помощью возведения в степень матрицы смежности вершин

Определение экспонента графа связано с возведением в степень матрицы M смежности вершин графа и с проверкой положительности её элементов.

Известна [9, 10] достижимая оценка экспонента матрицы $\exp M \leqslant n^2 - 2n + 2$, где n — порядок матрицы. То есть если матрица M^t имеет при $t > n^2 - 2n + 2$ хотя бы один нулевой элемент, то соответствующий граф не примитивен. Если $M^t > 0$, то матрица и граф примитивны и $\exp M = \exp \Gamma \leqslant t$.

Для оценки вычислительной сложности алгоритма рассмотрим однопроцессорную вычислительную модель. Элементарными операциями считаем сложение и умножение в кольце целых чисел. Рассмотрим операцию умножения в полугруппе G_n. Размер необходимой памяти оценим в битах.

Сложность умножения квадратных матриц размера n имеет порядок $\mathrm{O}(n^3)$. При этом для распознавания примитивности достаточно возвести матрицу в степень не выше 2^r, где $r = \lceil \log_2(n^2 - 2n + 2) \rceil$. С помощью алгоритма быстрого возведения в степень [8] потребуется $\mathrm{O}(rn^3) = \mathrm{O}(n^3 \log_2 n)$ операций для определения примитивности матрицы.

Опишем подробнее алгоритм быстрого возведения в степень для точного вычисления экспонента матрицы. Возведем матрицу M в квадрат. Полученную после первого возведения матрицу M^2 ещё раз возведем в квадрат, получим M^4 и т. д. Пусть матрица

M^k положительна. Тогда вернемся к матрице $M^{k/2}$ и умножим её на матрицу $M^{k/4}$, и далее будем делить пополам отрезок, содержащий значение $\exp M$, пока не определится наименьшая степень t, при которой матрица M^t положительна.

Оценим ёмкостную сложность алгоритма. В памяти достаточно хранить матрицы M^t, где $t = 2^0, 2^1, \ldots, 2^{r-1}$. Таким образом, ёмкостная сложность алгоритма составляет $\mathrm{O}(n^2 \log_2 n)$.

Сравним эти значения с оценками сложности алгоритма распознавания примитивности n-вершинного ориентированного графа с помощью поиска в глубину, полученными в п. 5. Результаты сравнения приведены в табл. 3.

Таблица 3

Сложность алгоритмов распознавания примитивности графа

Алгоритм	Временная сложность	Ёмкостная сложность
Поиск в глубину	$\mathrm{O}(n^2 \log n)$ обращений к памяти	$\mathrm{O}(n^2)$ бит
Возведение в степень матрицы смежности	$\mathrm{O}(n^3 \log n)$ сложений и умножений единиц и нулей	$\mathrm{O}(n^2 \log_2 n)$ бит

Отметим, что, в отличие от первого алгоритма, второй определяет значение показателя примитивности.

ЛИТЕРАТУРА

1. *Фомичев В. М.* Оценки экспонентов примитивных графов // Прикладная дискретная математика. 2011. № 2(11). С. 101–112.
2. *Сачков В. Н., Тараканов В. Е.* Комбинаторика неотрицательных матриц. М.: ТВП, 2000.
3. *Биркгоф Г.* Теория решеток. М.: Наука, 1984.
4. *Арнольд В. И.* Экспериментальное наблюдение математических фактов. М.: МЦНМО, 2006.
5. *Коблиц Н.* Курс теории чисел и криптографии. М.: ТВП, 2001.
6. *Rosser B.* The n-th prime is greater than $n \log n$ // Proc. London Math. Soc. 1939. V. 45. P. 21–44.
7. *Лахно А. П.* Поиск в глубину и его применение // Московские олимпиады по информатике. М.: МЦНМО, 2006.
8. *Порублев И. Н., Ставровский А. Б.* Алгоритмы и программы. Решение олимпиадных задач. М.: Вильямс, 2007
9. *Wielandt H.* Unzerlegbare nicht negative Matrizen // Math. Zeitschr. 1950. No. 52. S. 642–648.
10. *Сачков В. Н., Ошкин И. Б.* Экспоненты классов неотрицательных матриц // Дискретная математика. 1993. № 2. С. 150–159.