ISSN 2949-5598

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

Том 30 № 3 2023

Новосибирск
Издательство Института математики

Учредители журнала Сибирское отделение РАН
Институт математики им. С. Л. Соболева СО РАН

Журнал включён в базу данных Russian Science Citation Index (RSCI) на платформе Web of Science. Переводы статей на английский язык публикуются в *Journal of Applied and Industrial Mathematics* и доступны по ссылке `www.springer.com/mathematics/journal/11754`.

СИБИРСКОЕ ОТДЕЛЕНИЕ РОССИЙСКОЙ АКАДЕМИИ НАУК
ИНСТИТУТ МАТЕМАТИКИ им. С. Л. СОБОЛЕВА СО РАН

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

Выпускается с 1994 г. Научный журнал 4 номера в год

Том 30, № 3 (157) **Июль–сентябрь 2023**

СОДЕРЖАНИЕ

Бахарев А. О. *Оценки сложности реализации квантового криптоанализа постквантовых криптосистем, основанных на решётках* **5**

Береснев В. Л., Мельников А. А. *Дополнительные ограничения для динамической задачи конкурентного размещения* **43**

Быков Д. А., Коломеец Н. А. *О нижней оценке числа бент-функций на минимальном расстоянии от бент-функции из класса Мэйорана — МакФарланда* **57**

Вялый М. Н., Карпов В. Е. *Представления рёбер гиперграфов обобщёнными путями* **81**

Пяткин А. В. *Полиномиальные аппроксимационные схемы для задач выбора векторов и кластеризации с разными центрами* **96**

Талецкий Д. С. *О деревьях заданного диаметра с экстремальным количеством k-дистанционных независимых множеств* **111**

НОВОСИБИРСК
ИЗДАТЕЛЬСТВО ИНСТИТУТА МАТЕМАТИКИ

В журнале публикуются оригинальные научные статьи и обзоры теоретической и прикладной направленности по следующим разделам дискретного анализа, исследования операций и информатики:

- дискретная оптимизация
- комбинаторика
- контроль и надёжность дискретных устройств
- математические модели и методы принятия решений
- математическое программирование
- модели экономики
- моделирование процессов управления
- построение и анализ алгоритмов
- синтез и сложность управляющих систем
- теория автоматов
- теория графов
- теория игр и её приложения
- теория кодирования
- теория расписаний и размещений

Адрес редакции:
Институт математики
им. С. Л. Соболева СО РАН,
пр. Академика Коптюга, 4,
630090 Новосибирск, Россия
Телефон: +7 (383) 329–75–79
E-mail: discopr@math.nsc.ru

SIBERIAN BRANCH OF THE RUSSIAN ACADEMY OF SCIENCES
SOBOLEV INSTITUTE OF MATHEMATICS

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII

/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/

Published since 1994 | Scientific journal | 4 issues per year

Vol. 30, No. 3 (157) | **July–September, 2023**

CONTENTS

A. O. Bakharev. *Estimates of implementation complexity for quantum cryptanalysis of post-quantum lattice-based cryptosystems* **5**

V. L. Beresnev and **A. A. Melnikov.** *Additional constraints for dynamic competitive facility location problem* **43**

D. A. Bykov and **N. A. Kolomeec.** *On a lower bound for the number of bent functions at the minimum distance from a bent function in the Maiorana–McFarland class* **57**

M. N. Vyalyi and **V. E. Karpov.** *Hypergraph edge representations with the use of homological paths* **81**

A. V. Pyatkin. *PTAS for vector choice and clustering with different centers problems* **96**

D. S. Taletskii. *On trees with given diameter and extremal number of k-distance independent sets* **111**

NOVOSIBIRSK
SOBOLEV INSTITUTE PRESS

In this journal we publish original research papers and survey papers of both theoretical and practical importance on the following topics of discrete analysis, operations research and informatics:

- discrete optimization
- combinatorics
- control and reliability of discrete devices
- decision making models and methods
- mathematical programming
- economic models
- management modeling
- design and analysis of algorithms
- synthesis and complexity of control systems
- automata theory
- graph theory
- game theory and its applications
- coding theory
- theory of scheduling and facility location

Editorial office address:
Sobolev Institute of Mathematics,
4 Acad. Koptyug Avenue,
630090 Novosibirsk, Russia
Phone: +7 (383) 329–75–79
E-mail: discopr@math.nsc.ru

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль–сентябрь 2023. Т. 30, № 3. С. 5–42

УДК 519.7 DOI 10.33048/daio.2023.30.756

ОЦЕНКИ СЛОЖНОСТИ РЕАЛИЗАЦИИ КВАНТОВОГО КРИПТОАНАЛИЗА ПОСТКВАНТОВЫХ КРИПТОСИСТЕМ, ОСНОВАННЫХ НА РЕШЁТКАХ

А. О. Бахарев

Новосибирский гос. университет,
ул. Пирогова, 2, 630090 Новосибирск, Россия
E-mail: a.bakharev@g.nsu.ru

Аннотация. В силу развития квантовых вычислений возникает необходимость в разработке и анализе криптосистем, устойчивых к атакам с использованием квантового компьютера — алгоритмов постквантовой криптографии. Стойкость многих известных постквантовых криптосистем, основанных на теории решёток, базируется на сложности решения проблемы нахождения кратчайшего вектора в решетке (SVP). В настоящей статье разработана и описана модель квантового оракула из алгоритма Гровера для реализации гибридного квантово-классического алгоритма на основе GaussSieve, который может быть использован для атак на криптосистемы, стойкость которых зависит от решения задачи SVP. Получены выражения для верхних оценок числа кубитов и глубины схемы двух реализаций предложенной модели квантового оракула: минимизирующей число кубитов и минимизирующей глубину схемы. Проанализирована сложность реализации предложенной модели квантового оракула для атаки на постквантовые криптосистемы, основанные на решётках и являющиеся финалистами конкурса постквантовой криптографии NIST. Табл. 6, ил. 12, библиогр. 34.

Ключевые слова: квантовый поиск, криптография с открытым ключом, криптография на решётках, постквантовая криптография, алгоритм Гровера, квантовые вычисления.

Введение

Квантовые вычисления — это быстроразвивающаяся область компьютерных исследований, которая ставит под угрозу криптографическую

Исследование выполнено при поддержке Математического центра в Академгородке в рамках соглашения с Министерством науки и высшего образования России (соглашение № 075–15–2022–282).

стойкость стандартов шифрования, используемых в настоящее время, поэтому возникает необходимость в разработке и анализе криптосистем, которые будут устойчивы к атакам с использованием квантового компьютера. Такие криптосистемы называются постквантовыми. Важными направлениями анализа постквантовых криптосистем являются построение и исследование квантовых схем, которые могут быть использованы для атак на данные криптосистемы. Ключевыми параметрами квантовых схем являются число кубитов, число вентилей и глубина схемы. Известен ряд работ, в которых анализируется сложность реализации квантовых атак на симметричные криптосистемы, такие как действующий стандарт шифрования РФ ГОСТ Р 34.12–2015 («Кузнечик» и «Магма») [1, 2], действующий стандарт шифрования США AES (Advanced Encryption Standard) [3–6], предыдущий стандарт шифрования США DES (Data Encryption Standard) [7, 8].

Для построения постквантовых криптосистем с открытым ключом используются математические объекты разной природы. Большинство задач, на которых основывается стойкость постквантовых криптосистем, являются частными случаями NP-трудных задач. Популярные подходы [9] основаны на использовании решёток, кодов, хеш-функций и многочленов от многих переменных. Однако нельзя утверждать, что криптосистемы, построенные на этих подходах, полностью защищены от атак с использованием квантового или обычного компьютера. Именно поэтому криптографическое сообщество тратит большое количество времени, проводя криптоанализ различных криптосистем. Иногда эти исследования приводят к нахождению атаки, которая демонстрирует, что система не защищена, или вынуждает увеличить размер ключей.

В 2016 г. Национальный институт стандартов и технологий США (NIST), принимающий новые стандарты шифрования, объявил конкурс Post-Quantum Cryptography Competition, по завершении которого будет принят новый — квантово-устойчивый — стандарт асимметричного шифрования. В первом раунде, начавшемся 30 ноября 2017 г., приняли участие 69 схем шифрования и цифровой подписи из разных стран. В финал конкурса [10], который начался 22 июня 2020 г., прошли четыре схемы шифрования, три из которых основаны на теории решёток и одна — на кодах, исправляющих ошибки. Известен ряд работ, в которых исследуется сложность реализации квантовых схем, используемых при криптоанализе постквантовых криптосистем [11, 12].

В настоящей работе для анализа были выбраны криптосистемы, основанные на решётках. Одной из задач в теории решёток является задача нахождения кратчайшего вектора (shortest vector problem, SVP), которая заключается в нахождении ненулевого вектора наименьшей длины, в решётке, заданной своим базисом. В общем случае задача SVP NP-

трудна [13]. Стойкость систем, основанных на решётках, в большинстве случаев зависит от эффективности решения задачи SVP, так как большинство известных атак сводятся к решению этой проблемы. Известен ряд алгоритмов, решающих данную задачу: GaussSieve [14], ListSieve-Birthday [15], NV-Sieve [16], HashSieve [17], диаграммы Вороного [18], Гауссова выборка [19], Triple sieve [20], Spherical LSF [21]. Перспективными являются разработка и анализ квантовых алгоритмов, которые позволяют ускорить решение задачи SVP.

1. Основные определения и понятия

Пусть $\mathbb{F}_2$ — поле, состоящее из двух элементов, а $\mathbb{F}_2^n$ — векторное пространство размерности n над полем $\mathbb{F}_2$. *Весом* $wt(x)$ двоичного вектора $x \in \mathbb{F}_2^n$ называется число его ненулевых координат. Введём отношение частичного порядка $\preceq$ на множестве $\mathbb{F}_2^n$ следующим образом:

$$x \preceq y \Leftrightarrow x_i \leqslant y_i \text{ для любого } i \in \overline{1,n}, \quad x, y \in \mathbb{F}_2^n.$$

Произвольная функция $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ называется *булевой функцией* от n переменных. Через $\oplus$ будем обозначать операцию сложения по модулю 2, т. е. $a \oplus b = (a+b) \bmod 2$. Любая булева функция f от n переменных единственным образом представляется в виде

$$f(x_1, \ldots, x_n) = \bigoplus_{k=1}^{n} \bigoplus_{i_1,\ldots,i_k} a_{i_1 \ldots i_k} x_{i_1} \ldots x_{i_k} \oplus a_0,$$

где при каждом k все индексы $i_1, \ldots, i_k$ различны и $a_0, a_{i_1 \ldots i_k} \in \mathbb{F}_2$. Такое представление называется *полиномом Жегалкина*, или *алгебраической нормальной формой* функции f. Степенью булевой функции $\deg(f)$ называется число переменных в самом длинном слагаемом АНФ булевой функции f. *Векторной булевой функцией* называется произвольное отображение $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$. Векторную булеву функцию $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ всегда можно представить в виде $F(x) = (f_1(x), \ldots, f_m(x))$, где f_j — булева функция от n переменных. Функции f_j называют *координатными*.

1.1. Решётки. Термин «решётка» встречается в различных областях математики: алгебре, геометрии, теории графов и др. В настоящей работе используется

Определение 1. Пусть $u_1, \ldots, u_d \in \mathbb{R}^n$, $d \leqslant n$, — линейно независимые векторы. *Решёткой* размерности d называется множество

$$\Lambda = \left\{ \sum_{i=1}^{d} b_i u_i \mid b_i \in \mathbb{Z} \right\}.$$

Линейно независимая система векторов, порождающая решётку, называется *базисом решётки*.

Пусть $p \geqslant 1$ — вещественное число. Тогда для вектора $x = (x_1, \ldots, x_n)$ в $\mathbb{R}^n$ *норма* l_p равна

$$\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p \right)^{1/p}.$$

Определение 2. *Задача нахождения кратчайшего вектора* (SVP) — найти в заданной своим базисом решётке ненулевой вектор, имеющий наименьшую длину относительно нормы l_p.

В данной работе рассматривается вариант задачи SVP в евклидовой норме l_2. Далее для удобства вместо $\|\cdot\|_2$ будем писать $\|\cdot\|$. Перспективными являются разработка и анализ квантовых алгоритмов, которые позволяют ускорить решение данной задачи.

В 2010 г. в [14] был предложен алгоритм GaussSieve — один из самых эффективных классических алгоритмов, решающих задачу SVP.

Алгоритм 1. Алгоритм GaussSieve (Миччанчо, Вулгарис, 2010)

Вход: B — базис решётки
Выход: v — кратчайший вектор решётки

Инициализировать пустой неупорядоченный список L и пустой стек S
repeat
  Получить вектор v из стека (или сгенерировать новый)
  while $w \leftarrow$ ПОИСК$\{w \in L \mid \|v \pm w\| \leqslant \|v\|\}$ **do**
    Уменьшить v с помощью w $(v \leftarrow v \pm w)$
  while $w \leftarrow$ ПОИСК$\{w \in L \mid \|w \pm v\| \leqslant \|w\|\}$ **do**
    Удалить w из списка L
    Уменьшить w с помощью v $(w \leftarrow w \pm v)$
    Добавить w в стек S
  if v изменился **then**
    Добавить v в стек S
  else
    Добавить v в список L
until v — кратчайший вектор
return вектор v

На вход алгоритма 1 поступает базис решётки, на основе которого будут строиться новые векторы при условии пустого стека S. Функция «ПОИСК» перебирает векторы w в списке и проверяет их на одно из условий поиска:

$$\|v \pm w\| \leqslant \|v\| \quad \text{или} \quad \|w \pm v\| \leqslant \|w\|. \tag{1}$$

Если такой вектор существует, то функция возвращает его, иначе функция прерывает первый цикл, в котором находится. Авторами [14] предложено эвристическое условие остановки, основанное на численных экспериментах. Таким образом, алгоритм работает до тех пор, пока не сработает эвристическое условие остановки.

Так как длина списка L увеличивается экспоненциально с ростом размерности решётки, самой трудозатратной операцией данного алгоритма является функция «ПОИСК». В рамках предложенного в [22] подхода ускорение достигается за счёт использования в функции «ПОИСК» квантового алгоритма поиска.

1.2. Задача поиска. Задача, решаемая в функции «ПОИСК», называется *задачей поиска*. Предполагается, что есть неупорядоченный список из K элементов, в котором как минимум один элемент удовлетворяет некоторому условию. Требуется найти по крайней мере один такой элемент. Другими словами, определена булева функция f, которая по номеру элемента (его двоичному представлению x) определяет, является ли элемент подходящим. Если элемент подходящий, то $f(x) = 1$, иначе $f(x) = 0$. В такой постановке задача поиска сводится к нахождению решения уравнения $f(x) = 1$.

В классическом варианте при условии, что решение одно, требуется $\sim K/2$ обращений к функции f для нахождения решения. Квантовый алгоритм поиска элемента в неупорядоченном списке (алгоритм Гровера [23]) решает данную задачу за $\sim \frac{\pi}{4}\sqrt{K}$ обращений к *оракулу* — квантовому аналогу функции f. О том, как булева функция моделируется на квантовом компьютере, будет рассказано далее.

2. Квантовые вычисления

2.1. Кубит. Квантовый компьютер, в отличие от обычного, оперирует *квантовыми битами* (*кубитами* [24]). Подобно классическому биту, который может находиться в состоянии 0 или 1, кубит имеет возможные состояния $|0\rangle$ и $|1\rangle$. Здесь используется *дираковское обозначение* $|\cdot\rangle$, которое является стандартным обозначением состояния в квантовой механике. Различие между битами и кубитами в том, что кубит может находиться в состоянии, отличном от $|0\rangle$ или $|1\rangle$. Можно составить *линейную комбинацию состояний* (*суперпозицию*):

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle.$$

Числа α и β комплексные, и $|\alpha|^2+|\beta|^2 = 1$. Иначе говоря, состояние одного кубита можно представить как единичный вектор из $\mathbb{C}^2$. Однако мы не можем измерить кубит, чтобы определить его квантовое состояние,

т. е. значения α и β. Из квантовой механики следует, что при измерении кубита мы получаем либо результат $|0\rangle$ с вероятностью $|\alpha|^2$, либо результат $|1\rangle$ с вероятностью $|\beta|^2$.

Подобно случаю одиночного кубита, система двух кубитов имеет четыре *состояния вычислительного базиса*, обозначаемых как $|00\rangle$, $|01\rangle$, $|10\rangle$ и $|11\rangle$, где для любых $x, y \in \mathbb{F}_2$ выполнено $|xy\rangle \equiv |x\rangle|y\rangle$. Тогда вектор состояния, описывающий два кубита, имеет вид

$$|\varphi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle,$$

где $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$. Систему с произвольным числом кубитов описывает

Постулат 1. С каждой изолированной физической системой связывается комплексное векторное пространство со скалярным произведением, которое называется *пространством состояний* системы. Система полностью описывается *вектором состояния*, который представляет собой единичный вектор в пространстве состояний системы.

Для квантовых схем будем применять обозначения, представленные на рис. 1. В этих обозначениях временная ось направлена слева направо.

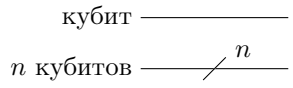


Рис. 1. Обозначения кубита и n кубитов

Постоянными будем называть кубиты, которые используются на протяжении всей работы оракула, а *временными* — те, которые используются только во время проведения промежуточных операций.

2.2. Эволюция квантовомеханической системы. Изменение состояния $|\psi\rangle$ квантовомеханической системы во времени описывает

Постулат 2. Эволюция замкнутой квантовой системы описывается унитарным преобразованием. Другими словами, состояние $|\psi\rangle$ системы в момент времени t_1 связано с её состоянием $|\psi'\rangle$ в момент времени t_2 посредством унитарного оператора U, зависящего только от моментов времени t_1 и t_2:

$$|\psi'\rangle = U|\psi\rangle. \tag{2}$$

Пусть $|\psi\rangle = |x_1, x_2, \ldots, x_k\rangle$ и $|\psi'\rangle = |y_1, y_2, \ldots, y_k\rangle$. Тогда равенство (2) можно переписать в обозначениях квантовых схем, как это показано на рис. 2.

Базовое преобразование квантового компьютера будем называть *вентилем*. Для того чтобы описать работу вентиля, достаточно указать

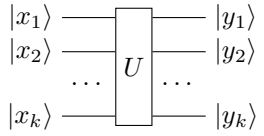
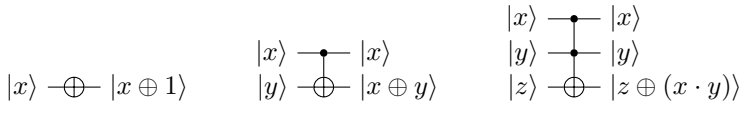


Рис. 2. Квантовая схема равенства (2)

принцип работы данного вентиля на вычислительном базисе кубитов, на которых он действует. В настоящей работе для построения всех операций и функций на квантовом компьютере используются базисные вентили, представленные на рис. 3 ($x, y, z \in \mathbb{F}_2$).



а) б) в)

Рис. 3. Используемые вентили: а) вентиль Паули-X (NOT); б) вентиль CNOT; в) вентиль Тоффоли (CCNOT)

Обозначение процесса измерения кубита в квантовых схемах изображено на рис. 4. Здесь $|\psi\rangle = |0\rangle$ с вероятностью $|\alpha|^2$ и $|\psi\rangle = |1\rangle$ с вероятностью $|\beta|^2$.

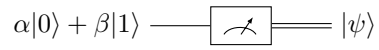


Рис. 4. Измерение кубита

Пусть $x \in \mathbb{F}_2^n$, $y \in \mathbb{F}_2$ и U_f — квантовый аналог булевой функции f от n переменных. Тогда действие U_f на кубитах $|x\rangle$ и $|y\rangle$

$$U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$$

моделируется схемой на рис. 5.

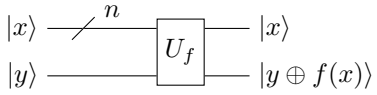


Рис. 5. Моделирование булевой функции на квантовом компьютере

Важным оператором, используемым во многих квантовых алгоритмах, является оператор Адамара H, изображённый на рис. 6. В матричном представлении данный оператор имеет следующий вид:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

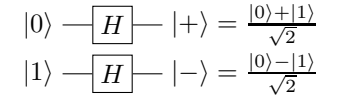


Рис. 6. Вентиль Адамара

Определим *глубину квантовой схемы* [24] как число слоёв, которые содержит схема. Один слой состоит из базисных вентилей, применённых к непересекающимся множествам кубитов.

Так как информация, записанная во временных кубитах, не нужна после получения результата промежуточной операции, данные кубиты нужно очистить для возможности их дальнейшего использования. *Очистка кубитов* заключается в применении в обратном порядке ранее используемых вентилей. Значит, для очистки всех временных кубитов необходимо после получения результата операции применить в обратном порядке все ранее применённые вентили, которые не участвуют в изменении выходных кубитов операции.

2.3. Алгоритм Гровера. *Квантовый параллелизм* — это фундаментальное свойство многих квантовых компьютеров, позволяющее вычислять функцию $f(x)$ для различных значений x одновременно. Для понимания работы квантового параллелизма рассмотрим следующие рассуждения. Результатом применения преобразования Адамара к n кубитам, изначально находящимися в состоянии $|0^{\otimes n}\rangle$, будет состояние

$$H^{\otimes n}|00\dots0\rangle = \frac{1}{2^{n/2}} \sum_{x\in\mathbb{F}_2^n} |x\rangle.$$

Это действие обозначим через $H^{\otimes n}$. Иными словами, преобразование Адамара приводит к суперпозиции всех состояний вычислительного базиса с одинаковыми коэффициентами. Тогда параллельное вычисление булевой функции $f(x)$ от n переменных может быть выполнено следующим образом. Приготавливаем $n+1$ кубитов в состоянии $|0^{\otimes n}, 0\rangle$, затем применяем к первым n кубитам преобразование Адамара, после чего задействуем квантовую схему, реализующую U_f. Это даёт состояние

$$U_f\left[\left(\frac{1}{2^{n/2}} \sum_{x\in\mathbb{F}_2^n} |x\rangle\right)|0\rangle\right] = \frac{1}{2^{n/2}} \sum_{x\in\mathbb{F}_2^n} |x, f(x)\rangle. \tag{3}$$

Однако этим параллелизмом нельзя воспользоваться непосредственно. Измерение состояния (3) даёт значение $f(x)$ только для одного x. Для получения пользы от квантового параллелизма нужно иметь возможность «извлекать» информацию о более чем одном значении $f(x)$ из суперпозиции состояний (3).

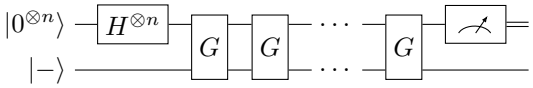


Рис. 7. Алгоритм Гровера

Квантовым алгоритмом, решающим задачу поиска, является алгоритм Гровера [23] (рис. 7), в котором G обозначает итерацию Гровера (рис. 8). Преобразование «Фаза» является известным вентилем, в отличие от вентиля «Оракул», который строится под каждую задачу отдельно.

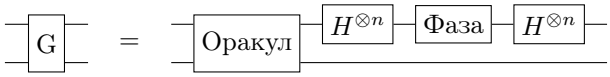


Рис. 8. Итерация Гровера

Рассмотрим принцип работы алгоритма Гровера. Обозначим через $|\psi\rangle$ состояние, поступившее на вход первой итерации Гровера. Пусть $\beta \in \mathbb{F}_2^n$ — это единственное решение уравнения $f(x) = 1$. Тогда «Оракул» O_f действует на состояние $|x\rangle|-\rangle$, где $x \in \mathbb{F}_2^n$, следующим образом:

$$O_f|x\rangle|-\rangle = \begin{cases} -|x\rangle|-\rangle, & \text{если } x = \beta, \\ |x\rangle|-\rangle & \text{иначе.} \end{cases}$$

Определим состояние $|\alpha\rangle$ так:

$$|\alpha\rangle \equiv \frac{1}{\sqrt{2^n - 1}} \sum_{x \neq \beta} |x\rangle.$$

Тогда состояние $|\psi\rangle$ можно записать в виде

$$|\psi\rangle = \sqrt{\frac{2^n - 1}{2^n}}|\alpha\rangle + \sqrt{\frac{1}{2^n}}|\beta\rangle.$$

Обозначим через $|\psi_k\rangle$ состояние, полученное из $|\psi\rangle$ после применения итерации Гровера G k раз. Тогда состояние $|\psi_k\rangle$ определяется через состояние $|\psi_{k-1}\rangle$, где $|\psi_0\rangle = |\psi\rangle$, следующим образом. «Оракул» производит отражение относительно вектора $|\alpha\rangle$ в плоскости, задаваемой векторами $|\alpha\rangle$ и $|\beta\rangle$, что соответствует состоянию $O|\psi_{k-1}\rangle$. Применение остальных преобразований итерации Гровера производит отражение в плоскости, задаваемой векторами $|\alpha\rangle$ и $|\beta\rangle$, относительно вектора $|\psi\rangle$, что соответствует состоянию $G|\psi_{k-1}\rangle = |\psi_k\rangle$. Если обозначить через $\theta/2$ угол между векторами $|\alpha\rangle$ и $|\psi\rangle$, то можно сказать, что итерация Гровера G представляет собой поворот вектора состояний на угол θ к вектору $|\beta\rangle$ в двумерном пространстве, натянутом на векторы $|\alpha\rangle$ и $|\beta\rangle$.

В [25] представлен подход, использующий алгоритм Гровера и позволяющий решать задачу поиска при неизвестном числе элементов, удовлетворяющих условию поиска.

3. Сложность реализации некоторых операций на квантовом компьютере

В данном разделе предлагаются и анализируются реализации операций, используемые для построения рассмотренной в разд. 4 модели квантового оракула.

Приведём определения прямого и дополнительного кодов числа. *Прямой код* — это представление числа в двоичной системе счисления, при котором последний (старший) бит отводится под знак числа. Если число положительное, то в старший бит записывается 0; если число отрицательное, то в старший бит записывается 1. Таким образом, запись числа $x = (x_1, \ldots, x_m)$ означает, что $x = (-1)^{x_m}(2^{m-2}x_{m-1} + \cdots + 2x_2 + x_1)$.

В *дополнительном коде*, так же как и прямом, последний бит отводится для записи знака числа. Представление положительных чисел в дополнительном коде совпадает с представлением в прямом коде, а представление отрицательных чисел получается из прямого кода следующим образом. Все биты, кроме старшего, в дополнительном коде сначала инвертируются, после чего к получившемуся числу прибавляется единица.

Рассмотрим, например, представление числа $-6 = (0111)$ в прямом коде. Инверсией всех битов, кроме старшего, получим (1001), после чего добавим единицу: $(1000) + (1001) = (0101)$. Полученная двоичная запись является представлением числа -6 в дополнительном коде. Если известно, что число положительное, то его двоичная запись рассматривается без бита, отведённого под запись знака.

Далее в этом разделе будем опускать скобки $|\cdot\rangle$ в обозначении кубитов, если из контекста ясно, о чём идёт речь

3.1. Сравнение двух целых положительных чисел. Необходимо сравнить два целых положительных числа a и b, двоичная запись которых занимает m битов. Рассмотрим предлагаемый вариант реализации операции сравнения (рис. 9). Определим

$$\text{flag} = \begin{cases} 0, & \text{если найден бит различия,} \\ 1 & \text{иначе.} \end{cases}$$

Начальная инициализация состояний $|\text{flag}\rangle$ и $|\text{output}\rangle$ равна $|1\rangle$, а начальная инициализация $|y\rangle$ равна $|0^{\otimes m}\rangle$. Первые два шага схемы заключаются в том, чтобы приравнять y_m к $a_m \oplus b_m$. Тогда $y_m = 1$, если $a_m \neq b_m$, и $y_m = 0$, если $a_m = b_m$. Далее с помощью оператора CNOT присваиваем flag значение $\text{flag} \oplus y_m$. Таким образом, $\text{flag} = 1$, если $a_m = b_m$, и $\text{flag} = 0$,

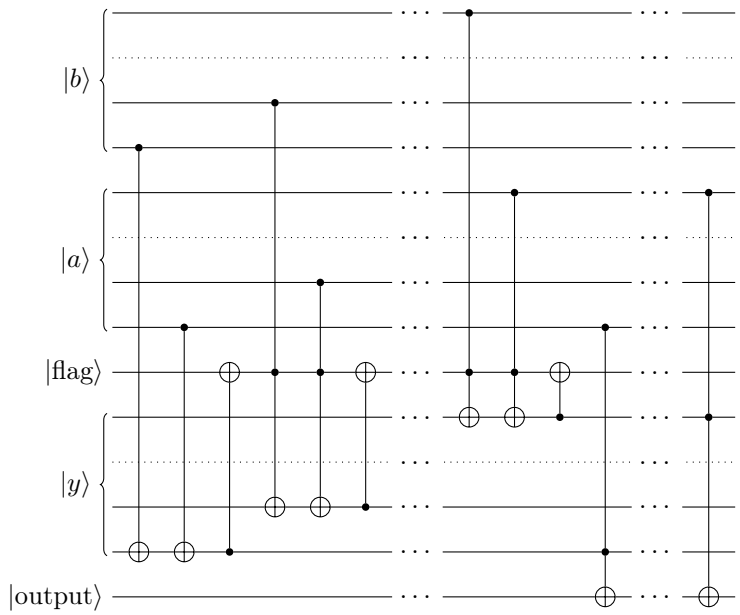


Рис. 9. Схема реализации операции сравнения

если $a_m \neq b_m$. Затем $y_{m-1} = \text{flag} \cdot (a_{m-1} \oplus b_{m-1})$ и $\text{flag} = \text{flag} \oplus y_{m-1}$. Дальше операции для y_i, $m-1 > i \geqslant 1$, аналогичны операциям для y_{m-1}. Такое построение гарантирует, что как только впервые найдётся бит i, в котором различаются a_i и b_i, на месте y_i появится 1, а остальные y_j, где $i \neq j$, останутся равными 0. Далее каждый a_i соединяем с y_i и направляем в output с помощью CCNOT. Тогда $\text{output} = 1 \oplus a_k$, где k — номер первого найденного бита различия a и b. Следовательно,

$$\text{output} = \begin{cases} 1, & \text{если } a \leqslant b, \\ 0 & \text{иначе.} \end{cases}$$

В конце операции проводим очистку всех кубитов, используемых для состояний $|\text{flag}\rangle$ и $|y\rangle$. Тогда данная операция будет использовать 1 постоянный и $m+1$ временных кубитов. При этом общее число вентилей равно $7m$ (из них $2m+4$ вентилей CNOT и $5m-4$ вентилей CCNOT). Глубина предложенной схемы совпадает с общим числом вентилей, так как все вентили действуют последовательно, и равна $7m$.

3.2. Отрицание и перевод в прямой код целого числа. Пусть есть число a, представленное в двоичном дополнительном коде длины m. Необходимо получить $-a$. Тогда все биты a нужно инвертировать, а затем прибавить к полученному представлению 1. Рассмотрим квантовую

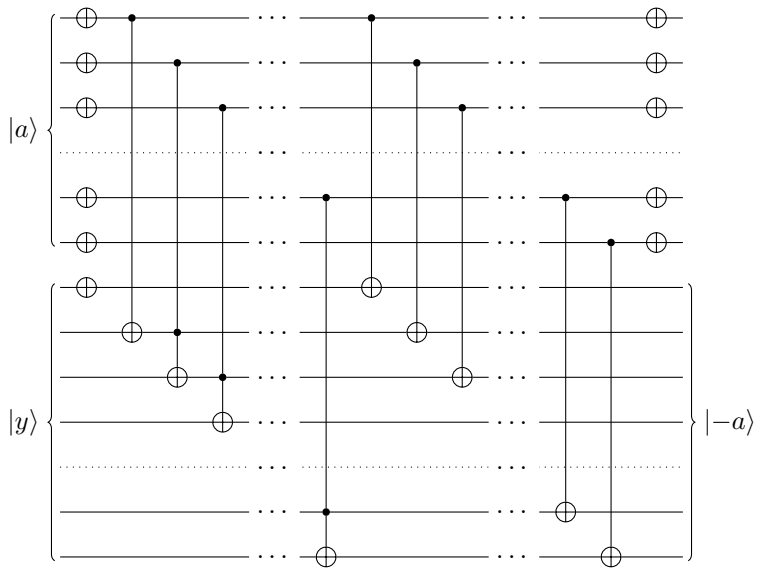


Рис. 10. Отрицание целого числа в дополнительном коде

схему на рис. 10. Здесь $|y\rangle = |0^{\otimes m}\rangle$ — кубиты, отведённые под выход функции. Сначала инвертируем кубиты числа a с помощью вентилей NOT. Затем присвоим $y_1 = 1$, $y_2 = a_1$, $y_3 = a_1 a_2$, $y_4 = y_3 a_3$, …, $y_m = y_{m-1} a_{m-1}$. После к каждому y_i добавим a_i, используя вентиль CNOT, $i = 1, \ldots, m$, и наконец, с помощью вентилей NOT инвертируем все кубиты числа a ещё раз. Тогда число постоянных и временных кубитов, достаточных для получения $-a$, равно m и 0 соответственно. При этом используется $4m$ вентилей (из них $2m + 1$ вентилей NOT, $m + 1$ вентилей CNOT и $m - 2$ вентилей CCNOT). Рассмотрим глубину данной схемы. Первый слой будет состоять из применения вентилей NOT ко всем кубитам числа a и кубиту y_1. Следующие $m - 1$ слоёв состоят из последовательного применения вентиля CNOT, направленного к кубиту y_2, и вентилей CCNOT, направленных к кубитам $y_3, \ldots, y_m$. Следующий слой состоит из вентилей CNOT, направленных к кубитам $y_1, \ldots, y_m$ и применённых одновременно. На последнем слое вентили NOT применяются ко всем кубитам числа a. В итоге глубина такой схемы равна $m + 2$.

Подобным образом можно перевести число из дополнительного кода в прямой. Так как для положительного числа эти коды совпадают, необходимо переводить только отрицательные числа. Тогда квантовая схема перевода двоичного числа из дополнительного кода в прямой будет подобна получению отрицательного числа, представленного в двоичном дополнительном коде, но с некоторыми изменениями:

• кубит a_m, отвечающий за знак, не меняется (убираем два вентиля NOT);

• отсутствует вентиль CCNOT, идущий в y_m;

• все вентили NOT заменяются вентилями CNOT, контролируемыми знаковым кубитом a_m;

• первый вентиль CNOT, контролируемый a_1, меняется на CCNOT, контролируемый a_1 и a_m.

Тогда число постоянных кубитов, необходимых для реализации операции перевода двоичного числа из дополнительного кода в прямой, равно m. При этом используется $4m-3$ вентилей (из них $3m-1$ вентилей CNOT и $m-2$ вентилей CCNOT). Рассмотрим глубину данной схемы. Первые m слоёв состоят из последовательно применённых вентилей CNOT, контролируемых кубитом a_m. Следующие $m-2$ слоёв состоят из последовательно применённых вентилей CCNOT, направленных к кубитам $y_2,\ldots,y_{m-1}$. Следующий слой состоит из вентилей CNOT, направленных к кубитам $y_1,\ldots,y_m$ и применённых одновременно. Последние $m-1$ слоёв состоят из последовательно применённых вентилей CNOT, направленных к кубитам $y_2,\ldots,y_{m-1}$. Следовательно, глубина такой схемы равна $3m-2$. Также данную схему можно распараллелить, тем самым уменьшая глубину схемы. Способ распараллеливания представлен в [26] и основан на копировании состояния кубита, от которого зависит оператор. Перед применением всех вентилей размножим состояние кубита a_m ещё на $\left\lceil\frac{m-1}{2}\right\rceil-1$ временных кубитов, после чего применением всех вентилей очистим временные кубиты. Тогда общее число вентилей будет равно $4m-5+2\left\lceil\frac{m-1}{2}\right\rceil$ (из них $3m-3+2\left\lceil\frac{m-1}{2}\right\rceil$ вентилей CNOT и $m-2$ вентилей CCNOT). Глубина такой схемы равна $2(\lceil\log_2(m-1)\rceil+1)+m$.

3.3. Сумма двух целых чисел. Пусть надо сложить два целых числа a и b, которые представлены в двоичном дополнительном коде длины $m\geqslant 3$. Тогда для записи ответа потребуется $m+1$ битов, иначе может произойти переполнение битовой строки. Так как бит a_m отвечает за знак, можно представлять чило $a=(a_1a_2\ldots a_{m-1}a_m)$ с помощью $m+1$ битов следующим образом: $a=(a_1a_2\ldots a_{m-1}a_ma_m)$. Пусть z — это вектор, в который записывается результат. Тогда

$$
\begin{gathered}
z_1=a_1\oplus b_1,\quad k_1=a_1b_1,\quad z_2=a_2\oplus b_2\oplus k_1,\\
k_2=a_2b_2\oplus k_1(a_2\oplus b_2),\quad z_3=a_3\oplus b_3\oplus k_2,\\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
k_{m-1}=a_{m-1}b_{m-1}\oplus k_{m-2}(a_{m-1}\oplus b_{m-1}),\quad z_m=a_m\oplus b_m\oplus k_{m-1},\\
k_m=a_mb_m\oplus k_{m-1}(a_m\oplus b_m),\quad z_{m+1}=a_m\oplus b_m\oplus k_m.
\end{gathered}
$$

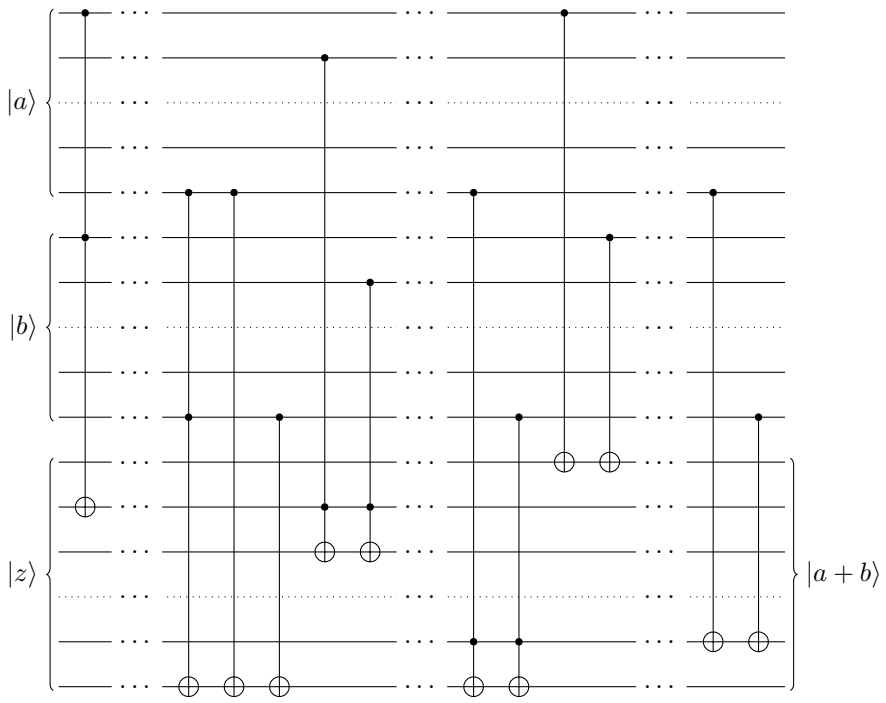


Рис. 11. Сложение двух целых чисел

Квантовая схема представлена на рис. 11. Сначала в место, отведённое под z_{i+1}, записываются значения k_i, $i = 1, \ldots, m$. После чего к каждому месту для z_i добавляются a_i и b_i, $i = 1, \ldots, m+1$. Обозначим место, отведённое под z_i, через r_i. Первые m вентилей CCNOT используются параллельно для вычисления слагаемого $a_i b_i$ в r_{i+1}, $i = 1, \ldots, m$. Затем последовательно в каждом r_{i+1} вычислим значение k_i, добавляя с помощью вентилей CCNOT $a_{i+1} r_{i+1}$ и $b_{i+1} r_{i+1}$ в r_{i+2}, $i = 1, \ldots, m-1$. После чего параллельно добавим a_i и b_i в r_i, $i = 1, \ldots, m+1$, где $a_{m+1} = a_m$, $b_{m+1} = b_m$, и получим $r_i = z_i$. Число кубитов, необходимое для реализации этой схемы, равно $m+1$, при этом число вентилей равно $5m$ (из них $2m+2$ вентилей CNOT и $3m-2$ вентилей CCNOT). Глубина данной схемы равна $2m+1$, так как предлагается проводить прибавление a_m и b_m к r_{m+1} параллельно с прибавлением $a_2 r_2$ и $b_2 r_2$ к r_3.

3.4. Возведение в квадрат целого числа. Рассмотрим операцию возведения в квадрат целого числа. Пусть есть целое число $a = a_m \ldots a_1$, $m \geqslant 3$ (если $m \leqslant 2$, то это целое число из множества $\{-1, 0, 1\}$, а для них операция возведения в степень очевидна), которое представлено в прямом двоичном коде длины m, и его нужно возвести в квадрат. Можно

не обращать внимание на кубит a_m, отвечающий за знак числа, так как после возведения в квадрат число станет положительным. Тогда можем считать, что работаем с положительным числом длины $n = m - 1$.

Первым шагом выделим для выхода операции $2n$ кубитов и обозначим их через z. Сразу же присвоим z_1 значение a_1 (1 вентиль CNOT). Далее выделим место под числа $(a_n \dots a_2)a_1$ и $(a_n \dots a_1)a_2$, которые будут складываться первыми, и скопируем их в выделенные места (1 вентиль CNOT и $2n - 2$ вентилей CCNOT). Наименьший значимый кубит результата этого сложения можно сразу записать в z_2, а для остальной части результата сложения выделим n кубитов. Так как после каждого сложения наименьший кубит уже равен соответствующему выходному, то будут складываться постоянно числа разряда n. Далее очистим место, которое отводилось для $(a_n \dots a_1)a_2$ и запишем туда $(a_n \dots a_1)a_3$ (2 вентиля CNOT и $2n - 2$ вентиля CCNOT). Далее подобно первому сложению просуммируем выход первого сложения и $(a_n \dots a_1)a_3$. Таким образом, проведём $n-1$ сложений, но результат последнего сложения можно сразу записывать в старшие $n + 1$ кубитов, предназначенные для выхода. В конце очистим все дополнительные кубиты.

В итоге, было выделено $2n$ кубитов для выхода операции, $2n - 1$ кубитов под числа $(a_n \dots a_2)a_1$ и $(a_n \dots a_1)a_2$, $n(n - 2)$ кубитов под хранение результатов промежуточных сумм. Тогда число постоянных кубитов равно $2n$, а число временных кубитов равно $n^2 - 1$. При этом общее число вентилей равно $14n^2 - 21n + 3$ (из них $4n^2 - 7$ вентилей CNOT и $10n^2 - 21n + 10$ вентилей CCNOT). Глубина схемы равна сумме глубин, используемых операций, а именно:

- 1 — глубина присвоения $z_1 = a_1$;
- $(2n + 1)(n - 1)$ — глубина $n - 1$ сложений двух чисел длины n;
- $n - 1$ — глубина копирования числа $(a_n \dots a_2)a_1$;
- $2n(n - 1)$ — глубина копирования и очистки чисел $(a_n \dots a_1)a_i$ для $i = 2, \dots, n$;
- $(2n+1)(n-2)$ — глубина очистки промежуточных результатов $n-2$ сложений двух чисел длины n;
- $2n(n - 2)$ — глубина копирования и очистки чисел $(a_n \dots a_1)a_i$ для $i = 2, \dots, n - 1$, необходимых для выполнения предыдущего пункта;
- $n - 1$ — глубина очистки числа $(a_n \dots a_2)a_1$.

В итоге глубина предложенной схемы возведения в квадрат целого числа равна $8n^2 - 8n - 4$.

3.5. Сумма нескольких целых положительных чисел.

Утверждение 1. *Пусть есть d целых положительных чисел, длина двоичного кода каждого из которых равна m. Тогда число кубитов, достаточное для их сложения, равно*

$$\sum_{i\in A}\left[\sum_{j=1}^{i-1}2^{i-j-1}(m+j)+(m+i)\right]-(m+l),$$

где $l=\min A$, $A=\left\{i\in\{1,\ldots,\lceil\log_2(d+1)\rceil\}\mid\left\lfloor\frac{d}{2^{i-1}}\right\rfloor\not\equiv 0\pmod 2\right\}$, *при этом глубина схемы равна* $2\lceil\log_2 d\rceil\left(m+\frac{\lceil\log_2 d\rceil+1}{2}\right)$.

Доказательство. Для того чтобы просуммировать все заданные числа, разобьём их на минимальное число подмножеств, мощности которых равны степеням двойки. Тогда у нас получится $wt(d)$ подмножеств. Просуммируем все элементы в каждом из них следующим образом. Разобьём элементы подмножества на пары и найдём сумму каждой пары. Далее полученные результаты разобьём на пары и найдём сумму каждой новой пары. Повторяем разбиение на пары и суммирование, пока не получим сумму всех элементов подмножества. Так как для суммы двух m-значных чисел достаточно $m+1$ кубитов, при каждом суммировании пар размер результата будет увеличиваться на 1. Тогда для нахождения суммы чисел каждого подмножества достаточно

$$\sum_{i\in A}\sum_{j=1}^{i-1}2^{i-j-1}(m+j)$$

кубитов. При этом согласно рассуждениям выше и формулам общего числа вентилей для суммы двух целых чисел находим, что общее число вентилей для суммирования чисел каждого подмножества равно

$$\sum_{i\in A}\sum_{j=1}^{i-1}2^{i-j-1}5(m+j-1).$$

Из них

$$\sum_{i\in A}\sum_{j=1}^{i-1}2^{i-j-1}(2(m+j-1)+2)$$

вентилей CNOT и

$$\sum_{i\in A}\sum_{j=1}^{i-1}2^{i-j-1}(3(m+j-1)-2)$$

вентилей CCNOT. Далее необходимо сложить суммы подмножеств. Будем складывать их, начиная с двух наименьших. Для этого достаточно

$$\sum_{i\in A}(m+i)-(m+l)$$

кубитов, при этом общее число вентилей будет равно

$$\sum_{i\in A} 5(m+i-1) - 5(m+l-1).$$

Из них

$$\sum_{i\in A} [2(m+i-1)+2] - 2(m+l-1) - 2$$

вентилей CNOT и

$$\sum_{i\in A} [3(m+i-1)-2] - 3(m+l-1) + 2$$

вентилей CCNOT. Глубина такой схемы равна $2\lceil \log_2 d \rceil \big(m + \frac{\lceil \log_2 d \rceil + 1}{2}\big)$. Утверждение 1 доказано.

3.6. Результаты раздела. В табл. 1 представлены выражения для числа кубитов и глубины схемы, достаточных для реализации на квантовом компьютере операций, рассмотренных в этом разделе и используемых для модели квантового оракула в разд. 4.

Таблица 1

Число кубитов и глубина схемы, достаточные для реализации некоторых операций на квантовом компьютере

Операции над целыми m-битными числами	Кубиты		Глубина
	постоянные	временные	
Сложение в дополнительном коде	$m+1$	—	$2m+1$
Возведение в квадрат в прямом коде	$2m-2$	m^2-2m	$8m^2-24m+12$
Отрицание в прямом коде	m	—	$m+2$
Перевод из дополнительного кода в прямой	m	$\lceil \frac{m-1}{2} \rceil - 1$	$2\lceil \log_2(m-1)+1 \rceil + m$
Сравнение положительных чисел	1	$m+1$	$7m$

Также были получены выражения для числа кубитов и глубины схемы, достаточных для реализации суммы нескольких целых положительных чисел (утверждение 1).

4. Модель квантового оракула

В данном разделе представлена модель квантового оракула, реализующая итерацию алгоритма Гровера для функции «поиск» из алгоритма 1, которая выполняет поиск элемента w в неупорядоченном списке L,

удовлетворяющего условию $\|v \pm w\| \leqslant \|v\|$ для заданного вектора v. Для условия $\|w \pm v\| \leqslant \|w\|$ модель квантового оракула и полученные оценки аналогичны. Обозначим через K длину списка L, содержащего векторы из пространства $\mathbb{Z}^d$, каждая координата которых кодируется строкой длины m.

4.1. Описание модели. Рассмотрим предлагаемую модель квантового оракула, у которой неупорядоченный список хранится в квантовой памяти.

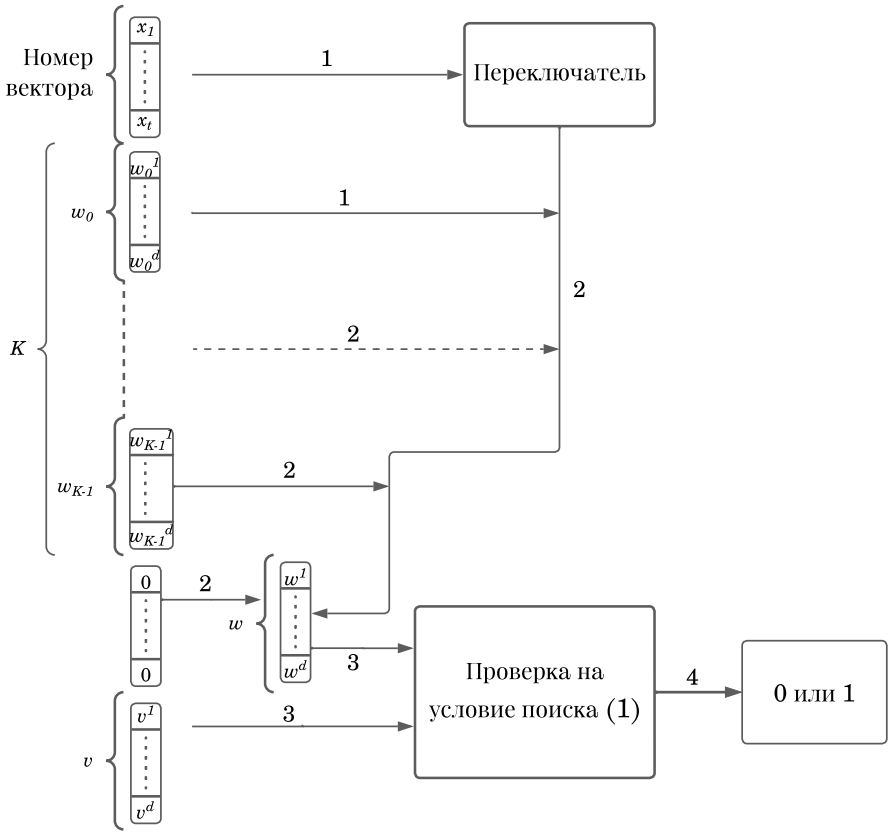


Рис. 12. Предлагаемая модель оракула с квантовым списком

Модель оракула, представленная на рис. 12, состоит из двоичного представления номера вектора в списке, K векторов из пространства $\mathbb{Z}^d$, каждая координата которых кодируется строкой длины m, переключателя, проверки на условие поиска и ответа. Работа модели состоит из четырёх этапов:

1) получение номера вектора на вход и передача его в переключатель;
2) выбор и копирование вектора из списка по выходу переключателя;
3) проверка скопированного вектора на условие поиска (1);
4) вывод ответа: 1 — если вектор удовлетворяет условию, 0 — если нет.

Проверка на условие поиска (1) содержит следующие операции: сумма целых чисел, возведение в квадрат целого числа, сравнение целых положительных чисел, получение отрицания и перевод в прямой код целого числа.

Переключатель. Переключатель — это векторная булева функция $F\colon \mathbb{F}_2^{\lceil \log_2 K\rceil} \to \mathbb{F}_2^{2^{\lceil \log_2 K\rceil}}$, которая номеру вектора i сопоставляет строку $(0,\dots,0,1,0,\dots,0)$, где 1 стоит на i-м месте. Рассмотрим две реализации переключателя.

Первая реализация не использует дополнительных кубитов. Распишем F как $(f_0, f_1, \dots, f_{2^{\lceil \log_2 K\rceil}-1})$, где f_k — булева функция из $\mathbb{F}_2^{\lceil \log_2 K\rceil}$ в $\mathbb{F}_2$, и рассмотрим каждую f_k отдельно. Представим номер функции f_k в двоичной системе счисления: $k = k_1 k_2 \dots k_{\lceil \log_2 K\rceil}$. Тогда

$$f_k = i_1^{k_1} i_2^{k_2} \dots i_{\lceil \log_2 K\rceil}^{k_{\lceil \log_2 K\rceil}} \times \bigoplus_{x \preceq \overline{k}} i_1^{x_1} i_2^{x_2} \dots i_{\lceil \log_2 K\rceil}^{x_{\lceil \log_2 K\rceil}}, \tag{4}$$

где $\overline{k} = (k_1, k_2, \dots, k_{\lceil \log_2 K\rceil})$ и $i_j^{x_j}$ — обычное возведение в степень.

Утверждение 2. *Число постоянных и временных кубитов, достаточное для реализации переключателя при входе длины m, равно 2^m и 0 соответственно, при этом глубина схемы равна $3^m - 1$.*

Доказательство. Число постоянных кубитов равно 2^m, что является длиной выхода. Ниже показано, что временные кубиты не потребуются.

Рассмотрим глубину схемы. Обозначим через $|f_k\rangle$ кубит, предназначенный для записи значения функции f_k. Начальная инициализация $|f_0\rangle$ равна $|1\rangle$, а начальная инициализация $|f_k\rangle$ при $k \neq 0$ равна $|0\rangle$. Сначала присвоим каждому $|f_k\rangle$ значение монома наименьшей степени функции f_k. Согласно (4) моном наименьшей степени в f_k равен $i_1^{k_1} i_2^{k_2} \dots i_{i_m}^{k_m}$. Так как начальная инициализация $|f_0\rangle$ равна $|1\rangle$, $|f_0\rangle$ уже равен значению наименьшего монома функции f_0. Мономы, у которых степень равна 1, можно получить из входа i вентилями CNOT. Таких мономов всего m штук, следовательно, на этом этапе задействовано m вентилей CNOT.

Опишем, как с помощью $2^m - m - 1$ вентилей CCNOT строятся все возможные мономы. Так как уже существуют все мономы степени 1, мономы степени 2 получаются с помощью применения вентиля CCNOT к соответствующему моному степени 1 и недостающей переменной из i. Таким образом, можно построить любой моном степени $t \in \{2,\dots,m\}$,

применяя вентиль CCNOT, если имеются все мономы степени 1 (вход переключателя) и все мономы степени $d-1$.

Теперь у нас есть все возможные мономы от переменных $i_1, i_2, \ldots, i_m$. Применяя вентиль CNOT, достроим функции $f_0, f_1, \ldots, f_{2^m-1}$. Будем достраивать функции в порядке уменьшения числа мономов в них, так как функции с меньшим числом слагаемых не содержат минимальных мономов функций с большим числом слагаемых. По построению функции f_k число слагаемых в ней равно двум в степени числа нулей в двоичном представлении номера функции. Тогда число вентилей, которые приходят в кубит $|f_k\rangle$, $k = 1, \ldots, 2^m - 1$, равно два в степени число нулей в двоичном представлении числа k. Так как функция f_0 содержит моном 1, а он не требует приходящих вентилей, число приходящих вентилей в кубит $|f_0\rangle$ равно $2^m - 1$. Число функций, в двоичной записи номера которых содержится t нулей, равно C_m^t. Тем самым общее число вентилей в переключателе равно

$$\sum_{t=0}^{m} C_m^t 2^t - 1 = (1+2)^m - 1 = 3^m - 1,$$

из них $2^m - m - 1$ вентилей CCNOT и $3^m - 2^m + m$ вентилей CNOT. Глубину схемы считаем равной общему числу вентилей. Утверждение 2 доказано.

Вторая реализация имеет меньшую глубину схемы, но использует временные кубиты. Сначала размножим входное состояние кубитов ещё на $(2^{\lceil \log_2 K \rceil} - 1)\lceil \log_2 K \rceil$ временных кубитов, получив $2^{\lceil \log_2 K \rceil}$ одинаковых состояний $|s_0\rangle$, $|s_1\rangle$, $\ldots$, $|s_{2^{\lceil \log_2 K \rceil}-1}\rangle$. Глубина размножения равна $\lceil \log_2 K \rceil$. Для каждого состояния применяем соответствующую функцию

$$N_i(s_i) = \begin{cases} 1, & \text{если } i = s_i, \\ 0 & \text{иначе,} \end{cases}$$

которая строится следующим образом. Пусть $i = i_1 i_2 \ldots i_{\lceil \log_2 K \rceil}$ — двоичное представление числа i. Тогда функция N_i сначала применяет вентиль NOT к кубиту с номером t состояния $|s_i\rangle$, если $i_t = 0$. Далее, аналогично реализации операции суммы нескольких целых положительных чисел разобьём состояние, полученное из $|s_i\rangle$ с помощью вентилей NOT, на минимальное число подмножеств, мощности которых равны степеням двойки, только вместо операции сложения используем вентиль CCNOT. Для одного выхода функции N_i понадобится $\lceil \log_2 K \rceil - 2$ временных кубитов и $\lceil \log_2 K \rceil - 1$ вентилей CCNOT. Глубина схемы, реализующей все применения функций N_i, $i = 0, 1, \ldots, 2^{\lceil \log_2 K \rceil} - 1$, равна $\lceil \log_2(\lceil \log_2 K \rceil) \rceil + 1$, так как функции N_i применяются одновременно. В конце очищаем временные кубиты. Глубина очистки равна $\lceil \log_2 K \rceil + \lceil \log_2(\lceil \log_2 K \rceil) \rceil$. Тогда для

данной реализации переключателя число постоянных кубитов совпадает с первой реализацией, число временных кубитов равно $(2^{\lceil \log_2 K \rceil} - 1) \times (2\lceil \log_2 K \rceil - 2)$, а глубина схемы равна $2\lceil \log_2 K \rceil + 2\lceil \log_2(\lceil \log_2 K \rceil) \rceil + 1$.

Копирование векторов из списка. Копирование вектора из списка использует выход переключателя длины $2^{\lceil \log_2 K \rceil}$. Рассмотрим две реализации данной операции.

Применяя вентиль CCNOT, копируем нужный вектор с номером i из списка. Для номеров, не имеющих соответствующих векторов, нужно применить вентили CNOT, направленные на выход оракула, — такие номера будут неподходящими. Тем самым в схеме применяется Kdm вентилей CCNOT и $2^{\lceil \log_2 K \rceil} - K$ вентилей CNOT для номеров, у которых нет соответствующего вектора. В этом случае глубина схемы составит Kdm, так как глубина вентилей, направленных на выход оракула, равна $2^{\lceil \log_2 K \rceil} - K < K$, и они выполняются параллельно копированию векторов из списка. Предлагается распараллелить эту операцию, используя $dm - 1$ временных кубитов. Тогда каждый кубит из выхода переключателя поочерёдно размножается до dm кубитов, используя $dm - 1$ вентилей CNOT, занимающих $\lceil \log_2 dm \rceil$ слоёв схемы. Затем выполняется параллельное копирование соответствующего вектора с использованием dm вентилей CCNOT одновременно. Далее размноженные кубиты чистятся для их применения следующим кубитом выхода переключателя. Чистка задействует $dm - 1$ вентилей CNOT, занимающих $\lceil \log_2 dm \rceil$ слоёв схемы. Глубина одного копирования равна $2\lceil \log_2(dm) \rceil + 1$, а число вентилей равно $3dm - 2$. Тогда глубина всей схемы копирования векторов составит $K(2\lceil \log_2(dm) \rceil + 1)$.

Вторая реализация имеет меньшую глубину схемы, но использует большее число временных кубитов, при этом длина списка $K \geqslant 2$. Подобно реализации операции сложения нескольких положительных чисел, разобьём список векторов на минимальное число подмножеств, мощности которых равны степеням двойки. Далее разделим элементы каждого множества на пары и скопируем их с помощью размножения выходного вектора переключателя и вентилей CCNOT в специально выделенное место для этой пары; если элемент в множестве один, то он остаётся на месте. Так как копирования разных пар происходят одновременно, глубина копирования всех пар совпадает с глубиной копирования одной пары и равна 2. Для этого необходимо размножить выходной вектор переключателя до $\lfloor \frac{K}{2} \rfloor$ состояний и выделить $\lfloor \frac{K}{2} \rfloor dm$ временных кубитов для записи результата копирования. Глубина размножения выхода переключателя до $\lfloor \frac{K}{2} \rfloor$ состояний равна $\lceil \log_2 \lfloor \frac{K}{2} \rfloor \rceil$. Тогда на этом этапе понадобится $\lfloor \frac{K}{2} \rfloor 2^{\lceil \log_2 K \rceil} + \lfloor \frac{K}{2} \rfloor dm$ временных кубитов и глубина схемы будет составлять $\lceil \log_2 \lfloor \frac{K}{2} \rfloor \rceil + 2$.

Таким образом, каждое из подмножеств, кроме подмножества из одного элемента, даёт новое подмножество, в котором вдвое меньше элементов. Далее снова разобьём элементы в каждом подмножестве на пары и с помощью вентилей CNOT скопируем их в специально отведённые места. Повторяем данное действие до тех пор, пока не получим два одноэлементных подмножества. Как только они получены, объединяем их в одно подмножество и продолжаем операцию копирования. По завершении останется одно подмножество, состоящее из одного элемента. Это и есть выход операции копирования векторов из списка. После получения выхода операции происходит очистка кубитов. Тогда число постоянных кубитов совпадает с предыдущей реализацией и равно выходу операции, число временных кубитов равно $\lfloor \frac{K}{2} \rfloor 2^{\lceil \log_2 K \rceil} + (K-2)dm$, а глубина схемы составляет $2\big(\lceil \log_2 \lfloor \frac{K}{2} \rfloor \rceil + 2\lceil \log_2 K \rceil - 1\big)$.

4.2. Оценки сложности реализации, минимизирующей число кубитов квантовой схемы. В данном пункте рассматривается реализация предложенной модели квантового оракула, использующая реализации переключателя и операции копирования вектора с меньшим числом кубитов.

Теорема 1. *Пусть имеется список длины K, содержащий векторы из пространства $\mathbb{Z}^d$, $d \geqslant 2$, каждая координата которых кодируется битовой строкой длины $m \geqslant 3$. Тогда для предложенной реализации модели квантового оракула, представленного на рис.* 12, *потребуется*

$$\lceil \log_2 K \rceil + 2^{\lceil \log_2 K \rceil} + Kdm + 3dm^2 + 14dm + 2d + 6m + 3\lceil \log_2 d \rceil + 3$$

кубитов, при этом глубина схемы равна

$$2 \cdot 3^{\lceil \log_2 K \rceil} + 2K(2\lceil \log_2 dm \rceil + 1) + 16m^2 + 8m + 4\lceil \log_2 m \rceil + \lceil \log_2 d \rceil (2\lceil \log_2 d \rceil + 8m + 10) + 17.$$

Доказательство. Разделим доказательство на две части: оценка числа кубитов и оценка глубины схемы.

Оценка числа кубитов. Сначала оценим число постоянных кубитов. Так как список имеет длину K, число кубитов, необходимых для написания номера вектора в двоичной системе, равно $\lceil \log_2 K \rceil$. Далее выделим $2^{\lceil \log_2 K \rceil}$ кубитов под выход переключателя. Также оракул содержит K векторов из пространства $\mathbb{Z}^d$, каждая координата которых кодируется битовой строкой длины m. Такой список будет занимать Kdm кубитов. Ещё необходимо $2dm$ кубитов под вектор v и копию вектора w из списка. Дальше отведём dm кубитов для получения и хранения вектора $-w$, а также отведём место для клонирования вектора v, получения

векторов $v+w$ и $v-w$ и их хранения. Для получения одного такого вектора нам понадобиться провести d операций сложения (так как операции сложения и вычитания в дополнительном коде одинаковы). Тогда число кубитов, отведённое для получения и хранения векторов $v+w$ и $v-w$, равно $2d(m+1)$. Теперь посчитаем число кубитов, необходимых для получения и хранения векторов $v^2, (v+w)^2$ и $(v-w)^2$ (это векторы $v, v+w$ и $v-w$, у которых координаты возведены в квадрат). Для этого нам понадобится $3d(m+1)$ кубитов для перевода координат в прямой код и $3d(2m)$ кубитов для получения квадрата координат. Далее суммируем координаты векторов $v^2, (v+w)^2$ и $(v-w)^2$. Это займёт $3(2m+\lceil\log_2 d\rceil)$ кубитов. Осталось добавить число кубитов, достаточных для реализации двух сравнений и выхода оракула. Сравниваемые числа $(2m+\lceil\log_2 d\rceil)$-значные. Тогда для реализации двух сравнений достаточно 2 кубита и ещё один кубит — для выхода оракула. Итоговое число постоянных кубитов равно

$$\lceil\log_2 K\rceil + 2^{\lceil\log_2 K\rceil} + Kdm + 14dm + 5d + 6m + 3\lceil\log_2 d\rceil + 3.$$

Оценим число временных кубитов. Для этого достаточно найти максимум всех временных кубитов, используемых в операциях:

- переключатель использует 0 кубитов;
- копирование использует $dm-1$ кубитов;
- получение $-w$ использует 0 кубитов;
- копирование v (для параллелизма) и получение $v\pm w$ использует dm кубитов;
- перевод в прямой код использует $3d\big(\lceil\frac{m}{2}\rceil - 1\big)$ кубитов;
- получение векторов длины d с квадратами координат использует $3d(m^2-1)$ кубитов;
- подсчёт квадрата норм векторов использует

$$3\left(\sum_{d_i\neq 0}\left[\sum_{j=1}^{i-1} 2^{i-j-1}(2m+j) + (2m+i)\right] - (2m+l) - (2m+\lceil\log_2 d\rceil)\right)$$

кубитов, где $l = \min\{k \mid d_k \neq 0\}$;

- копирование $\|v\|^2$ и сравнение использует $3(2m+\lceil\log_2 d\rceil)+2$ кубитов.

Видно, что максимум временных кубитов равен получению векторов с квадратами координат или подсчёту квадрата норм векторов. Оценим сверху число кубитов, используемых в операции подсчёта квадратов норм векторов. Будем называть слоем i все сложения, длина результата которых равна $2m+i$. Тогда на каждом i используется не более чем $\lfloor\frac{d}{2^i}\rfloor(2m+i)$ временных кубитов. Следовательно, общее число временных кубитов, используемых операцией подсчёта квадрата норм векторов,

не превосходит $2md + d\sum\limits_{i=1}^{\lceil \log_2 d\rceil} \frac{i}{2^i}$. Заметим, что $\sum\limits_{i=1}^{\lceil \log_2 d\rceil} \frac{i}{2^i} \leqslant \sum\limits_{i=1}^{\infty} \frac{i}{2^i}$. Чтобы найти сумму ряда в правой части, рассмотрим степенной ряд

$$\sum_{i=0}^{\infty} x^i = \frac{1}{1-x}, \quad |x| < 1. \tag{5}$$

Степенные ряды сходятся равномерно внутри своей области сходимости, значит, их можно почленно дифференцировать. Продифференцируем обе части равенства (5) и домножим на x:

$$x\sum_{i=1}^{\infty} ix^{i-1} = \frac{x}{(1-x)^2}.$$

Подставив $x = \frac{1}{2}$, получаем

$$\sum_{i=1}^{\infty} \frac{i}{2^i} = \frac{\frac{1}{2}}{\left(1-\frac{1}{2}\right)^2} = 2.$$

Тогда общее число временных кубитов для трёх операций подсчёта квадрата норм векторов не превосходит $3d(2m+2)$. Для $m > 2$ и $d \geqslant 2$ выполнено $6dm + 6d \leqslant 3dm^2 - 3d$. Следовательно, $3d(m^2-1)$ является максимум среди временных кубитов, используемых всеми операциями, и в сравнениях можно не очищать временные кубиты (это уменьшит общую глубину схемы). Общее число временных кубитов равно $3d(m^2-1)$. Значит, общее число используемых оракулом кубитов равно

$$\lceil \log_2 K\rceil + 2^{\lceil \log_2 K\rceil} + Kdm + 3dm^2 + 14dm + 2d + 6m + 3\lceil \log_2 d\rceil + 3.$$

Оценка глубины схемы. Сначала используется переключатель — его глубина составляет $3^{\lceil \log_2 K\rceil} - 1$. Далее происходит копирование векторов из списка при помощи выхода переключателя. Глубина данной операции равна $K(2\lceil \log_2(dm)\rceil + 1)$. Теперь вычислим вектор $-w$. Это займёт $m+2$ слоёв схемы. Далее — операции копирования вектора v, вычисления $v+w$ и $v-w$, очистка скопированного вектора v, глубина которых составляет $2m+3$. Вычислим глубину схемы, необходимую для получения и хранения векторов $v^2, (v+w)^2$ и $(v-w)^2$. Для этого каждую координату переведём в прямой код и возведём в квадрат. Глубина этих операций равна $2\lceil \log_2(m)\rceil + m + 3$ и $8m^2 - 8m - 4$ соответственно. Далее суммируем координаты векторов $v^2, (v+w)^2$ и $(v-w)^2$. Глубина этого вычисления равна $2\lceil \log_2 d\rceil\left(2m + \frac{\lceil \log_2 d\rceil + 1}{2}\right)$. Дальше происходят копирование вектора $\|v\|^2$ и два сравнения $\|v+w\|^2 \leqslant \|v\|^2$ и $\|v-w\|^2 \leqslant \|v\|^2$. Глубина этих операций составляет $4(2m + \lceil \log_2 d\rceil) + 1$. В завершение добавим три вентиля, реализующие операцию «или» для выходов двух операций сравнения. Они применяются последовательно, и их глубина

равна 3. Для очистки постоянных кубитов, как и для временных кубитов, нужно применить в обратном порядке все используемые вентили, которые не участвуют в изменении выходных кубитов оракула. Следовательно, при подсчёте значения глубины схемы нужно удвоить глубину каждой операции, кроме последней операции «или». Тогда итоговая глубина схемы равна

$$2 \cdot 3^{\lceil \log_2 K \rceil} + 2K(2\lceil \log_2 dm \rceil + 1) + 16m^2 + 8m + 4\lceil \log_2 m \rceil + \lceil \log_2 d \rceil (2\lceil \log_2 d \rceil + 8m + 10) + 17.$$

Теорема 1 доказана.

Обозначим через $\mathcal{Q}$ схему, реализующую модель квантового оракула, представленного на рис. 12, и имеющую минимальное число кубитов. Тогда из теоремы 1 получим

Следствие 1. *В условиях теоремы* 1 *число кубитов, используемых в схеме* $\mathcal{Q}$*, не превосходит*

$$\lceil \log_2 K \rceil + 2^{\lceil \log_2 K \rceil} + Kdm + 3dm^2 + 14dm + 2d + 6m + 3\lceil \log_2 d \rceil + 3.$$

4.3. Оценки сложности реализации, минимизирующей глубину квантовой схемы. В этом пункте рассматривается реализация предложенной модели квантового оракула, использующая реализации переключателя и операции копирования вектора с меньшей глубиной схем.

Теорема 2. *Пусть имеется список длины* $K \geqslant 2$*, содержащий векторы из пространства* $\mathbb{Z}^d$*,* $d \geqslant 2$*, каждая координата которых кодируется битовой строкой длины* $m \geqslant 3$*. Тогда для предложенной реализации модели квантового оракула, представленного на рис.* 12*, потребуется квантовая схема глубины*

$$12\lceil \log_2 K \rceil + 4\lceil \log_2(\lceil \log_2 K \rceil) \rceil + 4\left\lceil \log_2 \left\lfloor \frac{K}{2} \right\rfloor \right\rceil + 16m^2 + 8m + 4\lceil \log_2 m \rceil + \lceil \log_2 d \rceil (2\lceil \log_2 d \rceil + 8m + 10) + 17,$$

при этом число кубитов данной схемы равно

$$\lceil \log_2 K \rceil + 2^{\lceil \log_2 K \rceil} + Kdm + 14dm + 5d + 6m + 3\lceil \log_2 d \rceil + 3 + \max\left\{ \left\lfloor \frac{K}{2} \right\rfloor 2^{\lceil \log_2 K \rceil} + (K-2)dm,\ 3d(m^2-1) \right\}.$$

Доказательство. Аналогично теореме 1 разделим доказательство на две части: оценка глубины схемы и оценка числа кубитов.

Оценка глубины схемы. Заменим реализации переключателя и копирования векторов на минимизирующие глубину. В схеме из теоремы 1 глубина переключателя и копирования векторов из списка равнялась $(3^{\lceil \log_2 K \rceil} - 1)$ и $K(2\lceil \log_2 dm \rceil + 1)$ соответственно. Тогда, заменив в

$$2 \cdot 3^{\lceil \log_2 K \rceil} + 2K(2\lceil \log_2 dm \rceil + 1) + 16m^2 + 8m \\ + 4\lceil \log_2 m \rceil + \lceil \log_2 d \rceil (2\lceil \log_2 d \rceil + 8m + 10) + 17$$

эти слагаемые на

$$2\lceil \log_2 K \rceil + 2\lceil \log_2 (\lceil \log_2 K \rceil) \rceil + 1 \quad \text{и} \quad 2\left(\left\lceil \log_2 \left\lfloor \frac{K}{2} \right\rfloor \right\rceil + 2\lceil \log_2 K \rceil - 1 \right)$$

(глубину реализации переключателя и копирования векторов, минимизирующей глубину), получим итоговую глубину схемы квантового оракула с учётом очистки кубитов:

$$12\lceil \log_2 K \rceil + 4\lceil \log_2 (\lceil \log_2 K \rceil) \rceil + 4\left\lceil \log_2 \left\lfloor \frac{K}{2} \right\rfloor \right\rceil + 16m^2 + 8m \\ + 4\lceil \log_2 m \rceil + \lceil \log_2 d \rceil (2\lceil \log_2 d \rceil + 8m + 10) + 17.$$

Оценка числа кубитов. Число постоянных кубитов будет совпадать с числом постоянных кубитов из теоремы 1 и будет равно

$$\lceil \log_2 K \rceil + 2^{\lceil \log_2 K \rceil} + Kdm + 14dm + 5d + 6m + 3\lceil \log_2 d \rceil + 3,$$

так как реализации, минимизирующие число кубитов, и реализации, минимизирующие глубину схемы, имеют одинаковое число постоянных кубитов. Однако число временных кубитов отличается от предложенной реализации в теореме 1. Теперь максимум среди всех временных кубитов нужно искать в следующем списке используемых операций:

- переключатель использует $(2^{\lceil \log_2 K \rceil} - 1)(2\lceil \log_2 K \rceil - 2)$ кубитов;
- копирование использует $\lfloor \frac{K}{2} \rfloor 2^{\lceil \log_2 K \rceil} + (K-2)dm$ кубитов;
- получение $-w$ использует 0 кубитов;
- клонирование v (для параллелизма) и получение $v \pm w$ использует dm кубитов;
- перевод в прямой код использует $3d\big(\lceil \frac{m}{2} \rceil - 1\big)$ кубитов;
- получение векторов длины d с квадратами координат использует $3d(m^2 - 1)$ кубитов;
- подсчёт квадрата норм векторов использует

$$3\left(\sum_{d_i \neq 0} \left[\sum_{j=1}^{i-1} 2^{i-j-1}(2m + j) + (2m + i) \right] - (2m + l) - (2m + \lceil \log_2 d \rceil) \right)$$

кубитов, где $l = \min\{k \mid d_k \neq 0\}$;

• копирование $\|v\|^2$ и сравнение использует $3(2m + \lceil \log_2 d \rceil) + 2$ кубитов.

Заметим, что при $K \geqslant 2$

$$(2^{\lceil \log_2 K \rceil} - 1)(2\lceil \log_2 K \rceil - 2) \leqslant \left\lfloor \frac{K}{2} \right\rfloor 2^{\lceil \log_2 K \rceil} + (K-2)dm.$$

Тогда общее число временных кубитов равно максимуму из временных кубитов переключателя и временных кубитов операции получения векторов с квадратами координат, т. е.

$$\max\left\{ \left\lfloor \frac{K}{2} \right\rfloor 2^{\lceil \log_2 K \rceil} + (K-2)dm,\, 3d(m^2-1) \right\}.$$

Следовательно, общее число кубитов, используемых оракулом, равно

$$\lceil \log_2 K \rceil + 2^{\lceil \log_2 K \rceil} + Kdm + 14dm + 5d + 6m + 3\lceil \log_2 d \rceil + 3 + \max\left(\left\lfloor \frac{K}{2} \right\rfloor 2^{\lceil \log_2 K \rceil} + (K-2)dm, 3d(m^2-1) \right).$$

Теорема 2 доказана.

Обозначим через $\mathcal{D}$ схему, реализующую модель квантового оракула, представленного на рис. 12, и имеющую минимальную глубину. Тогда из теоремы 2 получим

Следствие 2. *В условиях теоремы 2 глубина схемы $\mathcal{D}$ не превосходит*

$$12\lceil \log_2 K \rceil + 4\lceil \log_2(\lceil \log_2 K \rceil) \rceil + 4\left\lceil \log_2 \left\lfloor \frac{K}{2} \right\rfloor \right\rceil + 16m^2 + 8m + 4\lceil \log_2 m \rceil + \lceil \log_2 d \rceil (2\lceil \log_2 d \rceil + 8m + 10) + 17.$$

4.4. Асимптотики. Как видно из теоремы 1, верхняя асимптотическая оценка сложности реализации, минимизирующей число кубитов, представленной модели оракула равна $O(Kdm + dm^2)$ кубитов. Таким образом, хранение списка в квантовой памяти приводит к линейному росту используемого числа кубитов от длины K, которая может расти экспоненциально с увеличением размерности решётки ($K \sim 2^{0{,}21d}$). Аналогично из теоремы 2 следует, что верхняя асимптотическая оценка глубины реализации, минимизирующей глубину схемы, равна $O(\log_2 K + (m + \log_2 d)^2)$, но верхняя асимптотическая оценка числа кубитов остаётся полиномиально зависящей от длины списка K.

5. Связь числа кубитов и глубины схемы квантового оракула с параметрами постквантовых криптосистем

В данном разделе рассмотрим криптосистемы, основанные на решётках и являющиеся финалистами конкурса NIST, и получим верхние оценки сложности реализации модели квантового оракула для атаки на данные криптосистемы. Обозначим через R кольцо $\mathbb{Z}[x]/(x^n-1)$, а через R_q — кольцо $\mathbb{F}_q[x]/(x^n-1)$. Любой элемент

$$a = \sum_{i=0}^{n-1} a_i x^i \in R_q, \quad a_i \in \mathbb{F}_q,$$

можно представить как вектор $a = (a_0, a_1, \ldots, a_{n-1})$. Операция умножения в R определяется как результат циклической свёртки: для $f, g \in R$ полагаем $f * g = h \in R$, где

$$h_k = \sum_{i=0}^{k} f_i g_{k-i} + \sum_{i=k+1}^{n-1} f_i g_{n+k-i} = \sum_{(i+j) \bmod n = k} f_i g_j, \quad k \in \overline{0, n-1}.$$

Если выполняется умножение полиномов по модулю числа, то коэффициенты приводятся по этому модулю.

Через R_q^i и $R_q^{i \times j}$ обозначим векторное пространство размерности i и множество матриц размеров $i \times j$ с элементами из кольца R_q.

5.1. Асимметричная криптография. В [27] впервые были предложены принципы работы асимметричной криптографии. Опишем принцип работы асимметричной криптосистемы. Первый абонент (А) формирует свой открытый ключ K_{pub} и секретный ключ K_{priv}. Основным требованием к секретному ключу является «трудность» вычисления по открытому ключу. Открытый ключ (А) сообщается всем абонентам. Любой другой абонент (Б) может зашифровать секретное сообщение m для (А), воспользовавшись его открытым ключом K_{pub} и функцией зашифрования $E(m, K_{\text{pub}})$ и вычислив шифртекст $c = E(m, K_{\text{pub}})$. После чего (Б) отправляет (А) шифртекст c по открытому каналу связи. Получив шифртекст c, (А) расшифрует его с помощью своего секретного ключа K_{priv} и функции расшифрования $D(c, K_{\text{priv}})$, восстановив секретное сообщение $m = D(c, K_{\text{priv}})$.

Таблица 2

Параметры и принцип работы RSA

Секретный ключ: p, q простые, $d \in \mathbb{N}$	Открытый ключ: $n = pq, e \in \mathbb{N}$
Зашифрование: $c = m^e \bmod n$	Расшифрование: $m = c^d \bmod n$

В качестве примера криптосистемы с открытым ключом рассмотрим RSA [28], параметры и принцип работы которой представлены в табл. 2. Данная криптосистема разработана в 1978 г. и широко используется даже сейчас. Приведём её краткое описание:

- p, q — большие простые числа, $n = pq$;
- $\varphi(n) = (p-1)(q-1)$ — функция Эйлера числа n;
- e, d — натуральные, взаимно простые с $\varphi(n)$, $ed \equiv 1 \pmod{\varphi(n)}$.

Однако в 1994 г. в [29] был представлен квантовый алгоритм, который за полиномиальное время решает задачи дискретного логарифмирования и факторизации, что переводит RSA в класс незащищённых криптосистем. Оценки сложности реализации квантового криптоанализа RSA приведены в [30].

5.2. NTRU. Криптосистема NTRU [31] представлена в 1996 г. в качестве альтернативы RSA [28] — существующему стандарту асимметричного шифрования. В табл. 3 представлен принцип её работы.

Таблица 3

Параметры и принцип работы NTRU

Секретный ключ: $f, g \in R$ с коэффициентами из $\{-1, 0, 1\}$	Открытый ключ: $N, p, q \in \mathbb{Z}$, НОД$(p, q) = 1$, $h = f_q * g \bmod q$, где $f_q * f = 1 \bmod q$
Зашифрование: $c = (p\varphi * h + m) \bmod q$, где $\varphi, m \in R$ с коэффициентами из $\{-1, 0, 1\}$ при некоторых ограничениях на них	Расшифрование: $a = f * c \bmod q$, $m' = f_q * a \bmod p$

Одна из самых эффективных атак на NTRU сводится к решению задачи SVP в решётке, базис которой образован строками матрицы M, построенной на основе открытого ключа:

$$M = \begin{pmatrix} 1 & 0 & \dots & 0 & h_0 & h_1 & \dots & h_{N-1} \\ 0 & 1 & \dots & 0 & h_{N-1} & h_0 & \dots & h_{N-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & h_1 & h_2 & \dots & h_0 \\ 0 & 0 & \dots & 0 & q & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & q \end{pmatrix}.$$

Кратчайший вектор решётки, порождённой этим базисом, с большой вероятностью имеет вид $r = (f, g)$.

5.3. SABER. Рассмотрим криптосистему SABER [32]. Пусть $\mathrm{bits}(x, i, j)$ — некоторая специально определённая функция, которая для $x \in \mathbb{Z}_2^k$ и $i \geqslant j \geqslant k$ возвращает целое число из $\mathbb{Z}_2^j$. Применение операции bits к вектору из R_q^l означает применение операции bits к коэффициентам многочлена в каждой координате вектора. Параметры и принцип работы криптосистемы Saber представлены в табл. 4.

Таблица 4

Параметры и принцип работы SABER

Секретный ключ:	Открытый ключ:
$s \in R_q^l$	N, l, $p = 2^{\varepsilon_p}$, $q = 2^{\varepsilon_q}$, $t = 2^{\varepsilon_t} \in \mathbb{N}$, $\varepsilon_q > \varepsilon_p > \varepsilon_t + 1$, $h \in R_q^l$, $A \in R_q^{l \times l}$, $b = \mathrm{bits}(As + h, \varepsilon_q, \varepsilon_p)$
Зашифрование:	Расшифрование:
$s' \in R_q^l$, $b' = \mathrm{bits}(A^\top s' + h, \varepsilon_q, \varepsilon_p)$, $v' = b^\top \mathrm{bits}(s', \varepsilon_p, \varepsilon_p)$, $m \in R_2$, $c_m = \mathrm{bits}(v' + 2^{\varepsilon_p - 1} m, \varepsilon_p, \varepsilon_t)$, $c = (c_m, b')$	$v = b'^\top \mathrm{bits}(s, \varepsilon_p, \varepsilon_p)$, $m' = \mathrm{bits}(v - 2^{\varepsilon_p - \varepsilon_t - 1} c_m + h, \varepsilon_p, 1)$

5.4. CRYSTALS-Kyber. Рассмотрим криптосистему CRYSTALS-Kyber [33]. Возьмём $x \in \mathbb{F}_q$ и натуральное число $d < \lceil \log_2 q \rceil$. Определим операции

$$\mathrm{Compress}_q(x, d) = \left\lceil \frac{2^d}{q} x \right\rfloor \bmod x^d,$$

$$\mathrm{Decompress}_q(x, d) = \left\lceil \frac{q}{2^d} x \right\rfloor,$$

которые используются в криптосистеме CRYSTALS-Kyber, представленной в табл. 5.

Таблица 5

Параметры и принцип работы CRYSTALS-Kyber

Секретный ключ:	Открытый ключ:
$s \in R_q^k$	$N, k, q, d_t, d_u, d_v \in \mathbb{N}$, $p \in R_2$, $e \in R_q^k$, $A \in R_q^{k \times k}$, $t = \mathrm{Compress}_q(As + e, d_t)$
Зашифрование:	Расшифрование:
$t = \mathrm{Decompress}_q(t, d_t)$, $r, e_1 \in R_q^k$, $e_2 \in R_q$, $m \in R_2$, $u = \mathrm{Compress}_q(A^\top r + e_1, d_u)$, $v = \mathrm{Compress}_q\big(t^\top r + e_2 + \lceil \frac{q}{2} \rfloor m, d_v\big)$, $c = (u, v)$	$u = \mathrm{Decompress}_q(u, d_u)$, $v = \mathrm{Decompress}_q(v, d_v)$, $m' = \mathrm{Compress}_q(v - s^\top u, 1)$

5.5. Верхние оценки реализации квантового оракула для атаки на постквантовые криптосистемы. В табл. 6 указана связь уровней защищённости криптосистем с числом кубитов и глубиной схемы предложенных реализаций представленной модели квантового оракула. Стоит отметить, что 5 июля 2022 г. закончился третий этап конкурса NIST [34], по итогом которого новым стандартом асимметричного шифрования была выбрана криптосистема CRYSTALS-Kyber.

Таблица 6

Число кубитов и глубина схемы, достаточные для предложенных реализаций квантового оракула

		Минимизация кубитов			Минимизация глубины		
Уровень защищённости		1	3	5	1	3	5
NTRU	кубиты	$2^{227,48}$	$2^{298,45}$	$2^{359,31}$	$2^{426,78}$	$2^{568,34}$	$2^{688,82}$
	глубина	$2^{340,19}$	$2^{452,72}$	$2^{547,82}$	$2^{12,9}$	$2^{13,13}$	$2^{13,35}$
SABER	кубиты	$2^{228,95}$	$2^{337,06}$	$2^{444,99}$	$2^{430,04}$	$2^{644,57}$	$2^{860,08}$
	глубина	$2^{343,36}$	$2^{512,95}$	$2^{684,12}$	$2^{13,1}$	$2^{13,38}$	$2^{13,6}$
CRYSTALS-Kyber	кубиты	$2^{228,85}$	$2^{336,96}$	$2^{444,89}$	$2^{430,04}$	$2^{644,57}$	$2^{860,08}$
	глубина	$2^{343,36}$	$2^{512,95}$	$2^{684,12}$	2^{13}	$2^{13,3}$	$2^{13,53}$

Как видно из табл. 6, экспоненциальная длина списка L накладывает ограничения на возможность реализации гибридных атак на полноразмерные постквантовые криптосистемы. Однако стоит заметить, что для атак на NTRU используются циклические решётки, исследования которых может помочь уменьшить длину списка L, что приведёт к меньшим верхним оценкам на сложность реализации квантового оракула.

Заключение

Разработана и описана модель квантового оракула, применимая в алгоритме Гровера для реализации гибридного квантово-классического алгоритма на основе GaussSieve. Такой алгоритм может быть использован для атак на криптосистемы, стойкость которых зависит от решения задачи SVP. Получены выражения для верхних оценок числа кубитов и глубины схемы двух предложенных реализаций модели квантового

оракула: минимизирующей число кубитов и минимизирующей глубину схемы. Проанализирована сложность реализации предложенных моделей квантового оракула для атаки на постквантовые криптосистемы, основанные на решётках и являющиеся финалистами третьего раунда конкурса постквантовой криптографии NIST.

ЛИТЕРАТУРА

1. ГОСТ Р 34.12—2015. Информационная технология. Криптографическая защита информации. Блочные шифры. Введ. 1.01.2016. М.: Стандартинформ, 2015. 25 с.
2. **Денисенко Д. В., Маршалко Г. Б., Никитенкова М. В., Рудской В. И., Шишкин В. А.** Оценка сложности реализации алгоритма Гровера для перебора ключей алгоритмов блочного шифрования ГОСТ Р 34.12—2015 // Журн. эксперим. и теор. физики. 2019. Т. 155, № 4. С. 645–653.
3. **Grassl M., Langenberg B., Roetteler M., Steinwandt R.** Applying Grover's algorithm to AES: Quantum resource estimates // Post-quantum cryptography. Proc. 7th Int. Workshop (Fukuoka, Japan, Feb. 24–26, 2016). Cham: Springer, 2016. P. 29–43. (Lect. Notes Comput. Sci.; V. 9606).
4. **Jaques S., Naehrig M., Roetteler M., Virdia F.** Implementing Grover oracles for quantum key search on AES and LowMC // Advances in cryptology — EUROCRYPT 2020. Proc. 39th Annu. Int. Conf. Theory Appl. Cryptogr. Techn. (Zagreb, Croatia, May 10–14, 2020). Pt. II. Cham: Springer, 2020. P. 280–310. (Lect. Notes Comput. Sci.; V. 12106).
5. **Langenberg B., Pham H., Steinwandt R.** Reducing the cost of implementing the advanced encryption standard as a quantum circuit // IEEE Trans. Quantum Eng. 2020. V. 1. P. 1–12.
6. **Zou J., Wei Z., Sun S., Liu X., Wu W.** Quantum circuit implementations of AES with fewer qubits // Advances in cryptology — ASIACRYPT 2020. Proc. 26th Int. Conf. Theory Appl. Cryptol. Inf. Secur. (Daejeon, South Korea, Dec. 7–11, 2020). Pt. II. Cham: Springer, 2020. P. 697–726. (Lect. Notes Comput. Sci.; V. 12492).
7. **Almazrooie M., Samsudin A., Abdullah R., Mutter K. N.** Quantum exhaustive key search with simplified-DES as a case study // SpringerPlus. 2016. V. 5, No. 1. P. 1–19.
8. **Денисенко Д. В., Никитенкова М. В.** Применение квантового алгоритма Гровера в задаче поиска ключа блочного шифра SDES // Журн. эксперим. и теор. физики. 2019. Т. 155, № 1. С. 32–53.
9. **Bernstein D. J.** Introduction to post-quantum cryptography // Post-quantum cryptography. Heidelberg: Springer, 2009. P. 1–14.
10. **Alagic G., Alperin-Sheriff J., Apon D.** [et al.]. Status report on the second round of the NIST post-quantum cryptography standardization process. Nat. Inst. Stand. Technol. interag. intern. rep. 8309. Gaithersburg, MD: NIST, 2020. 39 p. Available at `doi.org/10.6028/NIST.IR.8309` (accessed Apr. 26, 2023).

11. **Albrecht M. R., Gheorghiu V., Postlethwaite E. W., Schanck J. M.** Estimating quantum speedups for lattice sieves // Advances in cryptology — ASIACRYPT 2020. Proc. 26th Int. Conf. Theory Appl. Cryptol. Inf. Secur. (Daejeon, South Korea, Dec. 7–11, 2020). Pt. II. Cham: Springer, 2020. P. 583–613. (Lect. Notes Comput. Sci.; V. 12492).
12. **Perriello S., Barenghi A., Pelosi G.** A complete quantum circuit to solve the information set decoding problem // Proc. 2021 IEEE Int. Conf. Quantum Comput. Eng. (Broomfield, CO, USA, Oct. 17–22, 2021). Los Alamitos, CA: IEEE Comput. Soc., 2021. P. 366–377.
13. **Micciancio D.** Inapproximability of the shortest vector problem: Toward a deterministic reduction // Theory Comput. 2012. V. 8, No. 1. P. 487–512.
14. **Micciancio D., Voulgaris P.** Faster exponential time algorithms for the shortest vector problem // Proc. 21st Annu. ACM-SIAM Symp. Discrete Algorithms (Austin, TX, USA, Jan. 17–19, 2010). Philadelphia, PA: SIAM, 2010. P. 1468–1480.
15. **Pujol X., Stehlé D.** Solving the shortest lattice vector problem in time $2^{2,465n}$. San Diego: Univ. California, 2009. (Cryptol. ePrint Archive; Pap. 2009/605). Available at `eprint.iacr.org/2009/605` (accessed Apr. 26, 2023).
16. **Nguyen P. Q., Vidick T.** Sieve algorithms for the shortest vector problem are practical // J. Math. Cryptol. 2008. V. 2, No. 2. P. 181–207.
17. **Laarhoven T.** Sieving for shortest vectors in lattices using angular locality-sensitive hashing // Advances in cryptology — CRYPTO 2015. Proc. 35th Annu. Cryptol. Conf. (Santa Barbara, CA, USA, Aug. 16–20, 2015). Pt. I. Heidelberg: Springer, 2015. P. 3–22. (Lect. Notes Comput. Sci.; V. 9215).
18. **Micciancio D., Voulgaris P.** A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations // Proc. 42nd ACM Symp. Theory Comput. (Cambridge, MA, USA, June 5–8, 2010). New York: ACM, 2010. P. 351–358.
19. **Aggarwal D., Dadush D., Regev O., Stephens-Davidowitz N.** Solving the shortest vector problem in 2^n time using discrete Gaussian sampling // Proc. 47th ACM Symp. Theory Comput. (Portland, OR, USA, June 14–17, 2015). New York: ACM, 2015. P. 733-742.
20. **Bai S., Laarhoven T., Stehlé D.** Tuple lattice sieving // LMS J. Comput. Math. 2016. V. 19, No. A. P. 146–162.
21. **Becker A., Ducas L., Gama G., Laarhoven T.** New directions in nearest neighbor searching with applications to lattice sieving // Proc. 27th Annu. ACM-SIAM Symp. Discrete Algorithms (Arlington, VA, USA, Jan. 10–12, 2016). Philadelphia, PA: SIAM, 2016. 2016. P. 10–24.
22. **Laarhoven T., Mosca M., van de Pol J.** Finding shortest lattice vectors faster using quantum search // Des. Codes Cryptogr. 2015. V. 77, No. 2. P. 375–400.
23. **Grover L. K.** A fast quantum mechanical algorithm for database search // Proc. 28th ACM Symp. Theory Comput. (Philadelphia, PA, USA, May 22–24, 1996). New York: ACM, 1996. P. 212–219.

24. Nielsen M. A., Chuang I. L. Quantum computation and quantum information. Cambridge: Camb. Univ. Press, 2010.
25. Boyer M., Brassard G., Hoyer P., Tapp A. Tight bounds on quantum searching // Fortschr. Phys. 1998. V. 46, No. 4–5. P. 493–505.
26. Moore C., Nilsson M. Parallel quantum computation and quantum codes // SIAM J. Comput. 2001. V. 31, No. 3. P. 799–815.
27. Diffie W., Hellman M. E. New directions in cryptography // IEEE Trans. Inf. Theory. 1976. V. 22, No. 6. P. 644–654.
28. Rivest R. L., Shamir A., Adleman L. A method for obtaining digital signatures and public-key cryptosystems // Commun. ACM. 1978. V. 21, No. 2. P. 120–126.
29. Shor P. W. Algorithms for quantum computation: Discrete logarithms and factoring // Proc. 35th Annu. Symp. Found. Comput. Sci. (Santa Fe, USA, Nov. 20–22, 1994). Los Alamitos, CA: IEEE Comput. Soc., 1994. P. 124–134.
30. Gidney C., Ekerå M. How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits // Quantum. 2021. V. 5. P. 433.
31. Chen C., Danba O., Hoffstein J. [et al.]. NTRU. Algorithm specifications and supporting documentation. Eindhoven: Eindh. Univ. Technol., 2019. Available at `ntru.org/f/ntru-20190330.pdf` (accessed Apr. 26, 2023).
32. D'Anvers J.-P., Karmakar A., Roy S. S., Vercauteren F. SABER: Mod-LWR based KEM. Leuven: KU Leuven, 2017. Available at `esat.kuleuven.be/cosic/pqcrypto/saber/files/saberspecround1.pdf` (accessed Apr. 26, 2023).
33. Avanzi R., Bos J., Ducas L. [et al.]. CRYSTALS-Kyber. Algorithm specifications and supporting documentation. Amsterdam: Cent. Wiskd. Inform., 2021. Available at `pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf` (accessed Apr. 26, 2023).
34. Alagic G., Apon D., Cooper D. [et al.]. Status report on the third round of the NIST post-quantum cryptography standardization process. Nat. Inst. Stand. Technol. interag. intern. rep. NIST IR 8413-upd1. Gaithersburg, MD: NIST, 2022. 102 p. Available at `doi.org/10.6028/NIST.IR.8413-upd1` (accessed Apr. 26, 2023).

Бахарев Александр Олегович

Статья поступила
18 ноября 2022 г.
После доработки —
21 марта 2023 г.
Принята к публикации
3 апреля 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
July–September 2023. Vol. 30, No. 3. P. 5–42

UDC 519.7 DOI 10.33048/daio.2023.30.756

ESTIMATES OF IMPLEMENTATION COMPLEXITY FOR QUANTUM CRYPTANALYSIS OF POST-QUANTUM LATTICE-BASED CRYPTOSYSTEMS

A. O. Bakharev

Novosibirsk State University,
2 Pirogov Street, 630090 Novosibirsk, Russia
E-mail: a.bakharev@g.nsu.ru

Abstract. Due to the development of quantum computing, there is a need for the development and analysis of cryptosystems resistant to attacks using a quantum computer (post-quantum cryptography algorithms). The security of many well-known post-quantum cryptosystems based on lattice theory depends on the complexity of solving the shortest vector problem (SVP). In this paper, a model of quantum oracle developed from Grover's algorithm is described to implement a hybrid quantum-classical algorithm based on GaussSieve. This algorithm can be used for attacks on cryptosystems, the security of which depends on solving the SVP problem. Upper bounds for the number of qubits and the depth of the circuit were obtained for two implementations of the proposed quantum oracle model: minimizing the number of qubits and minimizing the circuit depth. The complexity of implementing the proposed quantum oracle model to attack post-quantum lattice-based cryptosystems that are finalists of the NIST post-quantum cryptography competition is analyzed. Tab. 6, illustr. 12, bibliogr. 34.

Keywords: quantum search, public-key cryptography, lattice-based cryptography, post-quantum cryptography, Grover's algorithm, quantum computation.

REFERENCES

1. Information technology. Cryptographic data security. Block ciphers, *GOST R 34.12—2015* (Standartinform, Moscow, 2015) [Russian].

This research is supported by the Mathematical Center in Akademgorodok under the agreement with the Ministry of Science and Higher Education of Russia (Agreement 075–15–2022–282).

English version: Journal of Applied and Industrial Mathematics **17** (3) (2023).

2. **D. V. Denisenko, G. B. Marshalko, M. V. Nikitenkova, V. I. Rudskoi,** and **V. A. Shishkin,** Estimating the complexity of Grover's algorithm for key search of block ciphers defined by GOST R 34.12—2015, *Zh. Éksp. Teor. Fiz.* **155** (4), 645–653 (2019) [Russian] [*J. Exp. Theor. Phys.* **128** (4), 552–559 (2019)].
3. **M. Grassl, B. Langenberg, M. Roetteler,** and **R. Steinwandt,** Applying Grover's algorithm to AES: Quantum resource estimates, in *Post-Quantum Cryptography* (Proc. 7th Int. Workshop, Fukuoka, Japan, Feb. 24–26, 2016) (Springer, Cham, 2016), pp. 29–43 (Lect. Notes Comput. Sci., Vol. 9606).
4. **S. Jaques, M. Naehrig, M. Roetteler,** and **F. Virdia,** Implementing Grover oracles for quantum key search on AES and LowMC, in *Advances in Cryptology — EUROCRYPT 2020* (Proc. 39th Annu. Int. Conf. Theory Appl. Cryptogr. Techn., Zagreb, Croatia, May 10–14, 2020), Pt. II (Springer, Cham, 2020), pp. 280–310 (Lect. Notes Comput. Sci., Vol. 12106).
5. **B. Langenberg, H. Pham,** and **R. Steinwandt,** Reducing the cost of implementing the advanced encryption standard as a quantum circuit, *IEEE Trans. Quantum Eng.* **1**, 1–12 (2020).
6. **J. Zou, Z. Wei, S. Sun, X. Liu,** and **W. Wu,** Quantum circuit implementations of AES with fewer qubits, in *Advances in Cryptology — ASIACRYPT 2020* (Proc. 26th Int. Conf. Theory Appl. Cryptol. Inf. Secur., Daejeon, South Korea, Dec. 7–11, 2020), Pt. II (Springer, Cham, 2020), pp. 697–726 (Lect. Notes Comput. Sci., Vol. 12492).
7. **M. Almazrooie, A. Samsudin, R. Abdullah,** and **K. N. Mutter,** Quantum exhaustive key search with simplified-DES as a case study, *SpringerPlus* **5** (1), 1–19 (2016).
8. **D. V. Denisenko** and **M. V. Nikitenkova,** Application of Grover's quantum algorithm for SDES key searching, *Zh. Éksp. Teor. Fiz.* **155** (1), 32–53 (2019) [Russian] [*J. Exp. Theor. Phys.* **128** (1), 25–44 (2019)].
9. **D. J. Bernstein,** Introduction to post-quantum cryptography, in *Post-Quantum Cryptography* (Springer, Heidelberg, 2009), pp. 1–14.
10. **G. Alagic, J. Alperin-Sheriff, D. Apon,** [et al.], Status report on the second round of the NIST post-quantum cryptography standardization process, *Nat. Inst. Stand. Technol. Interag. Intern. Rep. 8309* (NIST, Gaithersburg, MD, 2020). Available at `doi.org/10.6028/NIST.IR.8309` (accessed Apr. 26, 2023).
11. **M. R. Albrecht, V. Gheorghiu, E. W. Postlethwaite,** and **J. M. Schanck,** Estimating quantum speedups for lattice sieves, in *Advances in Cryptology — ASIACRYPT 2020* (Proc. 26th Int. Conf. Theory Appl. Cryptol. Inf. Secur., Daejeon, South Korea, Dec. 7–11, 2020), Pt. II (Springer, Cham, 2020), pp. 583–613 (Lect. Notes Comput. Sci., Vol. 12492).
12. **S. Perriello, A. Barenghi,** and **G. Pelosi,** A complete quantum circuit to solve the information set decoding problem, in *Proc. 2021 IEEE Int. Conf. Quantum Comput. Eng., Broomfield, CO, USA, Oct. 17–22, 2021* (IEEE Comput. Soc., Los Alamitos, CA, 2021), pp. 366–377.

13. **D. Micciancio,** Inapproximability of the shortest vector problem: Toward a deterministic reduction, *Theory Comput.* **8** (1), 487–512 (2012).
14. **D. Micciancio** and **P. Voulgaris,** Faster exponential time algorithms for the shortest vector problem, in *Proc. 21st Annu. ACM-SIAM Symp. Discrete Algorithms, Austin, TX, USA, Jan. 17–19, 2010* (SIAM, Philadelphia, PA, 2010), pp. 1468–1480.
15. **X. Pujol** and **D. Stehlé,** Solving the shortest lattice vector problem in time $2^{2,465n}$ (Univ. California, San Diego, 2009) (Cryptol. ePrint Archive, Pap. 2009/605). Available at `eprint.iacr.org/2009/605` (accessed Apr. 26, 2023).
16. **P. Q. Nguyen** and **T. Vidick,** Sieve algorithms for the shortest vector problem are practical, *J. Math. Cryptol.* **2** (2), 181–207 (2008).
17. **T. Laarhoven,** Sieving for shortest vectors in lattices using angular locality-sensitive hashing, in *Advances in Cryptology — CRYPTO 2015* (Proc. 35th Annu. Cryptol. Conf., Santa Barbara, CA, USA, Aug. 16–20, 2015), Pt. I (Springer, Heidelberg, 2015), pp. 3–22 (Lect. Notes Comput. Sci., Vol. 9215).
18. **D. Micciancio** and **P. Voulgaris,** A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations, in *Proc. 42nd ACM Symp. Theory Comput., Cambridge, MA, USA, June 5–8, 2010* (ACM, New York, 2010), pp. 351–358.
19. **D. Aggarwal, D. Dadush, O. Regev,** and **N. Stephens-Davidowitz,** Solving the shortest vector problem in 2^n time using discrete Gaussian sampling, in *Proc. 47th ACM Symp. Theory Comput., Portland, OR, USA, June 14–17, 2015* (ACM, New York, 2015), pp. 733-742.
20. **S. Bai, T. Laarhoven,** and **D. Stehlé,** Tuple lattice sieving, *LMS J. Comput. Math.* **19** (A), 146–162 (2016).
21. **A. Becker, L. Ducas, G. Gama,** and **T. Laarhoven,** New directions in nearest neighbor searching with applications to lattice sieving, in *Proc. 27th Annu. ACM-SIAM Symp. Discrete Algorithms, Arlington, VA, USA, Jan. 10–12, 2016* (SIAM, Philadelphia, PA, 2016), pp. 10–24.
22. **T. Laarhoven, M. Mosca,** and **J. van de Pol,** Finding shortest lattice vectors faster using quantum search, *Des. Codes Cryptogr.* **77** (2), 375–400 (2015).
23. **L. K. Grover,** A fast quantum mechanical algorithm for database search, in *Proc. 28th ACM Symp. Theory Comput., Philadelphia, PA, USA, May 22–24, 1996* (ACM, New York, 1996), pp. 212–219.
24. **M. A. Nielsen** and **I. L. Chuang,** *Quantum Computation and Quantum Information* (Camb. Univ. Press, Cambridge, 2010).
25. **M. Boyer, G. Brassard, P. Hoyer,** and **A. Tapp,** Tight bounds on quantum searching, *Fortschr. Phys.* **46** (4–5), 493–505 (1998).
26. **C. Moore** and **M. Nilsson,** Parallel quantum computation and quantum codes, *SIAM J. Comput.* **31** (3), 799–815 (2001).
27. **W. Diffie** and **M. E. Hellman,** New directions in cryptography, *IEEE Trans. Inf. Theory* **22** (6), 644–654 (1976).
28. **R. L. Rivest, A. Shamir,** and **L. Adleman,** A method for obtaining digital signatures and public-key cryptosystems, *Commun. ACM* **21** (2), 120–126 (1978).

29. P. W. Shor, Algorithms for quantum computation: Discrete logarithms and factoring, in *Proc. 35th Annu. Symp. Found. Comput. Sci., Santa Fe, USA, Nov. 20–22, 1994* (IEEE Comput. Soc., Los Alamitos, CA, 1994), pp. 124–134.

30. C. Gidney and **M. Ekerå,** How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, *Quantum* **5**, 433 (2021).

31. C. Chen, O. Danba, J. Hoffstein, [et al.], NTRU. Algorithm specifications and supporting documentation (Eindh. Univ. Technol., Eindhoven, 2019). Available at `ntru.org/f/ntru-20190330.pdf` (accessed Apr. 26, 2023).

32. J.-P. D'Anvers, A. Karmakar, S. S. Roy, and **F. Vercauteren,** SABER: Mod-LWR based KEM (KU Leuven, Leuven, 2017). Available at `esat.kuleuven.be/cosic/pqcrypto/saber/files/saberspecround1.pdf` (accessed Apr. 26, 2023).

33. R. Avanzi, J. Bos, L. Ducas, [et al.], CRYSTALS-Kyber. Algorithm specifications and supporting documentation (Cent. Wiskd. Inform., Amsterdam, 2021). Available at `pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf` (accessed Apr. 26, 2023).

34. G. Alagic, D. Apon, D. Cooper, [et al.], Status report on the third round of the NIST post-quantum cryptography standardization process, *Nat. Inst. Stand. Technol. Interag. Intern. Rep. NIST IR 8413-upd1* (NIST, Gaithersburg, MD, 2022). Available at `doi.org/10.6028/NIST.IR.8413-upd1` (accessed Apr. 26, 2023).

Aleksandr O. Bakharev

Received November 18, 2022
Revised March 21, 2023
Accepted April 3, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль–сентябрь 2023. Т. 30, № 3. С. 43–56

УДК 519.8+518.25 DOI 10.33048/daio.2023.30.774

ДОПОЛНИТЕЛЬНЫЕ ОГРАНИЧЕНИЯ ДЛЯ ДИНАМИЧЕСКОЙ ЗАДАЧИ КОНКУРЕНТНОГО РАЗМЕЩЕНИЯ

В. Л. Береснев[1, 2, a], *А. А. Мельников*[1, 2, b]

[1] Институт математики им. С. Л. Соболева,
пр. Акад. Коптюга, 4, 630090 Новосибирск, Россия

[2] Новосибирский гос. университет,
ул. Пирогова, 2, 630090 Новосибирск, Россия

E-mail: [a] beresnev@math.nsc.ru, [b] melnikov@math.nsc.ru

Аннотация. Рассматривается математическая модель конкурентного размещения объектов (предприятий), в которой соперничающие стороны (Лидер и Последователь) принимают решения с учётом изменяющегося множества потребителей на рассматриваемом горизонте планирования, состоящего из заданного числа периодов времени. При этом предполагается, что Лидер принимает решение об открытии своих объектов в начале горизонта планирования, а Последователь имеет возможность обновлять своё решение на каждом из периодов времени. В работе исследуется возможность применения для рассматриваемой динамической задачи конкурентного размещения способа построения наилучшего решения, базирующегося на использовании HP-релаксации исследуемой двухуровневой модели. Основным элементом этого подхода является построение дополнительных ограничений для усиления HP-релаксации исследуемой двухуровневой задачи и вычисления верхних границ значений целевой функции этой задачи. В работе предлагаются семейства дополнительных ограничений для усиления HP-релаксации рассматриваемой динамической задачи, позволяющие вычислять нетривиальные верхние границы. Библиогр. 13.

Ключевые слова: игра Штакельберга, двухуровневое программирование, конкурентное размещение, правильные неравенства.

Исследование выполнено при финансовой поддержке Российского научного фонда (проект № 23–21–00082).

Введение

Задачи конкурентного размещения формируют семейство оптимизационных моделей, рассматривающих решения о выборе состава средств обслуживания потребителей в рамках игрового взаимодействия нескольких сторон. Наиболее изученным является случай двух сторон. Среди таких моделей выделяется класс моделей, основанных на игре Штакельберга, в которых предполагается, что одна из конкурирующих сторон (Лидер) принимает решение первой, после чего вторая сторона (Последователь), зная решение Лидера, принимает своё решение. В условиях, когда решение каждой из сторон предполагается окончательным, а возможность его модификации при моделировании не учитывается, будем говорить о *статических* моделях конкурентного размещения. Описание состояния развития области статического конкурентного размещения можно найти в соответствующих обзорах [1–6].

Динамические задачи размещения объектов обобщают свои статические аналоги на случай, когда существенные параметры задачи (например, множество потребителей) не фиксированы, а могут изменяться на рассматриваемом горизонте планирования. Обычно предполагается, что горизонт планирования разбит на периоды, на каждом из которых происходят изменения внешних факторов и, возможно, корректировка принятых ранее решений по размещению объектов.

Следует различать явные и неявные динамические модели [7]. В неявных динамических моделях учитываются изменения существенных факторов, влияющих на принятие решений, однако они статичны в том смысле, что объекты открываются в начале горизонта планирования и используются на всех периодах. Явные динамические модели предполагают определённую стратегию изменения состава используемых объектов на горизонте планирования.

Этот класс задач не нов для исследователей. Подробный обзор этой области можно найти в [8]. Для решения динамических задач с учётом различных дополнительных факторов и особенностей рассматриваемых средств обслуживания используются в основном эвристические алгоритмы. Предлагаются также алгоритмы, построенные на другой основе. В [9] для решения достаточно общей динамической задачи (без закрытия объектов) рассматривается подход, базирующийся на декомпозиции Бендерса и использующий специальный метод внутренних точек для решения подзадач Бендерса. Предложенный подход позволяет построить алгоритм с впечатляющими вычислительными возможностями.

В предлагаемой работе рассматривается динамическая задача конкурентного размещения, сформулированная на основе динамической задачи размещения и задачи конкурентного размещения [10, 11]. В модели конкурентного размещения объектов (предприятий) присутствуют две

соперничающие стороны (Лидер и Последователь), которые последовательно открывают свои предприятия для обслуживания заданного множества потребителей. Первым открывает свои предприятия Лидер, а затем Последователь, зная размещение предприятий Лидера, принимает решение о размещении своих объектов.

При этом каждая из сторон стремится захватить потребителей и получить максимальную прибыль от их обслуживания. В случае динамической задачи конкурентного размещения указанное взаимодействие сторон происходит в каждом периоде горизонта планирования. При этом каждая из сторон стремится получить наибольшую суммарную прибыль на всём горизонте планирования.

Одна из первых постановок динамической задачи конкурентного размещения содержится в [12]. Работ с постановками явных динамических задач конкурентного размещения, содержащих предложения по способам их решения, нам не известно.

В настоящей работе объектом исследования является модель, в которой Лидер открывает свои предприятия в начальный момент горизонта планирования и не изменяет состава своих открытых предприятий внутри самого горизонта. Последователь, напротив, имеет возможность пополнять состав своих предприятий на каждом из периодов времени, составляющих горизонт планирования. Исследование такой постановки оправданно, поскольку алгоритм построения решения такой модели может быть использован как основная процедура последовательного построения решения динамической задачи общего вида, в которой Лидер в начале каждого периода рассматриваемого горизонта планирования может реконструировать множество открытых им предприятий.

Предлагаемая модель есть задача двухуровневого математического программирования, включающая задачу верхнего уровня (задачу Лидера) и задачу нижнего уровня (задачу Последователя). В качестве решения этой задачи принимается пессимистическое оптимальное решение [13]. Такое решение предполагает, что выбираемое Последователем оптимальное решение задачи нижнего уровня является наихудшим решением для Лидера.

Для построения пессимистических допустимых решений задач конкурентного размещения используются алгоритмы на основе вычислительной схемы неявного перебора. При построении таких алгоритмов важное значение имеют процедуры вычисления верхних границ целевой функции задачи Лидера на подмножествах решений. В [11] предложен способ вычисления таких границ, базирующийся на так называемой high-point relaxation (HP-релаксации). Релаксированная задача получается из исходной двухуровневой модели исключением из неё целевой функции задачи нижнего уровня. Полученная задача даёт завышенную верхнюю

границу, и для её улучшения релаксированную задачу необходимо усилить дополнительными ограничениями, стимулирующими переменные нижнего уровня принимать ненулевые значения. Такими ограничениями являются так называемые c-отсечения [11], которые выполняются на пессимистических допустимых решениях исходной задачи и позволяют получать эффективные верхние границы. Модифицированные c-отсечения могут быть использованы как дополнительные ограничения для HP-релаксации рассматриваемой динамической задачи. Однако только таких ограничений недостаточно для получения нетривиальных верхних границ. Необходимы ещё ограничения, которые стимулируют открытие предприятий Последователя в начальные периоды горизонта планирования. Основным результатом работы является построение нового семейства дополнительных ограничений (d-отсечений), отвечающих указанным требованиям. Совместное использование c- и d-отсечений для усиления HP-релаксации имеет перспективы получения качественных верхних границ для целевой функции исследуемой динамической задачи.

1. Формулировка задачи

Для формальной записи динамической задачи конкурентного размещения DCompFLP используем следующие обозначения множеств, рассматриваемых в модели:

I — множество мест возможного размещения предприятий;

T — множество периодов рассматриваемого горизонта планирования;

J_t, $t \in T$ — множество потребителей в период времени t. Без ограничения общности предполагаем, что $J_{t_1} \cap J_{t_2} = \varnothing$ при любых t_1, $t_2 \in T$, $t_1 \neq t_2$. Обозначим через $J = \bigcup\limits_{t \in T} J_t$ множество потребителей на всех периодах.

Используем также следующие обозначения параметров, рассматриваемых в модели:

f_i, $i \in I$ — фиксированные затраты на открытие Лидером объекта i;

g_{it}, $i \in I$, $t \in T$ — фиксированные затраты на открытие Последователем объекта i в период t (для всякого $i \in I$ будем считать, что величины (g_{it}) убывают с ростом номера t);

p_{ij}, $i \in I$, $j \in J$ — величина дохода, получаемого Лидером от обслуживания открытым им объектом i потребителя j;

q_{ij}, $i \in I$, $j \in J$ — величина дохода, получаемого Последователем от обслуживания открытым им объектом i потребителя j.

В модели используются следующие двоичные переменные:

x_i, $i \in I$, равна единице, если Лидер открывает предприятие i, и нулю в противном случае;

z_{it}, $i \in I$, $t \in T$, равна единице, если Последователь открывает объект i в период t, и нулю в противном случае;

x_{ij}, $i \in I$, $j \in J$, равна единице, если объект i, открытый Лидером, назначен для обслуживания потребителя j, и нулю в противном случае;

z_{ij}, $i \in I$, $j \in J$, равна единице, если предприятие i, открытое Последователем, назначено для обслуживания потребителя j, и нулю в противном случае.

Будем полагать, что выбор объекта для обслуживания потребителя $j \in J$ подчинён предпочтениям этого потребителя, заданным линейным порядком $\succeq_j$ на множестве I. Для потребителя $j \in J$ и пары объектов $i_1, i_2 \in I$ будем писать $i_1 \succ_j i_2$ в случае, если i_1 предпочтителен для j по отношению i_2. Обозначение $i_1 \succeq_j i_2$ эквивалентно тому, что $i_1 \succ_j i_2$ или $i_1 = i_2$. В модели полагаем, что для обслуживания потребителя может быть назначено любое открытое предприятие одной из сторон, которое более предпочтительно, чем каждое из предприятий, открытых другой стороной.

С использованием введённых обозначений динамическая задача конкурентного размещения DCompFLP записывается в виде двухуровневой задачи математического программирования:

$$\max_{(x_i),(x_{ij})} \Big(-\sum_{i\in I} f_i x_i + \sum_{t\in T}\sum_{i\in I}\sum_{j\in J_t} p_{ij} x_{ij} \Big), \tag{1}$$

$$\sum_{\tau=1}^{t} z^0_{i\tau} + \sum_{k:\, i\succeq_j k} x_{kj} \leqslant 1, \quad i \in I,\, t \in T,\, j \in J_t, \tag{2}$$

$$x_i \geqslant x_{ij}, \quad i \in I,\, j \in J, \tag{3}$$

$$x_i, x_{ij} \in \{0,1\}, \quad i \in I,\, j \in J, \tag{4}$$

$$(z^0_{it}), (z^0_{ij}) \text{ — оптимальное решение задачи}, \tag{5}$$

$$\max_{(z_{it}),(z_{ij})} \Big(-\sum_{t\in T}\sum_{i\in I} g_{it} z_{it} + \sum_{t\in T}\sum_{i\in I}\sum_{j\in J_t} q_{ij} z_{ij} \Big), \tag{6}$$

$$x_i + \sum_{t\in T} z_{it} \leqslant 1, \quad i \in I, \tag{7}$$

$$x_i + \sum_{k:\, i\succeq_j k} z_{kj} \leqslant 1, \quad i \in I,\, j \in J, \tag{8}$$

$$\sum_{\tau=1}^{t} z_{i\tau} \geqslant z_{ij}, \quad i \in I,\, t \in T,\, j \in J_t, \tag{9}$$

$$z_{it}, z_{ij} \in \{0,1\}, \quad i \in I,\, t \in T,\, j \in J. \tag{10}$$

Эта модель, как и всякая двухуровневая задача, включает задачу верхнего уровня (1)–(5) (задачу Лидера) и задачу нижнего уровня (6)–(10) (задачу Последователя).

Целевая функция (1) задачи Лидера выражает величину прибыли Лидера на рассматриваемом горизонте планирования. Прибыль складывается из затрат на открытие предприятий и суммарных доходов от обслуживания потребителей по всем периодам рассматриваемого горизонта планирования. Неравенства (2) гарантируют, что Лидер использует для обслуживания потребителей только предприятия, которые для этого потребителя более предпочтительны, чем любое открытое предприятия Последователя. Кроме того, из этих ограничений следует, что для обслуживания каждого потребителя может быть назначено не более одного открытого предприятия Лидера. Условия (3) означают, что для обслуживания потребителя может быть назначено только открытое предприятие Лидера.

Целевая функция (6) задачи Последователя аналогична целевой функции задачи Лидера и выражает величину суммарной прибыли Последователя на рассматриваемом горизонте планирования. Ограничения (7) означают, что Последователь не может открыть предприятие, открытое Лидером. Условия (8) и (9) аналогичны ограничениям (2) и (3).

Обозначим задачу верхнего уровня (1)–(5) через $\mathcal{L}$, а задачу нижнего уровня (6)–(10) — через $\mathcal{F}$. Для модели (1)–(10) в целом будем использовать обозначение $(\mathcal{L}, \mathcal{F})$.

С учётом специфики рассматриваемой задачи уточним понятия её допустимого и оптимального решений. Будем рассматривать пары (X, Z), где $X = ((x_i), (x_{ij}))$ — допустимое решение задачи $\mathcal{L}$ при заданных векторах $z_t = (z_{it})$, $t \in T$, а $Z = ((z_{it}), (z_{ij}))$ — допустимое решение задачи $\mathcal{F}$ при заданном векторе $x = (x_i)$. При заданных векторах размещения x и (z_t) будем рассматривать только такие пары (X, Z), в которых переменные (x_{ij}) принимают оптимальные значения. Тогда через $L(X, Z)$ обозначим значение целевой функции задачи $\mathcal{L}$ на допустимом решении (X, Z), а через $F(Z)$ — значение целевой функции задачи $\mathcal{F}$ на допустимом решении Z.

Пару (X, Z), где $X = ((x_i), (x_{ij}))$, назовём *допустимым решением* задачи $(\mathcal{L}, \mathcal{F})$, порождённым двоичным вектором $x = (x_i)$, если X — допустимое решение задачи $\mathcal{L}$, а Z — оптимальное решение задачи $\mathcal{F}$. Допустимое решение (X, Z), порождённое вектором x, назовём *пессимистическим* допустимым решением задачи $(\mathcal{L}, \mathcal{F})$, если $L(X, Z) \leqslant L(X', Z')$ для любого допустимого решения (X', Z'), порождённого вектором x. Наилучшее пессимистическое допустимое решение назовём *пессимистическим оптимальным* решением задачи $(\mathcal{L}, \mathcal{F})$.

2. Усиленная оценочная задача

Построение алгоритмов неявного перебора для вычисления оптимального решения задачи $(\mathcal{L}, \mathcal{F})$ предполагает наличие способа вычисления

верхних границ целевой функции исследуемой задачи на рассматриваемых подмножествах решений. В качестве основы для вычисления таких границ для задачи $(\mathcal{L}, \mathcal{F})$ используем её НР-релаксацию. Эта задача получается из рассматриваемой двухуровневой задачи исключением из неё требования оптимальности значений переменных задачи нижнего уровня. Понятно, что в оптимальном решении релаксации задачи $(\mathcal{L}, \mathcal{F})$ все переменные задачи $\mathcal{F}$ будут принимать нулевые значения, и получаемая верхняя граница будет существенно завышенной. Стало быть, наша цель состоит в построении дополнительных ограничений релаксированной задачи, «заставляющих» эти переменные принимать ненулевые значения. Если дополнительные ограничения выполняются для всех пессимистических допустимых решений, то оптимальное значение целевой функции релаксированной задачи с такими дополнительными ограничениями будет давать требуемую верхнюю границу. Такую задачу будем называть *усиленной оценочной задачей* (strong estimating problem, SEP).

2.1. Релаксированная задача. Исключение из задачи $(\mathcal{L}, \mathcal{F})$ условия (5) означает исключение из неё целевой функции (6) и переменных z_{ij}, $i \in I$, $j \in J$. Эти переменные перестают играть всякую роль, поскольку их нулевые значения оказываются допустимыми при любых значениях остальных переменных. В результате получаем, что НР-релаксация задачи $(\mathcal{L}, \mathcal{F})$ есть задача целочисленного программирования вида

$$\max_{(x_i),(x_{ij}),(z_{it})} \Big(-\sum_{i\in I} f_i x_i + \sum_{t\in T} \sum_{i\in I} \sum_{j\in J_t} p_{ij} x_{ij} \Big),$$

$$\sum_{\tau=1}^{t} z_{i\tau} + \sum_{k\colon i \succ_j k} x_{kj} \leqslant 1, \quad i \in I,\ t \in T,\ j \in J_t,$$

$$x_i \geqslant x_{ij}, \quad i \in I,\ j \in J,$$

$$x_i + \sum_{t\in T} z_{it} \leqslant 1, \quad i \in I,$$

$$x_i, x_{ij}, z_{it} \in \{0, 1\}, \quad i \in I,\ t \in T,\ j \in J.$$

Для этой задачи построим дополнительные ограничения двух видов. При этом будем иметь в виду, что ограничения должны не только выполняться на допустимых решениях исходной задачи $(\mathcal{L}, \mathcal{F})$, но и нарушаться на оптимальном решении текущей задачи SEP. В таком случае дополнительные ограничения будут отсекать оптимальное решение задачи SEP, что в конечном итоге приведёт к уменьшению оптимального значения целевой функции и, следовательно, к улучшению верхней границы.

При построении дополнительных ограничений используем следующие обозначения. Пусть $x = (x_i)$, $i \in I$ — ненулевой двоичный вектор, $J' \subseteq J$, $J' \neq \varnothing$ и $j \in J'$. Положим

$I^1(x) = \{i \in I \mid x_i = 1\}$;

$\alpha_j(x)$ — такой элемент $i' \in I^1(x)$, что $i' \succeq_j k$ для всех $k \in I^1(x)$;

$\alpha_{J'}(x) = \{\alpha_j(x) \mid j \in J'\}$;

$N_j(x) = \{i \in I \mid i \succ_j \alpha_j(x)\}$;

$N_{J'}(x) = \bigcup\limits_{j \in J'} N_j(x)$.

2.2. c-Отсечения. Пусть (X', z') — допустимое решение задачи SEP для задачи $(\mathcal{L}, \mathcal{F})$. Пусть $u'_{it} = \sum\limits_{\tau=1}^{t} z'_{i\tau}$, $i \in I$, $t \in T$, и пусть $x' = (x'_i)$, $u'_t = (u'_{it})$, $t \in T$. Для всякого $t \in T$ положим $J_{0t} = J_t$, если u'_t — нулевой вектор, и $J_{0t} = \{j \in J_t \mid \alpha_j(x') \succ_j \alpha_j(u'_t)\}$ в противном случае. Исходными данными для построения с-отсечения являются элементы $t_0 \in T$ и $k \in N_{J_{0t_0}}(x')$. Рассмотрим множества $J_{0t}(k) = \{j \in J_{0t} \mid k \succ_j \alpha_j(x')\}$, $t \in T$, $t \geqslant t_0$, и пусть $l_0 \geqslant t_0$ — наименьший номер, для которого справедливы неравенства

$$\sum_{t=t_0}^{l_0} \sum_{j \in J_{0t}(k)} p_{\alpha_j(x')j} > 0, \tag{11}$$

$$\sum_{t=t_0}^{l_0} \sum_{j \in J_{0t}(k)} q_{kj} \geqslant g_{kt_0}. \tag{12}$$

Пусть $J' \subseteq \bigcup\limits_{t=t_0}^{l_0} J_{0t}(k)$ — подмножество, для которого выполняются аналогичные условия

$$\sum_{j \in J'} p_{\alpha_j(x')j} > 0, \tag{13}$$

$$\sum_{j \in J'} q_{kj} \geqslant g_{kt_0}. \tag{14}$$

Тогда неравенство

$$\sum_{i \in N_{J'}(x')} u_{il_0} \geqslant 1 + \sum_{i \in \alpha_{J'}(x')} (x_i - 1) - \sum_{i \in N_{J'}(x')} x_i \tag{15}$$

назовём *с-отсечением* решения (X', z'), порождённым номерами $t_0 \in T$ и $k \in N_{J_{0t_0}}(x')$.

Понятно, что по построению с-отсечения решения (X', z') неравенство (15) нарушается на этом решении. Выполняется также следующее свойство с-отсечения.

Утверждение 1. *Пусть* (X, Z), $X = ((x_i), (x_{ij}))$, $Z = ((z_{it}), (z_{ij}))$, *есть пессимистическое допустимое решение задачи* $(\mathcal{L}, \mathcal{F})$. *Тогда неравенство* (15) *выполняется на этом решении.*

Доказательство. Пусть вектор $x = (x_i)$, порождающий рассмотренное решение (X, Z), такой, что $x_i = 0$ для некоторого $i \in \alpha_{J'}(x')$ или $x_i = 1$ для некоторого $i \in N_{J'}(x')$. Тогда правая часть неравенства (15) не больше нуля и, следовательно, неравенство справедливо.

Пусть $x_i = 1$ для всех $i \in \alpha_{J'}(x')$ и $x_i = 0$ для $i \in N_{J'}(x')$. Предположим, что неравенство (15) нарушается. Тогда его левая часть равна нулю. Это означает, что для всякого $i \in N_{J'}(x')$ не только $x_i = 0$, но и $z_{it} = 0$ для $t = 1, \ldots, l_0$. В частности, $z_{kt} = 0$, $t = 1, \ldots, l_0$. При заданном векторе $x = (x_i)$ рассмотрим допустимое решение $Z' = ((z'_{it}), (z'_{ij}))$ задачи $\mathcal{F}$, которое отличается от решения $Z = ((z_{it}), (z_{ij}))$ только тем, что $z_{kt_0} = 1$ и $z_{kj} = 1$ для $j \in J'$. Для решений Z' и Z в силу условия (14) справедливо соотношение

$$F(Z') - F(Z) \geqslant \sum_{j \in J'} q_{kj} - g_{kt_0} \geqslant 0.$$

Если $F(Z') > F(Z)$, то решение Z не есть оптимальное решение задачи $\mathcal{F}$ и, следовательно, (X, Z) не является допустимым решением задачи $(\mathcal{L}, \mathcal{F})$. Если $F(Z') = F(Z)$, то Z' — оптимальное решение задачи $\mathcal{F}$. Тогда (X', Z') есть допустимое решение задачи $(\mathcal{L}, \mathcal{F})$, порождённое вектором $x = (x_i)$. Это решение отличается от исходного допустимого решения (X, Z) тем, что $x'_{ij} = 0$, $i \in I$, когда $j \in J'$. Тем самым для решений (X, Z) и (X', Z') с учётом условий (13) получаем

$$L(X, Z) - L(X', Z') \geqslant \sum_{j \in J'} \sum_{i \in I} p_{ij} x_{ij} > 0.$$

Это означает, что решение (X, Z) не есть пессимистическое допустимое решение. Таким образом, в обоих случаях приходим к противоречию с тем, что (X, Z) — пессимистическое допустимое решение.

2.3. d-Отсечения. Рассмотренные c-отсечения (15) стимулируют величины u_{it} принимать единичные значения. Если для оптимального решения задачи SEP правая часть неравенства (15) равна единице, то величина u_{il_0} для некоторого $i \in I$ также равна единице. Однако, поскольку целевая функция задачи SEP максимизируется по переменным z_{it}, единичное значение получает переменная z_{il_0}. Такой выбор единичной компоненты ослабляет верхнюю границу. Ниже рассматриваются дополнительные ограничения (d-отсечения), принуждающие в случае, когда в оптимальном решении $u_{il} = 1$, принимать единичное значение переменную z_{it} для $t < l$.

Пусть (X', z') — допустимое решение задачи SEP для задачи $(\mathcal{L}, \mathcal{F})$, в котором $z_{kl} = 1$, $l \in T$, $l > 1$. Пусть $u'_{it} = \sum\limits_{\tau=1}^{t} z'_{i\tau}$, $i \in I$, $t \in T$, и $x' = (x'_i)$, $u'_t = (u'_{it})$, $i \in I$, $t \in T$. Для всякого $t \in T$ положим $J_{0t} = J_t$, если u'_t — нулевой вектор, и $J_{0t} = \{j \in J_t \mid \alpha_j(x') \succ_j \alpha_j(u'_t)\}$ в противном случае. Исходными данными для построения d-отсечения являются номер k и элемент $t_0 \in T$, $t_0 < l$, если $k \in N_{J_{0t_0}}(x')$. Если элемента t_0 с указанными свойствами не существует, то для данного k d-отсечение не строится. Для заданного k рассмотрим множество $J_{0t}(k) = \{j \in J_{0t} \mid k \succ_j \alpha_j(x')\}$, $t = t_0, \ldots, l-1$, и пусть l_0, $t_0 \leqslant l_0 < l$ — наименьший номер, для которого справедливы неравенства

$$\sum_{t=t_0}^{l_0} \sum_{j \in J_{0t}(k)} p_{\alpha_j(x')j} > 0, \tag{16}$$

$$\sum_{t=t_0}^{l_0} \sum_{j \in J_{0t}(k)} q_{kj} \geqslant g_{kt_0} - g_{kl}. \tag{17}$$

Пусть $J' \subseteq \bigcup\limits_{t=t_0}^{l_0} J_{0t}(k)$ — подмножество, для которого выполняются аналогичные условия

$$\sum_{j \in J'} p_{\alpha_j(x')j} > 0, \quad \sum_{j \in J'} q_{kj} \geqslant g_{kt_0} - g_{kl}. \tag{18}$$

Тогда неравенство

$$\sum_{i \in N_{J'}(x')} u_{il_0} \geqslant 1 - (1 - u_{kl}) + \sum_{i \in \alpha_{J'}(x')} (x_i - 1) - \sum_{i \in N_{J'}(x')} x_i \tag{19}$$

назовём *d-отсечением* решения (X', z'), сгенерированного номерами $t_0 \in T$ и $k \in N_{J_{0t_0}}(x')$.

По построению это неравенство нарушается на решении (X', z'), и для него, как и в случае c-отсечения, справедливо

Утверждение 2. *Пусть* (X, Z), $X = ((x_i), (x_{ij}))$, $Z = ((z_{it}), (z_{ij}))$, *есть пессимистическое допустимое решение задачи* $(\mathcal{L}, \mathcal{F})$. *Тогда неравенство* (19) *выполняется на этом решении.*

Доказательство. Действительно, пусть $x_i = 1$ для $i \in \alpha_{J'}(x')$, $x_i = 0$ для $i \in N_{J'}(x')$ и $u_{kl} = 1$. Если $z_{kt} = 1$ для некоторого $t \leqslant l_0$, то неравенство выполняется. Пусть $z_{kl'} = 1$, где $l_0 < l' \leqslant l$. Предположим, что неравенство нарушается и $z_{it} = 0$, $t = 1, \ldots, l_0$, для всякого

$i \in N_{J'}(x')$. В частности, $z_{kt_0} = 0$. При заданном векторе $x = (x_i)$ рассмотрим допустимое решение $Z' = ((z'_{it}), (z'_{ij}))$ задачи $\mathcal{F}$, которое отличается от решения $Z = ((z_{it}), (z_{ij}))$ тем, что $z'_{kt_0} = 1$, $z'_{kl'} = 0$ и $z'_{kj} = 1$ для $j \in J'$. Для решений Z и Z' в силу условия (18) и убывания величин g_{kt} с ростом t справедливы соотношения

$$F(Z') - F(Z) \geqslant \sum_{j \in J'} q_{kj} - g_{kt_0} + g_{kl'} \geqslant \sum_{j \in J'} q_{kj} - g_{kt_0} + g_{kl} \geqslant 0.$$

Отсюда, повторяя рассуждения из доказательства утверждения 1, приходим к противоречию с тем, что (X, Z) — пессимистическое допустимое решение задачи $(\mathcal{L}, \mathcal{F})$.

Завершая описание процедуры построения дополнительных ограничений, отметим, что при заданных номерах $t_0 \in T$ и $k \in N_{J_{0t_0}}(x')$ выполнение неравенств (11) и (12) можно рассматривать как необходимые и достаточные условия существования подмножества J', для которого верны неравенства (13) и (14), а следовательно, как условие существования c-отсечения, сгенерированного номерами t_0 и k. Насколько сильным будет это отсечение, зависит от выбора множества J', которое определяет число элементов в множествах $\alpha_{J'}(x')$ и $N_{J'}(x')$. Для построения множества J' с наименьшим числом элементов в этих множествах можно сформулировать вспомогательную задачу ЦЛП, оптимальное решение которой будет генерировать наилучшее множество J'. Аналогично при заданных элементах $t_0 \in T$ и $k \in N_{J_{0t_0}}(x')$ выполнение условий (16) и (17) можно рассматривать как критерий существования d-отсечения, сгенерированного элементами t_0 и k. Качество этого отсечения определяется выбором множества J'. Для построения наилучшего множества J', как и в случае c-отсечения, может быть использована вспомогательная оптимизационная задача.

Заключение

В работе предложена постановка динамической задачи конкурентного размещения. Особенность рассмотренной модели определяется условием для Лидера принимать решение об открытии своих предприятий на первом промежутке рассматриваемого горизонта планирования. Для этой модели предложены ограничения для усиления её HP-релаксации.

На следующих этапах исследования динамических задач конкурентного размещения полученные результаты будут использованы при построении процедур вычисления верхних границ значений целевой функции рассматриваемой задачи и алгоритма ветвей, границ и отсечений для поиска её пессимистического оптимального решения. Кроме того, алгоритм поиска решения предложенной задачи будет использован как

процедура в алгоритмах решения более общих динамических задач конкурентного размещения.

ЛИТЕРАТУРА

1. **Aras N., Küçükaydın H.** Bilevel models on the competitive facility location problem // Spatial interaction models: Facility location using game theory. Cham: Springer, 2017. P. 1–19.
2. **Ashtiani M. G., Makui A., Ramezanian R.** A robust model for a leader-follower competitive facility location problem in a discrete space // Appl. Math. Model. 2013. V. 37, No. 1–2. P. 62–71.
3. **Drezner T.** A review of competitive facility location in the plane // Logist. Res. 2014. V. 7, No. 1. Paper ID 114. 12 p.
4. **Karakitsiou A.** Modeling discrete competitive facility location. Cham: Springer, 2015. 54 p.
5. **Kress D., Pesch E.** Sequential competitive location on networks // Eur. J. Oper. Res. 2012. V. 217, No. 3. P. 483–499.
6. **Mishra M., Singh S. P., Gupta M. P.** Location of competitive facilities: A comprehensive review and future research agenda // Benchmarking. 2022. V. 30, No. 4. P. 1171–1230.
7. **Current J., Ratick S., ReVelle C.** Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach // Eur. J. Oper. Res. 1998. V. 110, No. 3. P. 597–609.
8. **Eiselt H. A., Marianov V., Drezner T.** Competitive location models // Location science. Cham: Springer, 2019. P. 391–429.
9. **Castro J., Nasini S., Saldanha-da-Gama F.** A cutting-plane approach for large-scale capacitated multi-period facility location using a specialized interior-point method // Math. Program. 2017. V. 163, No. 1. P. 411–444.
10. **Береснев В. Л.** О задаче конкурентного размещения предприятий со свободным выбором поставщиков // Автоматика и телемеханика. 2014. № 4. С. 94–105.
11. **Beresnev V. L., Melnikov A. A.** Approximation of the competitive facility location problem with MIPs // Comput. Oper. Res. 2019. V. 104. P. 139–148.
12. **Santos-Peñate D. R., Suárez-Vega R., Dorta-González P.** The leader-follower location model // Netw. Spat. Econ. 2007. V. 7, No. 1. P. 45–61.
13. **Dempe S.** Bilevel optimization: Theory, algorithms, applications and a bibliography // Bilevel optimization: Advances and next challenges. Cham: Springer, 2020. P. 581–672.

Береснев Владимир Леонидович
Мельников Андрей Андреевич

Статья поступила
17 мая 2023 г.
После доработки —
25 мая 2023 г.
Принята к публикации
29 мая 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
July–September 2023. Vol. 30, No. 3. P. 43–56

UDC 519.8+518.25 DOI 10.33048/daio.2023.30.774

ADDITIONAL CONSTRAINTS FOR DYNAMIC COMPETITIVE FACILITY LOCATION PROBLEM

V. L. Beresnev[1, 2, a] and *A. A. Melnikov*[1, 2, b]

[1] Sobolev Institute of Mathematics,
4 Koptyug Avenue, 630090 Novosibirsk, Russia
[2] Novosibirsk State University,
2 Pirogov Street, 630090 Novosibirsk, Russia
E-mail: [a] beresnev@math.nsc.ru, [b] melnikov@math.nsc.ru

Abstract. We consider a competitive facility location model, where competing parties (Leader and Follower) make decisions considering changes of the set of customers happening during the planing horizon, having known number of time periods. It is supposed that the Leader makes a decision on opening their facilities at the beginning of the planning horizon, while the Follower can revise their decision in each time period. In the present paper, we study perspectives to apply a method for finding the best solution which is based on using HP-relaxation of the bi-level problem considered. The key element of this method is construction of additional inequalities strengthening the HP-relaxation and computation of upper bounds for the objective function of the problem. In the work, new families of additional constraints are proposed to strengthen the HP-relaxation, that allow us to compute non-trivial upper bounds. Bibliogr. 13.

Keywords: Stackelberg game, bi-level programming, competitive location, valid inequalities.

REFERENCES

1. **N. Aras** and **H. Küçükaydın,** Bilevel models on the competitive facility location problem, in *Spatial Interaction Models: Facility Location Using Game Theory* (Springer, Cham, 2017), pp. 1–19.
2. **M. G. Ashtiani, A. Makui,** and **R. Ramezanian,** A robust model for a leader-follower competitive facility location problem in a discrete space, *Appl. Math. Model.* **37** (1–2), pp. 62–71 (2013).

This research is supported by the Russian Science Foundation (Project 23–21–00082).

English version: Journal of Applied and Industrial Mathematics **17** (3) (2023).

3. **T. Drezner,** A review of competitive facility location in the plane, *Logist. Res.* **7** (1), ID 114, 12 p. (2014).
4. **A. Karakitsiou,** *Modeling Discrete Competitive Facility Location* (Springer, Cham, 2015).
5. **D. Kress** and **E. Pesch,** Sequential competitive location on networks, *Eur. J. Oper. Res.* **217** (3), 483–499 (2012).
6. **M. Mishra, S. P. Singh,** and **M. P. Gupta,** Location of competitive facilities: A comprehensive review and future research agenda, *Benchmarking* **30** (4), 1171-1230 (2022).
7. **J. Current, S. Ratick,** and **C. ReVelle,** Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach, *Eur. J. Oper. Res.* **110** (3), 597–609 (1998).
8. **H. A. Eiselt, V. Marianov,** and **T. Drezner,** Competitive location models, in *Location Science* (Springer, Cham, 2019), pp. 391–429.
9. **J. Castro, S. Nasini,** and **F. Saldanha-da-Gama,** A cutting-plane approach for large-scale capacitated multi-period facility location using a specialized interior-point method, *Math. Program.* **163** (1), 411–444 (2017).
10. **V. L. Beresnev,** On the competitive facility location problem with a free choice of suppliers, *Avtom. Telemekh.,* No. 4, 94–105 (2014) [Russian] [*Autom. Remote Control* **75** (4), 668–676 (1997)].
11. **V. L. Beresnev** and **A. A. Melnikov,** Approximation of the competitive facility location problem with MIPs, *Comput. Oper. Res.* **104**, 139–148 (2019).
12. **D. R. Santos-Peñate, R. Suárez-Vega,** and **P. Dorta-González,** The leader follower location model, *Netw. Spat. Econ.* **7** (1), 45–61 (2007).
13. **S. Dempe,** Bilevel optimization: Theory, algorithms, applications and a bibliography, in *Bilevel Optimization: Advances and Next Challenges* (Springer, Cham, 2020), pp. 581–672.

Vladimir L. Beresnev
Andrey A. Melnikov

Received May 17, 2023
Revised May 25, 2023
Accepted May 29, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль–сентябрь 2023. Т. 30, № 3. С. 57–80

УДК 519.7 DOI 10.33048/daio.2023.30.764

О НИЖНЕЙ ОЦЕНКЕ ЧИСЛА БЕНТ-ФУНКЦИЙ НА МИНИМАЛЬНОМ РАССТОЯНИИ ОТ БЕНТ-ФУНКЦИИ ИЗ КЛАССА МЭЙОРАНА — МАКФАРЛАНДА

Д. А. Быков[1, a], *Н. А. Коломеец*[2, b]

[1] Новосибирский гос. университет,
ул. Пирогова, 2, 630090 Новосибирск, Россия

[2] Институт математики им. С. Л. Соболева,
пр. Акад. Коптюга, 4, 630090 Новосибирск, Россия

E-mail: [a] den.bykov.2000i@gmail.com, [b] kolomeec@math.nsc.ru

Аннотация. Рассматриваются бент-функции, находящиеся на минимальном расстоянии 2^n от функции из класса Мэйорана — МакФарланда $\mathcal{M}_{2n}$, содержащего бент-функции от $2n$ переменных. Для функции, полученной из бент-функции класса $\mathcal{M}_{2n}$ прибавлением индикатора аффинного подпространства размерности n, доказан критерий того, что она также является бент-функцией. Другими словами, охарактеризованы все бент-функции на минимальном расстоянии от функции из $\mathcal{M}_{2n}$. Показано, что не достигается нижняя оценка $2^{2n+1} - 2^n$ на число бент-функций на минимальном расстоянии от функции из $\mathcal{M}_{2n}$, если перестановка, по которой построена исходная бент-функция, не является APN-функцией. Доказано, что при простых $n \geqslant 5$ существуют функции из $\mathcal{M}_{2n}$, для которых данная нижняя оценка точна, приведены примеры таких бент-функций. Также установлено, что перестановки EA-эквивалентных функций из $\mathcal{M}_{2n}$ аффинно эквивалентны, если вторые производные хотя бы одной из перестановок не тождественно нулевые. Библиогр. 31.

Ключевые слова: бент-функция, булева функция, минимальное расстояние, класс Мэйорана — МакФарланда, нижняя оценка, аффинная эквивалентность.

Исследование выполнено в рамках государственного задания ИМ СО РАН (проект № FWNF–2022–0018).

Введение

Бент-функции — это булевы функции от чётного числа переменных, обладающие максимальной нелинейностью. Они были введены О. Ротхаусом [1], однако также изучались в СССР в 1960-х гг. В. А. Елисеевым и О. П. Степченковым (см. [2]). Эти функции интересны своими приложениями в криптографии, теории кодирования, теории символьных последовательностей. Например, нелинейность является одним из важнейших криптографических свойств булевых функций: её низкое значение влечёт уязвимость к линейному криптоанализу. Бент-функции имеют максимальное значение нелинейности, однако не обладают некоторыми другими важными криптографическими свойствами. Тем не менее они используются в S-блоках шифра CAST [3]. Одним из вариантов построения функции с высокой нелинейностью является модификация имеющейся бент-функции, также существуют различные обобщения этих функций (см. [4]). Нелинейности и связанным с ней свойствам посвящены олимпиадные задачи (см., например, NSUCRYPTO [5, 6]).

Подробную информацию о бент-функциях можно найти в [2, 4, 7–11]. Несмотря на обширные исследования, в этой области до сих пор остаётся множество открытых вопросов. Неизвестно точное число бент-функций, более того, нет его хороших оценок (см. [12–14]). Существуют различные конструкции, но не ясно, какую часть от всех бент-функций они покрывают. Также непонятно, как соотносятся различные известные классы бент-функций. Например, классы $\mathcal{C}$ и $\mathcal{D}$, которые были определены в работе [15], построены из бент-функций класса Мэйорана — МакФарланда $\mathcal{M}_{2n}$ прибавлением индикаторов определённых аффинных подпространств. Благодаря построенным примерам известно, что оба этих класса не лежат в замыкании $\mathcal{M}_{2n}$ относительно EA-эквивалентности (см., например, [15, 16]). Однако что-то более точное об их пересечении сказать сложно.

В данной работе изучаются метрические свойства класса бент-функций. Главным образом рассматривается достижимость нижней оценки $2^{2n+1}-2^n$ числа бент-функций, лежащих на минимальном расстоянии 2^n от функции из $\mathcal{M}_{2n}$. Такие бент-функции изучались в работах [17–20]. В [17] доказан критерий расположения бент-функций на данном расстоянии. В [20] получен явный вид всех функций из $\mathcal{M}_{2n}$, ближайших к некоторой исходной функции из $\mathcal{M}_{2n}$, а также дана указанная нижняя оценка числа бент-функций. В случае достижимости этой оценки для некоторой функции из $\mathcal{M}_{2n}$ мы знаем точное число ближайших к ней бент-функций, более того, все они также будут из $\mathcal{M}_{2n}$. В противном случае на минимальном расстоянии от этой бент-функции будет лежать хотя бы одна функция не из $\mathcal{M}_{2n}$. Известно немного бент-функций, для которых

найдено число бент-функций, лежащих на минимально возможном расстоянии 2^n. В [18] установлено, что для квадратичной бент-функции это число равно $2^n(2^1+1)\dots(2^n+1)$, при этом в [19] показано, что это значение является верхней оценкой числа бент-функций на расстоянии 2^n от исходной, достигающейся только для квадратичных бент-функций. От любой не слабо нормальной бент-функции [21, 22] не лежит ни одной бент-функции на расстоянии 2^n. Отметим, что метрические свойства изучались также для класса самодуальных бент-функций (см. [23]).

Опишем структуру работы. В разд. 1 и 2 приведены базовые определения и используемые утверждения о бент-функциях на минимальном расстоянии от исходной. Условия, при которых функция на минимальном расстоянии от функции из $\mathcal{M}_{2n}$ также будет бент-функцией, рассмотрены в разд. 3, где доказан соответствующий критерий, который является основой для последующих разделов. В разд. 4 показано, что оценка не достигается, если перестановка в исходной бент-функции из $\mathcal{M}_{2n}$ не APN-функция. Достижимость оценки при простых $n \geqslant 5$ показана в разд. 5, при этом приведён набор бент-функций, для которых оценка достигается, и оценено число таких функций. В разд. 6 доказано, что если вторые производные перестановки в функции из $\mathcal{M}_{2n}$ не тождественно нулевые, то любая EA-эквивалентная ей функция из $\mathcal{M}_{2n}$ имеет аффинно эквивалентную перестановку.

1. Основные определения

Рассмотрим векторное пространство $\mathbb{F}_2^n$ размерности n над полем $\mathbb{F}_2$ из двух элементов. Для *двоичных векторов* $x, y \in \mathbb{F}_2^n$ определены $x \oplus y = (x_1 \oplus y_1, x_2 \oplus y_2, \dots, x_n \oplus y_n)$ и $\langle x, y\rangle = x_1y_1 \oplus x_2y_2 \oplus \dots \oplus x_ny_n$. Напомним, что непустое множество $E \subseteq \mathbb{F}_2^n$ называется *линейным подпространством* $\mathbb{F}_2^n$, если для любых $a, b \in E$ выполнено $a \oplus b \in E$; линейное подпространство $E^\perp = \{x \in \mathbb{F}_2^n \mid \langle x, y\rangle = 0 \text{ для всех } y \in E\}$ называется *ортогональным* к E; множество $U = a \oplus E = \{a \oplus x \mid x \in E\}$, $a \in \mathbb{F}_2^n$, называется *аффинным подпространством* $\mathbb{F}_2^n$, размерность U определяется равенством $\dim U = \dim E$.

Пусть $\mathcal{F}_n$ — множество всех *булевых функций* от n переменных, т. е. функций вида $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$. Характеристическую булеву функцию множества $S \subseteq \mathbb{F}_2^n$ обозначим через *индикатор* Ind_S. Любая функция $f \in \mathcal{F}_n$ может быть единственным образом представлена в виде *полинома Жегалкина* (*алгебраической нормальной формы,* АНФ):

$$f(x_1, x_2, \dots, x_n) = \bigoplus_{a \in \mathbb{F}_2^n} g_a x_1^{a_1} x_2^{a_2} \dots x_n^{a_n},$$

где $g_a \in \mathbb{F}_2$, $x_i^{a_i}$ — обычное возведение в степень. *Степенью* $\deg f$ булевой функции f называется степень её полинома Жегалкина. Функции

степени не больше 1, т. е. представимые в виде $f(x) = \langle a, x\rangle \oplus c$, где $a \in \mathbb{F}_2^n$, $c \in \mathbb{F}_2$, называются *аффинными*. Функции степени 2 называются *квадратичными*.

Расстояние $\operatorname{dist}(f, g)$ между функциями $f, g \in \mathcal{F}_n$ — число аргументов, на которых их значения различаются. *Нелинейностью* f называется расстояние от f до ближайшей к ней аффинной функции от n переменных. Булева функция от чётного числа переменных $2n$, нелинейность которой максимальна и равна $2^{2n-1} - 2^{n-1}$, называется *бент-функцией*. Через $\mathcal{B}_{2n}$ обозначим множество всех бент-функций от $2n$ переменных.

Функции вида

$$f(x, y) = \langle x, \pi(y)\rangle \oplus \varphi(y),$$

где $\varphi(y) \in \mathcal{F}_n$ и $\pi\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ взаимно однозначная, являются бент-функциями и образуют *класс Мэйорана — МакФарланда* $\mathcal{M}_{2n}$.

Функция $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ называется *векторной булевой функцией;* можно полагать, что это набор из m булевых функций. Например, F называется *аффинной*, если она представима в виде $F(x) = xA \oplus b$, где A — двоичная (т. е. над полем $\mathbb{F}_2$) матрица размера $m \times n$, $b \in \mathbb{F}_2^m$. Функция $\mathrm{D}_a F(x) = F(x) \oplus F(x \oplus a)$ называется *производной* функции F по направлению $a \in \mathbb{F}_2^n$. Множество $\{F(x) \mid x \in S\}$, где $S \subseteq \mathbb{F}_2^n$, будем обозначать через $F(S)$.

Сужением функции $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ на аффинное подпространство $U \subseteq \mathbb{F}_2^n$ называется функция $F|_U\colon U \to \mathbb{F}_2^m$ такая, что $F|_U(y) = F(y)$ для всех $y \in U$. Будем использовать сужения в том числе и для булевых функций, т. е. при $m = 1$. $F|_U$ *аффинна*, если $F|_U = G|_U$ для некоторой аффинной функции $G\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$, в этом случае будем также говорить, что F *аффинна* на U. Если $a \in U$ и M — двоичная матрица размера $\dim U \times n$, строки которой образуют базис $a \oplus U$, то аффинность $F|_U$ означает обычную аффинность функции $F'(u) = F(uM \oplus a)$, $u \in \mathbb{F}_2^{\dim U}$.

2. Бент-функции на минимально возможном расстоянии от исходной бент-функции

Любую булеву функцию на расстоянии $|S|$ от исходной $f \in \mathcal{F}_m$ можно представить следующей конструкцией:

$$f \mapsto f \oplus \mathrm{Ind}_S, \tag{1}$$

т. е. значения f изменяются ровно в $|S|$ местах, определяемых множеством $S \subseteq \mathbb{F}_2^m$.

Рассмотрим произвольную бент-функцию $f \in \mathcal{B}_{2n}$ и минимально возможное расстояние 2^n. Тогда конструкция (1) позволяет в удобном виде описать все бент-функции на этом расстоянии от f (см. [17]).

Теорема 1 [17]. *Пусть $f \in \mathcal{B}_{2n}$, $U \subseteq \mathbb{F}_2^{2n}$, при этом $|U| = 2^n$. Тогда $f \oplus \mathrm{Ind}_U \in \mathcal{B}_{2n}$ в том и только том случае, когда U — аффинное подпространство $\mathbb{F}_2^{2n}$ и f аффинна на U.*

Таким образом, все бент-функции на расстоянии 2^n от f получаются прибавлением индикатора некоторого аффинного подпространства $\mathbb{F}_2^{2n}$ размерности n. Отметим, что в [15] можно найти критерий принадлежности функции $f \oplus \mathrm{Ind}_S$ к классу бент-функций в случае, если S — произвольное аффинное подпространство $\mathbb{F}_2^{2n}$. С его помощью, например, были построены классы $\mathcal{C}$ и $\mathcal{D}$ бент-функций при использовании в качестве исходных бент-функций подмножества класса $\mathcal{M}_{2n}$. Некоторые свойства данной конструкции см. также в [24].

Далее в работе будем рассматривать бент-функции на расстоянии 2^n от бент-функций из $\mathcal{M}_{2n}$. Для любой бент-функции из $\mathcal{M}_{2n}$ известны все бент-функции из того же класса $\mathcal{M}_{2n}$, лежащие на минимально возможном расстоянии от неё (см. [20]).

Теорема 2 [20]. *Пусть $f, g \in \mathcal{M}_{2n}$, причём $f(x, y) = \langle x, \pi(y)\rangle \oplus \varphi(y)$, $g(x, y) = \langle x, \pi'(y)\rangle \oplus \varphi'(y)$. Тогда $\mathrm{dist}(f, g) = 2^n$, если и только если выполняется одно из двух следующих условий:*

(1) $\pi = \pi'$ и $\mathrm{dist}(\varphi, \varphi') = 1$,

(2) существуют $a, b \in \mathbb{F}_2^n$, $a \neq b$, такие, что $\pi(a) = \pi'(b)$ и $\pi(b) = \pi'(a)$, при этом $\pi(y) = \pi'(y)$ и $\varphi(y) = \varphi'(y)$ для всех $y \in \mathbb{F}_2^n \backslash \{a, b\}$.

Поскольку для любой f из $\mathcal{M}_{2n}$ существует бент-функция (причём из класса $\mathcal{M}_{2n}$), находящаяся на расстоянии 2^n, можем называть рассматриваемые бент-функции *бент-функциями на минимальном расстоянии от f* или *ближайшими к f бент-функциями*. Теорема 2 даёт нижнюю оценку числа таких бент-функций.

Утверждение 1 [20]. *Имеется не менее $2^{2n+1} - 2^n$ бент-функций на минимальном расстоянии от функции из $\mathcal{M}_{2n}$. Все учтённые здесь бент-функции также лежат в классе $\mathcal{M}_{2n}$.*

Оценку $2^{2n+1} - 2^n$ также можно найти в [18]. В последующих разделах мы докажем ряд утверждений, связанных с её достижимостью. В частности, покажем, что она достигается при простых $n \geqslant 5$.

3. Критерий расположения бент-функции на минимальном расстоянии от бент-функции из $\mathcal{M}_{2n}$

Здесь мы уточним теорему 1 для случая $f \in \mathcal{M}_{2n}$. Другими словами, опишем все аффинные подпространства $\mathbb{F}_2^{2n}$, на которых аффинна произвольная бент-функция из класса $\mathcal{M}_{2n}$. Будем использовать для этого базисные матрицы специального вида.

3.1. GJB-матрицы. Пусть M — двоичная матрица размера $k \times n$. Обозначим через $\langle M \rangle$ линейное подпространство $\mathbb{F}_2^n$, состоящее из всех линейных комбинаций строк M.

Будем называть M *базисной матрицей Гаусса — Жордана* (GJB-*матрицей*), если она является приведённой ступенчатой матрицей ранга k. *Ведущим элементом строки* называется первая слева единица в строке матрицы M, а столбец его содержащий — *ведущим столбцом*. Таким образом, ведущий элемент каждой следующей строки находится правее предыдущего, а каждый ведущий столбец содержит ровно одну единицу. Например, матрица

$$M = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

является GJB-матрицей размера 3×8, ведущими столбцами которой являются столбцы под номерами 2, 5 и 6.

Подход к перечислению подпространств с использованием GJB-матриц использовался, например, в работах [18, 21]. Он удобен тем, что любое линейное подпространство $\mathbb{F}_2^n$ имеет единственную GJB-матрицу, т. е. GJB-матрица M однозначно соответствует $\langle M \rangle$.

Любое аффинное подпространство $U = a \oplus \langle M \rangle$, $a \in \mathbb{F}_2^n$, также можно задать единственным образом, наложив на вектор a следующее ограничение: все его элементы, соответствующие ведущим столбцам M, должны быть нулевыми.

3.2. Уточнение критерия. В теореме ниже приведён критерий расположения бент-функций на минимальном расстоянии от бент-функции $\langle y, \pi(x) \rangle \oplus \varphi(x)$. Критерий для бент-функций непосредственно из $\mathcal{M}_{2n}$ является его простым следствием: для этого достаточно поменять местами x и y. Такой переход нужен для удобного представления подпространств с помощью GJB-матриц.

Теорема 3. *Пусть функция* $f(x, y) = \langle y, \pi(x) \rangle \oplus \varphi(x)$ *такая, что* $f'(x, y) = f(y, x) \in \mathcal{M}_{2n}$, $U = (a, b) \oplus \langle M \rangle$ *для* $a, b \in \mathbb{F}_2^n$ *и двоичной матрицы* $M = \begin{pmatrix} L & T \\ 0 & R \end{pmatrix}$, *где* L *и* R — *полноранговые матрицы размеров* $k \times n$ *и* $(n - k) \times n$ *соответственно,* $k \in \{0, \ldots, n\}$, T — *произвольная матрица подходящих размеров. Тогда функция*

$$g(x, y) = \langle y, \pi(x) \rangle \oplus \varphi(x) \oplus \mathrm{Ind}_U(x, y)$$

является бент-функцией в том и только том случае, если выполнены условия

(1) $\pi(a \oplus \langle L \rangle) = \pi(a) \oplus \langle R \rangle^{\perp}$,

(2) $\langle uT \oplus b,\, \pi(uL \oplus a)\rangle \oplus \varphi(uL \oplus a)$ — *аффинная функция от* k *переменных* $u = (u_1, \ldots, u_k) \in \mathbb{F}_2^k$.

Доказательство. Согласно теореме 1 требуется аффинность функции f на U. В аффинном подпространстве U можно перейти к новым переменным $u = (u_1, u_2, \ldots, u_k)$, $v = (v_1, v_2, \ldots, v_{n-k})$. Старые переменные выражаются через них так:

$$(x, y) = (u, v)\begin{pmatrix} L & T \\ 0 & R \end{pmatrix} \oplus (a, b), \quad \text{т. е.} \quad \begin{cases} x = uL \oplus a, \\ y = uT \oplus vR \oplus b. \end{cases}$$

Перепишем сужение f на U в новых переменных:

$$\begin{aligned} f|_U(x, y) &= \langle uT \oplus vR \oplus b,\, \pi(uL \oplus a)\rangle \oplus \varphi(uL \oplus a) \\ &= \underbrace{\langle vR,\, \pi(uL \oplus a)\rangle}_{\text{зависит от } u \text{ и } v} \oplus \underbrace{\langle uT \oplus b,\, \pi(uL \oplus a\rangle \oplus \varphi(uL \oplus a)}_{\text{зависит от } u} \\ &= f_1(u, v) \oplus f_2(u). \end{aligned}$$

Любой член в АНФ функции $f_1(u, v)$ содержит некоторый v_i, а члены АНФ функции $f_2(u)$ состоят лишь из u_i. Значит, нелинейные члены $f_2(u)$ не могут быть занулены с помощью $f_1(u, v)$, а нелинейные члены $f_1(u, v)$ — с помощью $f_2(u)$. Таким образом, f аффинна на U, если и только если f_1 и f_2 — аффинные функции.

Перепишем функцию f_1:

$$f_1(u, v) = \langle vR,\, \pi(uL \oplus a)\rangle = \langle v,\, \pi(uL \oplus a)R^T\rangle.$$

Она аффинна тогда и только тогда, когда функция $t(u) = \pi(uL \oplus a)R^T$ тождественно равна константе. Рассмотрим уравнение для $z \in \mathbb{F}_2^n$:

$$zR^T = \pi(a)R^T.$$

Обозначим пространство его решений через $\pi(a) \oplus Z$. Далее заметим, что $\dim Z = n - \operatorname{rk} R^T = k$. В свою очередь,

$$\big|\{\pi(uL \oplus a) \mid u \in \mathbb{F}_2^k\}\big| = |\pi(a \oplus \langle L\rangle)| = 2^k.$$

Значит, $t \equiv \pi(a)R^T$, если и только если $\pi(a \oplus \langle L\rangle) = \pi(a) \oplus Z$.

Поскольку пространством решений $zR^T = \pi(a)R^T$ является $\pi(a) \oplus Z$, имеем $eR^T = 0$ для всех $e \in Z$ и только для них. Получаем

$$\begin{aligned} Z = \big\{e \in \mathbb{F}_2^n \mid eR^T = 0\big\} &= \big\{e \in \mathbb{F}_2^n \mid \langle eR^T,\, p\rangle = 0 \text{ для всех } p \in \mathbb{F}_2^{n-k}\big\} \\ &= \big\{e \in \mathbb{F}_2^n \mid \langle e,\, pR\rangle = 0 \text{ для всех } p \in \mathbb{F}_2^{n-k}\big\} \\ &= \big\{e \in \mathbb{F}_2^n \mid \langle e,\, t\rangle = 0 \text{ для всех } t \in \langle R\rangle\big\} = \langle R\rangle^{\perp}. \end{aligned}$$

Таким образом, f_1 аффинна тогда и только тогда, когда $\pi(a \oplus \langle L\rangle) = \pi(a) \oplus \langle R\rangle^{\perp}$. Теорема 3 доказана.

Использование критерия в приведённом виде не очень удобно для подсчёта числа подпространств, порождающих бент-функции. Однако ситуация меняется, если использовать GJB-матрицы для подпространств.

Замечание 1. Пусть U определено в теореме 3 и M — GJB-матрица для $\langle M\rangle$. Это равносильно тому, что L и R — GJB-матрицы произвольных линейных подпространств $\mathbb{F}_2^n$ размерностей k и $n-k$ соответственно, а T — матрица размера $k\times n$ с единственным ограничением: её столбцы, являющиеся ведущими для R, нулевые, т. е. оставшиеся k^2 её элементов произвольны. Добавим также ограничение на (a,b): координаты a и b, соответствующие ведущим столбцам L и R соответственно, должны быть нулевыми. Тогда наборы a,b,L,R и T однозначно сопоставляются всем аффинным подпространствам $U\subseteq\mathbb{F}_2^n$.

Более того, если такие a,b,L,R и T удовлетворяют условию теоремы 3, то R определено однозначно по π, a и L: это GJB-матрица для $\big(\pi(a)\oplus\pi(a\oplus\langle L\rangle)\big)^{\perp}$. Таким образом, для перебора всех U, порождающих бент-функции, достаточно перебрать все a, b, L и T, которые удовлетворяют условию теоремы (первое условие можно заменить тем, что $\pi(a\oplus\langle L\rangle)$ является аффинным подпространством $\mathbb{F}_2^n$) и приведённым выше ограничениям.

Следующее утверждение получается из теоремы 3 при $T=0$. В этом случае $U=U_1\times U_2$, где U_1, U_2 — аффинные подпространства $\mathbb{F}_2^n$.

Следствие 1. *Пусть* $f(x,y)=\langle x,\,\pi(y)\rangle\oplus\varphi(y)\in\mathcal{M}_{2n}$, $a,b\in\mathbb{F}_2^n$, $E_1,E_2\subseteq\mathbb{F}_2^n$ *— линейные подпространства такие, что* $\dim E_1+\dim E_2=n$. *Тогда*

$$g(x,y)=\langle x,\,\pi(y)\rangle\oplus\varphi(y)\oplus\mathrm{Ind}_{(a\oplus E_1)\times(b\oplus E_2)}(x,y)$$

является бент-функцией тогда и только тогда, когда выполнены условия

(1) $\pi(b\oplus E_2)=\pi(b)\oplus E_1^{\perp}$,

(2) $\langle a,\,\pi(y)\rangle\oplus\varphi(y)$ *аффинна на* $b\oplus E_2$.

Карле в работе [15] определил $\mathcal{D}$ как класс бент-функций вида

$$f(x,y)=\langle x,\,\pi(y)\rangle\oplus\mathrm{Ind}_{E_1\times E_2}(x,y),$$

где $E_1,E_2\subseteq\mathbb{F}_2^n$ — линейные подпространства, $\dim E_1+\dim E_2=n$ и π — перестановка на $\mathbb{F}_2^n$ такая, что $\pi(E_2)=E_1^{\perp}$. Заметим, что следствие 1 обобщает определение класса $\mathcal{D}$, так как даёт критерий для аффинных подпространств и произвольной $\varphi\in\mathcal{F}_n$ вместо линейных подпространств и тождественно нулевой φ.

3.3. Использование критерия для нижней оценки. Применяя обозначения теоремы 3, можно разбить критерий на $n+1$ случаев по параметру k. Далее будем называть их случаями размерности k, поскольку $k=\dim\langle L\rangle$.

Покажем, что случаи размерностей 0 и 1 дают в точности все бент-функции, описанные в теореме 2. Другими словами, эти два случая составляют бент-функции из нижней оценки $2^{2n+1}-2^n$ (см. утверждение 1).

Следствие 2. *Пусть функция* $f(x,y) = \langle y,\, \pi(x)\rangle \oplus \varphi(x)$ *такая, что* $f'(x,y) = f(y,x) \in \mathcal{M}_{2n}$, $a, b \in \mathbb{F}_2^n$ *и матрица* M *представлена как в условии теоремы* 3, *причём* $k \in \{0,1\}$. *Тогда*

$$g(x,y) = \langle y,\, \pi(x)\rangle \oplus \varphi(x) \oplus \mathrm{Ind}_{(a,b)\oplus\langle M\rangle}(x,y)$$

является бент-функцией тогда и только тогда, когда $g'(x,y) = g(y,x)$ *принадлежит классу* $\mathcal{M}_{2n}$. *Более того, существует ровно* $2^{2n+1} - 2^n$ *различных бент-функций* g.

Таким образом, все такие бент-функции описаны в теореме 2, *и все они учтены в нижней оценке из утверждения* 1.

Доказательство. Как и в теореме 3, перейдём к новым переменным $u = (u_1, u_2, \ldots, u_k)$, $v = (v_1, v_2, \ldots, v_{n-k})$:

$$\begin{cases} x = uL \oplus a, \\ y = uT \oplus vR \oplus b. \end{cases}$$

Пусть $\dim\langle L\rangle = 0$. В этом случае $\langle R\rangle = \mathbb{F}_2^n$. Тогда

$$\mathrm{Ind}_U(x,y) = \begin{cases} 1, & \text{если } x = a, \\ 0 & \text{иначе.} \end{cases}$$

Значит, $g(x,y) = \langle y,\, \pi(x)\rangle \oplus \varphi'(x)$, где φ' отличается от φ только на векторе a.

Пусть $\dim\langle L\rangle = 1$. В этом случае $\dim\langle R\rangle = n-1$. Тогда

$$\begin{cases} x = u_1 \cdot l \oplus a, \\ y = vR \oplus u_1 \cdot t \oplus b, \end{cases} \qquad l, t \in \mathbb{F}_2^n.$$

Если $\langle R\rangle^\perp = \{0, r\}$, то $y \in \langle R\rangle$ тогда и только тогда, когда $\langle y,\, r\rangle = 0$. При этом из первого условия теоремы 3 имеем $r = \pi(a\oplus l) + \pi(a)$. Таким образом,

$$\mathrm{Ind}_U(x,y) = \begin{cases} 0, & \text{если } x \notin \{a, a\oplus l\}, \\ \langle x,\, \pi(a\oplus l) + \pi(a)\rangle \oplus c_{u_1} & \text{иначе.} \end{cases}$$

Здесь $c_{u_1} = \langle u_1 \cdot t \oplus b,\, \pi(a\oplus l) + \pi(a)\rangle$. Значит,

$$g(x,y) = \begin{cases} \langle y,\, \pi(x)\rangle \oplus \varphi(x), & \text{если } x \notin \{a, a\oplus l\}, \\ \langle y,\, \pi(a\oplus l)\rangle \oplus \varphi(a) \oplus c_0, & \text{если } x = a, \\ \langle y,\, \pi(a)\rangle \oplus \varphi(a\oplus l) \oplus c_1, & \text{если } x = a\oplus l. \end{cases}$$

Таким образом, случаи $\dim\langle L\rangle \in \{0,1\}$ в точности соответствуют случаям теоремы 2 для $g'(x,y) = g(y,x)$. Следствие 2 доказано.

Также отметим, что в случае размерности n условие теоремы 3 очевидным образом упрощается.

Следствие 3. *Пусть функция $f(x,y) = \langle y,\, \pi(x)\rangle \oplus \varphi(x)$ такая, что $f'(x,y) = f(y,x) \in \mathcal{M}_{2n}$, $b \in \mathbb{F}_2^n$ и $M = (I_n\ T)$ для единичной матрицы I_n и произвольной двоичной матрицы T порядка n. Тогда*

$$g(x,y) = \langle y,\, \pi(x)\rangle \oplus \varphi(x) \oplus \mathrm{Ind}_{(0,b)\oplus\langle M\rangle}(x,y)$$

является бент-функцией в том и только том случае, если аффинна функция $\langle xT \oplus b,\, \pi(x)\rangle \oplus \varphi(x)$.

Согласно следствию 2 для достижимости нижней оценки необходимо, чтобы все случаи, кроме $k \in \{0,1\}$, не порождали подпространств, удовлетворяющих условию теоремы 3. А именно, это случай $k = n$ при $n \geqslant 2$, упрощённое условие для которого приведено в следствии 3, и «общий» случай $2 \leqslant k \leqslant n-1$ при $n > 2$.

4. Бент-функции, для которых нижняя оценка не достигается

Основной результат данного раздела, т. е. случаи, для которых оценка гарантированно не достигается, связан с одним из важнейших классов криптографических векторных булевых функций. Функция $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ называется APN-*функцией*, если для любых $a, b \in \mathbb{F}_2^n$, $a \neq 0$, уравнение $F(x \oplus a) \oplus F(x) = b$ имеет не более двух решений (см. [25]). Имеется множество открытых вопросов, связанных с APN-функциями [26], главный из которых — существование взаимно однозначных APN-функций от чётного числа переменных. Максимальной размерностью, для которой ответ на этот вопрос известен, является $n = 6$ [27]. Наиболее изученными являются квадратичные APN-функции (см., например, [28]). Известен следующий критерий: взаимно однозначная $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ является APN-функцией, если и только если $F(U)$ не является аффинным подпространством $\mathbb{F}_2^n$ для любого аффинного подпространства $U \subseteq \mathbb{F}_2^n$ размерности 2 (см. [9]).

Для начала докажем вспомогательное утверждение, не имеющее отношения к APN-функциям.

Лемма 1. *Пусть $\pi\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ и $\varphi \in \mathcal{F}_n$. Если U и $\pi(U)$ — аффинные подпространства $\mathbb{F}_2^n$ размерности 2, то существует аффинная функция $H\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ такая, что $\langle H(x),\, \pi(x)\rangle \oplus \varphi(x)$ аффинна на U.*

Доказательство. Пусть $a \in U$ и M — матрица размера $2\times n$, строки которой являются базисом $a \oplus U$. Перейдём к функциям от двух переменных $u = (u_1, u_2) \in \mathbb{F}_2^2$, используя $x = uM \oplus a$ для всех $x \in U$. Будем строить аффинную $H|_U$. Ясно, что любое такое сужение можно доопределить до аффинной функции H.

Полином Жегалкина для функции $\varphi'(u) = \varphi(uM \oplus a)$ может содержать только u_1u_2 в качестве неаффинного слагаемого. Если такого монома нет, т. е. если $\varphi|_U$ аффинна, то положим

$$H(uM \oplus a) \equiv (0, \ldots, 0),$$

что влечёт аффинность $\langle H(x),\, \pi(x)\rangle \oplus \varphi(x)$ на U.

Пусть моном u_1u_2 присутствует в полиноме Жегалкина. Рассмотрим

$$\pi'(u) = \pi(uM \oplus a) = (\pi'_1(u), \pi'_2(u), \ldots, \pi'_n(u)).$$

По условию в $\pi(U)$ содержится 4 различных вектора, поэтому найдётся функция $\pi'_k \not\equiv \mathrm{const}$, $k \in \{1, \ldots, n\}$. Значит, полином Жегалкина для π'_k содержит хотя бы один из следующих мономов: u_1, u_2 или u_1u_2. Если имеется u_1u_2, то положим

$$H(uM \oplus a) = (0, \ldots, 0, \underset{k}{1}, 0, \ldots, 0).$$

Иначе, имея u_j, $j \in \{1, 2\}$, положим

$$H(uM \oplus a) = (0, \ldots, 0, \underset{k}{u_i}, 0, \ldots, 0), \quad \text{где } i \in \{1, 2\},\ i \neq j.$$

Таким образом, полином Жегалкина для $\langle H(uM \oplus a),\, \pi(uM \oplus a)\rangle$ тоже содержит моном u_1u_2, что означает аффинность $\langle H(x),\, \pi(x)\rangle \oplus \varphi(x)$ на U. Лемма 1 доказана.

Теперь можем доказать, что для достижимости нижней оценки необходимы взаимно однозначные APN-функции.

Теорема 4. *Пусть $f(x, y) = \langle x,\, \pi(y)\rangle \oplus \varphi(y) \in \mathcal{M}_{2n}$, где π не является* APN*-функцией. Тогда число бент-функций, лежащих на минимальном расстоянии от f, строго больше нижней оценки $2^{2n+1} - 2^n$.*

Доказательство. Согласно следствию 2 (см. также теорему 3), случаи, когда $\langle L\rangle$ имеют размерность 0 и 1, дают в точности $2^{2n+1} - 2^n$ бент-функций из нижней оценки.

Так как π не является APN-функцией, существует аффинное подпространство U размерности 2, которое под действием этой перестановки переходит в аффинное подпространство размерности 2. Для U выполняется п. (1) теоремы 3, не зависящий от выбора базиса (см. также замечание 1). По лемме 1 существует аффинная функция $H(x)$, т. е. $H(x) = xT \oplus a$ для некоторой двоичной матрицы T размера $2 \times n$ и $a \in \mathbb{F}_2^n$, которая обеспечивает выполнение п. (2) теоремы 3. Таким образом, если π не является APN-функцией, то случай размерности 2 даст как минимум одну дополнительную бент-функцию. Теорема 4 доказана.

Отметим, что теорема 4 гарантирует следующее: всегда существует бент-функция за пределами класса $\mathcal{M}_{2n}$, лежащая на минимальном расстоянии от бент-функции из $\mathcal{M}_{2n}$, построенной по перестановке, не являющейся APN-функцией.

5. Достижимость нижней оценки

Как уже было отмечено выше, достижимость нижней оценки для некоторой бент-функции $f(x,y) = \langle x,\, \pi(y)\rangle \oplus \varphi(y) \in \mathcal{M}_{2n}$ означает, что все случаи размерностей $k \in \{2,3,\ldots,n\}$ не дают аффинных подпространств $\mathbb{F}_2^{2n}$, удовлетворяющих условию теоремы 3. Для случаев размерности $2 \leqslant k \leqslant n-1$ этого можно добиться, выбрав π такую, что она не переводит никакие аффинные подпространства размерностей $2,3,\ldots,n-1$ в аффинные подпространства, т. е. п. (1) теоремы 3 не будет выполнен. Примером такой функции при простых n является функция обращения элементов конечного поля, что приводит к необходимости рассматривать булевы функции в алгебраическом представлении.

5.1. Алгебраическое представление булевых функций. Ввиду того, что конечное поле $\mathbb{F}_{2^n}$ порядка 2^n является векторным пространством над полем $\mathbb{F}_2$, выберем в нём некоторый базис. Координаты любого элемента $\mathbb{F}_{2^n}$ в этом базисе образуют вектор из $\mathbb{F}_2^n$, т. е. произвольный элемент из $\mathbb{F}_{2^n}$ можно отождествить с вектором из $\mathbb{F}_2^n$. Таким образом, булевы функции можно рассматривать как функции $f\colon \mathbb{F}_{2^n} \to \mathbb{F}_2$, а векторные булевы функции — как функции $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^m}$. Функция $\mathrm{tr}\colon \mathbb{F}_{2^n} \to \mathbb{F}_2$, определённая как

$$\mathrm{tr}(x) = x^{2^0} + x^{2^1} + \cdots + x^{2^{n-1}},$$

называется *следом*. Заметим, что $\mathrm{tr}(x^2) = \mathrm{tr}(x)$ и $\mathrm{tr}(x+y) = \mathrm{tr}(x) + \mathrm{tr}(y)$ для всех $x, y \in \mathbb{F}_{2^n}$. Произвольная аффинная функция $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$, соответствующая функции вида $xT \oplus a$ в $\mathbb{F}_2^n$, представима единственным образом в виде

$$F(x) = \alpha_0 x^{2^0} + \alpha_1 x^{2^1} + \cdots + \alpha_{n-1} x^{2^{n-1}} + \alpha_n, \quad \alpha_0, \alpha_1, \ldots, \alpha_n \in \mathbb{F}_{2^n}.$$

Циклотомическим классом $\mathcal{C}_{n,s}$ над $\mathbb{F}_2$ по модулю $2^n - 1$ называется множество $\mathcal{C}_{n,s} = \{s \cdot 2^i \bmod (2^n - 1) \mid i = 0,1,\ldots,n-1\}$, где $s \in \{0,1,\ldots,2^n-2\}$. Наименьший из элементов циклотомического класса будем называть его *представителем*. Отметим, что $s \cdot 2^k \bmod (2^n-1)$ соответствует циклическому сдвигу вектора двоичной записи числа s на k разрядов влево. Множество всех ненулевых представителей циклотомических классов обозначим через P_n. Произвольная булева функция f представима в виде

$$f(x) = t_0 + \mathrm{tr}\Big(\sum_{k\in P_n} c_k x^k\Big) + t_{2^n-1}x^{2^n-1}, \quad t_0, t_{2^n-1} \in \mathbb{F}_2,\ c_k \in \mathbb{F}_{2^n} \qquad (2)$$

(см., например, [8, 11]). Для простых n это представление единственно, однако отметим, что при произвольных n потребуются дополнительные ограничения на коэффициенты c_k [11]. Представлением произвольной аффинной булевой функции является $t_0 + \mathrm{tr}(\alpha x)$, где $t_0 \in \mathbb{F}_2$ и $\alpha \in \mathbb{F}_{2^n}$. В представлении над полем бент-функции класса $\mathcal{M}_{2n}$ записываются в виде

$$f(x, y) = \mathrm{tr}(x\pi(y)) + \varphi(y),$$

где $\pi\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ взаимно однозначна и $\varphi\colon \mathbb{F}_{2^n} \to \mathbb{F}_2$ — произвольная функция. В зависимости от выбора базиса данные функции либо полностью соответствуют классическому $\mathcal{M}_{2n}$, либо аффинно эквивалентны этим функциям (см. разд. 6).

Линейными подпространствами поля $\mathbb{F}_{2^n}$ (как векторного пространства над $\mathbb{F}_2$) являются его аддитивные подгруппы. Соответственно *аффинным подпространством* U *поля* $\mathbb{F}_{2^n}$ назовём $U = a + E$, где $a \in \mathbb{F}_{2^n}$ и E — линейное подпространство $\mathbb{F}_{2^n}$. Вне зависимости от выбранного базиса аффинные подпространства $\mathbb{F}_{2^n}$ соответствуют аффинным подпространствам $\mathbb{F}_2^n$.

5.2. Бент-функции, для которых нижняя оценка достижима. В [29] (см. также [30]) приведена теорема 2, описывающая все аффинные подпространства $\mathbb{F}_{p^n}$, которые под действием функции обращения в поле $\mathbb{F}_{p^n}$ переходят также в аффинные подпространства. Далее приведён частный случай этой теоремы для $p = 2$ и простого n.

Теорема 5 [29]**.** *Пусть n простое и $F(x) = x^{2^n-2}$ определена на $\mathbb{F}_{2^n}$. Тогда под действием F в аффинные подпространства переходят все аффинные подпространства $\mathbb{F}_{2^n}$ размерностей $0, 1, n$ и только они.*

Таким образом, при простом n для построения функций из $\mathcal{M}_{2n}$, на которых нижняя оценка достигается, можно взять в качестве перестановки функцию $F(x) = x^{2^n-2}$. Отметим, что F является взаимно однозначной APN-функцией при нечётных n, что согласуется с теоремой 4.

Теорема 6. *Пусть $n \geqslant 5$ простое, $f(x, y) = \mathrm{tr}(xy^{2^n-2}) + \varphi(y) \in \mathcal{M}_{2n}$, $x, y \in \mathbb{F}_{2^n}$, такая, что φ не представима в виде*

$$\theta(y) = c_0 + \mathrm{tr}(\beta_1 y^{2^1-1} + \beta_2 y^{2^2-1} + \cdots + \beta_{n-1} y^{2^{n-1}-1}) + c_{2^n-1} y^{2^n-1} \in \mathcal{F}_n,$$

где $y \in \mathbb{F}_{2^n}$, $c_0, c_{2^n-1} \in \mathbb{F}_2$, $\beta_1, \beta_2, \ldots, \beta_{n-1} \in \mathbb{F}_{2^n}$. Тогда на минимальном расстоянии от f существует ровно $2^{2n+1} - 2^n$ бент-функций.

Доказательство. По следствию 2 случаи размерностей 0 и 1 дают в точности $2^{2n+1} - 2^n$ бент-функций из нижней оценки. Согласно теореме 5 перестановка $\pi(x) = x^{2^n-2}$ не переводит аффинные подпространства размерностей $2, 3, \ldots, n-1$ в аффинные подпространства. Значит, случаи размерностей $2 \leqslant k \leqslant n-1$ не дают аффинных подпространств, удовлетворяющих теореме 3, так как не выполнен п. (1). Остаётся показать, что при указанном выборе φ и случай размерности $k = n$ не даёт подпространств, удовлетворяющих условию следствия 3.

Любая аффинная функция $H\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ единственным образом представима в виде

$$H(x) = \alpha_0 x^{2^0} + \alpha_1 x^{2^1} + \cdots + \alpha_{n-1} x^{2^{n-1}} + \alpha_n, \quad \text{где } \alpha_i \in \mathbb{F}_{2^n}.$$

Значит, функция из условия следствия 3 имеет вид $g + \varphi$, где $g \in \mathcal{G}_n$ и

$$\begin{aligned}
\mathcal{G}_n &= \{\mathrm{tr}((\alpha_0 x^{2^0} + \alpha_1 x^{2^1} + \cdots + \alpha_{n-1} x^{2^{n-1}} + \alpha_n) x^{2^n-2}) \mid \alpha_i \in \mathbb{F}_2^n\} \\
&= \{\mathrm{tr}(\alpha_0 x^{2^n-2+2^0} + \cdots + \alpha_{n-1} x^{2^n-2+2^{n-1}} + \alpha_n x^{2^n-2}) \mid \alpha_i \in \mathbb{F}_2^n\} \\
&= \{\mathrm{tr}(\alpha_0 x^{2^n-1} + \alpha_1 x^{2^1-1} + \cdots + \alpha_{n-1} x^{2^{n-1}-1} + \alpha_n x^{2^n-2}) \mid \alpha_i \in \mathbb{F}_2^n\} \\
&= \{\mathrm{tr}(\alpha_0 x^{11\ldots1_2} + \alpha_1 x^{00\ldots01_2} + \cdots + \alpha_{n-1} x^{011\ldots1_2} + \alpha_n x^{11\ldots10_2}) \mid \alpha_i \in \mathbb{F}_2^n\}.
\end{aligned}$$

Используя $\mathrm{tr}(y^2) = \mathrm{tr}(y)$ для $y = \alpha_n^{2^{n-1}} x^{011\ldots1_2}$, получаем

$$\mathrm{tr}\left(\alpha_n^{2^{n-1}} x^{011\ldots1_2}\right) = \mathrm{tr}\left(\left(\alpha_n^{2^{n-1}}\right)^2 (x^{011\ldots1_2})^2\right) = \mathrm{tr}(\alpha_n x^{11\ldots10_2}).$$

Таким образом,

$$\begin{aligned}
\mathcal{G}_n = \{&\mathrm{tr}(\alpha_1 x^{00\ldots01_2} + \cdots + \alpha_{n-2} x^{001\ldots1_2} \\
&+ [\alpha_{n-1} + \alpha_n^{2^{n-1}}] x^{011\ldots1_2} + \alpha_0 x^{11\ldots1_2}) \mid \alpha_i \in \mathbb{F}_2^n\}.
\end{aligned}$$

Положим $\alpha_i = \beta_i$, $i = 1, 2, \ldots, n-2$, $\alpha_{n-1} + \alpha_n^{2^{n-1}} = \beta_{n-1}$, $\mathrm{tr}(\alpha_0) = c_{2^n-1}$. При этом для всевозможных $\alpha_0, \alpha_1, \ldots, \alpha_n \in \mathbb{F}_2^n$ получаем всевозможные $\beta_1, \beta_2, \ldots, \beta_{n-1} \in \mathbb{F}_{2^n}$ и $c_{2^n-1} \in \mathbb{F}_2$. Также заметим, что $\mathrm{tr}(\alpha_0 x^{2^n-1}) = \mathrm{tr}(\alpha_0) x^{2^n-1}$, так как $x^{2^n-1} \in \mathbb{F}_2$ для любого $x \in \mathbb{F}_{2^n}$ и $\mathrm{tr}(0) = 0$. Значит,

$$\mathcal{G}_n = \{\mathrm{tr}(\beta_1 x^{2^1-1} + \cdots + \beta_{n-1} x^{2^{n-1}-1}) + c_{2^n-1} x^{2^n-1} \mid \beta_i \in \mathbb{F}_{2^n},\ c_{2^n-1} \in \mathbb{F}_2\}.$$

Заметим, что существует $g \in \mathcal{G}_n$ такая, что $g + \varphi$ аффинна, если и только если существует аффинная h такая, что $\varphi + h \in \mathcal{G}_n$. Тем самым, чтобы нижняя оценка достигалась, φ не должна быть представима в виде функций из $\mathcal{G}_n$ с точностью до аффинной части.

Нетрудно видеть, что θ и функции из $\mathcal{G}_n$ записаны в представлении (2), обладающем свойством единственности при простом n. Осталось заметить, что если φ удовлетворяет условию теоремы, то для любой $g \in \mathcal{G}_n$ функция $g + \varphi$ имеет в представлении (2) некоторое слагаемое $\mathrm{tr}(\gamma x^k)$, $\gamma \in \mathbb{F}_{2^n}$, $\gamma \neq 0$, где $k \neq 1$, т. е. является представителем другого

циклотомического класса. Значит, $g + \varphi$ не аффинная. Таким образом, условие следствия 3 не выполнено. Теорема 6 доказана.

Отметим, что на минимальном расстоянии от бент-функций из условия теоремы 6 лежат только бент-функции из класса $\mathcal{M}_{2n}$.

Замечание 2. В условие теоремы 6 не включено $n = 3$, так как при этом n не существует подходящих функций φ. Действительно, любая булева функция $\varphi\colon \mathbb{F}_{2^3} \to \mathbb{F}_2$ единственным образом представима в виде (2):

$$\varphi(y) = c_0 + \mathrm{tr}(\beta_1 y^{001_2} + \beta_2 y^{011_2}) + c_7 y^{111_2}, \quad \text{где } c_0, c_7 \in \mathbb{F}_2,\ \beta_1, \beta_2 \in \mathbb{F}_{2^3}.$$

Согласно экспериментальным данным при $n = 3$ для любых других взаимно однозначных функций, отличных от $\pi(y) = y^{2^n-2}$, нижняя оценка также не достигается.

Пример 1. Условию теоремы 6 при любом простом $n \geqslant 5$ удовлетворяют, в частности, функции

$$f(x, y) = \mathrm{tr}(xy^{2^n-2}) + \mathrm{tr}(y^5),$$
$$g(x, y) = \mathrm{tr}(xy^{2^n-2}) + \mathrm{tr}(y^{11}).$$

В силу единственности представления функции θ в теореме 6 несложно подсчитать число функций φ, подходящих под условие теоремы. Этим числом можно оценить количество бент-функций из $\mathcal{M}_{2n}$, для которых нижняя оценка достигается.

Следствие 4. *Пусть $n \geqslant 5$ простое. Тогда существует не менее $2^{2^n} - 2^{n^2-n+2}$ бент-функций из $\mathcal{M}_{2n}$, для которых достигается нижняя оценка $2^{2n+1} - 2^n$.*

Доказательство. Функции θ из условия теоремы 6 образуют множество

$$\mathcal{O}_n = \{c_0 + \mathrm{tr}(\beta_1 y^{2^1-1} + \beta_2 y^{2^2-1} + \cdots + \beta_{n-1} y^{2^{n-1}-1}) + c_{2^n-1} y^{2^n-1} \mid c_0, c_{2^n-1} \in \mathbb{F}_2,\ \beta_1, \beta_2, \ldots, \beta_{n-1} \in \mathbb{F}_{2^n}\}.$$

Как упоминалось в доказательстве теоремы 6, функции из множества $\mathcal{O}_n$ записаны в представлении (2), обладающем свойством единственности при простых n. Легко видеть, что

$$|\mathcal{O}_n| = 2^2 \cdot (2^n)^{n-1} = 2^{n^2-n+2},$$

так как это число различных способов выбрать $\beta_1, \beta_2, \ldots, \beta_{n-1} \in \mathbb{F}_{2^n}$, $c_0, c_{2^n-1} \in \mathbb{F}_2$. Значит, число функций φ, подходящих под условие теоремы 6, равно $2^{2^n} - 2^{n^2-n+2}$. Следствие 4 доказано.

6. Использование критерия для класса Мэйорана — МакФарланда

Здесь мы покажем, что уточнённый критерий, доказанный в разд. 3, может быть полезен и для аффинной классификации класса $\mathcal{M}_{2n}$. Начнём с необходимых определений.

Функции $F, G\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ называются *аффинно эквивалентными*, если существуют невырожденные двоичные матрицы A, B размеров $n \times n$ и $m \times m$ соответственно и векторы $a \in \mathbb{F}_2^n$, $b \in \mathbb{F}_2^m$ такие, что $F(x) = G(xA \oplus a)B \oplus b$ для всех $x \in \mathbb{F}_2^n$. Функции F и G *EA-эквивалентны*, если F аффинно эквивалентна $G \oplus H$ для некоторой аффинной функции $H\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$.

Для булевых функций, т. е. в случае $m = 1$, матрица B может быть только единичной, поэтому её можно убрать из определения. Аффинная классификация является важной составляющей исследований класса бент-функций в частности и криптографических булевых функций в целом, особенно если речь идёт о функциях от малого числа переменных. Известно, что эквивалентность, определённая по группе автоморфизмов класса бент-функций, в точности является EA-эквивалентностью (см. [31]).

Поскольку действие элементов группы автоморфизмов изометрично, доказанные ранее результаты справедливы не только для функций из класса $\mathcal{M}_{2n}$, но и для всех функций, EA-эквивалентных им.

Начнём с доказательства вспомогательной леммы.

Лемма 2. *Пусть $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^m$ аффинна на $a \oplus E$, где E — линейное подпространство $\mathbb{F}_2^n$ и $a \in \mathbb{F}_2^n$. Тогда для любых $t, s \in E$ справедливо $\mathrm{D}_t\mathrm{D}_s F(x) = 0$ для всех $x \in a \oplus E$.*

Доказательство. Пусть $t, s \in E$. Поскольку F аффинна, по определению существует двоичная матрица A размера $m \times n$ и вектор $b \in \mathbb{F}_2^m$ такие, что для всех $x \in a \oplus E$ справедливо $F(x) = xA \oplus b$. Тогда

$$F(x \oplus t \oplus s) = (x \oplus t \oplus s)A \oplus b = ((x \oplus t)A \oplus b) \\ \oplus ((x \oplus s)A \oplus b) \oplus (xA \oplus b) = F(x \oplus t) \oplus F(x \oplus s) \oplus F(x).$$

Таким образом, $F(x \oplus t \oplus s) \oplus F(x \oplus t) \oplus F(x \oplus s) \oplus F(x) = 0$. Это означает, что $\mathrm{D}_t\mathrm{D}_s F(x) = 0$. Лемма 2 доказана.

Теорема 7. *Пусть $f(x, y) = \langle x, \pi(y)\rangle \oplus \varphi(y)$ и $f'(x, y) = \langle x, \pi'(y)\rangle \oplus \varphi'(y)$ — бент-функции из класса $\mathcal{M}_{2n}$, EA-эквивалентные друг другу, при этом $\mathrm{D}_t\mathrm{D}_s\pi \not\equiv 0$ для любых $t, s \in \mathbb{F}_2^n$, где $t, s \neq 0$ и $t \neq s$. Тогда существуют обратимые двоичные матрицы A, B размера $n \times n$ и векторы $a, b, c \in \mathbb{F}_2^n$ такие, что*

(1) $\pi(y) = \pi'(yA \oplus a)B \oplus b$ *для всех* $y \in \mathbb{F}_2^n$,

(2) $\varphi(y) = \varphi'(yA \oplus a) \oplus \langle yS \oplus c,\, \pi'(yA \oplus a)\rangle \oplus \ell(y)$ *для всех* $y \in \mathbb{F}_2^n$, *где* S *— некоторая двоичная матрица размера* $n \times n$, *а* $\ell \in \mathcal{F}_n$ *— некоторая аффинная функция.*

Доказательство. Пусть $E = \big\{(x, 0) \mid x \in \mathbb{F}_2^n\big\}$. Ясно, что функция f аффинна на $z \oplus E$ для всех $z \in \mathbb{F}_2^{2n}$. Докажем от противного, что E — единственное подпространство $\mathbb{F}_2^n$ размерности n, удовлетворяющее данному свойству.

Пусть существует подпространство $E' \neq E$ размерности n такое, что функция f аффинна на $z \oplus E'$ для всех $z \in \mathbb{F}_2^{2n}$. Согласно теоремам 1 и 3 подпространство E' имеет базис

$$\begin{pmatrix} T & L \\ R & 0 \end{pmatrix},$$

где L и R — полноранговые матрицы размеров $k \times n$ и $(n-k) \times n$ соответственно, при этом для всех $a, b \in \mathbb{F}_2^n$

(1) $\pi(a \oplus \langle L\rangle) = \pi(a) \oplus \langle R\rangle^{\perp}$ и

(2) $\langle uT \oplus b,\, \pi(uL \oplus a)\rangle \oplus \varphi(uL \oplus a)$ — аффинная функция от $u = (u_1, \ldots, u_k) \in \mathbb{F}_2^k$.

Поскольку $E \neq E'$, то $k > 0$. Более того, $k \neq 1$. Действительно, если $k = 1$, то условие (1) гарантирует, что $\pi(a \oplus \langle L\rangle) = \pi(a) \oplus \langle R\rangle^{\perp}$. Так как $\dim\langle R\rangle^{\perp} = 1$, для $t \in \langle L\rangle$ и $s \in \langle R\rangle^{\perp}$, $t, s \neq 0$ справедливо $\mathrm{D}_t\pi(a) = \pi(a) \oplus \pi(a \oplus t) = (\pi(a) \oplus 0) \oplus (\pi(a) \oplus s) = s$. Поскольку R фиксировано, $\mathrm{D}_t\pi \equiv s$. Это означает, что существует нулевая вторая производная.

Таким образом, $k \geqslant 2$. Рассмотрим условие (2) при $b = 0$: получаем, что функция $\langle uT,\, \pi(uL \oplus a)\rangle \oplus \varphi(uL \oplus a)$ аффинна. Подставляя другие b, получаем, что $\langle b,\, \pi(uL \oplus a)\rangle \oplus \langle uT,\, \pi(uL \oplus a)\rangle \oplus \varphi(uL \oplus a)$ тоже аффинна. Следовательно, $\langle b,\, \pi(uL \oplus a)\rangle$ аффинна. Подставляя в b векторы из стандартного базиса, получаем, что векторная функция $\pi(uL \oplus a)$ аффинна от переменных u. Согласно лемме 2 для любых $t, s \in \langle L\rangle$ справедливо $\mathrm{D}_t\mathrm{D}_s\pi(x) = 0$ для всех $x \in a \oplus \langle L\rangle$. Это означает, что $\mathrm{D}_t\mathrm{D}_s\pi \equiv 0$. В то же время $k \geqslant 2$, поэтому всегда можно выбрать $t, s \neq 0$, $t \neq s$, что противоречит условию теоремы.

Далее, так как f и f' EA-эквивалентны, для некоторой обратимой двоичной матрицы A размера $2n \times 2n$ и $c, d \in \mathbb{F}_2^n$ справедливо $f(x, y) = f'((x, y)A \oplus (c, d)) \oplus \ell(x, y)$ для всех $x, y \in \mathbb{F}_2^n$, где ℓ — некоторая аффинная функция. Поскольку f' аффинна на $z \oplus E$ для всех $z \in \mathbb{F}_2^{2n}$, функция f аффинна на $z \oplus A^{-1}(E) = \{z \oplus xA^{-1} \mid x \in E\}$ для всех $z \in \mathbb{F}_2^{2n}$. Однако E является единственным подпространством $\mathbb{F}_2^{2n}$, обладающим данным свойством. Значит, $A^{-1}(E) = E$, что эквивалентно тому, что $E = A(E) = \{xA \mid x \in E\}$. Другими словами, матрицу A можно представить в следующем виде:

$$A = \begin{pmatrix} P & 0 \\ S & Q \end{pmatrix}, \quad \text{т. е. } (x,y)A = (xP \oplus yS, yQ),$$

где P, Q, S — двоичные матрицы размера $n \times n$, причём P, Q обратимы. Значит,

$$\begin{aligned} f(x,y) \oplus \ell(x,y) &= f'((x,y)A \oplus (c,d)) = f'(xP \oplus yS \oplus c, yQ \oplus d) \\ &= \langle xP \oplus yS \oplus c,\, \pi'(yQ \oplus d)\rangle \oplus \varphi'(yQ \oplus d) \\ &= \langle xP,\, \pi'(yQ \oplus d)\rangle \oplus \varphi'(yQ \oplus d) \oplus \langle yS \oplus c,\, \pi'(yQ \oplus d)\rangle \\ &= \langle x,\, \pi'(yQ \oplus d)P^T\rangle \oplus \varphi'(yQ \oplus d) \oplus \langle yS \oplus c,\, \pi'(yQ \oplus d)\rangle. \end{aligned}$$

Так как ℓ аффинна, то $\ell(x,y) = \langle v,\, x\rangle \oplus \langle w,\, y\rangle \oplus e$, где $v, w \in \mathbb{F}_2^n$, $e \in \mathbb{F}_2$. Это означает, что $\pi(y) = \pi'(yQ \oplus d)P^T \oplus v$. В то же время получаем

$$\varphi(y) = \varphi'(yQ \oplus d) \oplus \langle yS \oplus c,\, \pi'(yQ \oplus d)\rangle \oplus \langle w,\, y\rangle \oplus e.$$

Теорема 7 доказана.

Таким образом, если перестановка π не имеет нулевых вторых производных, то любая функция из класса Мэйорана — МакФарланда, которая EA-эквивалентна $f(x,y) = \langle x,\, \pi(y)\rangle \oplus \varphi(y)$, должна иметь перестановку, аффинно эквивалентную π.

Заключение

Полученные в работе результаты связывают рассматриваемые метрические свойства класса бент-функций со свойствами APN-функций, которые также образуют один из важнейших классов криптографических функций. Например, необходимым условием достижимости нижней оценки числа ближайших бент-функций к функции из $\mathcal{M}_{2n}$ является использование взаимно однозначной APN-функции от n переменных. Выбор таких функций может обеспечить достижимость оценки не только в случае простых $n \geqslant 5$, как в теореме 6. Однако, при чётном n вопрос их существования является открытым, начиная с $n = 8$, и называется «The big APN problem».

ЛИТЕРАТУРА

1. **Rothaus O.** On «bent» functions // J. Comb. Theory. Ser. A. 1976. V. 20, No. 3. P. 300–305.
2. **Tokareva N. N.** Bent functions: Results and applications to cryptography. London: Acad. Press, 2015. 220 p.
3. **Adams C.** The CAST-128 encryption algorithm. RFC 2144. Wilmington, DE: IETF, 1997. Available at `doi.org/10.17487/RFC2144` (accessed Mar. 4, 2023).
4. **Токарева Н. Н.** Обобщения бент-функций. Обзор работ // Дискрет. анализ и исслед. операций. 2010. Т. 17, № 1. С. 34–64.

5. **Agievich S. V., Gorodilova A. A., Idrisova V. A., Kolomeec N. A., Shushuev G. I., Tokareva N. N.** Mathematical problems of the Second International Students' Olympiad in Cryptography // Cryptologia. 2017. V. 41, No. 6. P. 534–565.
6. **Tokareva N. N., Gorodilova A. A., Agievich S. V.** [et al.]. Mathematical methods in solutions of the problems presented at the Third International Students' Olympiad in Cryptography // Прикл. дискрет. математика. 2018. № 40. С. 34–58.
7. **Токарева Н. Н.** Бент-функции: результаты и приложения. Обзор работ // Прикл. дискрет. математика. 2009. № 1. С. 15–37.
8. **Carlet C.** Boolean functions for cryptography and error-correcting codes // Boolean models and methods in mathematics, computer science, and engineering. New York: Camb. Univ. Press, 2010. P. 257–397. (Encycl. Math. Appl.; V. 134).
9. **Carlet C.** Vectorial Boolean functions for cryptography // Boolean models and methods in mathematics, computer science, and engineering. New York: Camb. Univ. Press, 2010. P. 398–472. (Encycl. Math. Appl.; V. 134).
10. **Cusick T. W., Stănică P.** Cryptographic Boolean functions and applications. London: Acad. Press, 2017. 275 p.
11. **Логачёв О. А., Сальников А. А., Смышляев С. В., Ященко В. В.** Булевы функции в теории кодирования и криптологии. М.: МЦНМО, 2012. 584 с.
12. **Potapov V. N., Taranenko A. A., Tarannikov Yu. V.** Asymptotic bounds on numbers of bent functions and partitions of the Boolean hypercube into linear and affine subspaces. Ithaca, NY: Cornell Univ., 2021. (Cornell Univ. Libr. e-Print Archive; arXiv:2108.00232).
13. **Shaporenko A. S.** Derivatives of bent functions in connection with the bent sum decomposition problem // Des. Codes Cryptogr. 2023. V. 91, No. 5. P. 1607–1625.
14. **Tokareva N. N.** On the number of bent functions from iterative constructions: lower bounds and hypothesis // Adv. Math. Commun. 2011. V. 5, No. 4. P. 609–621.
15. **Carlet C.** Two new classes of bent functions // Advances in cryptology — EUROCRYPT'93. Proc. Workshop Theory and Applications of Cryptographic Techniques (Lofthus, Norway, May 23–27, 1993). Heidelberg: Springer, 1994. P. 77–101. (Lect. Notes Comput. Sci.; V. 765).
16. **Zhang F., Cepak N., Pasalic E., Wei Y.** Further analysis of bent functions from $\mathcal{C}$ and $\mathcal{D}$ which are provably outside or inside $\mathcal{M}^{\#}$ // Discrete Appl. Math. 2020. V. 285. P. 458–472.
17. **Коломеец Н. А., Павлов А. В.** Свойства бент-функций, находящихся на минимальном расстоянии друг от друга // Прикл. дискрет. математика. 2009. № 4. С. 5–20.
18. **Коломеец Н. А.** Перечисление бент-функций на минимальном расстоянии от квадратичной бент-функции // Дискрет. анализ и исслед. операций. 2012. Т. 19, № 1. С. 41–58.

19. Коломеец Н. А. Верхняя оценка числа бент-функций на расстоянии 2^k от произвольной бент-функции от $2k$ переменных // Прикл. дискрет. математика. 2014. № 3. С. 28–39.
20. Kolomeec N. A. The graph of minimal distances of bent functions and its properties // Des. Codes Cryptogr. 2017. V. 85, No. 3. P. 395–410.
21. Canteaut A., Daum M., Dobbertin H., Leander G. Finding nonnormal bent functions // Discrete Appl. Math. 2006. V. 154, No. 2. P. 202–218.
22. Leander G., McGuire G. Construction of bent functions from near-bent functions // J. Comb. Theory. Ser. A. 2009. V. 116, No. 4. P. 960–970.
23. Kutsenko A. V. Metrical properties of self-dual bent functions // Des. Codes Cryptogr. 2020. V. 88, No. 1. P. 201–222.
24. Kolomeec N. A. Some general properties of modified bent functions through addition of indicator functions // Cryptogr. Commun. 2021. V. 13, No. 6. P. 909–926.
25. Nyberg K. Differentially uniform mappings for cryptography // Advances in cryptology — EUROCRYPT'93. Proc. Workshop Theory and Applications of Cryptographic Techniques (Lofthus, Norway, May 23–27, 1993). Heidelberg: Springer, 1994. P. 55–64. (Lect. Notes Comput. Sci.; V. 765).
26. Carlet C. Open questions on nonlinearity and on APN functions // Arithmetic of finite fields. Rev. Sel. Pap. 5th Int. Workshop (Gebze, Turkey, Sept. 27–28, 2014). Cham: Springer, 2015. P. 83–107. (Lect. Notes Comput. Sci.; V. 9061).
27. Browning K. A., Dillon J. F., McQuistan M. T., Wolfe A. J. An APN permutation in dimension six // Finite fields: Theory and applications. Proc. 9th Int. Conf. Finite Fields and Applications (Dublin, Ireland, July 13–17, 2009). Providence, RI: AMS, 2010. P. 33–42. (Contemp. Math.; V. 518).
28. Kalgin K. V., Idrisova V. A. The classification of quadratic APN functions in 7 variables and combinatorial approaches to search for APN functions // Cryptogr. Commun. 2023. V. 15, No. 2. P. 239–256.
29. Kolomeec N. A., Bykov D. A. On the image of an affine subspace under the inverse function within a finite field. Ithaca, NY: Cornell Univ., 2022. (Cornell Univ. Libr. e-Print Archive; arXiv:2206.14980).
30. Коломеец Н. А., Быков Д. А. Об инвариантных подпространствах функций, аффинно эквивалентных обращению элементов конечного поля // Прикл. дискрет. математика. Прил. 2022. № 15. С. 5–8.
31. Токарева Н. Н. Группа автоморфизмов множества бент-функций // Дискрет. математика. 2010. Т. 22, № 4. С. 34–42.

Быков Денис Александрович
Коломеец Николай Александрович

Статья поступила
6 марта 2023 г.
После доработки —
2 мая 2023 г.
Принята к публикации
5 мая 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
July–September 2023. Vol. 30, No. 3. P. 57–80

UDC 519.7 DOI 10.33048/daio.2023.30.764

ON A LOWER BOUND FOR THE NUMBER OF BENT FUNCTIONS AT THE MINIMUM DISTANCE FROM A BENT FUNCTION IN THE MAIORANA–MCFARLAND CLASS

D. A. Bykov[1, a] and *N. A. Kolomeec*[2, b]

[1] Novosibirsk State University,
2 Pirogov Street, 630090 Novosibirsk, Russia

[2] Sobolev Institute of Mathematics,
4 Acad. Koptyug Avenue, 630090 Novosibirsk, Russia

E-mail: [a] den.bykov.2000i@gmail.com, [b] kolomeec@math.nsc.ru

Abstract. Bent functions at the minimum distance 2^n from a given bent function in $2n$ variables belonging to the Maiorana–McFarland class $\mathcal{M}_{2n}$ are investigated. We provide a criterion for a function obtained using the addition of the indicator of an n-dimensional affine subspace to a given bent function from $\mathcal{M}_{2n}$ to be a bent function as well. In other words, all bent functions at the minimum distance from a Maiorana–McFarland bent function are characterized. It is shown that the lower bound $2^{2n+1} - 2^n$ for the number of bent functions at the minimum distance from $f \in \mathcal{M}_{2n}$ is not attained if the permutation used for constructing f is not an APN function. It is proven that for any prime $n \geqslant 5$ there are functions from $\mathcal{M}_{2n}$ for which this lower bound is accurate. Examples of such bent functions are found. It is also established that the permutations of EA-equivalent functions from $\mathcal{M}_{2n}$ are affinely equivalent if the second derivatives of at least one of the permutations are not identically zero. Bibliogr. 31.

Keywords: bent function, Boolean function, minimum distance, Maiorana–McFarland class, lower bound, affine equivalence.

This research is carried out within the framework of the state contract of the Sobolev Institute of Mathematics (Project FWNF–2022–0018).

English version: Journal of Applied and Industrial Mathematics **17** (3) (2023).

REFERENCES

1. **O. Rothaus,** On «bent» functions, *J. Comb. Theory, Ser. A,* **20** (3), 300–305 (1976).
2. **N. N. Tokareva,** *Bent Functions: Results and Applications to Cryptography* (Acad. Press, London, 2015).
3. **C. Adams,** The CAST-128 encryption algorithm, *RFC 2144* (IETF, Wilmington, DE, 1997). Available at `doi.org/10.17487/RFC2144` (accessed Mar. 4, 2023).
4. **N. N. Tokareva,** Generalizations of bent functions. A survey, *Diskretn. Anal. Issled. Oper.* **17** (1), 34–64 (2010) [Russian] [*J. Appl. Ind. Math.* **5** (1), 110–129 (2011)].
5. **S. V. Agievich, A. A. Gorodilova, V. A. Idrisova, N. A. Kolomeec, G. I. Shushuev,** and **N. N. Tokareva,** Mathematical problems of the Second International Students' Olympiad in Cryptography, *Cryptologia* **41** (6), 534–565 (2017).
6. **N. N. Tokareva, A. A. Gorodilova, S. V. Agievich,** [et al.], Mathematical methods in solutions of the problems presented at the Third International Students' Olympiad in Cryptography, *Prikl. Diskretn. Mat.,* No. 40, 34–58 (2018).
7. **N. N. Tokareva,** Bent Functions: Results and Applications. A survey, *Prikl. Diskretn. Mat.,* No. 1, 15–37 (2009) [Russian].
8. **C. Carlet,** Boolean functions for cryptography and error-correcting codes, in *Boolean Models and Methods in Mathematics, Computer Science, and Engineering* (Camb. Univ. Press, New York, 2010), pp. 257–397. (Encycl. Math. Appl., Vol. 134).
9. **C. Carlet,** Vectorial Boolean functions for cryptography, in *Boolean Models and Methods in Mathematics, Computer Science, and Engineering* (Camb. Univ. Press, New York, 2010), pp. 398–472. (Encycl. Math. Appl., Vol. 134).
10. **T. W. Cusick** and **P. Stănică,** *Cryptographic Boolean Functions and Applications* (Acad. Press, London, 2017).
11. **O. A. Logachyov, A. A. Sal'nikov, S. V. Smyshlyaev,** and **V. V. Yashchenko,** *Boolean Functions in Coding Theory and Cryptology* (MTsNMO, Moscow, 2012) [Russian].
12. **V. N. Potapov, A. A. Taranenko,** and **Yu. V. Tarannikov,** Asymptotic bounds on numbers of bent functions and partitions of the Boolean hypercube into linear and affine subspaces (Cornell Univ., Ithaca, NY, 2021) (Cornell Univ. Libr. e-Print Archive; arXiv:2108.00232).
13. **A. S. Shaporenko,** Derivatives of bent functions in connection with the bent sum decomposition problem, *Des. Codes Cryptogr.* **91** (5), 1607–1625 (2023).
14. **N. N. Tokareva,** On the number of bent functions from iterative constructions: lower bounds and hypothesis, *Adv. Math. Commun.* **5** (4), 609–621 (2011).
15. **C. Carlet,** Two new classes of bent functions, in *Advances in Cryptology — EUROCRYPT'93* (Proc. Workshop Theory and Applications of Cryptographic Techniques, Lofthus, Norway, May 23–27, 1993) (Springer, Heidelberg, 1994), pp. 77–101 (Lect. Notes Comput. Sci., Vol. 765).

16. **F. Zhang, N. Cepak, E. Pasalic,** and **Y. Wei,** Further analysis of bent functions from $\mathcal{C}$ and $\mathcal{D}$ which are provably outside or inside $\mathcal{M}^{\#}$, *Discrete Appl. Math.* **285**, 458–472 (2020).
17. **N. A. Kolomeec** and **A. V. Pavlov,** Properties of bent functions at the minimal distance from each other, *Prikl. Diskretn. Mat.,* No. 4, 5–20 (2009) [Russian].
18. **N. A. Kolomeec,** Enumeration of the bent functions of least deviation from a quadratic bent function, *Diskretn. Anal. Issled. Oper.* **19** (1), 41–58 (2012) [Russian] [*J. Appl. Ind. Math.* **6** (3), 306–317 (2012)].
19. **N. A. Kolomeec,** An upper bound for the number of bent functions at the distance 2^k from an arbitrary bent function in $2k$ variables, *Prikl. Diskretn. Mat.,* No. 3, 28–39 (2014) [Russian].
20. **N. A. Kolomeec,** The graph of minimal distances of bent functions and its properties, *Des. Codes Cryptogr.* **85** (3), 395–410 (2017).
21. **A. Canteaut, M. Daum, H. Dobbertin,** and **G. Leander,** Finding nonnormal bent functions, *Discrete Appl. Math.* **154** (2), 202–218 (2006).
22. **G. Leander** and **G. McGuire,** Construction of bent functions from near-bent functions, *J. Comb. Theory. Ser. A.* **116** (4), 960–970 (2009).
23. **A. V. Kutsenko,** Metrical properties of self-dual bent functions, *Des. Codes Cryptogr.* **88** (1), 201–222 (2020).
24. **N. A. Kolomeec,** Some general properties of modified bent functions through addition of indicator functions, *Cryptogr. Commun.* **13** (6), 909–926 (2021).
25. **K. Nyberg,** Differentially uniform mappings for cryptography, in *Advances in Cryptology — EUROCRYPT'93* (Proc. Workshop Theory and Applications of Cryptographic Techniques, Lofthus, Norway, May 23–27, 1993) (Springer, Heidelberg, 1994), pp. 55–64 (Lect. Notes Comput. Sci., Vol. 765).
26. **C. Carlet,** Open questions on nonlinearity and on APN functions, in *Arithmetic of Finite Fields* (Rev. Sel. Pap. 5th Int. Workshop, Gebze, Turkey, Sept. 27–28, 2014) (Springer, Cham, 2015), pp. 83–107 (Lect. Notes Comput. Sci., Vol. 9061).
27. **K. A. Browning, J. F. Dillon, M. T. McQuistan,** and **A. J. Wolfe,** An APN permutation in dimension six, in *Finite Fields: Theory and Applications* (Proc. 9th Int. Conf. Finite Fields and Applications, Dublin, Ireland, July 13–17, 2009) (AMS, Providence, RI, 2010), pp. 33–42 (Contemp. Math., Vol. 518).
28. **K. V. Kalgin** and **V. A. Idrisova,** The classification of quadratic APN functions in 7 variables and combinatorial approaches to search for APN functions, *Cryptogr. Commun.* **15** (2), 239–256 (2023).
29. **N. A. Kolomeec** and **D. A. Bykov,** On the image of an affine subspace under the inverse function within a finite field (Cornell Univ., Ithaca, NY, 2022) (Cornell Univ. Libr. e-Print Archive; arXiv:2206.14980).
30. **N. A. Kolomeec** and **D. A. Bykov,** Invariant subspaces of functions affine equivalent to the finite field inversion, *Prikl. Diskretn. Mat., Prilozh.,* No. 15, 5–8 (2022) [Russian].

31. N. N. Tokareva, The group of automorphisms of the set of bent functions, *Diskretn. Mat.* **22** (4), 34–42 (2010) [Russian] [*Discrete Math. Appl.* **20** (5–6), 655–664 (2010)].

Denis A. Bykov
Nikolay A. Kolomeec

Received March 6, 2023
Revised May 2, 2023
Accepted May 5, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль–сентябрь 2023. Т. 30, № 3. С. 81–95

УДК 519.171 DOI 10.33048/daio.2023.30.757

ПРЕДСТАВЛЕНИЯ РЁБЕР ГИПЕРГРАФОВ ОБОБЩЁННЫМИ ПУТЯМИ

М. Н. Вялый[1,2,3,a], *В. Е. Карпов*[1,b]

[1] Московский физико-технический институт (гос. университет), Институтский пер., 9, 141701 Долгопрудный, Россия

[2] Федеральный исследовательский центр «Информатика и управление» РАН, ул. Вавилова, 42, 119333 Москва, Россия

[3] Национальный исследовательский университет «Высшая школа экономики», Покровский б-р, 11, 109028 Москва, Россия

E-mail: [a] vyalyi@gmail.com, [b] ve.karpov@yandex.ru

Аннотация. Изучается задача реализации гиперграфа на графе с условием, что каждое ребро гиперграфа реализуется подграфом, в котором ровно две вершины имеют нечётную степень. Установлена связь такой задачи реализации гиперграфов и гипотезы о двойном покрытии циклами. Доказана алгоритмическая трудность проверки существования реализации в различных постановках: реализации на всех графах, на простых графах и на графах из нескольких очень узких классов. Библиогр. 11.

Ключевые слова: гиперграф, покрытие циклами, эйлеров граф, NP-полнота.

Введение

Одной из важных задач в теории гиперграфов является задача построения *носителя* гиперграфа — такого графа на том же множестве вершин, что и гиперграф, в котором каждое гиперребро индуцирует связный подграф. Интерес представляют как структурные результаты, так и результаты об алгоритмической сложности проверки существования носителя в данном классе графов. Например, хорошо известен критерий существования носителя в виде дерева [1, гл. XI]. На основе этого критерия легко строится алгоритм проверки существования дерева-

Работа первого автора выполнена при поддержке Программы фундаментальных исследований НИУ ВШЭ, а также в рамках государственного задания ФИЦ ИУ РАН (проект № 0063–2019–0003).

носителя. В [2] приводится линейный алгоритм для этой задачи. Линейные алгоритмы также найдены для проверки существования носителя в виде пути и в виде цикла [3]. Известны также полиномиальные алгоритмы проверки существования носителя в виде дерева ограниченной степени [3] и в виде кактуса [4].

Многие задачи такого рода NP-полны, например, задача проверки существования планарного [5] или 2-внешнепланарного носителя [3].

Здесь рассматривается другая задача реализации гиперграфа на графе. А именно, нас интересуют такие отображения вершин гиперграфа в рёбра некоторого графа, при которых образ каждого гиперребра является подграфом, у которого степени ровно двух вершин нечётные; далее называем такие подграфы обобщёнными путями (см. разд. 1). Если носитель реализации является деревом, то обобщённые пути — это простые пути по дереву, поэтому существование носителя-дерева такого, что каждое гиперребро является путём, равносильно реализации гиперграфа на дереве. Из [6] следует полиномиальный алгоритм для реализации гиперграфов путями на деревьях.

Рассматриваемая здесь постановка мотивирована связью с гипотезой о двойном покрытии циклами (CDC). В разд. 2 доказано, что гипотеза CDC равносильна гипотезе о возможности реализации гиперграфов специального вида — двойственных к гиперграфам, реализованным на дереве (определения см. в разд. 1). В силу этой связи нас интересуют реализации на графах с кратными рёбрами, в отличие от работ о существовании носителя, а условие связности подграфа, реализующего гиперребро, для нас избыточно.

В разд. 3 на языке графических реализаций сформулированы усиления гипотезы CDC. Эти сильные гипотезы, скорее всего, неверны, хотя и подтверждаются анализом графов с небольшим числом вершин (до 26). Опровержение этих гипотез представляется интересной задачей.

В разд. 4 анализируется алгоритмическая сложность задач, связанных с реализациями гиперграфов в указанном выше смысле. Большинство из них оказываются NP-полными.

В заключении обсуждаются открытые вопросы, возникшие в ходе этой работы.

1. Определения и обозначения

(Простой) *гиперграф* (V, E) — это семейство непустых подмножеств множества V, элементы которого называются (гипер-) *вершинами*. Элементы E называются (гипер-) *рёбрами*. *Граф* — это гиперграф, у которого размер каждого ребра равен 2.

Нам потребуются графы и гиперграфы с кратными рёбрами. В этом случае удобны следующие обозначения. Гиперграф задаётся функцией

из множества рёбер E в непустые подмножества вершин. В этом случае гиперграф удобно представлять матрицей инцидентности. Для гиперграфа с кратными рёбрами $\mathcal{H}(V, E)$ матрица инцидентности — это двоичная матрица размера $|V|\times|E|$, элемент $\mathcal{H}(v, e)$ которой равен 1 тогда и только тогда, когда v принадлежит ребру $\mathcal{H}(e)$. Также используем для отношения инцидентности обозначение $v \in \mathcal{H}(e)$.

Двойственный гиперграф $\mathcal{H}^T(V^T, E^T)$ определяется транспонированием матрицы инцидентности гиперграфа:

$$E^T = V, \quad V^T = E, \quad \mathcal{H}^T(e, v) = \mathcal{H}(v, e).$$

Утолщением $G'(V, E')$ простого графа (V, E) будем называть граф с тем же множеством вершин и добавленными кратными рёбрами, т. е. каждое $e \in E$ является ребром $G'(V, E')$ и $G'(e') \in E$ для любого $e' \in E'$.

Обобщённым путём (*обобщённым циклом*) в графе называем не обязательно связный рёберный подграф, у которого ровно две вершины имеют нечётную степень (в случае цикла все вершины чётной степени). Для простоты обозначений отождествляем рёберный подграф и множество входящих в него рёбер. Заметим, что если в графе нет циклов, то обобщённый путь обязательно является простым путём в графе — связным графом, у которого степени двух вершин равны 1, а степени остальных равны 2.

Нас интересуют представления гиперграфов в виде семейств обобщённых путей в некотором графе (возможно, с кратными рёбрами). Гиперграф $\mathcal{H}(X, E)$ называем *графическим*, если существует его *реализация* — такой граф $G(V, X)$, что $\{x \mid x \in \mathcal{H}(e)\}$ является обобщённым путём для каждого $e \in E$. Тем самым вершинам гиперграфа соответствуют рёбра в его реализации, а гиперрёбрам — обобщённые пути. Будем также говорить, что $\mathcal{H}(X, E)$ реализуется на $G(V, X)$.

Представляют интерес реализации на ограниченных классах графов. В частности, будем называть гиперграф *сильно графическим*, если для него существует реализация на простом графе (без кратных рёбер), *сильно древесным*, если для него существует реализация на дереве, и *древесным*, если для него существует реализация на утолщении дерева.

Заметим, что кратные гиперрёбра не влияют на свойства графичности гиперграфа. Однако наличие кратных рёбер в реализации существенно: класс графических гиперграфов шире класса сильно графических.

Пример 1. Гиперграф $\mathcal{H}_0$, заданный матрицей

$$\begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix},$$

древесный, но не сильно графический. Его можно реализовать на утолщении пути длины 2. Занумеруем вершины гиперграфа сверху вниз, а столбцы — слева направо. Реализация гиперграфа $\mathcal{H}_0$ на утолщении пути с вершинами a, b, c (a, c — концы пути), в котором ребро $\{a, b\}$ имеет кратность 3, а ребро $\{b, c\}$ — кратность 1, задаётся так: первым трём вершинам соответствует ребро $\{a, b\}$, а последней — ребро $\{b, c\}$. При таком соответствии любое гиперребро соответствует обобщённому пути — утолщению пути длины 2, в котором одно ребро простое, а другое имеет кратность 3 (для первого столбца) или 2 (для остальных столбцов).

Докажем, что у $\mathcal{H}_0$ нет реализаций без кратных рёбер. Первое гиперребро из $\mathcal{H}_0$ содержит все вершины. Значит, в любой реализации образ всего множества вершин является обобщённым путём. Если кратных рёбер нет, то вариантов всего два: цикл на трёх вершинах с прикреплённым висячим ребром или же путь P_5 (на 5 вершинах с 4 рёбрами). В каждом из вариантов есть всего два обобщённых пути из трёх рёбер (заметим, что в простом графе обобщённый путь из трёх рёбер является простым путём). Значит, реализация остальных трёх гиперрёбер из $\mathcal{H}_0$ ни в одном из этих вариантов невозможна.

2. Связь с гипотезой о двойном покрытии циклами

Напомним гипотезу о двойном покрытии. *Мостом* в графе называется ребро, удаление которого увеличивает число компонент связности. Нетрудно видеть, что мост не принадлежит никакому циклу.

Гипотеза двойного покрытия циклами (CDC) [7, 8]. *Для любого графа G без мостов существует двойное покрытие обобщёнными циклами — множество обобщённых циклов $\{E_1, E_2, \ldots, E_k\}$ такое, что каждое ребро графа входит ровно в два из них.*

Без ограничения общности в гипотезе CDC достаточно рассматривать только связные графы, в дальнейшем предполагаем это условие выполненным.

Обобщённый цикл является эйлеровым графом, и его рёбра разбиваются в дизъюнктное объединение рёбер простых циклов. Тем самым гипотеза CDC равносильна утверждению о двойном покрытии рёбер графа простыми циклами. Для покрытий обобщёнными циклами существует более сильная гипотеза 5-CDC, которая утверждает, что каждый граф без мостов имеет двойное покрытие пятью обобщёнными циклами [9, гипотеза 7.4.1, с. 162].

Гипотеза о двойном покрытии циклами выражается в терминах графичности гиперграфов.

Гипотеза двойственности. *Гиперграф, двойственный к сильно древесному гиперграфу без изолированных вершин, является графическим.*

Теорема 1. *Гипотеза CDC равносильна гипотезе двойственности.*

Прежде чем перейти к доказательству, выпишем некоторые понятия линейной алгебры, в терминах которых его удобно провести. Для графа $G(V, E)$ через $\mathcal{E}(G)$ обозначим векторное пространство над $\mathbb{F}_2$, базис которого индексирован рёбрами графа G. В дальнейшем отождествляем рёбра с векторами этого выделенного базиса. Подграфу графа G отвечает вектор, который является суммой по всем рёбрам, входящим в подграф, поэтому подмножества рёбер графа отождествляются с векторами $\mathcal{E}(G)$. Аналогично введём пространство $\mathcal{V}(G)$ с выделенным базисом, индексированным вершинами графа, и отождествим подмножества вершин с векторами $\mathcal{V}(G)$.

Выделенные базисы пространств $\mathcal{E}(G)$ и $\mathcal{V}(G)$ задают в этих пространствах скалярное произведение обычным образом: сумма произведений координат в выделенном базисе.

Как известно (см. [10, разд. 1.9]), обобщённые циклы образуют подпространство $\mathcal{C}(G)$ пространства $E(G)$, разрезы (дополненные пустым множеством) также образуют подпространство $\mathcal{C}^*(G)$ и $\mathcal{C}^*(G) = \mathcal{C}(G)^\perp$ (пространство разрезов является ортогональным дополнением пространства циклов).

Если задано двойное покрытие $C = \{E_1, E_2, \ldots, E_k\}$ рёбер связного графа $G = (V, E)$, то однозначно определён *геометрически двойственный*[1)] *граф* $G^*(\{1, \ldots, k\}, E)$ с тем же множеством рёбер, так что $\mathcal{E}(G) = \mathcal{E}(G^*)$. Для $e \in E$ полагаем $G^*(e) = \{i, j\}$, где E_i, E_j содержат e. Таким образом, 1-вершинный разрез $E(i)$ в G^* совпадает с обобщённым циклом E_i в G. Поскольку 1-вершинные разрезы образуют базис в пространстве разрезов связного графа, получаем $\mathcal{C}^*(G^*) \subseteq \mathcal{C}(G)$. Ортогональные дополнения антимонотонны: если $L_1 \subseteq L_2$, то $L_1^\perp \supseteq L_2^\perp$, поэтому выполняется также включение $\mathcal{C}(G^*) \supseteq \mathcal{C}^*(G)$.

Верно и обратное: если $\mathcal{C}^*(H) \subseteq \mathcal{C}(G)$ для графов $G = (V, E)$, $H = (I, E)$, то разрезы $E(i)$, $i \in I$, являются эйлеровыми циклами в G и образуют двойное покрытие циклами.

Доказательство теоремы 1. Предположим, что выполнена гипотеза CDC. Рассмотрим дерево $T = (V, E')$ и гиперграф $\mathcal{H}(E', E'')$, реализованный на этом дереве. Это означает, что каждому $e'' \in E''$ соответствует простой путь $\tau(e'')$ по дереву T. Построим граф $G(V, E)$, $E = E' \cup E''$, добавляя рёбра $e'' \in E''$. Ребру e'' соответствует пара $\{u, v\}$, где u и v — концы пути $\tau(e'')$. Таким образом, гиперрёбрам $\mathcal{H}$ соответствуют рёбра из E, не входящие в дерево. Так как изолированных вершин

[1)] Название оправдано тем, что в случае плоского графа и циклов, которые являются границами граней, получается обычная конструкция геометрически двойственного графа (см. [1, § 40, с. 169]).

в $\mathcal{H}(E', E'')$ нет, то в графе G нет мостов. Пусть $\{E_1, \ldots, E_k\}$ — двойное покрытие циклами рёбер графа G и G^* — соответствующий геометрически двойственный граф.

Двойственный гиперграф $\mathcal{H}^T(E'', E')$ реализуется на G^* обобщёнными путями. Для любого $e' \in E'$ множество C, состоящее из тех $e'' \in E''$, для которых $e' \in \mathcal{H}(e'')$, и самого e', порождает разрез в графе G. Так как $\mathcal{C}(G) \supseteq \mathcal{C}^*(G^*)$, из антимонотонности следует, что $\mathcal{C}(G^*) \supseteq \mathcal{C}^*(G)$. Значит, в геометрически двойственном графе C — обобщённый цикл. Поскольку гиперребро двойственного гиперграфа получается из этого цикла удалением e', оно является обобщённым путём в G^*.

С другой стороны, пусть выполнена гипотеза двойственности. В связном графе $G = (V, E)$ без мостов выберем остовное дерево T, для краткости так же обозначаем множество его рёбер. Поскольку между любыми двумя вершинами дерева есть ровно один простой путь и в G нет мостов, корректно определён гиперграф базисных путей $\mathcal{B}(T, E \setminus T)$, в котором $\mathcal{B}(e)$, где $e \in E \setminus T$, $e = \{u, v\}$, состоит из рёбер пути по T с теми же концами. Это сильно древесный гиперграф. Из гипотезы двойственности следует, что двойственный гиперграф $\mathcal{B}^T(E \setminus T, T)$ графический; пусть он реализуется на графе $H(C, E \setminus T)$. Расширим этот граф до графа $H'(C, E)$, добавляя ребро $e \in T$ с концами c_1, c_2 — вершинами нечётной степени в реализации e на графе H.

В дереве каждое ребро — мост, поэтому каждому ребру e остовного дерева отвечает разрез E_e в исходном графе (разбиение вершин, задаваемое разрезом в остовном дереве, задаваемым ребром e). Легко видеть, что такие разрезы образуют базис пространства разрезов, так как каждый содержит лишь одно ребро дерева.

В графе H' разрезы, отвечающие рёбрам остовного дерева, являются обобщёнными циклами, поэтому $\mathcal{C}^*(G) \subseteq \mathcal{C}(H')$ и $\mathcal{C}(G) \supseteq \mathcal{C}^*(H')$ (из антимонотонности ортогональных дополнений). Значит, 1-вершинные разрезы H' задают двойное покрытие циклами рёбер G. Теорема 1 доказана.

3. Усиления гипотезы CDC

Во второй части доказательства теоремы 1 остовное дерево в графе выбиралось произвольно. Это означает, что гипотеза CDC следует уже из графичности гиперграфов, двойственных к гиперграфам базисных путей для такого класса остовных деревьев, который реализуется в любом связном графе. Например, достаточно ограничиться деревьями обхода в глубину или деревьями обхода в ширину и т. п.

Деревья обхода в глубину корневые и обладают таким свойством: концы любого ребра графа лежат на некотором пути из корня в лист. Получаем равносильное CDC ослабление гипотезы двойственности: требуется графическая реализация лишь гиперграфов, двойственных к *монотонно*

сильно древесным гиперграфам — тем которые реализуются на корневом дереве подпутём пути из корня в лист.

Другое наблюдение состоит в том, что вторая часть доказательства теоремы не только строит геометрически двойственный граф $H'(C, E)$ для некоторого двойного покрытия циклами, но и разделяет его на части: на рёбра из остовного дерева T и на рёбра из дополнения. Последние задают реализацию $H(C, E \setminus T)$ двойственного гиперграфа. Это позволяет усилить гипотезу двойственности, требуя реализации двойственного гиперграфа на ограниченном классе графов.

Например, рассмотрим следующие классы графов:

- утолщения звезды $K_{1,4}$;
- утолщения пути P_6 длины 5;
- утолщения графа-вилки F_5 (дерево на 5 вершинах с вершиной степени 3).

Компьютерные эксперименты показывают, что для каждого из этих классов и для каждого графа с не более чем 26 вершинами существует такое остовное дерево, что гиперграф, двойственный к гиперграфу базисных циклов, реализуется в этом классе.

Это позволяет сформулировать три усиления гипотезы CDC в виде усиления гипотезы двойственности.

Гипотеза $K_{1,4}$. *В любом графе без мостов существует такое остовное дерево, что гиперграф, двойственный к гиперграфу базисных путей относительно этого дерева, реализуется на утолщении $K_{1,4}$.*

Гипотеза P_6. *В любом графе без мостов существует такое остовное дерево, что гиперграф, двойственный к гиперграфу базисных путей относительно этого дерева, реализуется на утолщении P_6.*

Гипотеза F_5. *В любом графе без мостов существует такое остовное дерево, что гиперграф, двойственный к гиперграфу базисных путей относительно этого дерева, реализуется на утолщении F_5.*

Заметим, что из гипотез $K_{1,4}$ и F_5 следует гипотеза 5-CDC.

4. Алгоритмическая сложность задач графической реализации

Назовём ГРАФ алгоритмическую задачу, в которой на входе дан гиперграф $(X, \mathcal{H})$ и нужно узнать, графический ли он. Аналогично на входе задачи С-ГРАФ дан гиперграф $(X, \mathcal{H})$ и нужно узнать, сильно графический ли он, т. е. существует ли его реализация на простом графе.

Для класса графов $\mathcal{G}$ алгоритмическая задача ГРАФ($\mathcal{G}$) имеет следующий вид. Дан гиперграф $(X, \mathcal{H})$. Нужно узнать, существует ли его реализация на утолщении графа $G \in \mathcal{G}$. Если класс состоит из одного графа G, то задачу обозначаем ГРАФ(G).

Для маленьких графов G задача ГРАФ(G) эффективно решается. Рассмотрим реализации на утолщениях графа K_2, т. е. на графах, у которых все рёбра имеют одинаковые концы. В этом случае очевидно, что реализация $(\mathcal{H}, X)$ существует тогда и только тогда, когда каждое гиперребро содержит нечётное число элементов. Это даёт и простой структурный критерий, и простой алгоритм решения ГРАФ(K_2), работающий за полиномиальное время.

Утверждение 1. *Задача* ГРАФ$(K_{1,2})$ = ГРАФ(P_3) *решается за полиномиальное время.*

Доказательство. Зададим реализацию гиперграфа $(X, \mathcal{H})$ на утолщении пути $P_3 = (\{1,2,3\}, \{\{1,2\},\{2,3\}\})$ двоичным вектором размерности $|X|$: $n_x = 0$, если вершина x реализована ребром с концами $\{1,2\}$, и $n_x = 1$, если x реализована ребром с концами $\{2,3\}$.

Степени вершин в подграфе, отвечающем ребру $H \in \mathcal{H}$, равны соответственно

$$d_1 = |H| - \sum_{x \in H} n_x, \quad d_2 = |H|, \quad d_3 = \sum_{x \in H} n_x.$$

Если $|H|$ нечётно, то ровно две степени будут нечётными, что и требуется для обобщённого пути. Если $|H|$ чётно, то и d_2 чётно, так что две другие степени обязаны быть нечётными. Таким образом, существование реализации равносильно разрешимости системы уравнений в поле $\mathbb{F}_2$ из двух элементов

$$\sum_{x \in H} n_x = 1, \quad H \in \mathcal{H},\ |H| \equiv 0 \pmod 2.$$

Разрешимость такой системы проверяется стандартными методами линейной алгебры за полиномиальное время. Утверждение 1 доказано.

Утверждение 2. *Для любого гиперграфа из существования реализации на утолщении P_4 следует существование реализации на утолщении P_3.*

Доказательство. Предположим, что имеется реализация гиперграфа $(X, \mathcal{H})$ на утолщении пути P_4 с множеством вершин $\{1,2,3,4\}$ и рёбрами, соединяющими различающиеся на 1 вершины. Обозначим через X_i множество вершин гиперграфа, соответствующих ребру $\{i, i+1\}$. Построим реализацию на утолщении пути P_3 с множеством вершин $\{1,2,3\}$ и рёбрами, соединяющими различающиеся на 1 вершины, по правилу: вершинам из $X_1 \cup X_3$ соответствует ребро $\{1,2\}$, а вершинам из X_2 соответствует ребро $\{2,3\}$.

Докажем, что такое соответствие задаёт реализацию. Для гиперребра $H \in \mathcal{H}$ обозначим через h_i число его вершин, сопоставленных ребру

$\{i, i+1\}$ в исходной реализации. Тогда в новом соответствии H отвечает подграф со степенями вершин $h_1 + h_3, h_1 + h_2 + h_3, h_2$. Нужно доказать, что среди этих чисел ровно два нечётных. Предположим противное. Сумма степеней вершин чётна, поэтому все указанные числа чётные, т. е. h_2 чётно, а h_1 и h_3 имеют одинаковую чётность. В реализации на P_4 гиперребру H отвечает подграф со степенями вершин h_1, $h_1 + h_2$, $h_2 + h_3$, h_3, и при сделанном предположении чётности всех этих чисел одинаковы, что противоречит условию реализации (ровно два из этих чисел нечётные). Утверждение 2 доказано.

Утверждение 3. *Задачи* ГРАФ($K_{1,3}$) *и* ГРАФ(C_3) NP-*полны.*

Доказательство. Сводимость NP-полной задачи о 3-раскраске к задаче ГРАФ($K_{1,3}$) сопоставляет графу G его сам, но рассматриваемый как гиперграф.

Рассмотрим простой граф $G = (V, E)$ как гиперграф и его графическую реализацию с носителем на утолщении звезды $K(\{0, 1, 2, \ldots, k\}, V)$, где рёбра звезды имеют вид $\{0, i\}$. Вершинам G соответствуют рёбра утолщённой звезды K. Получаем раскраску вершин G: если v соответствует ребро $\{0, i\}$, то красим v в цвет i. Рёбрам G отвечают простые пути длины 2 в K. Это означает, что построенная раскраска в k цветов правильная. Верно и обратное: по любой правильной k-раскраске получаем реализацию на утолщении звезды.

Заметим, что множества простых путей длины 2 в звезде $K_{1,3}$ и цикле C_3 одинаковы. Значит, описанная выше сводимость даёт также доказательство NP-полноты задачи ГРАФ(C_3). Утверждение 3 доказано.

Заметим, что гиперграфы в построенной сводимости являются простыми графами, поэтому всегда реализуются на утолщении некоторой звезды $K_{1,s}$ (используем раскраску в s цветов).

Теорема 2. *Задача* ГРАФ(дерево) NP-*полна.*

Доказательство. Строим сводимость задачи о 3-раскраске к задаче ГРАФ(дерево). По простому графу $G = (V, E)$ построим гиперграф $(X, \mathcal{H})$ на множестве $X = V \cup \{a, b, c_1, c_2, c_3\}$ размера $5 + |V|$. Гиперрёбра зададим так:

$$\{a, b\},\ \{a, c_i\},\ \{b, c_i\},\ \{c_i, c_j\}, \quad 1 \leqslant i, j \leqslant 3,\ i \neq j,$$
$$\{a, v\},\ \{b, v\}, \quad v \in V,$$
$$\{a, c_1, c_2, c_3, u, v\}, \quad \{u, v\} \in E.$$

Пусть $T(U, X)$ — реализация этого гиперграфа на утолщении дерева. Гиперребро размера 2 всегда реализуется простым путём, поэтому у рёбер $\pi(a)$, $\pi(b)$ есть ровно одна общая вершина, обозначим её u.

Если t_1t_2, t_2t_3, t_3t_1 — простые пути на утолщённом дереве, то у рёбер t_1, t_2, t_3 обязана быть общая вершина. Тройка простых путей $T(a)T(b)$, $T(a)T(c_i)$, $T(b)T(c_i)$, $1 \leqslant i \leqslant 3$, показывает, что u является концом $T(c_i)$. Тройка $T(a)T(b)$, $T(a)T(v)$, $T(b)T(v)$, $v \in V$, показывает, что u является концом $T(v)$. Вторые концы рёбер $T(a), T(b), T(c_i), T(v)$ обозначим через x_a, x_b, x_i, x_v соответственно; здесь $1 \leqslant i \leqslant 3$, $v \in V$. Вершины x_a, x_b, x_i различны, так как каждой паре гиперрёбер $\{a, b\}$, $\{a, c_i\}$, $\{b, c_i\}$ из $\mathcal{H}$ соответствует простой путь.

Пусть $\{p, q\} \in E$. Обобщённый путь

$$\{T(a), T(c_1), T(c_2), T(c_3), T(p), T(q)\}$$

содержит вершины x_a, x_i, $1 \leqslant i \leqslant 3$, степень хотя бы двух из них должна быть чётной. У x_a степень равна 1, так как $T(a)T(p)$, $T(a)T(q)$, $T(b)T(p)$, $T(b)T(q)$ — простые пути. Значит, $x_p, x_q \in \{x_1, x_2, x_3\}$ и $x_p \neq x_q$.

Покрасим неизолированную вершину p в цвет i, если $x_p = x_i$. Изолированные вершины красим произвольно. Получаем правильную раскраску графа G, так как $x_p \neq x_q$ для $\{p, q\} \in E$. И наоборот, любая 3-раскраска графа G даёт реализацию гиперграфа $(X, \mathcal{H})$, в которой $x_v = x_i$, где i — цвет v.

Таким образом, получаем искомую сводимость. Ясно, что она реализуется полиномиальным алгоритмом. Теорема 2 доказана.

Теорема 3. *Задача* ГРАФ(путь) NP-*полна.*

Доказательство. Задача о 4-раскраске графа также NP-полна: для сводимости 3-раскраски к 4-раскраске достаточно добавить к графу универсальную вершину, связанную со всеми остальными. Построим сводимость задачи о 4-раскраске к задаче ГРАФ(путь).

Ориентируем рёбра простого неориентированного графа (V, E) произвольным образом. Построим гиперграф $(X, \mathcal{H})$ на множестве

$$X = \{x, y\} \cup V \cup \{a_e, b_e \mid e \in E\}$$

размера $2 + |V| + 2|E|$. Его гиперрёбра зададим так: $\{x, y\}$,

$$\{x, y, v\}, \quad v \in V,$$
$$\{a_e, u\},\ \{b_e, v\},\ \{a_e, b_e, u, v\}, \quad e = \{u, v\} \in E.$$

В последнем случае нам требуется порядок на $\{u, v\}$. Считаем, что в выбранной ориентации u — начало ребра e, а v — конец.

Пусть этот гиперграф реализован на утолщении пути $P(U, X)$, где $U = \{1, 2, \ldots, k\}$, а рёбра пути имеют вид $\{i, i+1\}$, $1 \leqslant i < k$.

Поскольку $P(x)P(y)$ — простой путь длины 2, у рёбер $P(x)$, $P(y)$ есть ровно одна общая вершина, обозначим её d. Не ограничивая общности, считаем, что вторые концы этих рёбер суть $d-1$ и $d+1$ соответственно.

Так как $P(x)P(y)P(v)$ — обобщённый путь для любой $v \in V$, получаем для $P(v)$ четыре возможности: $\{d-2, d-1\}$, $\{d-1, d\}$, $\{d, d+1\}$, $\{d+1, d+2\}$. Покрасим вершину v в цвет $c(v)$, равный меньшему из концов $P(v)$.

Докажем, что если $e = \{p, q\} \in E$, то $c(p) \neq c(q)$. Достаточно проверить, что $P(p)$ и $P(q)$ имеют разные множества концов, так как концы $P(v)$, $v \in V$, различаются на 1. Предположим, напротив, что $P(p) = P(q) = \{i, i+1\}$. Так как $P(a_e)P(b_e)P(p)P(q)$ — обобщённый путь, а пары рёбер $P(a_e)$, $P(p)$ и $P(b_e)$, $P(q)$ имеют ровно один общий конец, получаем две возможности: либо оба ребра $P(a_e)$, $P(b_e)$ равны $\{i-1, i\}$ или $\{i+1, i+2\}$, либо одно из них — $\{i-1, i\}$, а другое — $\{i+1, i+2\}$. В первом случае в подграфе с рёбрами $P(a_e)$, $P(b_e)$, $P(p)$, $P(q)$ все вершины имеют чётную степень, во втором все вершины имеют нечётную степень. Тем самым этот подграф не является обобщённым путём; противоречие.

Обратно, пусть существует правильная раскраска $c\colon V \to \{1, 2, 3, 4\}$ графа G в 4 цвета. Построим реализацию гиперграфа $(X, \mathcal{H})$ на утолщении пути P_7 с множеством вершин $\{0, 1, \ldots, 6\}$ и рёбрами $\{i, i+1\}$, $0 \leqslant i < 6$. Полагаем $P(x) = \{2, 3\}$, $P(y) = \{3, 4\}$ и $P(v) = \{c(v), c(v)+1\}$ для $v \in V$. Для каждого ребра $e = \{p, q\}$ дополняем рёбра $P(p)$, $P(q)$ рёбрами $P(a_e)$, $P(b_e)$ до простого пути τ длины 4. С учётом условия $c(p) \neq c(q)$ есть три возможности:

1) если $|c(p) - c(q) = 1|$, то $P(a_e)$, $P(b_e)$ — висячие рёбра в τ;

2) если $|c(p) - c(q) = 2|$, то одно из рёбер $P(a_e)$, $P(b_e)$ висячее в τ, а другое лежит между $P(p)$ и $P(q)$;

3) если $|c(p) - c(q) = 3|$, то $P(p)$, $P(q)$ — висячие рёбра в τ, соединённые простым путём $P(a_e)P(b_e)$. Теорема 3 доказана.

Теорема 4. *Задача* ГРАФ NP-*полна.*

Доказательство. Применим ту же сводимость, что в теореме 2. Достаточно доказать, что в любой реализации вершинам a, b, c_i, $1 \leqslant i \leqslant 3$, гиперграфа соответствуют рёбра звезды $K_{1,5}$, а дальнейший анализ остаётся прежним.

Любая пара этих рёбер образует путь длины 2, поэтому вместе они образуют простой граф, в котором каждая пара рёбер инцидентна. Тройка попарно инцидентных рёбер может быть реализована как цикл длины 3. Однако уже четвёртое ребро не может быть инцидентно всем рёбрам этого цикла. Теорема 4 доказана.

Проверка сильной графичности также трудна, так как к ней сводится проверка графичности.

Теорема 5. *Задача* С-ГРАФ NP-*полна.*

Доказательство. Строим сводимость задачи ГРАФ к С-ГРАФ.

По гиперграфу $(X, \mathcal{H})$ построим гиперграф $(Y, \mathcal{K})$, где Y представляет собой дизъюнктное объединение двух копий X: $Y = X' \cup X''$, $X' \cap X'' = \varnothing$, а каждому элементу $x \in X$ соответствуют $x' \in X'$, $x'' \in X''$.

Множество гиперрёбер $\mathcal{K}$ включает в себя множества вида $\{x', x''\}$ для всех $x \in X$ и для каждого гиперребра $H \in \mathcal{H}$ обе его копии, т. е. множество $\{y \in \{x', x''\} \mid x \in H\}$.

Пусть $(X, \mathcal{H})$ реализуется на графе $G(V, E)$, возможно, с кратными рёбрами. Тогда $(Y, \mathcal{K})$ реализуется на простом графе G', полученном подразбиением рёбер графа G. Подразбиение ребра e с концами u, w состоит в том, что к множеству вершин графа добавляется новая вершина v_e, из множества рёбер удаляется ребро e и добавляются два ребра: e' (с концами u, v_e) и e'' (с концами v_e, w). После подразбиения всех рёбер кратных рёбер не остаётся. Если при реализации $(X, \mathcal{H})$ на G гипервершине x соответствует ребро $e \in E$, то реализация $(Y, \mathcal{K})$ сопоставляет паре x', x'' рёбра e', e'', полученные подразбиением e.

В обратную сторону: пусть $(Y, \mathcal{K})$ реализуется на простом графе $G = (V, E)$. Построим граф $G'(V', E')$, $V' \subseteq V$, по следующему правилу. Каждое гиперребро вида $\{x', x''\}$ реализуется как простой путь $v_0 v_1 v_2$. Вершины v_0, v_2 входят в V'. Каждая пара $\{v_0, v_2\}$ включается в E' (возможно, несколько раз). Других вершин и рёбер в G' нет. Вершине x гиперграфа $(X, \mathcal{H})$ сопоставим ребро G' с концами v_0, v_2 простого пути, реализующего $\{x', x''\}$ в $G = (V, E)$. Получаем реализацию $(X, \mathcal{H})$ на $G'(V', E')$. Теорема 5 доказана.

Заключение

Связь между графическими реализациями гиперграфов и гипотезой CDC позволяет сформулировать усиления гипотезы CDC, ограничивая классы графов для графических реализаций. Даже если эти гипотезы неверны, опровержение любой из них позволит лучше понять структуру двойных покрытий циклами.

Результаты об алгоритмической сложности проверки графичности гиперграфа получены для общих гиперграфов. Интересно установить их алгоритмическую сложность для сильно древесных гиперграфов.

Другой интересный открытый вопрос: можно ли для заданного класса деревьев (например, путей) указать один универсальный граф, на утолщениях которого реализуются все двойственные к гиперграфам путей на этом классе деревьев? Для путей ответ известен. В [11] доказана гипотеза CDC для графов, в которых есть гамильтонов путь. Конструкция из этой работы строит по любому гиперграфу, реализованному подпутями пути, реализацию двойственного гиперграфа на утолщении графа $P_2 \,\square\, P_3$ (декартово произведение путей на 2 и 3 вершинах, т. е. (2×3)-решётка). Это означает, что граф $P_2 \,\square\, P_3$ универсальный для путей.

Существуют ли универсальные графы для других классов деревьев, неизвестно.

Авторы благодарят рецензента за внимание к работе и ценные замечания.

ЛИТЕРАТУРА

1. **Емеличев В. А., Мельников О. И., Сарванов В. И., Тышкевич Р. И.** Лекции по теории графов. М.: Наука, 1990.
2. **Tarjan R. E., Yannakakis M.** Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs // SIAM J. Comput. 1984. V. 13, No. 3. P. 566–579.
3. **Buchin K., van Kreveld M., Meijer H., Speckmann B., Verbeek K.** On planar supports for hypergraphs // Graph drawing. Rev. Pap. 17th Int. Symp. (Chicago, USA, Sept. 22–25, 2009). Heidelberg: Springer, 2010. P. 345–356. (Lect. Notes Comput. Sci.; V. 5849).
4. **Brandes U., Cornelsen S., Pampel B., Sallaberry A.** Blocks of hypergraphs: Applied to hypergraphs and outerplanarity // Combinatorial algorithms. Rev. Sel. Pap. 21st Int. Workshop (London, UK, July 26–28, 2010). Heidelberg: Springer, 2011. P. 201–211. (Lect. Notes Comput. Sci.; V. 6460).
5. **Johnson D. S., Pollak H. O.** Hypergraph planarity and the complexity of drawing Venn diagrams // J. Graph Theory. 1987. V. 11, No. 3. P. 309–325.
6. **Brandes U., Cornelsen S., Pampel B., Sallaberry A.** Path-based supports for hypergraphs // Combinatorial algorithms. Rev. Sel. Pap. 21st Int. Workshop (London, UK, July 26–28, 2010). Heidelberg: Springer, 2011. P. 20–33. (Lect. Notes Comput. Sci.; V. 6460).
7. **Szekeres G.** Polyhedral decompositions of cubic graphs // Bull. Aust. Math. Soc. 1973. V. 8. P. 367–387.
8. **Seymour P. D.** Sums of circuits // Graph theory and related topics. New York: Acad. Press, 1979. P. 341–355.
9. **Zhang C. Q.** Integer flows and cycle covers of graphs. New York: Marcel Dekker, 1997.
10. **Дистель Р.** Теория графов. Новосибирск: Изд-во Ин-та математики, 2002. 336 с.
11. **Tarsi M.** Semi-duality and the cycle double cover conjecture // J. Comb. Theory. Ser. B. 1986. V. 41, No. 3. P. 332–340.

Вялый Михаил Николаевич
Карпов Владимир Евгеньевич

Статья поступила
21 ноября 2022 г.
После доработки —
27 февраля 2023 г.
Принята к публикации
3 марта 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
July–September 2023. Vol. 30, No. 3. P. 81–95

UDC 519.171 DOI 10.33048/daio.2023.30.757

HYPERGRAPH EDGE REPRESENTATIONS WITH THE USE OF HOMOLOGICAL PATHS

M. N. Vyalyi[1,2,3,a] and *V. E. Karpov*[1,b]

[1] Moscow Institute of Physics and Technology,
9 Institutskii Lane, 141701 Dolgoprudnyi, Russia

[2] Federal Research Center "Computer Science and Control" RAS
42 Vavilov Street, 119333 Moscow, Russia

[3] HSE University,
11 Pokrovskii Boulevard, 109028 Moscow, Russia

E-mail: [a] vyalyi@gmail.com, [b] ve.karpov@yandex.ru

Abstract. We consider a problem of realization of hypergraphs on graphs provided each hyperedge is realized by a subgraph in which exactly two vertices have odd degree. This problem is related to Cycle Double Cover conjecture. We prove that checking the existence of realization is computationally hard. The hardness is proved in various settings: for realizations on all graphs, on simple graphs, and on graphs from several restricted classes. Bibliogr. 11.

Keywords: hypergraph, cycle cover, Eulerian graph, NP-completeness.

REFERENCES

1. **V. A. Emelichev, O. I. Melnikov, V. I. Sarvanov,** and **R. I. Tyshkevich,** *Lectures on Graph Theory* (Nauka, Moscow, 1990 [Russian]; B. I. Wissenschaftsverlag, Mannheim, 1994).
2. **R. E. Tarjan** and **M. Yannakakis,** Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs, *SIAM J. Comput.* **13** (3), 566–579 (1984).

The work of the first author is supported by the HSE University Program for Basic Research and carried out within the framework of the state assignment of FRC CSC RAS (Project 0063–2019–0003).

English version: Journal of Applied and Industrial Mathematics **17** (3) (2023).

3. **K. Buchin, M. van Kreveld, H. Meijer, B. Speckmann,** and **K. Verbeek,** On planar supports for hypergraphs, in *Graph Drawing* (Rev. Pap. 17th Int. Symp., Chicago, USA, Sept. 22–25, 2009) (Springer, Heidelberg, 2010), pp. 345–356 (Lect. Notes Comput. Sci., Vol. 5849).
4. **U. Brandes, S. Cornelsen, B. Pampel,** and **A. Sallaberry,** Blocks of hypergraphs: Applied to hypergraphs and outerplanarity, in *Combinatorial Algorithms* (Rev. Sel. Pap. 21st Int. Workshop, London, UK, July 26–28, 2010) (Springer, Heidelberg, 2011), pp. 201–211 (Lect. Notes Comput. Sci., Vol. 6460).
5. **D. S. Johnson** and **H. O. Pollak,** Hypergraph planarity and the complexity of drawing Venn diagrams, *J. Graph Theory* **11** (3), 309–325 (1987).
6. **U. Brandes, S. Cornelsen, B. Pampel,** and **A. Sallaberry,** Path-based supports for hypergraphs, in *Combinatorial Algorithms* (Rev. Sel. Pap. 21st Int. Workshop, London, UK, July 26–28, 2010) (Springer, Heidelberg, 2011), pp. 20–33 (Lect. Notes Comput. Sci., Vol. 6460).
7. **G. Szekeres,** Polyhedral decompositions of cubic graphs, *Bull. Aust. Math. Soc.* **8**, 367–387 (1973).
8. **P. D. Seymour,** Sums of circuits, in *Graph Theory and Related Topics* (Acad. Press, New York, 1979), pp. 341–355.
9. **C. Q. Zhang,** *Integer Flows and Cycle Covers of Graphs* (Marcel Dekker, New York, 1997).
10. **R. Diestel,** *Graph Theory* (Springer, Heidelberg, 2000; Izd. Inst. Mat., Novosibirsk, 2002 [Russian]).
11. **M. Tarsi,** Semi-duality and the cycle double cover conjecture, *J. Comb. Theory, Ser. B,* **41** (3), 332–340 (1986).

Mikhail N. Vyalyi
Vladimir E. Karpov

Received November 21, 2022
Revised February 27, 2023
Accepted March 3, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль–сентябрь 2023. Т. 30, № 3. С. 96–110

УДК 519.8+518.25 DOI 10.33048/daio.2023.30.763

ПОЛИНОМИАЛЬНЫЕ АППРОКСИМАЦИОННЫЕ СХЕМЫ ДЛЯ ЗАДАЧ ВЫБОРА ВЕКТОРОВ И КЛАСТЕРИЗАЦИИ С РАЗНЫМИ ЦЕНТРАМИ

А. В. Пяткин

Институт математики им. С. Л. Соболева,
пр. Акад. Коптюга, 4, 630090 Новосибирск, Россия
E-mail: artempyatkin@gmail.com

Аннотация. Рассматриваются две близкие в постановочном плане задачи. Во-первых, общая задача кластеризации, т. е. разбиения множества d-мерных евклидовых векторов на заданное число кластеров с разными типами центров, при котором суммарная дисперсия будет минимальной. Под дисперсией понимается сумма квадратов расстояний между элементами кластера и его центром. При этом для одной части кластеров центр может быть выбран произвольно (очевидно, что в этом случае следует выбрать центроид), для другой части в качестве центра должен быть выбран один из векторов исходного множества, а для остальных кластеров центрами являются заранее заданные точки. Также на входе задаются размеры каждого кластера. Вторая рассматриваемая задача — это задача выбора подмножества векторов заданной мощности с минимальной суммой квадратов расстояний от его элементов до центроида. В статье построены полиномиальные аппроксимационные схемы (PTAS) для обеих задач. Библиогр. 23.

Ключевые слова: кластеризация, центроид, медоид, аппроксимация, PTAS-схема, задача MSSC.

Введение

В настоящей статье изучаются две задачи, связанные с кластеризацией множества векторов евклидова пространства на непустые подмножества похожих векторов и поиском в нём одного подмножества из наиболее похожих векторов. Такого рода задачи весьма актуальны для анализа и обработки данных, искусственного интеллекта, вычислительной геометрии, математической статистики и дискретной оптимизации [1–3].

Исследование выполнено в рамках государственного задания ИМ СО РАН (проект № FWNF–2022–0019).

По сути, это целый класс задач, отличающихся способами выбора критериев похожести, а также ограничений на число и размеры кластеров.

Одним из наиболее часто встречающихся критериев является минимизация дисперсии, под которой понимается сумма квадратов расстояний между элементами кластера и некоторой точкой, которая называется *центром* кластера. Определим для произвольного кластера $\mathcal{C}$ и его центра x функцию

$$f(\mathcal{C}, x) = \sum_{y \in \mathcal{C}} \|x - y\|^2.$$

Обычно рассматриваются следующие ограничения на выбор центра кластера:

а) произвольная точка (т. е. ограничений нет);

б) точка из исходного множества;

в) фиксированная (заданная) точка пространства.

Такой выбор центров можно обосновать следующим образом. Допустим, что в заданной географической области имеются несколько городков, в которых нужно мониторить ситуацию с помощью сенсоров. Некоторые сенсоры автономны, и их можно поместить куда угодно. Другие сенсоры требуют постоянного обслуживания людьми, и поэтому их следует помещать в поселениях (городках). Также в области могут иметься уже установленные ранее сенсоры (неважно каких типов), которые нельзя перемещать, но можно также использовать для мониторинга. Поскольку расход энергии пропорционален квадрату расстояния от сенсора до объекта, задачу кластеризации с разными типами сенсоров можно интерпретировать как проблему размещения сенсоров двух имеющихся типов с учётом ранее установленных сенсоров, минимизируя общий расход энергии.

Если все векторы из кластера $\mathcal{C}$ известны и центр кластера можно выбрать произвольно, то нетрудно показать с помощью частных производных, что оптимальным выбором будет так называемый *центроид*, определяемый как среднее значение всех векторов из кластера $\mathcal{C}$:

$$\overline{y}(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{y \in \mathcal{C}} y.$$

Заметим, что если это ограничение задано для всех центров кластеров, то получается классическая задача MSSC (minimum sum-squared clustering) [4, 5]. Если центр кластера должен быть выбран среди точек исходного множества, то такой центр назовём *медоидом*. Отметим, что термин медоид был введён в работе [6] как представитель кластера, чьё среднее отличие от остальных объектов кластера минимально. Хотя медоиды обычно ассоциируются с задачами, в которых используется сумма

расстояний, применение этого же термина для задач с квадратами расстояний не противоречит исходному определению. Наконец, третий тип центров кластеров (заданная точка пространства) будем называть *фиксированным*.

Основным объектом изучения является

Задача 1 (p центроидов, q медоидов, r фиксированных центров)**.** *Дано конечное множество векторов* $\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$, *множество точек* $\{z_1, \ldots, z_r\} \subset \mathbb{R}^d$ *и положительные целые числа* $m_1, \ldots, m_k$, *удовлетворяющие соотношениям* $m_1 + \ldots + m_k = n$ *и* $p+q+r = k$. *Найти разбиение множества* $\mathcal{Y}$ *на* k *кластеров* $\mathcal{C}_1, \ldots, \mathcal{C}_k$, *где* $|\mathcal{C}_i| = m_i$ *для всех* $i = 1, \ldots, k$ *такое, что значение*

$$F(\mathcal{C}_1, \ldots, \mathcal{C}_k; X) = \sum_{i=1}^{k} f(\mathcal{C}_i, x_i)$$

было бы минимально, где $X = \{x_1, \ldots, x_k\}$ *— множество центров кластеров, удовлетворяющих следующим ограничениям:*

- $x_i = \overline{y}(\mathcal{C}_i)$ *при* $1 \leqslant i \leqslant p$;
- $x_i \in \mathcal{Y}$ *при* $p + 1 \leqslant i \leqslant p + q$;
- $x_i = z_{i-p-q}$ *при* $p + q + 1 \leqslant i \leqslant k$.

Другими словами, центры первых p кластеров должны быть центроидами, центры следующих q кластеров — медоидами, а центры последних r кластеров — фиксированными точками $z_1, \ldots, z_r$.

Известно, что эта задача NP-трудна даже в случае $p \geqslant 1$ и $k = 2$. Действительно, при $p = 2$ и $q = r = 0$ получаем задачу MSSC с $k = 2$, NP-трудность которой была доказана в [7] для случая нефиксированных размеров кластеров (очевидно, что если вариант задачи с фиксированными размерами кластеров полиномиально разрешим, то и задача с нефиксированными размерами кластеров также полиномиально разрешима с помощью полного перебора всех размеров; отсюда NP-трудность задачи с нефиксированными мощностями кластеров влечёт NP-трудность задачи с фиксированными мощностями). Первая полиномиальная аппроксимационная схема (PTAS) для задачи MSSC с нефиксированными размерами кластеров при $k = 2$ была предложена в [8]; она находит $(1 + \varepsilon)$-приближённое решение за время $O(n(1/\varepsilon)^d)$. В работе [9] предложена PTAS-схема для той же задачи с произвольным числом кластеров k, находящая $(1+\varepsilon)$-приближённое решение за время $O(dn2^{(k/\varepsilon)^{O(1)}})$. Обе эти PTAS-схемы рандомизированные.

Случай $p = r = 1$, $q = 0$ был рассмотрен в [10–13]. NP-трудность задачи была сначала доказана для фиксированных [10, 11], а затем для нефиксированных [12, 13] размеров кластеров. Для обоих вариантов задачи известны 2-приближённые алгоритмы [14, 15] временной сложности

$O(n^2d)$. Кроме того, в [16] для задачи 1 с $p = r = 1$ и $q = 0$ построена PTAS-схема трудоёмкости $O(dn^{1+2/\varepsilon}(9/\varepsilon)^{3/\varepsilon})$.

Наконец, NP-трудность задачи 1 в случае $p = q = 1$, $r = 0$ была доказана в [17] для фиксированных размеров кластеров, а в [18] — для нефиксированных. В [18] также предложен 2-приближённый алгоритм трудоёмкости $O(n^2d + n^3)$ для последней задачи. Что касается случая $p = 0$, то он является полиномиально разрешимым для любого k, как показано в следующем разделе.

Основным результатом настоящей статьи является построение PTAS-схемы для общего случая задачи 1. Ранее для случая $p > 0$ и $q > 0$ таких схем известно не было.

Заметим, что в [19] была построена PTAS-схема для следующей близкой по смыслу задачи выбора подмножества похожих векторов.

Задача 2 (выбор подмножества векторов)**.** *Дано множество векторов* $\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$ *и положительное целое* $m \leqslant n$. *Найти подмножество* $\mathcal{C} \subseteq \mathcal{Y}$ *размера* m, *минимизирующее функцию* $f(\mathcal{C}, \overline{y}(\mathcal{C}))$.

Предложенная в [19] полиномиальная схема находит $(1 + \varepsilon)$-приближённое решение задачи 2 за время $O(dn^{1+2/\varepsilon}(9/\varepsilon)^{3/\varepsilon})$. Отметим, что прямое применение указанной схемы для решения задачи 1 не позволяет построить для неё PTAS-схему, что может быть проиллюстрировано на следующем простом примере. Пусть $d = 1$ и $\mathcal{Y}$ содержит четыре числа $y_1 = -2$, $y_2 = y_3 = 0$ и $y_4 = 2$. Пусть также $p = q = 1$, $r = 0$ и $m_1 = m_2 = 2$. Очевидно, что при достаточно малом $\varepsilon > 0$ PTAS-схема найдёт решение $\mathcal{C} = \{y_2, y_3\}$ (которое является оптимальным решением задачи 2 с таким множеством $\mathcal{Y}$ и $m = 2$). Тогда $F(\mathcal{C}, \mathcal{Y} \setminus \mathcal{C}; \{0, 0\}) = 8$ (в качестве центра второго кластера берётся точка y_2), в то время как оптимальным решением является $\mathcal{C}_1 = \{y_1, y_2\}$ и $\mathcal{C}_2 = \{y_3, y_4\}$, поскольку $F(\mathcal{C}_1, \mathcal{C}_2; \{-1, 0\}) = 6$. Таким образом, при малом $\varepsilon > 0$ относительная погрешность остаётся равной $1/3$. Однако, используя одно из геометрических свойств, доказанных в [19], можно предложить иную технику, которая позволяет построить как PTAS-схему для задачи 1, так и лучшую PTAS-схему для задачи 2.

Статья имеет следующую структуру. В разд. 1 приводятся предварительные результаты, а также улучшенная PTAS-схема для задачи 2. Основной результат статьи (а именно, PTAS-схема для задачи 1) доказывается в разд. 2. Заключение завершает статью. Отметим, что укороченная версия данной статьи (без улучшенной PTAS-схемы для задачи 2) подавалась в виде тезисов на конференцию MOTOR-2023.

1. Предварительные результаты

В данном разделе приводятся предварительные результаты, необходимые для обоснования PTAS-схемы. Отметим, что некоторые результаты известны (фольклорны), но их строгое доказательство приводится в тексте для полноты изложения.

Прежде всего покажем, что задача 1 с k фиксированными центрами полиномиально разрешима для любого заданного k. Это известный фольклорный результат, доказательство которого можно найти, например, в [20]. Однако, приведённое ниже доказательство более короткое, хотя и основано на той же идее. Через $x(j)$ обозначим j-ю координату вектора x.

Утверждение 1 [20]. *В случае $p = q = 0$ задача 1 разрешима за время $O(n^2d + n^3)$.*

Доказательство. Пусть $z_1, \ldots, z_r$ и $m_1, \ldots, m_r$ являются центрами и мощностями искомых кластеров соответственно. Заметим, что $m_1 + \ldots + m_r = n$. Для каждого $i = 1, \ldots, r$ зададим вектор $a_i \in \mathbb{R}^n$ по правилу $a_i(j) = \|z_i - y_j\|^2$ для всех $j = 1, \ldots, n$. Возьмём m_i копий вектора a_i для каждого $i = 1, \ldots, r$ и составим из этих n векторов квадратную матрицу A размера n. Тогда, очевидно, $F(\mathcal{Y})$ равно оптимальному решению задачи о назначениях для матрицы A. Поскольку задача о назначениях решается классическим Венгерским методом [21] за время $O(n^3)$, а построение матрицы A требует $O(n^2d)$ операций, исходная задача разрешима за время $O(n^2d + n^3)$. Утверждение 1 доказано.

Заметим, что если все элементы матрицы A являются целыми положительными числами, не превосходящими некоторой константы C, то задачу о назначениях можно решить быстрее с помощью алгоритма из работы [22], а именно, за время $O(M\sqrt{n}\log(nC))$, где через M обозначено число ненулевых элементов матрицы A. Однако, в нашем случае этот алгоритм может работать хуже, поскольку $M = n^2 - n$, и даже в случае целочисленных координат векторов, константа C может оказаться больше чем 2^n.

Также заметим, что из утверждения 1 вытекает полиномиальная разрешимость задачи 1 в случае $p = 0$. Действительно, задача о поиске q медоидов и r фиксированных центров сводится к решению не более чем n^q задач с $q + r$ фиксированными центрами (полным перебором всех вариантов выбора медоидов).

Ключевую роль играет

Лемма 1. *Пусть $\mathcal{C} \subset \mathcal{Y}$ — произвольный кластер мощности m. Тогда для любого положительного целого $t \leqslant m$ существует подмножество*

$\mathcal{T}^* \subset \mathcal{C}$ *мощности* t *такое, что*

$$\|\overline{y}(\mathcal{C}) - \overline{y}(\mathcal{T}^*)\|^2 \leqslant \frac{f(\mathcal{C}, \overline{y}(\mathcal{C}))}{mt}.$$

Доказательство. Рассмотрим функцию $g(\mathcal{T}) = \|\overline{y}(\mathcal{C}) - \overline{y}(\mathcal{T})\|^2$. Выберем произвольное подмножество $\mathcal{T} = \{y_1, \ldots, y_t\}$. Тогда

$$g(\mathcal{T}) = \left\| \overline{y}(\mathcal{C}) - \frac{1}{t}\sum_{i=1}^{t} y_i \right\|^2 = \frac{1}{t^2}\left\| t\overline{y}(\mathcal{C}) - \sum_{i=1}^{t} y_i \right\|^2 = \frac{1}{t^2}\left\| \sum_{i=1}^{t} (\overline{y}(\mathcal{C}) - y_i) \right\|^2$$
$$= \frac{1}{t^2}\Bigg(\sum_{y \in \mathcal{T}} \|\overline{y}(\mathcal{C}) - y\|^2 + \sum_{y \in \mathcal{T}} \sum_{\substack{y' \in \mathcal{T}, \\ y' \neq y}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle \Bigg).$$

Значит, среднее значение функции $g(\mathcal{T})$ по всем подмножествам $\mathcal{T} \subset \mathcal{C}$ мощности t равно

$$\frac{1}{\binom{m}{t} t^2} \sum_{\substack{\mathcal{T} \subset \mathcal{C}, \\ |\mathcal{T}| = t}} \Bigg(\sum_{y \in \mathcal{T}} \|\overline{y}(\mathcal{C}) - y\|^2 + \sum_{y \in \mathcal{T}} \sum_{\substack{y' \in \mathcal{T}, \\ y' \neq y}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle \Bigg).$$

Поскольку каждый $y \in \mathcal{C}$ лежит ровно в $\binom{m-1}{t-1}$ подмножествах $\mathcal{T} \subset \mathcal{C}$ размера t, имеем

$$\sum_{\substack{\mathcal{T} \subset \mathcal{C}, \\ |\mathcal{T}| = t}} \sum_{y \in \mathcal{T}} \|\overline{y}(\mathcal{C}) - y\|^2 = \binom{m-1}{t-1} \sum_{y \in \mathcal{C}} \|\overline{y}(\mathcal{C}) - y\|^2$$
$$= \binom{m-1}{t-1} f(\mathcal{C}, \overline{y}(\mathcal{C})). \quad (1)$$

Очевидно, что для любого $z \in \mathbb{R}^d$ выполняется тождество

$$\sum_{y \in \mathcal{C}} \langle \overline{y}(\mathcal{C}) - y, z \rangle = \left\langle m\overline{y}(\mathcal{C}) - \sum_{y \in \mathcal{C}} y,\, z \right\rangle = 0,$$

поэтому

$$0 = \sum_{y \in \mathcal{C}} \sum_{y' \in \mathcal{C}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle = \sum_{y \in \mathcal{C}} \|\overline{y}(\mathcal{C}) - y\|^2 + \sum_{y \in \mathcal{C}} \sum_{\substack{y' \in \mathcal{C}, \\ y' \neq y}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle,$$

а значит,

$$\sum_{y \in \mathcal{C}} \sum_{\substack{y' \in \mathcal{C}, \\ y' \neq y}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle = -\sum_{y \in \mathcal{C}} \|\overline{y}(\mathcal{C}) - y\|^2.$$

Отметим, что каждая упорядоченная пара $y, y' \in \mathcal{C}$, в которой $y \neq y'$, принадлежит ровно $\binom{m-2}{t-2}$ подмножествам $\mathcal{T} \subset \mathcal{C}$ размера t. Следовательно,

$$\sum_{\substack{\mathcal{T}\subset\mathcal{C},\\ |\mathcal{T}|=t}} \sum_{y\in\mathcal{T}} \sum_{\substack{y'\in\mathcal{T},\\ y'\neq y}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle = \binom{m-2}{t-2} \sum_{y\in\mathcal{C}} \sum_{\substack{y'\in\mathcal{C},\\ y'\neq y}} \langle \overline{y}(\mathcal{C}) - y,\, \overline{y}(\mathcal{C}) - y' \rangle$$
$$= -\binom{m-2}{t-2} \sum_{y\in\mathcal{C}} \|\overline{y}(\mathcal{C}) - y\|^2 = -\binom{m-2}{t-2} f(\mathcal{C}, \overline{y}(\mathcal{C})). \quad (2)$$

Объединяя (1) и (2), получим формулу для среднего значения функции $g(\mathcal{T})$ по всем подмножествам мощности t:

$$\frac{\binom{m-1}{t-1} - \binom{m-2}{t-2}}{\binom{m}{t} t^2} f(\mathcal{C}, \overline{y}(\mathcal{C})) = \left(\frac{1}{mt} - \frac{t-1}{(m-1)mt}\right) f(\mathcal{C}, \overline{y}(\mathcal{C})$$
$$= \frac{(m-t)}{(m-1)mt} f(\mathcal{C}, \overline{y}(\mathcal{C})).$$

Поскольку минимальное значение функции $g(\mathcal{T})$ не превосходит её среднего значения, найдётся подмножество $\mathcal{T}^* \subset \mathcal{C}$ размера t, для которого выполняется неравенство

$$g(\mathcal{T}^*) \leqslant \frac{(m-t)}{(m-1)mt} f(\mathcal{C}, \overline{y}(\mathcal{C})) \leqslant \frac{f(\mathcal{C}, \overline{y}(\mathcal{C}))}{mt}.$$

Лемма 1 доказана.

Следующее утверждение было доказано в [19].

Утверждение 2 [19]**.** *Для любого кластера $\mathcal{C} \subset \mathcal{Y}$ мощности m и точки $z \in \mathbb{R}^d$ выполняется тождество*

$$f(\mathcal{C}, z) = f(\mathcal{C}, \overline{y}(\mathcal{C})) + m\|\overline{y}(\mathcal{C}) - z\|^2.$$

Из утверждения 2 и леммы 1 следует основной инструмент для построения PTAS-схемы для обеих рассматриваемых задач.

Лемма 2. *Обозначим через $\mathcal{Z}$ множество центроидов всех кластеров мощности не более t в множестве $\mathcal{Y}$. Тогда для любого кластера $\mathcal{C} \subset \mathcal{Y}$ найдётся такая точка $z \in \mathcal{Z}$, что $f(\mathcal{C}, z) \leqslant (1 + 1/t) f(\mathcal{C}, \overline{y}(\mathcal{C}))$.*

Доказательство. Рассмотрим произвольный кластер $\mathcal{C} \subset \mathcal{Y}$ и положим $m = |\mathcal{C}|$. Ясно, что если $m \leqslant t$, то множество $\mathcal{Z}$ содержит центроид кластера $\mathcal{C}$ и можно положить $z = \overline{y}(\mathcal{C})$. В противном случае рассмотрим для кластера $\mathcal{C}$ подмножество $\mathcal{T}^*$ мощности t, существование которого

доказано в лемме 1. Тогда точка $z = \overline{y}(\mathcal{T}^*)$ лежит в $\mathcal{Z}$, и по утверждению 2 и лемме 1 получаем

$$f(\mathcal{C}, z) = f(\mathcal{C}, \overline{y}(\mathcal{C})) + m\|\overline{y}(\mathcal{C}) - z\|^2 \leqslant \left(1 + \frac{1}{t}\right) f(\mathcal{C}, \overline{y}(\mathcal{C})).$$

Лемма 2 доказана.

Таким образом, множество $\mathcal{Z}$, построенное в лемме 2, содержит не более чем $2n^t$ кандидатов на роль примерного центра каждого из первых p кластеров в задаче 1. Более того, для каждого такого кластера $\mathcal{C}$ по крайней мере один из этих центров даёт $(1+\varepsilon)$-приближённое решение для $f(\mathcal{C}, \overline{y}(\mathcal{C}))$, где $\varepsilon = 1/t$.

Лемма 2 также позволяет построить новую PTAS-схему для задачи 2 с лучшей оценкой трудоёмкости, чем в [19]. Действительно, рассмотрим следующий простой алгоритм.

Алгоритм 1

Вход: множество векторов $\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$, размер кластера $m \leqslant n$ и положительный целый параметр $t \leqslant m$.

Шаг 0. Положим $\mathcal{Z} = \varnothing$ и рекорд $R = \infty$.

Шаг 1. Рассмотрим все подмножества $\mathcal{T} \subset \mathcal{Y}$ размера t и добавим в $\mathcal{Z}$ центроид $\overline{y}(\mathcal{T})$ каждого из рассмотренных подмножеств.

Шаг 2. Для каждого $z \in \mathcal{Z}$ рассмотрим подмножество $\mathcal{C}_z$, состоящее из m ближайших к z векторов из $\mathcal{Y}$. Если $f(\mathcal{C}_z, z) < R$, то обновляем рекорд $R := f(\mathcal{C}_z, z)$.

Шаг 3. Обозначим через z^* и $\mathcal{C}_{z^*}$ точку и подмножество, на которых достигается рекорд R.

Выход: точка z^* и множество $\mathcal{C}_{z^*}$.

Теорема 1. *Для любого $\varepsilon > 0$ алгоритм 1 находит $(1 + \varepsilon)$-приближённое решение для задачи 2 за время $O(n^{1+1/\varepsilon}d)$.*

Доказательство. Обозначим через $\mathcal{C}^*$ оптимальный кластер в задаче 2 и положим $t = \lceil 1/\varepsilon \rceil$. Тогда по лемме 2 существует такая точка $z' \in \mathcal{Z}$, что $f(\mathcal{C}^*, z') \leqslant (1 + 1/t) f(\mathcal{C}^*, \overline{y}(\mathcal{C}^*))$. Обозначим через $\mathcal{C}'$ подмножество из m ближайших к z' векторов из $\mathcal{Y}$ (напомним, что это подмножество рассматривалось на шаге 2 алгоритма 1). Тогда

$$f(\mathcal{C}_{z^*}, \overline{y}(\mathcal{C}_{z^*})) \leqslant f(\mathcal{C}_{z^*}, z^*) \leqslant f(\mathcal{C}', z') \leqslant f(\mathcal{C}^*, z') \leqslant (1 + \varepsilon) f(\mathcal{C}^*, \overline{y}(\mathcal{C}^*)),$$

так как $\varepsilon \geqslant 1/t$.

Множество $\mathcal{Z}$ содержит не более чем n^t элементов. Поиск m ближайших к точке $z \in \mathcal{Z}$ векторов из $\mathcal{Y}$ занимает $O(nd)$ операций (детали можно найти, например, в [23]). Значит, временная сложность алгоритма 2 для решения задачи 2 не превосходит $O(dn^{1+1/\varepsilon})$. Теорема 1 доказана.

2. Основной результат

Предлагаемая PTAS-схема для решения задачи 1 действует следующим образом. После выбора параметра t строится множество $\mathcal{Z}$, содержащее центроиды всех подмножеств мощности не более t в $\mathcal{Y}$. Далее множество $\mathcal{Z}$ рассматривается как множество кандидатов на роль приближённого центра каждого из первых p кластеров (для которых центрами являются центроиды) в задаче 1, множество $\mathcal{Y}$ — как множество возможных центров для следующих q кластеров (медоиды) и одноэлементные множества $\{z_j\}$ — как множества центров оставшихся r кластеров (с фиксированными центрами). Далее рассматриваются все возможные способы выбора по одному центру для каждого кластера из соответствующего ему множества кандидатов, и для каждого такого выбранного набора центров решается задача с k фиксированными центрами с помощью утверждения 1. Тогда наилучшее из найденных решений является $(1+\varepsilon)$-приближённым решением задачи 1.

Приведём теперь формальное описание аппроксимационной схемы.

Алгоритм 2

Вход: множество векторов $\mathcal{Y} = \{y_1, \ldots, y_n\} \subset \mathbb{R}^d$, множество точек (фиксированных центров) $\{z_1, \ldots, z_r\} \subset \mathbb{R}^d$, положительные целые числа $p, q, r, m_1, \ldots, m_k$, удовлетворяющие соотношениям $m_1 + \ldots + m_k = n$ и $p + q + r = k$, а также положительный целый параметр t.

Шаг 0. Положим $\mathcal{Z} = \varnothing$ и рекорд $R = \infty$.

Шаг 1. Рассмотрим каждое подмножество $\mathcal{T} \subset \mathcal{Y}$ мощности не более t и добавим к $\mathcal{Z}$ его центроид $\overline{y}(\mathcal{T})$.

Шаг 2. Положим $\mathcal{Z}_i = \mathcal{Z}$ для всех $i = 1, \ldots, p$, а также $\mathcal{Z}_i = \mathcal{Y}$ для всех $i = p+1 \ldots, p+q$ и $\mathcal{Z}_i = \{z_{i-p-q}\}$ для всех $i = p+q+1, \ldots, k$.

Шаг 3. Рассмотрим все возможные способы выбора точек $x_i \in \mathcal{Z}_i$ для каждого $i = 1, \ldots, k$ и для каждого такого набора точек решим задачу 1 с k фиксированными центрами для множества $\mathcal{Y}$ с помощью алгоритма из утверждения 1. Обозначим через $\mathcal{C}_1, \ldots, \mathcal{C}_k$ найденные кластеры и положим $X = \{x_1, \ldots, x_k\}$. Если $F(\mathcal{C}_1, \ldots, \mathcal{C}_k; X) < R$, то обновляем рекорд $R := F(\mathcal{C}_1, \ldots, \mathcal{C}_k; X)$.

Шаг 4. Обозначим через $\mathcal{C}_1^A, \ldots, \mathcal{C}_k^A$ и $X^A = \{x_1, \ldots, x_k\}$ кластеры множество центров, на которых достигается рекорд R.

Выход: кластеры $\mathcal{C}_1^A, \ldots, \mathcal{C}_k^A$ и $X^A = \{x_1, \ldots, x_k\}$.

Осталось доказать корректность и оценить трудоёмкость алгоритма 2.

Теорема 2. *Алгоритм* 2 *находит* $(1+\varepsilon)$*-приближённое решение задачи* 1 *за время* $O((d+n)n^{q+2+p/\varepsilon})$, *где* $\varepsilon = 1/t$.

Доказательство. Докажем корректность алгоритма. Пусть кластеры $\mathcal{C}_1^A, \ldots, \mathcal{C}_k^A$ и множество их центров $X^A = \{x_1, \ldots, x_k\}$ получены на выходе алгоритма 2. Обозначим через $\mathcal{C}_1^*, \ldots, \mathcal{C}_k^*$ и $X^* = \{x_1^*, \ldots, x_k^*\}$ набор кластеров и их центров в оптимальном решении задачи 1. По лемме 2 для каждого $i = 1, \ldots, p$ существует такая точка $x_i' \in \mathcal{Z}_i$, что $f(\mathcal{C}_i^*, x_i') \leqslant (1+\varepsilon) f(\mathcal{C}_i^*, x_i^*)$. Положим $x_i' = x_i^*$ для всех $i = p+1, \ldots, k$, и пусть $X' = \{x_1', \ldots, x_k'\}$. Тогда, очевидно, что

$$F(\mathcal{C}_1^*, \ldots, \mathcal{C}_k^*; X') \leqslant (1+\varepsilon) F(\mathcal{C}_1^*, \ldots, \mathcal{C}_k^*; X^*). \tag{3}$$

Заметим, что набор центров X' рассматривался на шаге 3 алгоритма 2. Обозначим через $\mathcal{C}_1', \ldots, \mathcal{C}_k'$ множество кластеров, найденных для набора X' на шаге 3. Поскольку этот набор является оптимальным решением задачи с k фиксированными центрами (заданными множеством X') для множества $\mathcal{Y}$, а разбиение $\mathcal{C}_1^A, \ldots, \mathcal{C}_k^A$ и множество центров X^A — наилучшее решение, найденное алгоритмом 2, имеют место следующие неравенства:

$$F(\mathcal{C}_1^A, \ldots, \mathcal{C}_k^A; X^A) \leqslant F(\mathcal{C}_1', \ldots, \mathcal{C}_k'; X') \leqslant F(\mathcal{C}_1^*, \ldots, \mathcal{C}_k^*; X'). \tag{4}$$

Объединяя (3) и (4), получаем требуемое соотношение

$$F(\mathcal{C}_1^A, \ldots, \mathcal{C}_k^A; X^A) \leqslant (1+\varepsilon) F(\mathcal{C}_1^*, \ldots, \mathcal{C}_k^*; X^*).$$

Оценим трудоёмкость алгоритма. Очевидно, что шаги 0, 2 и 4 выполняются за константное время, а шаг 1 алгоритма 2 требует $O(dn^t)$ операций. Наиболее трудоёмким является шаг 3, на котором для каждого из возможных наборов центров решается задача с фиксированными центрами. Поскольку всего имеется не более $(2n^t)^p n^q$ возможных способов выборов центров, а решение задачи с k фиксированными центрами требует $O(n^2 d + n^3)$ операций по утверждению 1, трудоёмкость алгоритма 2 равна $O((d+n)n^{pt+q+2})$. Теорема 2 доказана.

Заключение

Основным результатом статьи является полиномиальная аппроксимационная схема (PTAS) трудоёмкости $O((d+n)n^{q+2+p/\varepsilon})$ для достаточно общей задачи кластеризации, в которой центры первых p кластеров являются центроидами, следующих q кластеров — медоидами, а центры оставшихся r кластеров фиксированы (числа p, q и r известны заранее, т. е. не являются частью входа). Мощности искомых кластеров составляют часть входа. Ключевая идея схемы заключается в построении не слишком большого множества кандидатов на роли центроидов, содержащего хорошее приближение для центров всех таких кластеров. Схема рассматривает все возможные комбинации выбора центров и решает

для каждого набора задачу с k фиксированными центрами сведением последней к задаче о назначениях.

В качестве дополнительного результата получена улучшенная PTAS-схема для задачи выбора подмножества векторов.

Автор благодарит рецензента за тщательное изучение работы и сделанные замечания, позволившие улучшить её представление.

ЛИТЕРАТУРА

1. **Berkhin P.** A survey of clustering data mining techniques // Grouping multidimensional data: Recent advances in clustering. Heidelberg: Springer, 2006. P. 25–71.
2. **Jain A. K., Dubes R. C.** Algorithms for clustering data. Englewood Cliffs, NJ: Prentice Hall, 1988.
3. **Ghoreyshi S., Hosseinkhani J.** Developing a clustering model based on K-means algorithm in order to creating different policies for policyholders // Int. J. Adv. Comput. Sci. Inf. Tech. 2015. V. 4, No. 2. P. 46–53.
4. **Fisher W. D.** On grouping for maximum homogeneity // J. Am. Stat. Assoc. 1958. V. 53, No. 284. P. 789–798.
5. **MacQueen J.** Some methods for classification and analysis of multivariate observations // Proc. 5th Berkeley Symp. Mathematics, Statistics and Probability (Berkeley, USA, June 21 — July 18, 1965; Dec. 27, 1965 — Jan. 7, 1966). V. 1. Berkeley: Univ. California Press, 1967. P. 281–297.
6. **Kaufman L., Rousseeuw P. J.** Clustering by means of medoids // Statistical data analysis based on the L_1-norm. Amsterdam: North-Holland, 1987. P. 405–416.
7. **Aloise D., Deshpande A., Hansen P., Popat P.** NP-hardness of Euclidean sum-of-squares clustering // Mach. Learn. 2009. V. 75, No. 2. P. 245–248.
8. **Inaba M., Katoh N., Imai H.** Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering // Proc. 10th Annu. Symp. Computational Geometry (Stony Brook, NY, USA, June 6–8, 1994). New York: ACM Press, 1994. P. 332–339.
9. **Kumar A., Sabharwal Y., Sen S.** A simple linear time $(1+\varepsilon)$-approximation algorithm for geometric k-means clustering in any dimensions // Proc. 45th Annu. IEEE Symp. Foundations of Computer Science (Rome, Italy, Oct. 17–19, 2004). Los Alamitos, CA: IEEE Comp. Soc., 2004. P. 454–462.
10. **Бабурин А. Е., Гимади Э. Х., Глебов Н. И., Пяткин А. В.** Задача отыскания подмножества векторов с максимальным суммарным весом // Дискрет. анализ и исслед. операций. Сер. 2. 2007. Т. 14, № 1. С. 32–42.
11. **Гимади Э. Х., Кельманов А. В., Кельманова М. А., Хамидуллин С. А.** Апостериорное обнаружение в числовой последовательности квазипериодического фрагмента при заданном числе повторов // Сиб. журн. индустр. математики. 2006. Т. 9, № 1. С. 55–74.

12. Кельманов А. В., Пяткин А. В. О сложности одного из вариантов задачи выбора подмножества «похожих» векторов // Докл. Акад. наук. 2008. Т. 421, № 5. С. 590–592.

13. Кельманов А. В., Пяткин А. В. Об одном варианте задачи выбора подмножества векторов // Дискрет. анализ и исслед. операций. 2008. Т. 15, № 5. С. 20–34.

14. Долгушев А. В., Кельманов А. В. Приближённый алгоритм решения одной задачи кластерного анализа // Дискрет. анализ и исслед. операций. 2011. Т. 18, № 2. С. 29–40.

15. Кельманов А. В., Хандеев В. И. Полиномиальный алгоритм с оценкой точности 2 для решения одной задачи кластерного анализа // Дискрет. анализ и исслед. операций. 2013. Т. 20, № 4. С. 36–45.

16. Долгушев А. В., Кельманов А. В., Шенмайер В. В. Полиномиальная аппроксимационная схема для одной задачи разбиения конечного множества на два кластера // Тр. Ин-та математики и механики. 2015. Т. 21, № 3. С. 100–109.

17. Кельманов А. В., Пяткин А. В., Хандеев В. И. NP-трудность квадратичной евклидовой задачи 2-кластеризации 1-mean и 1-median с ограничениями на размеры кластеров // Докл. Акад. наук. 2019. Т. 489, No. 4. С. 339–343.

18. Pyatkin A. V. 1-Mean and 1-medoid 2-clustering problem with arbitrary cluster sizes: Complexity and approximation // Yugoslav J. Oper. Res. 2023. V. 33, No. 1. P. 59–69.

19. Шенмайер В. В. Аппроксимационная схема для одной задачи поиска подмножества векторов // Дискрет. анализ и исслед. операций. 2012. Т. 19, № 2. С. 93–101.

20. Галашов А. Е., Кельманов А. В. 2-Приближённый алгоритм для одной задачи поиска семейства непересекающихся подмножеств векторов // Автоматика и телемеханика. 2014. № 4. С. 5–19.

21. Edmonds J., Karp R. M. Theoretical improvements in algorithmic efficiency for network flow problems // J. ACM. 1972. V. 19, No. 2. P. 248–264.

22. Gabow H. N., Tarjan R. E. Faster scaling algorithms for network problems // SIAM J. Comput. 1989. V. 18, No. 5. P. 1013–1036.

23. Wirth H. Algorithms + data structures = programs. Englewood Cliffs, NJ: Prentice Hall, 1976.

Пяткин Артём Валерьевич

Статья поступила
7 февраля 2023 г.

После доработки —
24 апреля 2023 г.

Принята к публикации
25 апреля 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
July–September 2023. Vol. 30, No. 3. P. 96–110

UDC 519.8+518.25 DOI 10.33048/daio.2023.30.763

PTAS FOR PROBLEMS OF VECTOR CHOICE AND CLUSTERING WITH DIFFERENT CENTERS

A. V. Pyatkin

Sobolev Institute of Mathematics,
4 Akad. Koptyug Avenue, 630090 Novosibirsk, Russia
E-mail: artempyatkin@gmail.com

Abstract. Two close in statements problems are considered. The first one is clustering, i. e. partitioning the set of d-dimensional Euclidean vectors into the given number of clusters with different types of centers so that the total dispersion would be minimum. By dispersion here we mean the sum of squared distances between the elements of the clusters and their centers. There are three types of centers: an arbitrary point (clearly, the centroid is the best choice), a point of the initial set (so-called medoid) or a fixed point of the space given in advance. The sizes of the clusters are also given as a part of the input. The second problem is the vector subset choice problem, which is finding a subset of vectors of fixed cardinality having the minimum sum of squared distances between its elements and the centroid. For each of these problems a PTAS is constructed. Bibliogr. 23.

Keywords: clustering, centroid, medoid, approximation, PTAS, MSSC.

REFERENCES

1. **P. Berkhin,** A survey of clustering data mining techniques, in *Grouping Multidimensional Data: Recent Advances in Clustering* (Springer, Heidelberg, 2006), pp. 25–71.
2. **A. K. Jain** and **R. C. Dubes,** *Algorithms for Clustering Data* (Prentice Hall, Englewood Cliffs, NJ, 1988).
3. **S. Ghoreyshi** and **J. Hosseinkhani,** Developing a clustering model based on K-means algorithm in order to creating different policies for policyholders, *Int. J. Adv. Comput. Sci. Inf. Tech.* **4** (2), 46–53 (2015).

This research is carried out within the framework of the state contract of the Sobolev Institute of Mathematics (Project FWNF–2022–0019).

English version: Journal of Applied and Industrial Mathematics **17** (3) (2023).

4. **W. D. Fisher,** On grouping for maximum homogeneity, *J. Am. Stat. Assoc.* **53** (284), 789–798 (1958).
5. **J. MacQueen,** Some methods for classification and analysis of multivariate observations, in *Proc. 5th Berkeley Symp. Mathematics, Statistics and Probability, Berkeley, USA, June 21 — July 18, 1965; Dec. 27, 1965 — Jan. 7, 1966,* Vol. 1 (Univ. California Press, Berkeley, 1967), pp. 281–297.
6. **L. Kaufman** and **P. J. Rousseeuw,** Clustering by means of medoids, in *Statistical Data Analysis Based on the L_1-Norm* (North-Holland, Amsterdam, 1987), pp. 405–416.
7. **D. Aloise, A. Deshpande, P. Hansen,** and **P. Popat,** NP-hardness of Euclidean sum-of-squares clustering, *Mach. Learn.* **75** (2), 245–248 (2009).
8. **M. Inaba, N. Katoh,** and **H. Imai,** Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering, in *Proc. 10th Annu. Symp. Computational Geometry, Stony Brook, NY, USA, June 6–8, 1994* (ACM Press, New York, 1994), pp. 332–339.
9. **A. Kumar, Y. Sabharwal,** and **S. Sen,** A simple linear time $(1 + \varepsilon)$-approximation algorithm for geometric k-means clustering in any dimensions, in *Proc. 45th Annu. IEEE Symp. Foundations of Computer Science, Rome, Italy, Oct. 17–19, 2004* (IEEE Comp. Soc., Los Alamitos, CA, 2004), pp. 454–462.
10. **A. E. Baburin, É. Kh. Gimadi, N. I. Glebov,** and **A. V. Pyatkin,** The problem of finding a subset of vectors with the maximum total weight, *Diskretn. Anal. Issled. Oper., Ser. 2,* **14** (1), 32–42 (2007) [Russian] [*J. Appl. Ind. Math.* **2** (1), 32–38 (2008)].
11. **É. Kh. Gimadi, A. V. Kel'manov, M. A. Kel'manova,** and **S. A. Khamidullin,** A posteriori detection of a quasiperiodic fragment with a given number of repetitions in a numerical sequence, *Sib. Zh. Ind. Mat.* **9** (1), 55–74 (2006) [Russian].
12. **A. V. Kel'manov** and **A. V. Pyatkin,** On the complexity of a search for a subset of «similar» vectors, *Dokl. Akad. Nauk* **421** (5), 590–592 (2008) [Russian] [*Dokl. Math.* **78** (1), 574–575 (2008)].
13. **A. V. Kel'manov** and **A. V. Pyatkin,** On a version of the problem of choosing a vector subset, *Diskretn. Anal. Issled. Oper.* **15** (5), 20–34 (2008) [Russian] [*J. Appl. Ind. Math.* **3** (4), 447–455 (2009)].
14. **A. V. Dolgushev** and **A. V. Kel'manov,** An approximation algorithm for solving a problem of cluster analysis, *Diskretn. Anal. Issled. Oper.* **18** (2), 29–40 (2011) [Russian] [*J. Appl. Ind. Math.* **5** (4), 551–558 (2011)].
15. **A. V. Kel'manov** and **V. I. Khandeev,** A 2-approximation polynomial algorithm for a clustering problem, *Diskretn. Anal. Issled. Oper.* **20** (4), 36–45 (2013) [Russian] [*J. Appl. Ind. Math.* **7** (4), 515–521 (2013)].
16. **A. V. Dolgushev, A. V. Kel'manov,** and **V. V. Shenmaier,** A polynomial-time approximation scheme for a problem of partitioning a finite set into two clusters, *Tr. Inst. Mat. Mekh.* **21** (3), 100–109 (2015) [Russian] [*Proc. Steklov Inst. Math.* **295** (Suppl. 1), 47–56 (2016)].

17. **A. V. Kel'manov, A. V. Pyatkin,** and **V. I. Khandeev,** NP-hardness of quadratic Euclidean 1-mean and 1-median 2-clustering problem with constraints on the cluster sizes, *Dokl. Akad. Nauk* **489** (4), 339–343 (2019) [Russian] [*Dokl. Math.* **100** (3), 545–548 (2019)].

18. **A. V. Pyatkin,** 1-Mean and 1-medoid 2-clustering problem with arbitrary cluster sizes: Complexity and approximation, *Yugoslav J. Oper. Res.* **33** (1), 59–69 (2023).

19. **V. V. Shenmaier,** An approximation scheme for a problem of search for a vector subset, *Diskretn. Anal. Issled. Oper.* **19** (2), 93–101 (2012) [Russian] [*J. Appl. Ind. Math.* **6** (3), 381–386 (2012)].

20. **A. E. Galashov** and **A. V. Kel'manov,** A 2-approximate algorithm to solve one problem of a family of disjoint vector subsets, *Avtom. Telemekh.,* No. 4, 5–19 (2014) [Russian] [*Autom. Remote Control* **75** (4), 595–606 (2014)].

21. **J. Edmonds** and **R. M. Karp,** Theoretical improvements in algorithmic efficiency for network flow problems, *J. ACM* **19** (2), 248–264 (1972).

22. **H. N. Gabow** and **R. E. Tarjan,** Faster scaling algorithms for network problems, *SIAM J. Comput.* **18** (5), 1013–1036 (1989).

23. **H. Wirth,** *Algorithms + Data Structures = Programs* (Prentice Hall, Englewood Cliffs, NJ, 1976).

Artem V. Pyatkin

Received February 7, 2023
Revised April 24, 2023
Accepted April 25, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль–сентябрь 2023. Т. 30, № 3. С. 111–131

УДК 519.17 DOI 10.33048/daio.2023.30.755

О ДЕРЕВЬЯХ ЗАДАННОГО ДИАМЕТРА С ЭКСТРЕМАЛЬНЫМ КОЛИЧЕСТВОМ k-ДИСТАНЦИОННЫХ НЕЗАВИСИМЫХ МНОЖЕСТВ

Д. С. Талецкий[1,2]

[1] Национальный исследовательский университет «Высшая школа экономики», ул. Большая Печёрская, 25/12, 603155 Нижний Новгород, Россия

[2] Нижегородский гос. университет им. Н. И. Лобачевского, пр. Гагарина, 23, 603950 Нижний Новгород, Россия

E-mail: dmitalmail@gmail.com

Аннотация. Множество вершин графа называется *k-дистанционным независимым*, если расстояние между любыми двумя его вершинами больше некоторого целого числа $k \geqslant 1$. В работе рассматривается задача описания n-вершинных деревьев фиксированного диаметра d, содержащих максимально и минимально возможное число k-дистанционных независимых множеств среди всех таких деревьев. Задача на максимум решается для случая $1 < k < d \leqslant 5$ при всех достаточно больших значениях n. Задача на минимум существенно более простая и решается для всех значений параметров $1 < k < d < n$. Ил. 4, библиогр. 10.

Ключевые слова: дерево, независимое множество, k-дистанционное независимое множество, диаметр.

Введение

Независимым множеством графа называется произвольное подмножество попарно не смежных его вершин. *k-Дистанционным независимым множеством* (сокращённо k-ДНМ) графа называется подмножество его вершин, любые две из которых находятся на расстоянии более $k \geqslant 1$ друг от друга. В частности, 1-ДНМ является обычным независимым множеством. *Диаметром* $\operatorname{diam}(G)$ связного графа G называется максимально возможное расстояние между двумя его вершинами. Через $i_k(G)$ обозначается число различных k-ДНМ, которое содержит граф G. Будем называть n-вершинное дерево диаметра d (i_k, d, n)-*максимальным*

Исследование выполнено при финансовой поддержке Российского научного фонда (проект № 21–11–00194).

((i_k, d, n)-*минимальным*), если оно содержит максимально (минимально) возможное число k-ДНМ среди всех таких деревьев.

Обозначим через $S_{d,n}$ n-вершинное дерево диаметра d, полученное из пути P_d присоединением $n-d$ листьев к одному из его концов. Очевидно, что звезда S_n, изоморфная $S_{2,n}$, является единственным $(i_1, 2, n)$-максимальным деревом. В [1] показано, что для всех $2 < d < n$ единственным (i_1, d, n)-максимальным деревом является граф $S_{d,n}$. С другой стороны, (i_1, d, n)-минимальные деревья устроены значительно сложнее, и задача их описания в случае $d \geqslant 8$ остаётся открытой. В работе [2] описаны (i_1, d, n)-минимальные деревья при $d \leqslant 4$ и произвольном n, а также при $d = 5$ и всех достаточно больших n. В [3] доказаны некоторые важные свойства (i_1, d, n)-минимальных деревьев при $d \geqslant 6$ и частично описана структура $(i_1, 6, n)$-минимальных деревьев. В недавней работе [4] описаны $(i_1, 6, n)$-минимальные и $(i_1, 7, n)$-минимальные деревья при $n > 160$ и $n > 400$ соответственно.

На сегодняшний день известно сравнительно мало результатов, связанных с k-ДНМ в графах. В [5–9] получены оценки на *число k-дистанционной независимости* графа (т. е. на наибольшую мощность его k-ДНМ). В [10] перечисляются k-ДНМ простого пути P_n при некоторых дополнительных ограничениях.

В настоящей работе исследуются (i_k, d, n)-максимальные и (i_k, d, n)-минимальные деревья в случае $k \geqslant 2$. Ясно, что при $k \geqslant d$ каждое n-вершинное дерево диаметра d содержит ровно $n+1$ k-ДНМ (каждое k-ДНМ в этом случае содержит не более одной вершины дерева), поэтому интерес представляет только нетривиальный случай $k < d$. Для всех $1 < k < d \leqslant 5$ и $n \geqslant 120$ найдено (i_k, d, n)-максимальное дерево $\widehat{T}_{k,d,n}$ и доказано, что оно единственно. Кроме того, для всех $1 < k < d < n$ найдено (i_k, d, n)-минимальное дерево $T_{k,d,n}$ и указаны все тройки (k, d, n), для которых это дерево единственно.

1. Некоторые определения и обозначения

Как обычно, через $N[v]$ обозначается *замкнутая окрестность* вершины v, т. е. множество, состоящее из v и всех смежных с ней вершин. Для $s \geqslant 1$ обозначим через $N_s[v]$ множество всех вершин, находящихся на расстоянии не более чем s от вершины v.

Вершина дерева T называется *предлистовой*, если она смежна хотя бы с одним его листом; *центральной*, если она находится на расстоянии не более $\lfloor \frac{\operatorname{diam}(T)+1}{2} \rfloor$ от всех его листьев. Как известно, деревья чётного диаметра содержат ровно одну центральную вершину, а деревья нечётного диаметра — ровно две. *Диаметральным путём* дерева T называется его простой путь, содержащий $\operatorname{diam}(T)+1$ вершин. Лист дерева T, являющийся концом диаметрального пути T, назовём *диаметральным*.

Напомним, что через $i_k(G)$ обозначается число различных k-ДНМ, которое содержит граф G. Через $i_k^+(G,v)$ ($i_k^-(G,v)$) обозначается число k-ДНМ графа G, содержащих (не содержащих) вершину v. Нетрудно видеть, что при всех $n \geqslant 2$ и $k \geqslant 1$ для любого n-вершинного дерева T и любой его вершины v имеет место строгое неравенство $i_k^+(T,v) < i_k^-(T,v)$. Действительно, обозначим через u произвольного соседа v. Тогда

$$i_k^-(T,v) \geqslant i_k(T \setminus N_1[v]) + i_k^+(T,u) > i_k(T \setminus N_k[v]) = i_k^+(T,v).$$

Обозначим через $M_{a,b}^l$ дерево диаметра 4, центральная вершина которого смежна с l листьями, a путями P_2 и b центральными вершинами путей P_3. Обозначим через $M_{a,b,c,d'}^{p,q}$ дерево диаметра 5, полученное из леса $M_{a,b}^p \cup M_{c,d'}^q$ соединением центральных вершин двух его поддеревьев. Будем использовать обозначение $M_{a,b}$ для дерева $M_{a,b}^0$ и M_n — для n-вершинного дерева $M_{a,b}$, где $a \leqslant 2$ (см. рис. 1). Кроме того, через $M'_{a,b,c,d'}$ обозначим дерево $M_{a,b,c,d'}^{1,0}$, а через $M_{a,b,c,d'}$ — дерево $M_{a,b,c,d'}^{0,0}$.

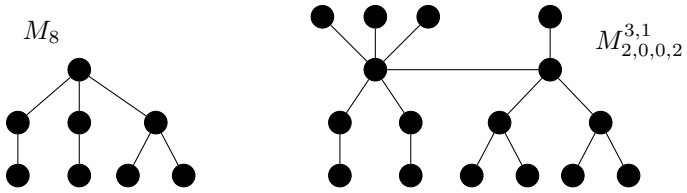


Рис. 1. Деревья M_8 и $M_{2,0,0,2}^{3,1}$

Будем называть (i_k,d,n)-максимальное ((i_k,d,n)-минимальное) дерево *максимальным* (*минимальным*), если значения параметров k, d и n понятны из контекста.

2. Случай $(i_{2k'}, 2k'+1, n)$-максимальных деревьев

Рассмотрим дерево T диаметра $2k'+1$ с центральными вершинами u и v. Обозначим через T_u максимальное по включению поддерево T, содержащее вершину u и не содержащее вершины v. Аналогичным образом определим поддерево T_v. Обозначим через l_u и l_v число диаметральных листьев T, содержащихся в поддеревьях T_u и T_v соответственно.

Теорема 1. *Для всех $n \geqslant 2k'+2$ и $k' \geqslant 1$ каждое $(i_{2k'}, 2k'+1, n)$-максимальное дерево единственно и изоморфно пути $P_{2k'}$, к концам которого присоединено $\left\lfloor \frac{n-2k'+1}{2} \right\rfloor$ и $\left\lfloor \frac{n-2k'}{2} \right\rfloor$ листьев соответственно.*

Доказательство. Очевидно, что каждое $2k'$-ДНМ дерева T содержит не более одной вершины из поддерева T_u и не более одной вершины из поддерева T_v. При этом если оно содержит ровно две вершины, то обе

они являются диаметральными листьями T. Тогда найдётся $(l_u+1)(l_v+1)$ k-ДНМ (включая пустое множество), все элементы которых являются диаметральными листьями. Кроме того, существует $(n-l_u-l_v)$ k-ДНМ, состоящих из одной вершины, не являющейся диаметральным листом. Таким образом,

$$i_{2k'}(T) = (n - l_u - l_v) + (l_u + 1)(l_v + 1) = n + l_u l_v + 1.$$

Ясно, что величина $l_u l_v$ будет принимать наибольшее значение в случае, если T содержит минимально возможное число нелистовых вершин и $|l_u - l_v| \leqslant 1$. Значит, T изоморфно простому пути $P_{2k'}$, к концам которого присоединено l_u и l_v листьев соответственно. Считаем, что $l_u \geqslant l_v$, тогда $l_u = \lfloor \frac{n-2k'+1}{2} \rfloor$ и $l_v = \lfloor \frac{n-2k'}{2} \rfloor$. Теорема 1 доказана.

Таким образом, в этом разделе описаны $(i_2, 3, n)$-максимальные деревья $\widehat{T}_{2,3,n}$ и $(i_4, 5, n)$-максимальные деревья $\widehat{T}_{4,5,n}$ (см. рис. 2). Показано, что при всех допустимых значениях n такие деревья единственны с точностью до изоморфизма.

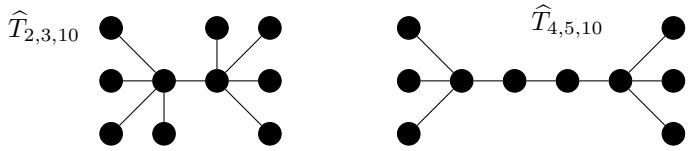


Рис. 2. Деревья $\widehat{T}_{2,3,10}$ и $\widehat{T}_{4,5,10}$

3. Случай $(i_k, 4, n)$-максимальных деревьев

Напомним, что граф $S_{4,n}$ является единственным с точностью до изоморфизма $(i_1, 4, n)$-максимальным деревом [1]. В этом разделе покажем, что при $k \in \{2, 3\}$ и $n \geqslant 53$ граф M_n является единственным $(i_k, 4, n)$-максимальным деревом.

3.1. Вариант $d = 4$, $k = 2$. Пусть центральная вершина u дерева T диаметра 4 смежна с l листьями, а также с $m \geqslant 2$ предлистовыми вершинами $u_1, \ldots, u_m$. Введём обозначение $\mathcal{L} = \prod_{i=1}^{m} \deg(u_i)$.

Лемма 1. *Имеет место равенство*

$$i_2(T) = 1 + \sum_{i=1}^{m} \frac{\mathcal{L}}{\deg(u_i)} + (l+1) \cdot \mathcal{L}.$$

Доказательство. Нетрудно видеть, что существует $\mathcal{L}$ 2-ДНМ, все элементы которых являются диаметральными листьями. Тогда найдётся $(l+1)\cdot\mathcal{L}$ 2-ДНМ, не содержащих ни одной нелистовой вершины дерева. Кроме того, для каждого $1 \leqslant i \leqslant m$ существует $\frac{\mathcal{L}}{\deg(u_i)}$ 2-ДНМ, содержащих вершину u_i, причём каждое из них не содержит других вершин из окрестности $N[u]$. Наконец, найдётся единственное 2-ДНМ $\{u\}$, содержащее центральную вершину u. Лемма 1 доказана.

Для деревьев вида $M^l_{a,b}$ имеет место равенство $\mathcal{L} = 2^a \cdot 3^b$, а значит,

$$i_2\big(M^l_{a,b}\big) = 1 + a\cdot 2^{a-1}\cdot 3^b + b\cdot 2^a\cdot 3^{b-1} + (l+1)\cdot 2^a\cdot 3^b.$$

Теорема 2. *Для всех $n \geqslant 53$ единственным $(i_2, 4, n)$-максимальным деревом является дерево M_n.*

Доказательство. Рассмотрим некоторое $(i_2, 4, n)$-максимальное дерево T и докажем по шагам, что оно совпадает с M_n.

Шаг 1. Покажем, что T имеет вид $M^l_{a,b}$. Пусть это не так. Тогда его центральная вершина u смежна хотя бы с одной вершиной u_0 степени $q_0 \geqslant 4$. Обозначим через w_1 и w_2 два произвольных листа, смежных с u_0. Через T_1 обозначим дерево, полученное удалением этих листьев из T, а через T_2 — дерево, полученное удалением вершины u_0 и всех смежных с ней листьев из T. Заменим в дереве T рёбра u_0w_1 и u_0w_2 на uw_1 и w_1w_2 и обозначим через T' получившееся дерево. Нетрудно видеть, что

$$i_2(T) = i_2(T_1) + i_2^+(T, w_1) + i_2^+(T, w_2) = i_2(T_1) + 2\cdot i_2^-(T_2, u),$$
$$i_2(T') = i_2(T_1) + i_2^+(T', w_2) + i_2^+(T', w_1) \geqslant i_2(T_1) + i_2^-(T_1, u) + 1.$$

Обозначим через w_3 произвольный лист, смежный с u_0 в T_1. Тогда

$$i_2^-(T_1, u) \geqslant (q_0 - 2)\cdot i_2^+(T_1, w_3) \geqslant (q_0 - 2)\cdot i_2^-(T_2, u),$$

откуда $i_2(T') > i_2(T)$, что противоречит максимальности T.

Шаг 2. Покажем, что T имеет вид $M_{a,b}$. Пусть это не так. Тогда T имеет вид $M^l_{a,b}$, где $l > 0$. Если при этом $a > 0$, то рассмотрим дерево $M^{l-1}_{a-1,b+1}$. Имеют место равенства

$$i_2\big(M^l_{a,b}\big) = 1 + a\cdot 2^{a-1}\cdot 3^b + b\cdot 2^a\cdot 3^{b-1} + (l+1)\cdot 2^a\cdot 3^b,$$
$$i_2\big(M^{l-1}_{a-1,b+1}\big) = 1 + (a-1)\cdot 2^{a-2}\cdot 3^{b+1} + (b+1)\cdot 2^{a-1}\cdot 3^b + l\cdot 2^{a-1}\cdot 3^{b+1}.$$

Легко проверить, что при $a, b, l \geqslant 0$ неравенство

$$i_2\big(M^{l-1}_{a-1,b+1}\big) > i_2\big(M^l_{a,b}\big)$$

равносильно неравенству $3a+2b+6l > 15$, которое верно при всех $n \geqslant 53$.

Если же $a = 0$, то T имеет вид $M^l_{0,b}$ и $i_2\big(M^l_{0,b}\big) = 1+b\cdot 3^{b-1}+(l+1)\cdot 3^b$. Тогда при всех $b, l \geqslant 1$ имеем

$$i_2\big(M^{l-1}_{2,b-1}\big) = 1 + 4\cdot 3^{b-1} + 4\cdot(b-1)\cdot 3^{b-2} + 4\cdot l\cdot 3^{b-1} > i_2\big(M^l_{0,b}\big).$$

Таким образом, при $l > 0$ дерево $M^l_{a,b}$ не максимальное.

Шаг 3. Покажем, что T изоморфно M_n. Достаточно доказать, что при $n \geqslant 53$ и $a \geqslant 3$ верно неравенство $i_2(M_{a,b}) < i_2(M_{a-3,b+2})$. Имеем

$$i_2(M_{a,b}) = 1 + a\cdot 2^{a-1}\cdot 3^b + b\cdot 2^a\cdot 3^{b-1} + 2^a\cdot 3^b,$$

$$i_2(M_{a-3,b+2}) = 1 + (a-3)\cdot 2^{a-4}\cdot 3^{b+2} + (b+2)\cdot 2^{a-3}\cdot 3^{b+1} + 2^{a-3}\cdot 3^{b+2}.$$

Поскольку при $n \geqslant 53$ и $a \geqslant 3$ верно $3a + 2b \geqslant 40$, требуемое неравенство выполнено. Теорема 2 доказана.

Заметим, что при $n = 52$ утверждение теоремы неверно, так как

$$i_2(M_{3,15}) = i_2(M_{0,17}).$$

3.2. Вариант $d = 4$, $k = 3$. Объектом нашего изучения по-прежнему является дерево T диаметра 4 с центральной вершиной u, смежной с $l \geqslant 0$ листьями, а также с предлистовыми вершинами $u_1, \ldots, u_m$, при этом $\mathcal{L} = \prod\limits_{i=1}^{m} \deg(u_i)$.

Лемма 2. *Имеет место равенство* $i_3(T) = 1 + \deg(u) + \mathcal{L}$.

Доказательство. Ясно, что существует ровно $1 + \deg(u)$ 3-ДНМ, содержащих хотя бы одну вершину из окрестности $N[u]$. Кроме того, найдётся $\mathcal{L}$ 3-ДНМ, не содержащих ни одной вершины из $N[u]$, откуда и следует требуемое равенство. Лемма 2 доказана.

Теорема 3. *При $n \geqslant 5$, $n \neq 7$ единственным $(i_3, 4, n)$-максимальным деревом является дерево M_n. При $n = 7$ деревья M_7 и $M_{3,0}$ и только они $(i_3, 4, n)$-максимальны.*

Доказательство. При $5 \leqslant n \leqslant 7$ выполнение условия теоремы легко проверить перебором всех n-вершинных деревьев диаметра 4. Предположим, что при $n \geqslant 8$ найдётся $(i_3, 4, n)$-максимальное дерево T, не изоморфное M_n. Аналогично предыдущей теореме проведём доказательство по шагам.

Шаг 1. Покажем, что T имеет вид $M^l_{a,b}$. Пусть это не так. Тогда его центральная вершина u смежна с некоторой вершиной u_0 такой, что $\deg(u_0) = q_0 \geqslant 4$. Обозначим через $u_1, \ldots, u_m$ других соседей вершины u и положим $\mathcal{L}' = \prod\limits_{i=1}^{m} \deg(u_i)$. Обозначим через w_1 и w_2 два произвольных

листа, смежных с u_0. Заменим в T рёбра u_0w_1 и u_0w_2 рёбрами uw_1 и w_1w_2. Обозначим через T' полученное дерево. Тогда

$$i_3(T') = 1 + (\deg(u) + 1) + 2 \cdot (q_0 - 2) \cdot \mathcal{L}' > 1 + \deg(u) + q_0 \cdot \mathcal{L}' = i_3(T);$$

противоречие с максимальностью T.

Шаг 2. Покажем, что T имеет вид $M_{a,b}$. Пусть это не так. Тогда T имеет вид $M^l_{a,b}$, где $l \geqslant 1$, т. е. вершина u смежна хотя бы с одним листом w. В этом случае $i_3^+(T, w) = 1$, поскольку все вершины дерева T находятся на расстоянии не более 3 от w. Рассмотрим некоторый диаметральный путь $w_1u_1uu_2w_2$ в T, заменим ребро uw ребром u_1w и обозначим через T' получившееся дерево. Ясно, что множество $\{w, w_2\}$ является 3-НДМ в T', откуда $i_3^+(T', w) \geqslant 2$. С другой стороны, $i_3^-(T, w) = i_3^-(T', w) = i_3(M^{l-1}_{a,b})$, откуда $i_3(T) < i_3(T')$, что противоречит максимальности T.

Шаг 3. Покажем, что T изоморфно M_n. Пусть это не так. Тогда T изоморфно дереву $M_{a,b}$, где $a \geqslant 3$. Покажем, что $i_3(M_{a,b}) < i_3(M_{a-3,b+2})$. Поскольку при $n \geqslant 8$ и $a \geqslant 3$ верно $2^{a-3} \cdot 3^b > 1$, то

$$i_3(M_{a,b}) = 1 + (a + b) + 2^a \cdot 3^b < 1 + (a + b - 1) + 2^{a-3} \cdot 3^{b+2} = i_3(M_{a-3,b+2});$$

противоречие с максимальностью T. Теорема 3 доказана.

4. Случай $(i_k, 5, n)$-максимальных деревьев

4.1. Вариант $d = 5$, $k = 2$. Обозначим через T'' дерево $M^{1,1}_{a,b,c,d'}$ с центральными вершинами u и v, которые смежны с листьями u' и v' соответственно. Обозначим через T' результат удаления листа v' из T'', а через T результат удаления листа u' из T'. Отметим, что деревья T и T' изоморфны деревьям $M_{a,b,c,d'}$ и $M'_{a,b,c,d'}$ соответственно.

Лемма 3. *Имеют место равенства*

$$i_2^-(T_u, u) = 2^a \cdot 3^b + a \cdot 2^{a-1} \cdot 3^b + b \cdot 2^a \cdot 3^{b-1},$$
$$i_2^-(T_v, v) = 2^c \cdot 3^{d'} + c \cdot 2^{c-1} \cdot 3^{d'} + d' \cdot 2^c \cdot 3^{d'-1},$$
$$i_2(T) = 2^a \cdot 3^b + 2^c \cdot 3^{d'} + i_2^-(T_u, u) \cdot i_2^-(T_v, v),$$
$$i_2(T') = i_2(T) + 2^a \cdot 3^b \cdot i_2^-(T_v, v),$$
$$i_2(T'') = i_2(T') + 2^c \cdot 3^{d'} \cdot i_2^-(T_u, u) + 2^{a+c} \cdot 3^{b+d'}.$$

Доказательство. Первые два равенства вытекают из леммы 1. Докажем третье равенство. Если некоторое 2-ДНМ I дерева T содержит его центральную вершину u (соответственно v), то все остальные вершины I являются диаметральными листьями поддерева T_v (T_u). Кроме того, для

каждого 2-НДМ I_u дерева T_u и каждого 2-ДНМ I_v дерева T_v, не содержащих вершин u и v соответственно, множество $I_u \cup I_v$ является 2-ДНМ дерева T. Четвёртое равенство следует из соотношений $i_2^-(T', u') = i_2(T)$ и $i_2^+(T', u') = 2^a \cdot 3^b \cdot i_2^-(T_v, v)$. Аналогично пятое равенство вытекает из соотношений $i_2^-(T'', v') = i_2(T')$ и $i_2^+(T'', v') = 2^c \cdot 3^{d'} \cdot i_2^-(T_u, u) + 2^{a+c} \cdot 3^{b+d'}$. Лемма 3 доказана.

Лемма 4. *При $n \geqslant 120$ каждое $(i_2, 5, n)$-максимальное дерево имеет вид $M_{a,b,c,d'}$.*

Доказательство. Предположим, что найдётся $(i_2, 5, n)$-максимальное дерево T, для которого утверждение леммы неверно. Рассмотрим четыре случая.

Случай 1. Хотя бы одна из центральных вершин T (считаем, что вершина u) смежна с вершиной u_0 степени $q_0 \geqslant 4$. В этом случае действуем аналогично шагу 1 теоремы 2. Обозначим через w_1 и w_2 два произвольных листа, смежных с u_0. Через T_1 обозначим дерево, полученное удалением этих листьев из T. Через T_2 обозначим дерево, полученное удалением вершины u_0 и всех смежных с ней листьев из T. Заменим в дереве T рёбра u_0w_1 и u_0w_2 на uw_1 и w_1w_2, обозначим через T' получившееся дерево. Нетрудно видеть, что

$$i_2(T) = i_2(T_1) + i_2^+(T, w_1) + i_2^+(T, w_2) = i_2(T_1) + 2 \cdot i_2^-(T_2, u),$$

$$i_2(T') = i_2(T_1) + i_2^+(T', w_2) + i_2^+(T', w_1) \geqslant i_2(T_1) + (q_0 - 2) \cdot i_2^-(T_2, u) + 1.$$

Таким образом, $i_2(T') > i_2(T)$, что противоречит максимальности T.

Случай 2. Хотя бы одна из центральных вершин T (считаем, что это вершина u) смежна с двумя различными листьями u' и u''. Удалим ребро uu'', добавим ребро $u'u''$ и обозначим получившееся дерево через T'. Очевидно, что $i_2^-(T, u'') = i_2^-(T', u'')$. Кроме того, любое 2-ДНМ дерева T, содержащее вершину u'', является 2-ДНМ дерева T', поскольку оно не содержит вершин u и u'. Однако множество $\{u'', v\}$ является 2-ДНМ T', но не является 2-ДНМ T, откуда $i_2^+(T', u'') > i_2^+(T, u'')$ и $i_2(T') > i_2(T)$, что противоречит максимальности T.

Случай 3. Дерево T имеет вид $M_{a,b,c,d'}^{1,1}$. Напомним, что через u и v обозначаются его центральные вершины, а через u' и v' — смежные с ними листья. Обозначим через $\widehat{T}$ результат удаления этих листьев из T. По лемме 3 имеем

$$i_2(T) = i_2(\widehat{T}) + 2^a \cdot 3^b \cdot i_2^-(\widehat{T}_v, v) + 2^c \cdot 3^{d'} \cdot i_2^-(\widehat{T}_u, u) + 2^{a+c} \cdot 3^{b+d'}.$$

Рассмотрим дерево T', полученное из дерева T заменой ребра vv' на $u'v'$. Тогда

$$i_2(T') = i_2(\widehat{T}) + i_2^+(T', v') + i_2^+(T', u') = i_2(\widehat{T}) + i_2^-(\widehat{T}, u) + 2^a \cdot 3^b \cdot i_2^-(\widehat{T}_v, v).$$

Покажем, что $i_2(T') > i_2(T)$. Предполагаем, что $n \geqslant 120$, тем самым $2 \cdot (a+c) + 3 \cdot (b+d') \geqslant 118$. В этом случае верно неравенство

$$i_2^-(\widehat{T}, u) > i_2^-(\widehat{T}_u, u) \cdot i_2^-(\widehat{T}_v, v) = 2^{a+c} \cdot 3^{b+d'} \cdot \left(1 + \frac{a}{2} + \frac{b}{3}\right) \cdot \left(1 + \frac{c}{2} + \frac{d'}{3}\right)$$
$$> 2^{a+c} \cdot 3^{b+d'} \cdot \left(2 + \frac{a}{2} + \frac{b}{3}\right) = 2^c \cdot 3^{d'} \cdot i_2^-(\widehat{T}_u, u) + 2^{a+c} \cdot 3^{b+d'}.$$

Таким образом, $i_2(T) < i_2(T')$, что противоречит максимальности T.

Случай 4. Дерево T имеет вид $M'_{a,b,c,d'}$. Рассмотрим все возможные варианты и покажем, что в каждом из них T не максимально.

Вариант $b \geqslant 1$. Рассмотрим дерево $M_{a+2,b-1,c,d'}$. Имеем

$$i_2(M'_{a,b,c,d'}) = 2^a \cdot 3^b + 2^c \cdot 3^{d'} + (2^{a+1} \cdot 3^b + a \cdot 2^{a-1} \cdot 3^b + b \cdot 2^a \cdot 3^{b-1}) \cdot i_2^-(T_v, v),$$
$$i_2(M_{a+2,b-1,c,d'}) = 2^{a+2} \cdot 3^{b-1} + 2^c \cdot 3^{d'} + (2^{a+2} \cdot 3^{b-1} + (a+2) \cdot 2^{a+1} \cdot 3^{b-1} + (b-1) \cdot 2^{a+2} \cdot 3^{b-2}) \cdot i_2^-(T_v, v).$$

Тогда для всех $b \geqslant 1$ верно $i_2(M'_{a,b,c,d'}) < i_2(M_{a+2,b-1,c,d'})$.

Вариант $a \geqslant 3$, $b = 0$. Рассмотрим дерево $M_{a-1,1,c,d'}$. Имеем

$$i_2(M'_{a,0,c,d'}) = 2^a + 2^c \cdot 3^{d'} + 2^{a-2} \cdot (2a+8) \cdot i_2^-(T_v, v),$$
$$i_2(M_{a-1,1,c,d'}) = 3 \cdot 2^{a-1} + 2^c \cdot 3^{d'} + 2^{a-2} \cdot (3a+5) \cdot i_2^-(T_v, v).$$

Тогда для всех $a \geqslant 3$ верно $i_2(M'_{a,0,c,d'}) < i_2(M_{a-1,1,c,d'})$.

Вариант $a \leqslant 2$, $b = 0$, $c = 1$. Рассмотрим дерево $M'_{a+1,0,0,d'}$. Имеем

$$i_2(M'_{a,0,1,d'}) = 2^a + 2 \cdot 3^{d'} + 2^{a-1} \cdot 3^{d'-1} \cdot (a+4) \cdot (2d'+9),$$
$$i_2(M'_{a+1,0,0,d'}) = 2^{a+1} + 3^{d'} + 2^{a-1} \cdot 3^{d'-1} \cdot (2a+10) \cdot (d'+3).$$

Тогда для всех $a \leqslant 2$ и $d' \geqslant 7$ верно $i_2(M'_{a,0,1,d'}) < i_2(M'_{a+1,0,0,d'})$.

Вариант $a \leqslant 2$, $b = 0$, $c = 2$. Рассмотрим дерево $M'_{a+2,0,0,d'}$. Имеем

$$i_2(M'_{a,0,2,d'}) = 2^a + 4 \cdot 3^{d'} + 2^{a-1} \cdot 3^{d'-1} \cdot (a+4) \cdot (4d'+24),$$
$$i_2(M'_{a+2,0,0,d'}) = 2^{a+2} + 3^{d'} + 2^{a-1} \cdot 3^{d'-1} \cdot (4a+24) \cdot (d'+3).$$

Тогда для всех $a \leqslant 2$ и $d' \geqslant 7$ верно $i_2(M'_{a,0,2,d'}) < i_2(M'_{a+2,0,0,d'})$.

Вариант $a \leqslant 2$, $b = 0$, $c \geqslant 3$. Рассмотрим дерево $M'_{a,0,c-3,d'+2}$. Имеем

$$i_2(M'_{a,0,c,d'}) = 2^a + 2^c \cdot 3^{d'} + 2^{c-4} \cdot 3^{d'-1} \cdot (48 + 24c + 16d') \cdot i_2^-(T_u, u),$$
$$i_2(M'_{a,0,c-3,d'+2}) = 2^a + 2^{c-3} \cdot 3^{d'+2} + 2^{c-4} \cdot 3^{d'-1} \cdot (27c + 18d' + 9) \cdot i_2^-(T_u, u).$$

Так как $a \leqslant 2$, то $3c + 2d' \geqslant 40$, а значит, $i_2(M'_{a,0,c,d'}) > i_2(M'_{a,0,c-3,d'+2})$.

Вариант $a = 1,\ b = c = 0$. Рассмотрим дерево $M_{0,3,0,d'-2}$. Имеем

$$i_2(M'_{1,0,0,d'}) = 2 + 3^{d'} + 5 \cdot (3^{d'} + d' \cdot 3^{d'-1}),$$
$$i_2(M_{0,3,0,d'-2}) = 27 + 3^{d'-2} + 54 \cdot (3^{d'-2} + (d'-2) \cdot 3^{d'-3}).$$

Тогда для всех $d' \geqslant 12$ верно $i_2(M'_{1,0,0,d'}) < i_2(M_{0,3,0,d'-2})$.

Вариант $a = 2,\ b = c = 0$. Рассмотрим дерево $M_{1,3,0,d'-2}$. Имеем

$$i_2(M'_{2,0,0,d'}) = 4 + 3^{d'} + 12 \cdot (3^{d'} + d' \cdot 3^{d'-1}),$$
$$i_2(M_{1,3,0,d'-2}) = 54 + 3^{d'-2} + 135 \cdot (3^{d'-2} + (d'-2) \cdot 3^{d'-3}).$$

Тогда для всех $d' \geqslant 3$ верно $i_2(M'_{2,0,0,d'}) < i_2(M_{1,3,0,d'-2})$. Лемма 4 доказана.

Лемма 5. *Пусть $(i_2, 5, n)$-максимальное дерево T имеет вид $M_{a,b,c,d'}$. Если при этом $3a + 2b \geqslant 40$, то $a \leqslant 2$. Если же $3c + 2d' \geqslant 40$, то $c \leqslant 2$.*

Доказательство. Предположим, что $3a + 2b \geqslant 40$ и $a \geqslant 3$. По лемме 3 имеют место соотношения

$$i_2(M_{a,b,c,d'}) = 2^a \cdot 3^b + 2^c \cdot 3^{d'} + 2^a \cdot 3^b \cdot \left(1 + \frac{a}{2} + \frac{b}{3}\right) \cdot i_2^-(T_v, v),$$

$$i_2(M_{a-3,b+2,c,d'}) = 2^{a-3} \cdot 3^{b+2} + 2^c \cdot 3^{d'} + 2^{a-3} \cdot 3^{b+2} \cdot \left(\frac{1}{6} + \frac{a}{2} + \frac{b}{3}\right) \cdot i_2^-(T_v, v).$$

Легко проверить, что в случае $3a + 2b \geqslant 40$ имеет место неравенство $i_2(M_{a,b,c,d'}) < i_2(M_{a-3,b+2,c,d'})$, что противоречит максимальности T. Случай $3c + 2d' \geqslant 40$ рассматривается аналогично. Лемма 5 доказана.

Лемма 6. *Если $n \geqslant 120$, то каждое $(i_2, 5, n)$-максимальное дерево имеет вид $M_{a,b,c,d'}$, при этом $|(3a + 2b) - (3c + 2d')| \leqslant 2$.*

Доказательство. По лемме 4 каждое максимальное дерево имеет вид $M_{a,b,c,d'}$. Предположим, что $3a+2b+3 \leqslant 3c+2d'$. Если $d' = 0$, то $c \leqslant 13$ по лемме 5, откуда $3a + 2b \leqslant 36$ и $n < 120$, что противоречит условию леммы. Если же $d' \geqslant 1$, то рассмотрим дерево $M_{a,b+1,c,d'-1}$. Имеем

$$i_2(M_{a,b,c,d'}) = 2^a \cdot 3^b + 2^c \cdot 3^{d'} + 2^{a+c} \cdot 3^{b+d'} \cdot \left(1 + \frac{a}{2} + \frac{b}{3}\right) \cdot \left(1 + \frac{c}{2} + \frac{d'}{3}\right),$$

$$i_2(M_{a,b+1,c,d'-1}) = 2^a \cdot 3^{b+1} + 2^c \cdot 3^{d'-1} + 2^{a+c} \cdot 3^{b+d'} \cdot \left(\frac{4}{3} + \frac{a}{2} + \frac{b}{3}\right) \cdot \left(\frac{2}{3} + \frac{c}{2} + \frac{d'}{3}\right).$$

Из этих соотношений следует неравенство

$$i_2(M_{a,b+1,c,d'-1}) - i_2(M_{a,b,c,d'}) > \\ 2^{a+c-1} \cdot 3^{b+d'-2}(3c + 2d' - 3a - 2b - 2) - 2^{c+1} \cdot 3^{d'-1}.$$

Предполагаем, что $2a+3b+2c+3d' \geqslant 118$ и $3c+2d' \geqslant 3a+2b+3$. Тогда, как нетрудно проверить, выполнено хотя бы одно из неравенств $2^{a-2} \cdot 3^{b-1} \geqslant 1$ и $3c + 2d' \geqslant 3a + 2b + 14$, откуда $i_2(M_{a,b+1,c,d'-1}) > i_2(M_{a,b,c,d'})$, что противоречит экстремальности $M_{a,b,c,d'}$. Лемма 6 доказана.

Теорема 4. *Для всех $n \geqslant 120$ максимальное дерево $\widehat{T}_{2,5,n}$ единственно, при этом*

$$\widehat{T}_{2,5,n} = \begin{cases} M_{1,p-1,1,p-1}, & \text{если } n = 6p, \\ M_{1,p-1,0,p}, & \text{если } n = 6p+1, \\ M_{0,p,0,p}, & \text{если } n = 6p+2, \\ M_{2,p-2,0,p+1}, & \text{если } n = 6p+3, \\ M_{1,p-1,0,p+1}, & \text{если } n = 6p+4, \\ M_{0,p+1,0,p}, & \text{если } n = 6p+5. \end{cases}$$

Доказательство. Назовём дерево вида $M_{a,b,c,d'}$ *подходящим*, если выполнены условия $|(3a + 2b) - (3c + 2d')| \leqslant 2$ и $\max(a, c) \leqslant 2$. Покажем, что при $n \geqslant 120$ искомое дерево $\widehat{T}_{2,5,n}$ подходящее. Достаточно проверить условие $\max(a, c) \leqslant 2$. Если $3a + 2b \leqslant 39$, то $3c + 2d' \leqslant 41$ по лемме 6. Тогда $2a+3b+2c+3d' \leqslant 117$, что противоречит предположению. Случай $3c+2d' \leqslant 39$ рассматривается аналогично. Если же $\min(3a+2b, 3c+2d') \geqslant 40$, то $\max(a, c) \leqslant 2$ по лемме 5, что и требовалось. Считаем, что $c \leqslant a \leqslant 2$ и, если $a = c$, то $b \geqslant d'$ (так как деревья $M_{a,b,c,d'}$ и $M_{a,d',c,b}$ в этом случае совпадают).

Случай $n = 6p$. Здесь сумма $2a + 3b + 2c + 3d' + 2$ кратна 3, откуда $a + c = 2$. Найдутся три подходящих дерева: $M_{1,p-1,1,p-1}$, $M_{2,p-2,0,p}$ и $M_{2,p-3,0,p+1}$. Тогда

$$i_2(M_{1,p-1,1,p-1}) = 4 \cdot 3^{p-1} + (3^p + 2 \cdot (p-1) \cdot 3^{p-2})^2,$$
$$i_2(M_{2,p-2,0,p}) = 13 \cdot 3^{p-2} + (8 \cdot 3^{p-2} + 4 \cdot (p-2) \cdot 3^{p-3}) \cdot (3^p + p \cdot 3^{p-1}),$$
$$i_2(M_{2,p-3,0,p+1}) = 85 \cdot 3^{p-3} + (8 \cdot 3^{p-3} + 4 \cdot (p-3) \cdot 3^{p-4}) \cdot (4 \cdot 3^p + p \cdot 3^p).$$

Нетрудно проверить, что при всех $p \geqslant 20$ дерево $M_{1,p-1,1,p-1}$ будет единственным $(i_2, 5, 6p)$-максимальным.

Случай $n = 6p + 1$. Здесь $a + c \in \{1, 4\}$. Найдутся два подходящих дерева: $M_{1,p-1,0,p}$ и $M_{2,p-1,2,p-2}$. Нетрудно проверить, что $i_2(M_{1,p-1,0,p}) > i_2(M_{2,p-1,2,p-2})$ при $p \geqslant 20$.

Случай $n = 6p+2$. В этом случае $a+c \in \{0,3\}$. Найдутся два подходящих дерева: $M_{0,p,0,p}$ и $M_{2,p-2,1,p}$. Нетрудно проверить, что $i_2(M_{0,p,0,p}) > i_2(M_{2,p-2,1,p})$ при $p \geqslant 20$.

Случай $n = 6p+3$. Здесь $a+c = 2$, деревья $M_{1,p,1,p-1}$ и $M_{2,p-2,0,p+1}$ подходящие и нетрудно проверить, что $i_2(M_{2,p-2,0,p+1}) > i_2(M_{1,p,1,p-1})$ при $p \geqslant 20$.

Случай $n = 6p+4$. В этом случае $a+c \in \{1,4\}$. Найдутся два подходящих дерева: $M_{1,p-1,0,p+1}$ и $M_{2,p-1,2,p-1}$. Нетрудно проверить, что $i_2(M_{1,p-1,0,p+1}) > i_2(M_{2,p-1,2,p-1})$ при $p \geqslant 20$.

Случай $n = 6p+5$. Здесь $a+c \in \{0,3\}$, деревья $M_{0,p+1,0,p}$ и $M_{2,p-1,1,p}$ подходящие и нетрудно проверить, что $i_2(M_{0,p+1,0,p}) > i_2(M_{2,p-1,1,p})$ при $p \geqslant 20$. Теорема 4 доказана.

4.2. Вариант $d = 5$, $k = 3$. Объектом нашего изучения по-прежнему является дерево T диаметра 5 с центральными вершинами u и v, которые смежны с вершинами $u_1, \ldots, u_p$ и $v_1, \ldots, v_q$ соответственно. Введём обозначения $\mathcal{L}_u = \prod_{i=1}^{p} \deg(u_i)$ и $\mathcal{L}_v = \prod_{i=1}^{q} \deg(v_i)$.

Лемма 7. *Имеет место равенство*

$$i_3(T) = 2 + (\deg(u)-1)\cdot\mathcal{L}_v + (\deg(v)-1)\cdot\mathcal{L}_u + \mathcal{L}_u\cdot\mathcal{L}_v.$$

Доказательство. Ясно, что существует ровно два 3-ДНМ, содержащих хотя бы одну из центральных вершин T. Кроме того, существует $(\deg(u)-1)\cdot\mathcal{L}_v$ 3-ДНМ, содержащих хотя бы одну из вершин $u_1, \ldots, u_p$ и $(\deg(v)-1)\cdot\mathcal{L}_u$ 3-ДНМ, содержащих хотя бы одну из вершин $v_1, \ldots, v_q$. Наконец, найдётся $\mathcal{L}_u\cdot\mathcal{L}_v$ 3-ДНМ, все элементы которых являются диаметральными листьями T. Лемма 7 доказана.

Заметим, что для дерева $M_{a,b,c,d'}$ выполнено соотношение

$$i_3(M_{a,b,c,d'}) = 2 + (a+b)\cdot 2^c\cdot 3^{d'} + (c+d')\cdot 2^a\cdot 3^b + 2^{a+c}\cdot 3^{b+d'}.$$

Лемма 8. *Каждое $(i_3, 5, n)$-максимальное дерево T имеет вид $M_{a,b,c,d'}$, при этом* $\max(a,c) \leqslant 2$.

Доказательство снова проведём по шагам.

Шаг 1. Покажем, что T имеет вид $M^{p',q'}_{a,b,c,d'}$. Пусть это не так. Тогда найдётся хотя бы одна предлистовая вершина u_0 степени $q_0 \geqslant 4$, смежная с одной из центральных вершин (считаем, что с вершиной u). Обозначим соседей u, отличных от v и u_0, через $u_1, \ldots, u_p$, а соседей v, отличных от u, через $v_1, \ldots, v_q$. Положим $\mathcal{L}'_u = \prod_{i=1}^{p} \deg(u_i)$, если $p \geqslant 1$, и $\mathcal{L}'_u = 1$, если $p = 0$. Обозначим через w_1 и w_2 два произвольных листа, смежных с u_0.

Заменим в дереве T рёбра u_0w_1 и u_0w_2 рёбрами uw_1 и w_1w_2, обозначим через T' получившееся дерево. Тогда

$$i_3(T) = 2 + (\deg(u) - 1) \cdot \mathcal{L}_v + (\deg(v) - 1) \cdot q_0 \cdot \mathcal{L}'_u + q_0 \cdot \mathcal{L}'_u \cdot \mathcal{L}_v,$$

$$i_3(T') = 2 + \deg(u) \cdot \mathcal{L}_v + (\deg(v) - 1) \cdot 2 \cdot (q_0 - 2) \cdot \mathcal{L}'_u + 2 \cdot (q_0 - 2) \cdot \mathcal{L}'_u \cdot \mathcal{L}_v.$$

Поскольку $q_0 \geqslant 4$, то $i_3(T') > i_3(T)$ и T не максимальное; противоречие.

Шаг 2. Покажем, что T имеет вид $M_{a,b,c,d'}$. Предположим, что это не так и хотя бы одна из центральных вершин (например, вершина u) смежна с листом u'. Рассмотрим в дереве T некоторый диаметральный путь $u_2u_1uvv_1v_2$. Удалим ребро uu' и добавим ребро u_1u', обозначим через T' получившееся дерево. Ясно, что $i_3^-(T, u') = i_3^-(T', u')$. Поскольку каждое 3-ДНМ дерева T, содержащее u', не содержит других вершин поддерева T_u и вершин из окрестности $N[v]$, оно является 3-ДНМ дерева T'. С другой стороны, множество $\{u', v_1\}$ не является 3-ДНМ дерева T, но является 3-ДНМ дерева T', откуда $i_3(T) < i_3(T')$; противоречие.

Шаг 3. Покажем, что $\max(a, c) \leqslant 2$. Предположим, что $a \geqslant 3$. Тогда

$$i_3(M_{a,b,c,d'}) = 2 + (a + b) \cdot 2^c \cdot 3^{d'} + (c + d') \cdot 2^a \cdot 3^b + 2^{a+c} \cdot 3^{b+d'},$$

$$i_3(M_{a-3,b+2,c,d'}) = 2 + (a + b - 1) \cdot 2^c \cdot 3^{d'} + (c + d') \cdot 2^{a-3} \cdot 3^{b+2} + 2^{a+c-3} \cdot 3^{b+d'+2}.$$

Ясно, что $i_3(M_{a-3,b+2,c,d'}) > i_3(M_{a,b,c,d'})$ при $a \geqslant 3$ и $\min(c, d') > 0$, что противоречит максимальности T. Случай $c \geqslant 3$ рассматривается аналогично. Лемма 8 доказана.

Лемма 9. *Для всех $n \geqslant 13$ каждое $(i_3, 5, n)-$максимальное дерево имеет вид $M_{a,b,c,d'}$, при этом либо $\min(a, c) = 0$ и $\min(b, d') = 1$, либо $\min(b, d') = 0$.*

Доказательство. Предположим, что некоторое максимальное дерево T имеет вид $M_{a,b,c,d'}$ и при этом либо $\min(a, c) = 0$ и $\min(b, d') \geqslant 2$, либо $\min(a, b, c, d') \geqslant 1$ и $\max(a, b, c, d') \geqslant 2$. Введём обозначение

$$j_3(M_{a,b,c,d'}) = i_3(M_{a,b,c,d'}) - 2 - 2^{a+c} \cdot 3^{b+d'} = (a+b) \cdot 2^c \cdot 3^{d'} + (c+d') \cdot 2^a \cdot 3^b.$$

По предположению деревья $M_{a,b-1,c,d'+1}$ и $M_{a,b+1,c,d'-1}$ существуют и имеют диаметр 5. При этом, так как дерево $M_{a,b,c,d'}$ максимально, имеем

$$i_3(M_{a,b,c,d'}) \geqslant \max(i_3(M_{a,b-1,c,d'+1}),\, i_3(M_{a,b+1,c,d'-1})).$$

Тем самым $j_3(M_{a,b,c,d'}) \geqslant \max(j_3(M_{a,b-1,c,d'+1}),\, j_3(M_{a,b+1,c,d'-1}))$ и выполняется система неравенств

$$\begin{cases} 2^c 3^{d'}(a + b) + 2^a 3^b (c + d') \geqslant 2^c 3^{d'+1}(a + b - 1) + 2^a 3^{b-1}(c + d' + 1), \\ 2^c 3^{d'}(a + b) + 2^a 3^b (c + d') \geqslant 2^c 3^{d'-1}(a + b + 1) + 2^a 3^{b+1}(c + d' - 1). \end{cases}$$

Преобразуя эту систему, получаем

$$\begin{cases}(2c+2d'-1)\cdot 2^a\cdot 3^{b-1}\geqslant(2a+2b-3)\cdot 2^c\cdot 3^{d'},\\(2a+2b-1)\cdot 2^c\cdot 3^{d'-1}\geqslant(2c+2d'-3)\cdot 2^a\cdot 3^b,\end{cases}$$

откуда следует неравенство

$$\frac{2a+2b-1}{2c+2d'-3}\geqslant 9\cdot\frac{2a+2b-3}{2c+2d'-1}.$$

Поскольку $\min(a+b,c+d')\geqslant 2$, данное неравенство имеет решения лишь в случае $a+b=c+d'=2$. По предположению это возможно только при $a=c=0$ и $b=d'=2$. В этом случае $i_3(M_{0,2,0,2})<i_3(M_{2,0,1,2})$. Таким образом, каждое дерево $M_{a,b,c,d'}$, не удовлетворяющее условию леммы, не максимальное, что и требовалось. Лемма 9 доказана.

Теорема 5. *Для всех $n\geqslant 11$ максимальное дерево $\widehat{T}_{3,5,n}$ единственно, при этом*

$$\widehat{T}_{3,5,n}=\begin{cases}M_{2,0,0,q-2}, & \text{если } n=3q,\\ M_{1,0,0,q-1}, & \text{если } n=3q+1,\\ M_{2,0,1,q-2}, & \text{если } n=3q+2.\end{cases}$$

Доказательство. Назовём максимальное дерево *подходящим*, если оно имеет вид $M_{a,b,c,d'}$, где $c\leqslant a\leqslant 2$ и либо $\min(a,c)=0$ и $\min(b,d')=1$, либо $\min(b,d')=0$. По леммам 8 и 9 при $n\geqslant 13$ искомое дерево $\widehat{T}_{3,5,n}$ подходящее. Считаем, что если $a=c$, то $b\leqslant d'$ (так как деревья $M_{a,b,c,d'}$ и $M_{a,d',c,b}$ в этом случае совпадают).

Случай $n=3q$.

Вариант $q=4$. По лемме 4 каждое $(i_3,5,12)$-максимальное дерево имеет вид $M_{a,b,c,d'}$, где $\max(a,c)\leqslant 2$. Поскольку число вершин дерева чётно, то $b+d'\in\{0,2\}$, откуда $b+d'=a+c=2$. Так как

$$i_3(M_{2,0,0,2})>\max(i_3(M_{1,1,1,1}),\, i_3(M_{1,0,1,2}),\, i_3(M_{2,1,0,1})),$$

дерево $M_{2,0,0,2}$ единственное $(i_3,5,12)$-максимальное.

Вариант $q\geqslant 5$. Если $a=c=1$, то единственным подходящим деревом является $M_{1,0,1,q-2}$, при этом $i_3(M_{1,0,1,q-2})=2\cdot 3^{q-1}+2q$. Если же $a=2$ и $c=0$, то найдутся три подходящих дерева: $M_{2,0,0,q-2}$, $M_{2,q-3,0,1}$ и $M_{2,1,0,q-3}$. При этом $i_3(M_{2,0,0,q-2})=2\cdot 3^{q-1}+4q-6$, $i_3(M_{2,q-3,0,1})=16\cdot 3^{q-3}+2q$ и $i_3(M_{2,1,0,q-3})=5\cdot 3^{q-2}+12q-34$. Поскольку при всех $q\geqslant 5$ верно неравенство

$$i_3(M_{2,0,0,q-2})>\max(i_3(M_{1,0,1,q-2}),\, i_3(M_{2,q-3,0,1}),\, i_3(M_{2,1,0,q-3})),$$

дерево $M_{2,0,0,q-2}$ единственное $(i_3,5,3q)$-максимальное.

Случай $n=3q+1$. Если $a=1$ и $c=0$, то найдутся три подходящих дерева: $M_{1,0,0,q-1}$, $M_{1,q-2,0,1}$ и $M_{1,1,0,q-2}$, при этом $i_3(M_{1,0,0,q-1})=3^q+2q$,

$i_3(M_{1,q-2,0,1}) = 8 \cdot 3^{q-2} + 2q$ и $i_3(M_{1,1,0,q-2}) = 8 \cdot 3^{q-2} + 6q - 10$. Если же $a = c = 2$, то единственным подходящим деревом является дерево $M_{2,0,2,q-3}$, при этом $i_3(M_{2,0,2,q-3}) = 8 \cdot 3^{q-2} + 4q - 2$. Поскольку при $q \geqslant 4$ верно

$$i_3(M_{1,0,0,q-1}) > \max(i_3(M_{1,q-2,0,1}),\, i_3(M_{1,1,0,q-2}),\, i_3(M_{2,0,2,q-3})),$$

дерево $M_{1,0,0,q-1}$ единственное $(i_3, 5, 3q+1)$-максимальное.

Случай $n = 3q + 2$.

Вариант $q = 3$. По лемме 8 каждое $(i_3, 5, 11)$-максимальное дерево имеет вид $M_{a,b,c,d'}$, где $\max(a, c) \leqslant 2$. Поскольку число вершин дерева нечётно, то $b + d' \in \{1, 3\}$, тогда $a + c \in \{0, 3\}$. Имеем

$$i_3(M_{2,0,1,1}) > \max(i_3(M_{0,1,0,2}),\, i_3(M_{2,1,1,0})),$$

а значит, дерево $M_{2,0,1,1}$ единственное $(i_3, 5, 11)$-максимальное.

Вариант $q \geqslant 4$. Если $a = c = 0$, то единственным подходящим деревом является $M_{0,1,0,q-1}$, при этом $i_3(M_{0,1,0,q-1}) = 4 \cdot 3^{q-1} + 3q - 1$. Если же $a = 2$ и $c = 1$, то найдутся два подходящих дерева: $M_{2,q-2,1,0}$ и $M_{2,0,1,q-2}$, при этом $i_3(M_{2,q-2,1,0}) = 4 \cdot 3^{q-1} + 2q + 2$ и $i_3(M_{2,0,1,q-2}) = 4 \cdot 3^{q-1} + 4q - 2$. Поскольку

$$i_3(M_{2,0,1,q-2}) > \max(i_3(M_{0,1,0,q-1}),\, i_3(M_{2,q-2,1,0})),$$

дерево $M_{2,0,1,q-2}$ единственное $(i_3, 5, 3q+2)$-максимальное. Теорема 5 доказана.

Заметим, что при $n = 10$ условие теоремы не выполнено, поскольку $i_3(M_{2,0,2,0}) > i_3(M_{1,0,0,2})$.

5. Случай (i_k, d, n)-минимальных деревьев

Напомним, что задача описания (i_1, d, n)-минимальных деревьев остаётся открытой при $d \geqslant 8$. В этом разделе для всех $1 < k < d < n$ построено (i_k, d, n)-минимальное дерево $T_{k,d,n}$ и указаны все тройки (k, d, n), при которых оно единственно. Кроме того, описаны все минимальные деревья в случае $1 < k < d \leqslant 5$.

Из определения k-ДНМ следует, что при $n, k \geqslant 1$ верно равенство

$$i_k(P_n) = i_k(P_{n-1}) + i_k(P_{n-k-1}),$$

где $i_k(P_{-s}) = 1$ при $0 \leqslant s \leqslant k$. Заметим, что $i_k(P_n) = n + 1$ при $0 \leqslant n \leqslant k + 1$.

Лемма 10. *Пусть $k \geqslant 2$ и $1 \leqslant m \leqslant n - 1$. Тогда*

$$i_k(P_n) < i_k(P_m) \cdot i_k(P_{n-m}).$$

Доказательство. Индукция по n при фиксированном $k \geqslant 2$. База индукции $n \leqslant k+1$ очевидна. По предположению индукции имеют место соотношения

$$\frac{i_k(P_{n-1})}{i_k(P_{n-m-1})} \leqslant i_k(P_m), \quad \frac{i_k(P_{n-k-1})}{i_k(P_{n-m-k-1})} \leqslant i_k(P_m).$$

Первое неравенство переходит в равенство лишь в случае $m = n-1$, но тогда, очевидно, второе неравенство строгое. Таким образом, верно строгое неравенство

$$\frac{i_k(P_n)}{i_k(P_{n-m})} = \frac{i_k(P_{n-1}) + i_k(P_{n-k-1})}{i_k(P_{n-m-1}) + i_k(P_{n-m-k-1})} < i_k(P_m).$$

Лемма 10 доказана.

Обозначим через $T_{k,d,n}$ дерево, полученное из пути P_{d+1} присоединением $n-d-1$ листьев либо к его k-ой от конца вершине, если $d > 2k-2$, либо к его центральной вершине, если $d \leqslant 2k-2$. Определим дерево $T'_{k,d,n}$ следующим образом. Если d чётно, то $T'_{k,d,n}$ получается из пути P_{d+1} присоединением $n-d-1$ листьев к одной из вершин, смежной с центральной вершиной пути (см. рис. 3). Если же d нечётно, то $T'_{k,d,n}$ получается из пути P_{d+1} присоединением листа к одной из его центральных вершин и $n-d-2$ листьев — к другой из его центральных вершин.

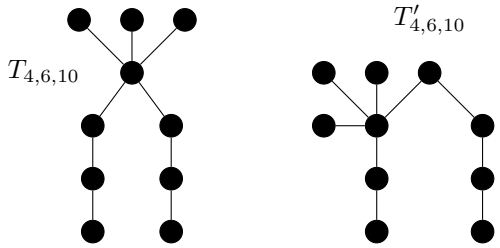


Рис. 3. Деревья $T_{4,6,10}$ и $T'_{4,6,10}$

Теорема 6. *При всех значениях $1 < k < d < n$ дерево $T_{k,d,n}$ будет (i_k, d, n)-минимальным. Кроме того, оно единственное минимальное, если и только если выполнено одно из следующих условий:*

1) $n = d+1$;
2) $n = d+2$ *и* $d \geqslant 2k-3$;
3) $n \geqslant d+3$ *и либо* $d = 2k-2$, *либо* $d \geqslant 3k-3$.

Доказательство. Рассмотрим три случая.

Случай $n = d+1$. Единственным $(d+1)$-вершинным деревом диаметра d является путь P_{d+1}, совпадающий с деревом $T_{k,d,d+1}$. Таким образом, это единственное $(i_k, d, d+1)$-минимальное дерево.

Случай $n \geqslant d+3$. Рассмотрим два варианта.

Вариант $d \leqslant 2k-2$. По определению дерева $T_{k,d,n}$ все его листья, не лежащие на диаметральном пути, смежны с одной из центральных вершин. Тогда каждый из этих листьев находится на расстоянии не более k от всех остальных вершин дерева и

$$i_k(T_{k,d,n}) = i_k(P_{d+1}) + (n-d-1).$$

Докажем, что $i_k(T) \geqslant i_k(T_{k,d,n})$ для любого n-вершинного дерева T диаметра d. Зафиксируем некоторый диаметральный путь P в дереве T. Ясно, что T содержит ровно $i_k(P_{d+1})$ k-ДНМ, содержащих только вершины P и не менее чем $n-d-1$ k-ДНМ, содержащих хотя бы одну вершину не из P, откуда сразу же вытекает требуемое неравенство.

Если $d = 2k-2$, то равенство $i_k(T) = i_k(T_{k,d,n})$ возможно лишь в том случае, когда все вершины T, не принадлежащие P, являются листьями, смежными с центральной вершиной T. Поскольку при чётном значении d центральная вершина единственна, дерево $T_{k,d,n}$ единственное минимальное. Если же $d < 2k-2$, то $i_k(T_{k,d,n}) = i_k(T'_{k,d,n})$, причём деревья $T_{k,d,n}$ и $T'_{k,d,n}$ не совпадают. Значит, дерево $T_{k,d,n}$ не единственное минимальное.

Вариант $d > 2k-2$. Поскольку для каждого листа дерева $T_{k,d,n}$, не лежащего на его диаметральном пути, вершины на расстоянии более k от него образуют путь P_{d-2k+2}, имеет место равенство

$$i_k(T'_{k,d,n}) = i_k(P_{d+1}) + (n-d-1) \cdot i_k(P_{d-2k+2}).$$

Докажем, что $i_k(T) \geqslant i_k(T_{k,d,n})$ для любого n-вершинного дерева T диаметра d. Зафиксируем в T некоторый диаметральный путь P. Для любой вершины u, не лежащей на P, найдётся хотя бы $d-2k+2$ вершин P на расстоянии более k от u. При этом такие вершины образуют либо один простой путь, либо два простых пути. Тогда по лемме 10 имеем

$$\begin{aligned} i_k(T) &\geqslant i_k(P_{d+1}) + (n-d-1) \cdot \min_{0 \leqslant m \leqslant d-2k+2} (i_k(P_m) \cdot i_k(P_{d-2k+2-m})) \\ &= i_k(P_{d+1}) + (n-d-1) \cdot i_k(P_{d-2k+2}) = i_k(T_{k,d,n}). \end{aligned}$$

Равенство $i_k(T) = i_k(T_{k,d,n})$ означает, что каждая вершина T, не лежащая на P, является листом, находящимся на расстоянии k от одного из концов P, и все такие листья находятся на расстоянии не более k

друг от друга. При $d > 2k-2$ путь P содержит две различные вершины, находящиеся на расстоянии $k-1$ от одного из его концов, обозначим их через v_1 и v_2. Тогда каждый лист T, не лежащий на P, смежен с одной из этих вершин. При этом если T не изоморфно $T_{k,d,n}$, то $\min(\deg(v_1), \deg(v_2)) \geqslant 3$. Поскольку расстояние между v_1 и v_2 не превосходит $k-2$, путь P содержит не более $3k-3$ вершин и T имеет диаметр не более $3k-4$. Значит, при $d \geqslant 3k-3$ дерево $T_{k,d,n}$ единственно, что и требовалось.

Случай $n = d+2$.

Вариант $d > 2k-2$. Из рассуждений предыдущего случая следует, что единственное $(i_k, d, d+2)$-минимальное дерево получается из пути P_{d+1} присоединением листа к его k-ой от конца вершине. Значит, оно совпадает с деревом $T_{k,d,d+2}$.

Вариант $d \in \{2k-2, 2k-3\}$. Ясно, что $i_k(T_{k,d,d+2}) = i_k(P_{d+1}) + 1$. Рассмотрим произвольное $(d+2)$-вершинное дерево T диаметра d. Оно состоит из диаметрального пути P_{d+1} и некоторого листа u, не лежащего на нём. При этом если T не совпадает с $T_{k,d,d+2}$, то лист u не смежен с центральной вершиной пути, откуда $i_k^+(T,u) > 1$ и $i_k(T) > i_k(T_{k,d,d+2})$, тогда дерево $T_{k,d,d+2}$ единственное минимальное.

Вариант $d < 2k-3$. Обозначим через $T''_{k,d,d+2}$ дерево, полученное из пути P_d присоединением листа к его вершине, которая смежна с одной из центральных вершин. Тогда $i_k(T_{k,d,d+2}) = i_k(T''_{k,d,d+2}) = i_k(P_{d+1}) + 1$ и дерево $T_{k,d,d+2}$ минимально, но не единственно. Теорема 6 доказана.

Дадим явное описание всех (i_k, d, n)-минимальных деревьев в случае $1 < k < d \leqslant 5$.

Следствие 1. *Для всех $n \geqslant 4$ единственным $(i_2, 3, n)$-минимальным деревом является граф $S_{3,n}$, при этом $i_2(S_{3,n}) = 2n-2$.*

Следствие 2. *Для всех $n \geqslant 5$ верны следующие утверждения.*

1) *Единственным $(i_2, 4, n)$-минимальным деревом является граф $S_{4,n}$, при этом $i_2(S_{4,n}) = 3n-6$.*

2) *Единственным $(i_3, 4, n)$-минимальным деревом является $M_{2,0}^{n-5}$, при этом $i_3(M_{2,0}^{n-5}) = n+2$.*

Следствие 3. *Для всех $n \geqslant 6$ верны следующие утверждения.*

1) *Единственным $(i_2, 5, n)$-минимальным деревом является граф $S_{5,n}$, при этом $i_2(S_{5,n}) = 4n-11$.*

2) *При $k \in \{3, 4\}$ каждое $(i_k, 5, n)$-максимальное дерево имеет вид $M_{1,0,1,0}^{p,q}$, где $p+q = n-6$, при этом $i_3\big(M_{1,0,1,0}^{p,q}\big) = 2n-2$ и $i_4\big(M_{1,0,1,0}^{p,q}\big) = n+2$.*

Отметим, что число попарно неизоморфных $(i_k, 5, n)$-максимальных деревьев при $k \in \{3, 4\}$ растёт линейно от n (см. рис. 4).

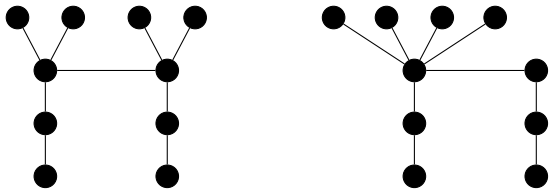


Рис. 4. Два попарно не изоморфных $(i_3, 5, 10)$-минимальных дерева

ЛИТЕРАТУРА

1. **Pedersen A. S., Vestergaard P. D.** An upper bound on the number of independent sets in a tree // Ars Comb. 2007. V. 84. P. 85–96.
2. **Frendrup A., Pedersen A. S., Sapozhenko A. A., Vestergaard P. D.** Merrifield–Simmons index and minimum number of independent sets in short trees // Ars Comb. 2013. V. 111. P. 85–95.
3. **Дайняк А. Б.** О числе независимых множеств в деревьях фиксированного диаметра // Дискрет. анализ и исслед. операций. 2009. Т. 16, № 2. С. 61–73.
4. **Талецкий Д. С.** Деревья диаметра 6 и 7 с минимальным количеством независимых множеств // Мат. заметки. 2021. Т. 109, № 2. С. 276–289.
5. **Atkinson G., Frieze A.** On the b-independence number of sparse random graphs // Comb. Probab. Comput. 2004. V. 13, No. 3. P. 295–309.
6. **Abiad A., Cioabă S. M., Tait M.** Spectral bounds for the k-independence number of a graph // Linear Algebra Appl. 2016. V. 510. P. 160–170.
7. **Bouchou A., Blidia M.** On the k-independence number in graphs // Australas. J. Comb. 2014. V. 59, No. 2. P. 311–322.
8. **O S., Shi Y., Taoqiu Z.** Sharp upper bounds on the k-independence number in graphs with given minimum and maximum degree // Graphs Comb. 2020. V. 37. P. 393–408.
9. **Li Z., Wu B.** The k-independence number of t-connected graphs // Appl. Math. Comput. 2021. V. 409. Paper ID 126412. 4 p.
10. **Jou M. J., Lin J. J.** Characterization of the distance-k independent dominating sets of the n-path // Int. J. Contemp. Math. Sci. 2018. V. 13, No. 6. P. 231–238.

Талецкий Дмитрий Сергеевич

Статья поступила
20 октября 2022 г.
После доработки —
2 мая 2023 г.
Принята к публикации
11 мая 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
July–September 2023. Vol. 30, No. 3. P. 111–131

UDC 519.17 DOI 10.33048/daio.2023.30.755

ON TREES WITH GIVEN DIAMETER AND EXTREMAL NUMBER OF DISTANCE-k INDEPENDENT SETS

D. S. Taletskii [1, 2]

[1] National Research University "Higher School of Economics",
25/12 Bolshaya Pechyorskaya Street, 603155 Nizhny Novgorod, Russia

[2] Lobachevsky Nizhny Novgorod State University,
23 Gagarin Avenue, 603950 Nizhny Novgorod, Russia

E-mail: dmitalmail@gmail.com

Abstract. The set of vertices of a graph is called *distance-k independent* if the distance between any two of its vertices is greater than some integer $k \geqslant 1$. In this paper we describe n-vertex trees with a given diameter d which have maximum and minimum possible number of distance-k independent sets among all such trees. The maximum problem is solved for the case $1 < k < d \leqslant 5$. The minimum problem is significantly more simple and is solved for all $1 < k < d < n$. Illustr. 4, bibliogr. 10.

Keywords: tree, independent set, distance-k independent set, diameter.

REFERENCES

1. **A. S. Pedersen** and **P. D. Vestergaard,** An upper bound on the number of independent sets in a tree, *Ars Comb.* **84**, 85–96 (2007).
2. **A. Frendrup, A. S. Pedersen, A. A. Sapozhenko,** and **P. D. Vestergaard,** Merrifield–Simmons index and minimum number of independent sets in short trees, *Ars Comb.* **111**, 85–95 (2013).
3. **A. B. Dainyak,** On the number of independent sets in the trees of a fixed diameter, *Diskretn. Anal. Issled. Oper.* **16** (2), 61–73 (2009) [Russian] [*J. Appl. Ind. Math.* **4** (2), 163–171 (2010)].
4. **D. S. Taletskii,** Trees of diameter 6 and 7 with minimum number of independent sets, *Mat. Zametki* **109** (2), 276–289 (2021) [Russian] [*Math. Notes* **109** (1–2), 280–291 (2021)].

This research is supported by the Russian Science Foundation (Project 21–11–00194).

English version: Journal of Applied and Industrial Mathematics **17** (3) (2023).

5. **G. Atkinson** and **A. Frieze,** On the b-independence number of sparse random graphs, *Comb. Probab. Comput.* **13** (3), 295–309 (2004).
6. **A. Abiad, S. M. Cioabă,** and **M. Tait,** Spectral bounds for the k-independence number of a graph, *Linear Algebra Appl.* **510**, 160–170 (2016).
7. **A. Bouchou** and **M. Blidia,** On the k-independence number in graphs, *Australas. J. Comb.* **59** (2), 311–322 (2014).
8. **S. O, Y. Shi,** and **Z. Taoqiu,** Sharp upper bounds on the k-independence number in graphs with given minimum and maximum degree, *Graphs Comb.* **37**, 393–408 (2020).
9. **Z. Li** and **B. Wu,** The k-independence number of t-connected graphs, *Appl. Math. Comput.* **409**, ID 126412, 4 p. (2021).
10. **M. J. Jou** and **J. J. Lin,** Characterization of the distance-k independent dominating sets of the n-path, *Int. J. Contemp. Math. Sci.* **13** (6), 231–238 (2018).

Dmitrii S. Taletskii

Received October 20, 2022
Revised May 2, 2023
Accepted May 11, 2023

ДИСКРЕТНЫЙ АНАЛИЗ
И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

2023. Том 30, № 3

Зав. редакцией Ю. В. Шамардин

Журнал подготовлен с использованием макропакета LaTeX 2_ε.
The present publication has been typeset using LaTeX 2_ε.

Журнал зарегистрирован в Федеральной службе по надзору
в сфере связи, информационных технологий и массовых коммуникаций.
Свидетельство о регистрации ЭЛ № ФС77-85978 от 26.09.2023 г.
Размещение в сети Интернет: `math-sobolev.ru`.

Дата размещения в сети Интернет 15.10.2023 г.
Формат $70 \times 100\,1/16$. Усл. печ. л. 10,7. Объём 1,04 МБ.

Издательство Института математики,
пр. Академика Коптюга, 4, 630090 Новосибирск, Россия