МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

УДК 510.52 DOI 10.17223/20710410/69/8

О ГЕНЕРИЧЕСКОЙ СЛОЖНОСТИ ПРОБЛЕМ 3-РАСКРАСКИ ГРАФОВ[1]

Д. П. Рузанова, А. Н. Рыбалов

Институт математики им. С. Л. Соболева СО РАН, г. Омск, Россия

E-mail: alexander.rybalov@gmail.com

Изучается генерическая сложность двух вариантов проблемы о 3-раскраске графов: проблема распознавания и проблема поиска 3-раскраски графа. Для обеих проблем эффективных полиномиальных алгоритмов неизвестно. Проблема поиска 3-раскраски используется в известном криптографическом протоколе Блюма для доказательства с нулевым разглашением. Предлагается полиномиальный генерический алгоритм для проблемы распознавания 3-раскраски графа. Для проблемы поиска 3-раскраски доказывается, что если $P \neq NP$ и $P = BPP$, то существует подпроблема этой проблемы, для которой нет полиномиального генерического алгоритма. Полученный результат является теоретическим обоснованием применения проблемы поиска 3-раскраски графа в криптографии, где нужно, чтобы проблема взлома криптоалгоритма была трудной для почти всех входов.

Ключевые слова: *генерическая сложность, 3-раскраска графа.*

ON THE GENERIC COMPLEXITY OF GRAPH 3-COLORING PROBLEMS

D. P. Ruzanova, A. N. Rybalov

Sobolev Institute of Mathematics, Omsk, Russia

In this paper, we study the generic complexity of two versions of the graph 3-coloring problem: the graph 3-coloring recognition problem and the graph 3-coloring search problem. For both problems, no efficient polynomial algorithms are known. The 3-coloring search problem is used in the well-known Blum zero-knowledge cryptographic protocol. We propose a polynomial generic algorithm for the graph 3-coloring recognition problem. For the 3-coloring search problem, we prove that if $P \neq NP$ and $P = BPP$, then there is a subproblem of this problem for which there is no polynomial generic algorithm. The obtained result provides theoretical justification for applying the 3-coloring search problem in cryptography, an application where breaking a cryptographic algorithm must be difficult for almost all inputs. To prove this theorem, we use the method of generic amplification, which allows to construct generically hard problems from the problems hard in the classical sense.

Keywords: *generic complexity, 3-coloring of graphs.*

[1]Работа поддержана грантом Российского научного фонда № 25-11-20023.

Введение

Проблема о 3-раскраске графов является классической комбинаторной проблемой, изучаемой многие десятилетия. Её формулировка состоит в следующем: по заданному произвольному графу определить, можно ли раскрасить все его вершины в три цвета так, чтобы любые две вершины, соединённые ребром, были раскрашены в разные цвета. Эта проблема содержится в классическом списке из двадцати одной NP-полной проблемы из знаменитой работы Р. Карпа [1], откуда следует, что при условии неравенства классов сложности P и NP для проблемы о 3-раскраске не существует эффективных полиномиальных алгоритмов. Этот факт открывает возможности для применения данной проблемы в криптографии. Например, проблема о 3-раскраске используется в протоколе Блюма [2] для доказательства с нулевым разглашением [3].

Однако трудноразрешимости в классическом смысле в современной криптографии недостаточно. Важно, чтобы алгоритмическая проблема, лежащая в основе стойкости криптографического протокола, являясь (гипотетически) трудной в классическом смысле, оставалась трудной и в генерическом смысле [4], т. е. для почти всех входов. Это объясняется тем, что при случайной генерации ключей в криптографическом алгоритме происходит генерация входа алгоритмической проблемы, лежащей в основе алгоритма. Если эта проблема будет легкоразрешимой почти всегда, то для почти всех таких входов её можно быстро решить и ключи почти всегда будут нестойкими. Поэтому проблема должна быть трудной для почти всех входов. Например, таким поведением обладают классические алгоритмические проблемы криптографии: распознавания квадратичных вычетов [5], дискретного логарифма [6], извлечения корня в группах вычетов [7] (проблема обращения функции RSA).

В данной работе изучается генерическая сложность двух вариантов проблемы о 3-раскраске графов. Первый вариант — проблема распознавания 3-раскраски графа. Здесь входом является произвольный граф, необходимо определить, существует ли для него 3-раскраска. Второй вариант — проблема поиска 3-раскраски графа. Для этой проблемы входом является граф, для которого заведомо существует 3-раскраска, нужно хотя бы одну такую 3-раскраску найти. Проблемы поиска, в отличие от проблем распознавания, находят применения в криптографии, где всегда известно, что решение есть и надо его найти. Для обеих проблем неизвестно эффективных полиномиальных алгоритмов. В работе предлагается полиномиальный генерический алгоритм для проблемы распознавания 3-раскраски графа. Для проблемы поиска 3-раскраски доказывается, что если $\mathrm{P} \neq \mathrm{NP}$ и $\mathrm{P} = \mathrm{BPP}$, то существует подпроблема этой проблемы, для которой нет полиномиального генерического алгоритма. Класс BPP состоит из проблем, разрешимых за полиномиальное время на вероятностных машинах Тьюринга. Считается, что класс BPP совпадает с классом P, то есть любой полиномиальный вероятностный алгоритм можно эффективно дерандомизировать, построив полиномиальный алгоритм, не использующий генератор случайных чисел и решающий ту же самую проблему. Хотя равенство $\mathrm{P} = \mathrm{BPP}$ до сих пор не доказано, имеются веские основания в его пользу [8].

1. Предварительные сведения

Рассмотрим неориентированные графы без петель и кратных рёбер. Граф G — это пара (V, E), где V — множество вершин; $E \subseteq V \times V$ — множество рёбер графа G. Для любого множества A через $|A|$ будем обозначать его мощность.

Пусть имеются два графа $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$. Биекция $\varphi : V_1 \to V_2$ называется *изоморфизмом* графов G_1 и G_2, если для любых $v_1, v_2 \in V_1$ имеет место

$(v_1, v_2) \in E_1$ тогда и только тогда, когда $(\varphi(v_1), \varphi(v_2)) \in E_2$. При этом графы G_1 и G_2 называются *изоморфными*, что обозначается как $G_1 \cong G_2$.

3-Раскраской графа G называется присваивание каждой вершине графа одного из трёх цветов, такое, что любые две вершины графа, соединённые ребром, раскрашены в разные цвета. Если для графа существует 3-раскраска, будем называть его *3-раскрашиваемым*. Существование 3-раскраски графа $G = (V, E)$ эквивалентно тому, что множество вершин можно разбить на три непересекающихся подмножества (возможно, пустые) $V = V_1 \cup V_2 \cup V_3$, такие, что любые две вершины из каждого подмножества V_i, $i = 1, 2, 3$, не соединены ребром.

Проблема распознавания 3-раскраски состоит в следующем: по произвольному заданному графу G определить, существует ли для него 3-раскраска. *Проблема поиска 3-раскраски* определяется несколько иначе: по произвольному заданному 3-раскрашиваемому графу G найти хотя бы одну 3-раскраску.

Под *размером* графа G будем понимать число вершин.

Определения полиномиальных детерминированных и вероятностных алгоритмов, а также вычислительных классов P, NP и BPP можно найти в [9].

Напомним основные определения генерического подхода [4]. Пусть I — некоторое множество входов, I_n — подмножество входов размера n. Для подмножества $S_n \subseteq I$ определим последовательность

$$\rho_n(S) = \frac{|S_n|}{|I_n|}, \quad n = 1, 2, 3, \ldots,$$

где $S_n = S \cap I_n$ — множество входов из S размера n. *Асимптотической плотностью* S назовём предел

$$\rho(S) = \lim_{n \to \infty} \rho_n(S).$$

Множество S называется *пренебрежимым*, если $\rho(S) = 0$.

Алгоритм $\mathcal{A}$ с множеством входов I и множеством выходов $J \cup \{\square\}$ $(\square \notin J)$ называется *генерическим*, если

1) $\mathcal{A}$ останавливается на всех входах из I;
2) множество $\{x \in I : \mathcal{A}(x) = \square\}$ является пренебрежимым.

Здесь символ $\square$ обозначает неопределённый ответ. Генерический алгоритм $\mathcal{A}$ *вычисляет* функцию $f : I \to \mathbb{N}$, если для всех $x \in I$ выполнено

$$(\mathcal{A}(x) \neq \square) \Rightarrow (f(x) = \mathcal{A}(x)).$$

Проблема распознавания множества $A \subseteq I$ *генерически разрешима за полиномиальное время*, если существует полиномиальный генерический алгоритм, вычисляющий характеристическую функцию множества A. Напомним, что *характеристической функцией* множества $A \subseteq I$ называется функция $\chi_A : I \to \{0, 1\}$, определённая следующим образом:

$$\chi_A(x) = \begin{cases} 1, & \text{если } x \in A, \\ 0, & \text{если } x \notin A. \end{cases}$$

2. Протокол Блюма

Доказательство с нулевым разглашением [3] — интерактивный вероятностный протокол между двумя акторами: Доказывающим (Prover) и Проверяющим (Verifier), позволяющий доказать, что некоторое утверждение верно, не предоставляя никакой информации о самом доказательстве данного утверждения. Данный криптографический протокол обладает тремя свойствами:

1) Полнотой: если утверждение действительно истинно, то Доказывающий убедит в этом Проверяющего с любой наперёд заданной точностью.
2) Корректностью: если утверждение неверно, то любой, даже «нечестный», Доказывающий не сможет убедить Проверяющего, за исключением пренебрежимо малой вероятности.
3) Нулевым разглашением: если утверждение верно, то любой, даже «нечестный», Проверяющий не узнает ничего, кроме самого факта, что утверждение верно.

М. Блюмом в [2] предложен протокол доказательства с нулевым разглашением, основанный на использовании вычислительно трудных проблем теории графов. Опишем этот протокол для проблемы 3-раскраски графов. Назовём проверяющую сторону Виктор (Verifier), а доказывающую — Полина (Prover). Полина знает 3-раскраску в некотором большом графе G. Виктору известен граф G, но он не знает его 3-раскраски. Полина хочет доказать Виктору, что она знает 3-раскраску, не выдавая при этом ни самой 3-раскраски, ни какой-либо другой информации о ней. Виктор хочет удостовериться, что Полина действительно знает 3-раскраску.

Для этого Виктор и Полина совместно выполняют несколько раундов протокола:

1) Полина генерирует граф H, изоморфный G. Преобразование 3-раскраски между изоморфными графами — простая задача, поэтому если Полине известна 3-раскраска G, то она также знает 3-раскраску в графе H.
2) Полина передаёт граф H Виктору.
3) Виктор выбирает случайный бит $b \in \{0, 1\}$.
4) Если $b = 0$, то Виктор просит Полину доказать изоморфизм G и H, то есть предоставить соответствующую биекцию вершин этих двух графов. Виктор может проверить, действительно ли G и H изоморфны.
5) Если $b = 1$, то Виктор просит Полину указать 3-раскраску H. Для проблемы изоморфизма графов на данный момент неизвестно полиномиальных алгоритмов, поэтому практически невозможно по 3-раскраске графа H найти 3-раскраску изоморфного ему графа G.

Проверим выполнение свойств доказательства с нулевым разглашением.

В каждом раунде Виктор выбирает новый случайный бит, который неизвестен Полине, поэтому, чтобы Полина могла ответить на оба вопроса, нужно, чтобы H был в самом деле изоморфен G и Полина должна знать 3-раскраску для H (а значит, и для G). Поэтому после достаточного числа раундов Виктор может быть уверен в том, что у Полины есть 3-раскраска G. С другой стороны, Полина не раскрывает никакой информации о 3-раскраске G. Более того, Виктору сложно будет доказать кому-либо ещё, что он сам или Полина знают 3-раскраску G.

Предположим, что Полине неизвестна 3-раскраска G, но она хочет обмануть Виктора. Тогда Полине необходим неизоморфный G граф G', в котором она знает 3-раскраску. В каждом раунде она может передавать Виктору либо граф H' — изоморфный G', либо граф H — изоморфный G. Если Виктор попросит доказать изоморфизм графов и ему был передан H, то обман не вскроется. Аналогично — если он попросит показать 3-раскраску и ему был передан H'. Вероятность того, что Полина обманет Виктора после k раундов, равна 2^{-k}, что близко к нулю при достаточном числе раундов.

Предположим, что Виктор не знает 3-раскраску, но хочет доказать третьей стороне (Борису), что Полина её знает. Если Виктор, например, заснял на видео все раунды протокола, Борис едва ли ему поверит: он может предположить, что Виктор и Полина в сговоре и в каждом раунде Виктор заранее сообщал Полине свой выбор случайного

бита, чтобы Полина могла передавать ему H для проверок изоморфизма и H' для проверок 3-раскраски. Таким образом, без участия Полины доказать, что она знает 3-раскраску, можно, лишь доказав, что во всех раундах протокола выбирались действительно случайные биты.

3. Генерический алгоритм распознавания 3-раскраски

Граф будем представлять матрицей смежности (достаточно только верхней её половины). Для краткости эти верхние половины матриц будем называть просто матрицами. Обозначим через $\mathcal{G}$ множество всех графов, через $\mathcal{G}_n$ — множество графов размера n. Легко подсчитать, что $|\mathcal{G}_n| = 2^{n(n-1)/2}$.

Теорема 1. Проблема распознавания 3-раскраски графов генерически разрешима за полиномиальное время.

Доказательство. Полиномиальный генерический алгоритм для распознавания 3-раскраски графов работает на графе G размера n следующим образом:

1) Ищет в графе G клику K_4 размера 4. Это делается перебором всевозможных подмножеств вершин G размера 4 и проверки того, что все они друг с другом соединены ребрами. Число таких подмножеств $\mathrm{C}_n^4 = O(n^4)$ полиномиально.
2) Если клика нашлась, то выдаёт ответ «НЕТ».
3) Если клики нет, то выдаёт ответ «НЕ ЗНАЮ».

Очевидно, что граф K_4 нельзя раскрасить в три цвета, а значит, если он является подграфом графа G, то и граф G также нельзя раскрасить тремя цветами.

Для доказательства генеричности алгоритма покажем, что множество графов, не содержащих подграфа K_4 (обозначим это множество S), является пренебрежимым. Рассмотрим множество графов S', в которых на вершинах $\{1,\ldots,n\}$ запрещены клики на вершинах $\{1,\ldots,4\}$, на вершинах $\{5,\ldots,8\}$, …, на вершинах $\{4([n/4]-1)+1, \ldots, 4[n/4]\}$. Так как для графов из S' запрещены не любые клики размера 4, то множество S содержится в S': $S \subseteq S'$.

Можно подсчитать, что

$$|S'_n| = 2^{n(n-1)/2-6[n/4]}(2^6-1)^{[n/4]}.$$

Это следует из того, что в матрицах смежности графов из множества S' над диагональю «запрещены» $[n/4]$ подматриц размера 4, состоящих из единиц. Эти 4×4-матрицы имеют $4(4-1)/2 = 6$ мест для расстановки нулей и единиц. Тогда

$$\rho(S') = \lim_{n\to\infty}\frac{|S'_n|}{|\mathcal{G}_n|} = \lim_{n\to\infty}\frac{2^{n(n-1)/2-6[n/4]}(2^6-1)^{[n/4]}}{2^{n(n-1)/2}} =$$
$$= \lim_{n\to\infty}\frac{(2^6-1)^{[n/4]}}{2^{6[n/4]}} = \lim_{n\to\infty}\left(1-\frac{1}{2^6}\right)^{[n/4]} = 0.$$

Это доказывает, что множество S' является пренебрежимым, а значит, его подмножество S тем более пренебрежимо. ■

4. Проблема поиска 3-раскраски

Напомним, что проблема поиска 3-раскраски состоит в том, что по произвольному 3-раскрашиваемому графу $G = (V, E)$ нужно найти хотя бы одну его 3-раскраску, то есть непересекающиеся подмножества V_1, V_2, V_3, такие, что $V_1 \sqcup V_2 \sqcup V_3 = V$ и $\forall x, y \in V_i\big((x,y) \notin E\big)$, $i \in \{1,2,3\}$. Будем обозначать эту проблему $\mathcal{SC}_3$. Для неё также неизвестно полиномиальных алгоритмов.

Рассмотрим бесконечную последовательность графов $\gamma = \{G_1, G_2, \ldots, G_n, \ldots\}$, такую, что G_n имеет n вершин, $n \in \mathbb{N}$. Для каждой последовательности γ определим подпроблему поиска 3-раскраски $\mathcal{SC}_3(\gamma)$ как ограничение исходной проблемы $\mathcal{SC}_3$ на множество входов $\{G : G \cong G_n, G_n \in \gamma, n \in \mathbb{N}\}$.

Лемма 1. Если не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{SC}_3$, то найдётся последовательность графов γ, такая, что не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{SC}_3(\gamma)$.

Доказательство. Пусть $P_1, P_2, \ldots$ — все полиномиальные вероятностные алгоритмы. Если не существует полиномиального вероятностного алгоритма для проблемы $\mathcal{SC}_3$, то для любого вероятностного полиномиального алгоритма P_n найдётся бесконечно много графов, для которых алгоритм P_n не может решить $\mathcal{SC}_3$. Из этого следует, что можно выбрать такую последовательность $\gamma' = \{G_1, G_2, \ldots, G_n, \ldots\}$, что алгоритм P_n не может решить $\mathcal{SC}_3$ для G_n для всех n. Более того, γ' упорядочена по возрастанию числа вершин в графах. Теперь можно расширить последовательность γ' до последовательности графов γ с графами G_n для всех размеров n. Из построения γ следует, что не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{SC}_3(\gamma)$. ■

Отметим, что множество всех входов размера n для проблемы $\mathcal{SC}_3(\gamma)$ выглядит так: $I_n = \{G : G \cong G_n, G_n \in \gamma\}$.

Теорема 2. Пусть γ — произвольная последовательность графов. Если существует генерический полиномиальный алгоритм, решающий проблему $\mathcal{SC}_3(\gamma)$, то существует вероятностный полиномиальный алгоритм, решающий эту проблему на всём множестве входов.

Доказательство. Допустим, что существует генерический полиномиальный алгоритм $\mathcal{A}$, решающий проблему поиска 3-раскраски графов $\mathcal{SC}_3(\gamma)$. Построим вероятностный полиномиальный алгоритм $\mathcal{B}$, решающий эту проблему на всём множестве входов. На графе G с n вершинами алгоритм $\mathcal{B}$ работает следующим образом:

1) Запускает алгоритм $\mathcal{A}$ на G.
2) Если $\mathcal{A}(G) \neq \square$, то $\mathcal{B}$ выдаёт ответ $\mathcal{A}(G)$ и останавливается, иначе идёт на шаг 3.
3) Генерирует случайно и равномерно перестановку π на вершинах $\{1, \ldots, n\}$ и вычисляет граф $G' = \pi(G)$.
4) Запускает алгоритм $\mathcal{A}$ на графе G'.
5) Если $\mathcal{A}(G') = \square$, то выдаёт ответ $V_1 = \{1, 2, \ldots, [n/3]\}$, $V_2 = \{[n/3] + 1, \ldots, [2n/3]\}$, $V_3 = \{[2n/3] + 1, \ldots, n\}$ (возможно, неправильный).
6) Пусть $\mathcal{A}(G') = \{V_1, V_2, V_3\}$ — решение задачи 3-раскраски для графа G'. Тогда

$$\pi(V) = \pi^{-1}(V_1) \sqcup \pi^{-1}(V_2) \sqcup \pi^{-1}(V_3)$$

является решением задачи 3-раскраски для исходного графа $G = \pi^{-1}(G')$.

Для доказательства корректности работы вероятностного алгоритма надо показать, что вероятность того, что $A(G') = \square$, меньше $1/3$. Заметим, что $\pi(G)$ при варьировании перестановки π пробегает всё множество входов размера n. Множество $\{G : A(G) = \square\}$ пренебрежимо, поэтому вероятность того, что $A(G') = \square$, стремится к 0 при увеличении n. ■

Теорема 3. Если P $\neq$ NP и P = BPP, то существует последовательность графов γ, такая, что для решения проблемы поиска 3-раскраски $\mathcal{SC}_3(\gamma)$ не существует генерического полиномиального алгоритма.

Доказательство. Покажем сначала, что при условиях P $\neq$ NP и P = BPP не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{SC}_3$. Действительно, пусть такой алгоритм существует. Так как проблема $\mathcal{SC}_3$ является NP-трудной, то существует полиномиально эквивалентная ей NP-проблема распознавания A. Из полиномиального вероятностного алгоритма для $\mathcal{SC}_3$ легко получается полиномиальный вероятностный алгоритм для решения проблемы A. А так как P = BPP, то существует и детерминированный полиномиальный алгоритм для A, откуда P = NP. Противоречие. Теперь нужное утверждение следует из леммы 1 и теоремы 2. ■

Авторы выражают благодарность рецензенту за полезные замечания и предложения по улучшению текста статьи.

ЛИТЕРАТУРА

1. *Karp R.* Reducibility among combinatorial problems // R. E. Miller, J. W. Thatcher, and J. D. Bohlinger (eds). Complexity of Computer Computations. The IBM Research Symposia Series. Boston: Springer, 1972. P. 85–103.
2. *Blum M.* How to prove a theorem so no one else can claim it // Proc. ICM'86. AMS, 1986. P. 1444–1451.
3. *Goldreich O., Micali S., and Wigderson A.* Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems // Proc. SFCS'86. IEEE Computer Society, 1986. P. 174–187.
4. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks // J. Algebra. 2003. V. 264. No. 2. P. 665–694.
5. *Рыбалов А. Н.* О генерической сложности проблемы распознавания квадратичных вычетов // Прикладная дискретная математика. 2015. № 2 (28). С. 54–58.
6. *Рыбалов А. Н.* О генерической сложности проблемы дискретного логарифма // Прикладная дискретная математика. 2016. № 3 (33). С. 93–97.
7. *Рыбалов А. Н.* О генерической сложности проблемы извлечения корня в группах вычетов // Прикладная дискретная математика. 2017. № 38. С. 95–100.
8. *Impagliazzo R. and Wigderson A.* P = BPP unless E has subexponential circuits: Derandomizing the XOR Lemma. Proc. 29th STOC. El Paso: ACM, 1997. P. 220–229.
9. *Китаев А., Шень А., Вялый М.* Классические и квантовые вычисления. М.: МЦНМО, ЧеРо. 1999. 192 с.

REFERENCES

1. *Karp R.* Reducibility among combinatorial problems. R. E. Miller, J. W. Thatcher, and J. D. Bohlinger (eds). Complexity of Computer Computations. The IBM Research Symposia Series, Boston, Springer, 1972, pp. 85–103.
2. *Blum M.* How to prove a theorem so no one else can claim it. Proc. ICM'86, AMS, 1986, pp. 1444–1451.
3. *Goldreich O., Micali S., and Wigderson A.* Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. Proc. SFCS'86, IEEE Computer Society, 1986, pp. 174–187.
4. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks. J. Algebra, 2003, vol. 264, no. 2, pp. 665–694.

5. *Rybalov A. N.* O genericheskoy slozhnosti problemy raspoznavaniya kvadratichnykh vychetov [On generic complexity of the quadratic residuosity problem]. Prikladnaya Diskretnaya Matematika, 2015, no. 2 (28), pp. 54–58. (in Russian)
6. *Rybalov A. N.* O genericheskoy slozhnosti problemy diskretnogo logarifma [On generic complexity of the discrete logarithm problem]. Prikladnaya Diskretnaya Matematika, 2016, no. 3 (33), pp. 93–97. (in Russian)
7. *Rybalov A. N.* O genericheskoy slozhnosti problemy izvlecheniya kornya v gruppakh vychetov [On generic complexity of the problem of finding roots in groups of residues]. Prikladnaya Diskretnaya Matematika, 2017, no. 38, pp. 95–100. (in Russian)
8. *Impagliazzo R. and Wigderson A.* P = BPP unless E has subexponential circuits: Derandomizing the XOR Lemma. Proc. 29th STOC, El Paso, ACM, 1997, pp. 220–229.
9. *Kitaev A., Shen' A., and Vyalyy M.* Klassicheskie i kvantovye vychisleniya [Classical and Quantum Computations]. Moscow, MCCME, 1999. 192 p. (in Russian)