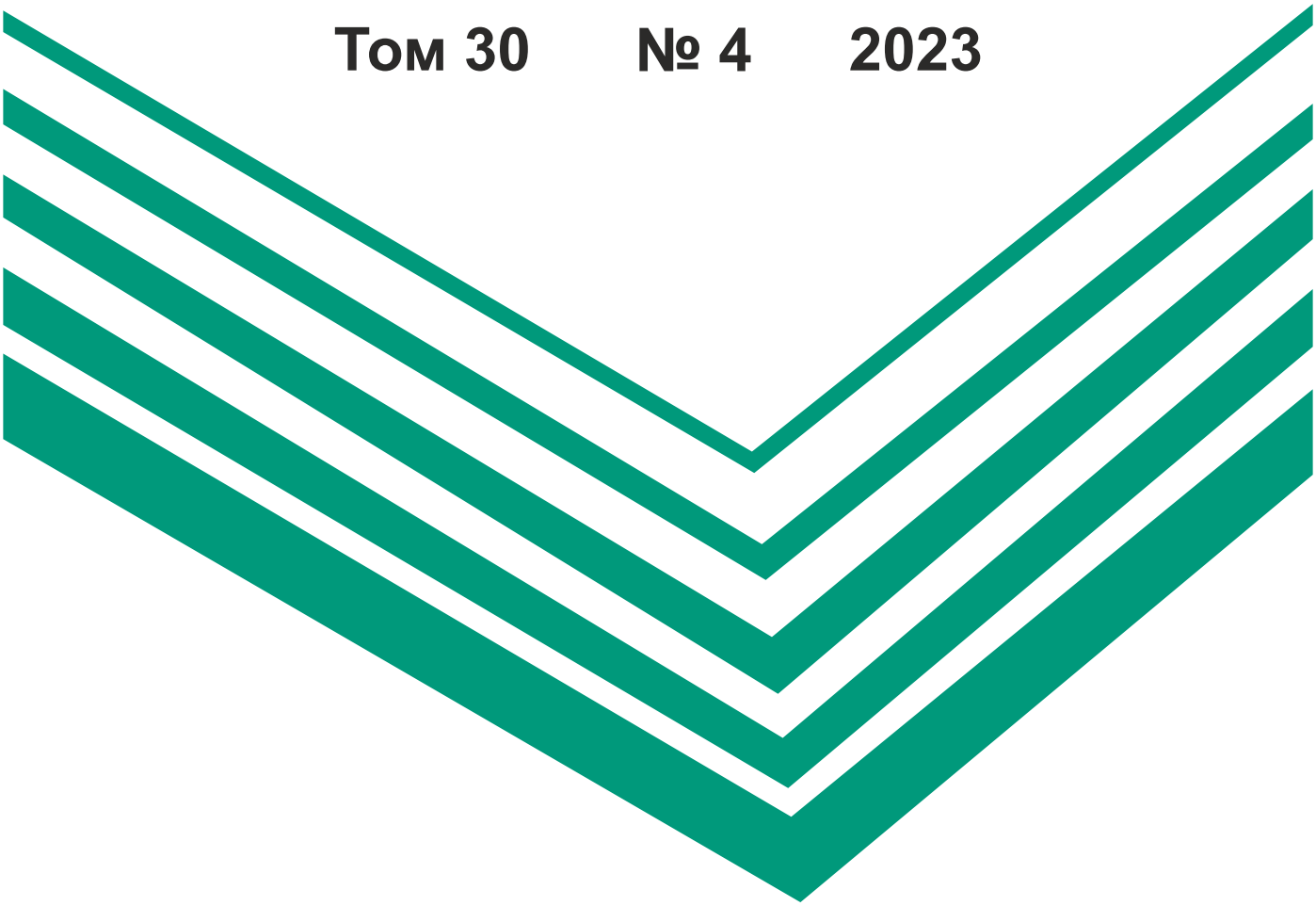
ISSN 2949-5598

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

Том 30 № 4 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, № 4 (158) **Октябрь–декабрь 2023**

СОДЕРЖАНИЕ

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. 4 (158) **October–December, 2023**

CONTENTS

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
Октябрь–декабрь 2023. Т. 30, № 4. С. 5–23

УДК 519.8 DOI 10.33048/daio.2023.30.773

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ «ИДИ С ПОБЕДИТЕЛЯМИ» ДЛЯ НЕКОТОРЫХ ЗАДАЧ СОСТАВЛЕНИЯ РАСПИСАНИЙ

П. А. Борисовский

Институт математики им. С. Л. Соболева,
пр. Акад. Коптюга, 4, 630090 Новосибирск, Россия
E-mail: pborisovsky@ofim.oscsbras.ru

Аннотация. Рассматривается подход к решению перестановочных задач составления расписаний с использованием графических ускорителей. Предложен параллельный эволюционный алгоритм на основе итеративного случайного локального поиска и алгоритма «иди с победителями». Проведён вычислительный эксперимент на тестовых примерах классической задачи Flow Shop и прикладной задачи составления производственного расписания с временны́ми окнами. Результаты показывают высокую скорость и хорошую точность получаемых решений по сравнению с различными вариантами генетического алгоритма, а также пакетом Gurobi. Предложенный подход отличается простотой реализации, удобством адаптации к особенностям высокопроизводительных графических вычислений и может применяться для решения практических задач. Табл. 3, библиогр. 18.

Ключевые слова: задача Flow Shop, производственное расписание, метаэвристика, графический процессор.

Введение

Задачи составления расписаний в промышленном производстве отличаются высокой сложностью и большим разнообразием. Несмотря на значительные успехи в решении классических задач теории расписаний, известные специализированные методы не всегда могут помочь на практике, так как даже небольшие различия в постановках могут создать серьёзные трудности для адаптации алгоритмов. Среди универсальных методов, подходящих для широкого класса задач, можно отметить целочисленное линейное программирование (ЦЛП), что связано с хорошо развитой теорией и наличием программного обеспечения (CPLEX,

Исследование выполнено за счёт гранта Российского научного фонда (проект № 22–71–10015).

Gurobi). К сожалению, на практике слишком высокая размерность зачастую не позволяет напрямую применять этот способ и требует дополнительных усилий по разработке алгоритмов (например, путём декомпозиции на задачи уменьшенной размерности). Вместе с тем этот подход полезен с точки зрения описания формальных математических моделей, верификации решений и получения оценок оптимума.

Другим направлением в разработке универсальных методов поиска решений являются различные метаэвристики, среди которых особо можно выделить алгоритмы на основе локальной оптимизации и эволюционные, в частности, генетические алгоритмы (ГА) (см., например, [1]). В связи с развитием компьютерной техники большой интерес вызывает применение параллельных вычислений в метаэвристиках, в том числе с использованием графических ускорителей (graphics processing unit, GPU).

Основой графического ускорителя является особый процессор, работающий по схеме «одиночный поток команд, множественный поток данных» (single instruction, multiple data, SIMD). Это позволяет достичь очень высокой степени параллелизма, но подходит не для любых алгоритмов, а для тех, в которых большое количество потоков одновременно выполняют один и тот же набор команд (в отличие от обычных многоядерных CPU, где каждое ядро может выполнять свои команды независимо от других). Такие вычисления особенно характерны при обработке графической информации, поэтому изначально GPU были предназначены для работы в видеокартах, но в дальнейшем стали применяться и для решения других задач [2].

Несмотря на то, что классическая схема популяционного ГА может быть сравнительно легко реализована на GPU, такой способ не представляется эффективным, так как высокой степени распараллеливания можно добиться только увеличением размера популяции, что не обязательно хорошо повлияет на качество поиска. Для преодоления этой проблемы известно множество подходов. Так, например, большой популярностью пользуется комбинирование ГА и локального поиска (memetic algorithm, MA), выполняемого на GPU [3, 4]. Также возможны варианты ГА с несколькими популяциями и определённым способом обмена данными между ними [5, 6]. Этот способ позволяет увеличить эффективность алгоритма, но отличается высокой сложностью разработки и настройки.

В данной статье предложен вариант эволюционного алгоритма, который достаточно просто реализуется на GPU. Работа является продолжением исследования [7], в котором обсуждаются особенности параллельной реализации итеративного случайного локального поиска на GPU для одной прикладной задачи составления расписаний с временны́ми окнами [8], возникающей в сборочном производстве. В дополнение к этому

здесь представлен алгоритм, сочетающий в себе идеи локального поиска и схемы «иди с победителями» (Go with the winners, GWW). Алгоритм GWW был предложен в [9], многие авторы отмечали его простоту и хорошую производительность (см., например, [10, 11]), а также возможность параллельной реализации [12]. Тем не менее на фоне стремительного роста интереса к другим метаэвристикам этот алгоритм не получил такого же широкого распространения. В данной работе предложенный подход реализован на GPU и опробован на указанной ранее задаче, а также на классической задаче Flow Shop. Показано, что алгоритм GWW хорошо адаптируется к особенностям GPU и обеспечивает высокую скорость и производительность по сравнению с различными вариантами ГА, а также пакетом Gurobi.

Статья организована следующим образом. В разд. 1 даны формулировки двух рассматриваемых задач. В разд. 2 описаны схемы классического параллельного ГА и предложенного гибридного алгоритма GWW. Разд. 3 содержит результаты экспериментального исследования, а в заключении обсуждаются результаты и возможные направления исследований.

1. Постановки задач

В перестановочной задаче Flow Shop заданы множество работ $J = \{j_1, \ldots, j_n\}$ и множество машин $I = \{i_1, \ldots, i_m\}$. Каждая работа состоит из m операций, которые должны быть выполнены соответственно на машинах $i_1, i_2, \ldots, i_m$ в фиксированном порядке, одинаковом для всех работ. Длительность операции работы j на машине i равна p_{ij}. Каждая операция может начаться после окончания предыдущей операции этой же работы и после освобождения нужной машины. Прерываний операций не допускается, при этом в данной статье не предполагается выполнения условия «no-wait», при котором запрещается промежуток времени между последовательными операциями одной работы. Необходимо найти такое упорядочение работ, при котором момент окончания последней работы (makespan) минимален.

В рассматриваемой здесь задаче составления производственного расписания с временны́ми окнами (production scheduling problem with time windows, PSPTW) задано множество работ J, каждая из которых должна быть выполнена на одной из машин множества I (т. е. в отличие от задачи Flow Shop каждая работа состоит из одной операции). Известны длительности выполнения работ на машинах p_{ij}. Для каждой работы j задано временно́е окно $[e_j, d_j]$ с условием, что момент окончания работы не может быть раньше e_j, при этом если он оказался позже d_j, то превышение времени считается запаздыванием и подлежит минимизации. Также заданы следующие параметры:

r_j — момент, начиная с которого может начаться работа j;

a_i — момент, начиная с которого может использоваться машина i;

s_j — предельно допустимое запаздывание работы j;

z_i — фиксированная стоимость выполнения произвольной работы на машине i.

Требуется назначить работы на машины и определить моменты начала и окончания выполнения каждой работы. Целевая функция состоит из двух критериев: главного — суммарное запаздывание L и дополнительного — общая стоимость выполнения работ Z:

$$f = L + \alpha Z \to \min,$$

где параметр α должен быть достаточно мал (в [8] предполагается равным 0,001). Задача NP-трудна. Она сформулирована в [8], где построена модель целочисленного программирования и разработан алгоритм локального поиска.

2. Алгоритмы решения

В данном разделе дано описание предложенного алгоритма GWW с итеративным локальным поиском, адаптированного для выполнения на GPU. Для сравнения приводится классическая схема ГА, которая также может быть реализована на GPU.

Эволюционные алгоритмы основаны на аналогиях с природными процессами эволюции живых организмов в соответствии с принципами наследственности и изменчивости. Пробные решения оптимизационной задачи понимаются как особи в популяции, к которым применяются операторы построения новых особей путём комбинации и изменения их элементов, и посредством селекции для следующих этапов выбираются преимущественно наиболее качественные особи с точки зрения исходной задачи.

2.1. Генетический алгоритм. В классическом ГА начальная популяция из N особей строится случайным образом, затем на каждой итерации строится новое поколение путём применения операторов кроссинговера и мутации к родительским особям, выбранным с помощью оператора селекции. Общая формальная схема ГА представлена ниже в алгоритме 1. Конкретная реализация отдельных операторов определяется особенностями решаемой задачи.

Приведённая схема допускает некоторые модификации. Например, чтобы не терять лучшие найденные решения, несколько лучших особей текущей популяции (элита) могут переходить в следующее поколение без изменений. Очевидным образом классический ГА может быть реализован в виде параллельной программы, так как применение кроссинговера

и мутации и вычисление функции пригодности может выполняться независимо в разных потоках.

Алгоритм 1. Генетический алгоритм

1: Построить начальную популяцию $\Pi^{(0)}$, содержащую N особей.
2: **for** $t = 1, 2, \ldots$ **do**
3: **loop** повторить $N/2$ раз:
4: Выбрать p_1, p_2 из $\Pi^{(t-1)}$ оператором селекции.
5: Применить к p_1, p_2 оператор кроссинговера и получить c_1, c_2.
6: Применить оператор мутации $c_1' := \mathrm{Mut}(c_1)$ $c_2' := \mathrm{Mut}(c_2)$.
7: Добавить c_1', c_2' в популяцию $\Pi^{(t)}$.

Заметим, что также существует другая *стационарная* (steady-state) схема ГА, в которой на каждой итерации рассматривается по одной паре родителей и потомков. Этот вариант считается довольно успешным, но, к сожалению, плохо подходит для распараллеливания.

Кодировка решений. Важным элементом ГА является кодировка, т. е. способ представления решения в виде особи. Для задачи Flow Shop любое решение естественным образом кодируется в виде перестановки работ $\pi = (\pi_1, \ldots, \pi_n)$.

В задаче PSPTW помимо перестановки в кодировку добавлен вектор $u = (u_1, \ldots, u_n)$, где u_j означает номер машины, на которую назначена работа π_j. Для упрощения процедуры восстановления решения предполагается, что значения u_j упорядочены по неубыванию. Например, перестановка $\pi = (4, 7, 3, 8, 6, 1, 5, 2)$ и вектор $u = (1, 1, 1, 2, 2, 3, 3, 3)$ представляют назначение работ: машина 1 — работы 4, 7, 3; машина 2 — работы 8, 6; машина 3 — работы 1, 5, 2.

При генерации начальных решений в данной работе не используются специальные методы. Перестановка π строится случайным образом с равномерным распределением. Вектор u также генерируется случайно с последующей сортировкой по возрастанию.

Операторы поиска. В качестве мутации используются известные операторы: оператор обмена Swap, меняющий между собой два случайно выбранных элемента массива i и j, и оператор вставки Insertion, обозначаемый далее Ins, в котором случайно выбранный элемент i переставляется в новую случайную позицию j, сдвигая промежуточные элементы вправо, если $i > j$, или влево, если $i < j$.

В случае кодировки задачи PSPTW оператор Swap изменяет только перестановку работ π, значения u не меняются. В операторе Ins элементы векторов π и u перемещаются вместе. Для того чтобы сохранить

порядок элементов вектора u, новое значение u_j задаётся таким же, как у соседних элементов, при этом, если они различаются, то выбор происходит случайным образом. Иначе говоря, работа i может переместиться на другую машину, и если новое место оказалось «между» машинами, то одна из них назначается случайно.

В качестве кроссинговера можно использовать известные операторы, разработанные для решения задачи коммивояжёра, например, оператор PMX [17].

2.2. Гибридный алгоритм «иди с победителями» с итеративным случайным локальным поиском. В основе классического детерминированного локального поиска лежит понятие *окрестности* решения — множества «соседних» решений, которое определяется по некоторому правилу. На итерациях алгоритма полностью или частично просматривается окрестность, и если в ней найдено решение лучше, чем предыдущее, то оно становится текущим, и поиск повторяется в его окрестности. Процесс продолжается, пока не будет найден локальный оптимум — решение, которое является наилучшим в своей окрестности. В методах оптимизации локальный поиск играет важнейшую роль как в качестве самостоятельного алгоритма, так и в виде вспомогательной процедуры для более сложных гибридных подходов.

Для задач большой размерности полный просмотр окрестности может оказаться слишком трудоёмким, тогда имеет смысл рассмотреть вариант, в котором просматривается некоторое случайное подмножество. Случайный выбор соседнего решения можно считать применением оператора мутации, и сам алгоритм можно отнести к классу эволюционных. Рассмотрим известную схему $(1+\lambda)$-ЭА, в которой на каждой итерации оператор мутации применяется λ раз к единственному родительскому решению, и если среди потомков найдено улучшающее решение, то оно переходит на следующую итерацию.

Алгоритм 2. Случайный локальный поиск $(1+\lambda)$-ЭА

1: Построить начальное решение x.
2: **for** $t = 1, 2, \ldots$ **do**
3: Построить λ потомков применением оператора мутации: $y_1 = \mathrm{Mut}(x)$, $\ldots$, $y_\lambda = \mathrm{Mut}(x)$.
4: Выбрать лучшего потомка y^* среди $y_1, \ldots, y_\lambda$.
5: Если $f(y^*) < f(x)$, то $x := y^*$.
6: Если выполнен критерий остановки, то вернуть решение x.

Очевидным образом шаг 3 может быть выполнен параллельно. Кроме того, можно параллельно запустить несколько независимых процессов

этого алгоритма, и после остановки каждого из них выдать лучший результат. Такой способ порождает большое число параллельных вычислительных потоков, что хорошо подходит для реализации на GPU. Большое количество независимых запусков позволяет преодолеть проблему «застревания» в локально оптимальных решениях невысокого качества. Помимо этого можно периодически выполнять операцию *встряски*, которая вносит сравнительно большие изменения в текущие решения, что даёт шанс перейти в область притяжения другого локального оптимума. На практике операция встряски может состоять в выполнении мутации несколько раз. Описанный подход широко известен как итеративный локальный поиск [13].

В данной работе предлагается улучшить качество поиска с помощью схемы «иди с победителями», идея которой состоит в следующем. Весьма вероятно, что среди большого количества экземпляров процесса локального поиска некоторые из них окажутся в областях решений невысокого качества, в которых продолжение вычислений скорее всего будет неперспективно. В этом случае можно удалить эти неудачные решения и заменить их лучшими, которые найдены другими процессами. На следующих итерациях поиск начнётся с копий лучших решений, но будет выполняться независимо, проходя те же более перспективные области, но другими случайными путями.

Алгоритм 3. Гибридный алгоритм GWW с итеративным локальным поиском

1: Построить начальные решения $x_1, \ldots, x_N$.
2: **while** не выполнен критерий остановки **do**
3: Запустить N экземпляров (1+λ)-ЭА, начиная с $x_1, \ldots, x_N$, с ограничением на количество итераций.
4: Среди текущих решений $x_1, \ldots, x_N$ выбрать R наилучших ($\overline{x}_1, \ldots, \overline{x}_R$) и R наихудших ($\underline{x}_1, \ldots, \underline{x}_R$).
5: Заменить ($\underline{x}_1, \ldots, \underline{x}_R$) копиями ($\overline{x}_1, \ldots, \overline{x}_R$).
6: При необходимости выполнить операцию встряски.

Критерием остановки может служить ограничение по времени или широко известное правило прекращать вычисления, если рекорд не обновлялся более половины прошедшего времени счёта. Для применения операции встряски можно задать простое правило, например, выполнять периодически через каждые h итераций, где h — настраиваемый параметр. Также необходимо задать другие параметры алгоритма: размер популяции N, значение λ, предельное количество итераций процедуры (1+λ)-ЭА и количество заменяемых решений R. Экспериментальное исследование показало, что это не представляет больших трудностей.

3. Экспериментальное исследование

Алгоритмы были запрограммированы на языке C++ с использованием CUDA. Вычисления проводились на устройстве Nvidia Tesla V100.

При реализации алгоритма на CUDA необходимо определить количество потоков или нитей (threads) и их объединение в блоки (blocks) [2]. Так как GPU производит одновременные вычисления группами из 32 нитей (warp), есть общая рекомендация задавать количество нитей кратным 32. Исходя их этого, в алгоритме ГА при вычислении пригодности в популяции из N особей создавалось $N/32$ блока и 32 нити в каждом блоке либо $N/64$ блока и 64 нити. Второй вариант показывал немного более высокую скорость работы. Очевидно, что размер популяции желательно задавать кратным 32 или 64. В алгоритме GWW требуется вычислять $N\lambda$ значений пригодности. Для этого выделялось N блоков и λ нитей. Из-за технических ограничений CUDA значение λ в этом случае не может превышать 1024. Очевидно, что это легко преодолеть, увеличив число блоков, но как показал эксперимент, увеличение λ нецелесообразно и не приводит к лучшим результатам.

Для уменьшения количества обращений к памяти векторы π и u в кодировке для PSPTW можно хранить в виде одного массива из n элементов типа `int`, так что в каждом элементе старшие 16 бит содержат значение j_k, младшие — u_k.

При хранении множества решений следует учитывать правило объединения запросов к памяти (memory coalescing, подробнее по ссылке `docs.nvidia.com/cuda/cuda-c-best-practices-guide`): если некоторая последовательность вычислительных потоков обращается к соседним ячейкам памяти по порядку (т. е. некоторый поток i читает или изменяет ячейку k, поток $i+1$ — ячейку $k+1$ и т. д.), то запросы могут объединяться в один, что в несколько раз ускоряет работу с памятью. Это означает, что популяцию из N особей (n-элементных массивов) следует хранить в виде матрицы $n \times N$, где особи соответствуют столбцам. Матрица затем записывается в память построчно. Иначе говоря, элементы популяции должны размещаться в следующей последовательности $(x_{11}, x_{21}, \ldots, x_{N1}, x_{12}, x_{22}, \ldots, x_{N2}, \ \ldots, x_{1n}, x_{2n}, \ldots, x_{Nn})$. Тогда при параллельном вычислении целевой функции потоки, считывая очередной элемент массива, будут обращаться к соседним ячейкам памяти, и большое количество запросов будет объединяться и выполняться значительно быстрее, чем при чтении по отдельности.

3.1. Эксперименты для задачи Flow Shop. Эксперименты проводились на известных тестовых примерах [14] и [15]. Для сравнения применён разработанный в [4] алгоритм $\mathrm{MA}_{\mathrm{PLS}}$, который представляет собой генетический алгоритм со стационарной схемой воспроизводства

и детерминированным параллельным локальным поиском на GPU. Экспериментальные результаты в [4] показали его преимущество по сравнению с другими разработанными ранее эволюционными и гибридными эвристиками. Можно также заметить, что эти результаты лучше, чем у более нового гибридного ГА с поиском с запретами [16]. Так как в [4] использовалось оборудование, которое уже считается устаревшим, расчёты были воспроизведены с такими же настройками на современном GPU Tesla V100. Полученные решения оказались сопоставимыми по качеству, время счёта сократилось примерно в три раза. Как и ранее, критерием остановки выбрано правило прекращать вычисления, если рекорд не обновлялся более чем половину времени работы.

В представленных в настоящей работе параллельных алгоритмах ГА и GWW используется такое же правило остановки. В качестве мутации в обоих алгоритмах применяется один из операторов Swap или Ins с вероятностями 0,25 и 0,75 соответственно (хорошо известно, что для задачи Flow Shop оператор Ins более предпочтителен). В качестве кроссинговера используется оператор PMX [17], применяемый с вероятностью 0,5.

Для настройки размера популяции и турнира в операторе селекции ГА многократно запускался на двух задачах с 50 и 100 работами, перебирая значения $N \in \{256, 512, 768, 1024, 1536, 2048, 4096\}$ и $s \in \{10, 20, 30, 40, 50, 100, 150, 200, 300, 500\}$. Для каждой пары значений было сделано 20 запусков. Качество работы оценивалось по значению относительного отклонения от нижней оценки целевой функции в среднем по всем запускам. Было замечено, что размер турнира сильно влияет на производительность. При $s \in [30, 50]$ среднее отклонение составляло от 1,3 до 1,5%, при других значениях — от 1,5 до 2,5%. Напротив, размер популяции оказывает небольшое влияние, и его увеличение не улучшает качества. Это говорит о том, что классическая схема ГА плохо использует возможности GPU с большим количеством ядер. Окончательно для дальнейших расчётов были выбраны значения $s = 40$, $N = 1024$. Также было опробовано несколько значений размера элиты: 1, 5 и 10% от размера популяции. Замечено, что использование элиты не приводит к значительному улучшению качества, поэтому далее рассматривается ГА без элиты.

В алгоритме GWW настраиваемых параметров больше, но замечено, что даже очень грубый подбор позволяет получить хорошую производительность. Часть параметров были назначены вручную в ходе предварительного тестирования. Так, для количества итераций процедуры $(1+\lambda)$-EA на шаге 3 алгоритма GWW выбрано значение $T = 100$, количество заменяемых решений на шагах 4–5 задано как $R = 0{,}1N$. Операция встряски была назначена на каждую 300-ю итерацию GWW (шаг 6) и состояла в пятикратном применении оператора мутации к каждому

Таблица 1

Сравнение алгоритмов на задаче Flow Shop

	Назв.	Верх.	GWW			ГА			MA_{PLS} [4]		
		оц.	мин.	сред.	t	мин.	сред.	t	мин.	сред.	t
50×10	rec31	3045	3045	3048	8	3053	3056	6	3048	3052	4
	rec33	3114	3114	3114	7	3114	3133	6	3114	3114	4
	rec35	3277	3277	3277	7	3277	3277	6	3277	3277	4
75×20	rec37	4951	4953	4973	41	5006	5047	36	4967	4987	36
	rec39	5087	5119	5120	16	5130	5141	36	5109	5119	26
	rec41	4960	4961	4979	46	5020	5061	35	4971	4991	42
50×20	ta51	3850	3862	3884	19	3893	3906	14	3868	3877	11
	ta52	3704	3713	3714	15	3722	3757	17	3713	3714	11
	ta53	3640	3654	3661	22	3665	3700	15	3658	3664	14
	ta54	3720	3732	3741	20	3753	3770	15	3731	3746	13
	ta55	3610	3614	3628	32	3662	3671	14	3616	3628	14
	ta56	3681	3687	3695	22	3724	3755	15	3687	3702	11
	ta57	3704	3707	3719	19	3732	3761	19	3708	3721	18
	ta58	3691	3699	3717	27	3746	3761	17	3697	3715	18
	ta59	3743	3756	3763	23	3777	3797	15	3753	3764	12
	ta60	3756	3767	3769	21	3781	3805	14	3767	3774	13
100×10	ta71	5770	5770	5770	11	5774	5791	17	5770	5770	17
	ta72	5349	5349	5351	12	5349	5381	19	5349	5357	12
	ta73	5676	5676	5677	12	5690	5691	17	5679	5679	11
	ta74	5781	5781	5789	18	5807	5824	16	5782	5811	14
	ta75	5467	5467	5473	15	5491	5503	16	5467	5485	18
	ta76	5303	5303	5306	11	5308	5316	17	5303	5307	11
	ta77	5595	5596	5597	13	5602	5626	15	5596	5598	11
	ta78	5617	5623	5625	13	5623	5651	19	5623	5635	14
	ta79	5871	5875	5875	11	5875	5883	18	5875	5875	15
	ta80	5845	5845	5847	11	5845	5878	18	5848	5848	12
100×20	ta81	6202	6236	6250	44	6291	6313	53	6230	6259	84
	ta82	6183	6210	6225	39	6242	6285	51	6199	6233	67
	ta83	6271	6296	6308	31	6342	6362	51	6298	6311	71
	ta84	6269	6303	6308	20	6323	6352	44	6303	6308	51
	ta85	6314	6329	6349	34	6383	6404	51	6324	6356	65
	ta86	6364	6394	6412	35	6439	6480	43	6394	6443	35
	ta87	6268	6287	6303	39	6328	6371	57	6298	6309	63
	ta88	6401	6423	6438	46	6506	6527	53	6438	6455	72
	ta89	6275	6286	6308	55	6355	6372	55	6301	6315	78
200×10	ta91	10862	10872	10872	20	10872	10883	58	10872	10872	55
	ta92	10480	10493	10497	25	10497	10521	75	10490	10495	75
	ta93	10922	10922	10922	22	10928	10956	58	10922	10922	69
	ta94	10889	10889	10891	22	10889	10896	62	10889	10890	49

Таблица 1 (продолжение)

Сравнение алгоритмов на задаче Flow Shop

	Назв.	Верх. оц.	GWW			ГА			MA_{PLS} [4]		
			мин.	сред.	t	мин.	сред.	t	мин.	сред.	t
200×10	ta95	10524	10527	10528	21	10527	10534	61	10527	10534	49
	ta96	10326	10330	10332	23	10331	10351	70	10331	10331	71
	ta97	10854	10857	10857	20	10857	10879	63	10857	10859	64
	ta98	10730	10731	10733	20	10731	10776	63	10731	10733	71
	ta99	10438	10438	10438	20	10438	10439	58	10438	10439	60
	ta100	10657	10676	10679	22	10676	10689	61	10676	10676	64
200×20	ta101	11195	11250	11292	82	11290	11339	87	11259	11280	202
	ta102	11203	11258	11308	81	11331	11369	139	11266	11301	206
	ta103	11281	11390	11414	61	11406	11475	87	11390	11409	193
	ta104	11275	11308	11359	96	11379	11454	97	11338	11354	202
	ta105	11259	11294	11314	65	11318	11353	93	11289	11313	200
	ta106	11176	11230	11293	49	11275	11324	102	11248	11289	185
	ta107	11360	11429	11443	58	11449	11478	137	11432	11441	183
	ta108	11334	11389	11428	75	11408	11473	112	11387	11413	192
	ta109	11192	11235	11299	86	11310	11355	100	11241	11282	195
	ta110	11288	11342	11374	65	11414	11449	112	11344	11374	203

решению. Настройка N и λ выполнялась, как и для ГА, на двух тестовых примерах перебором значений $N \in \{64, 128, 256, 512, 768, 1024, 1536, 2048, 4096\}$, $\lambda \in \{64, 128, 256, 512, 768, 1024, 1536\}$. Было замечено, что лучшие результаты получаются при значениях $N \cdot \lambda$ в интервале от 8000 до 25000, при которых отклонение составляло в среднем от 0,8 до 0,9%. С другими настройками отклонение преимущественно изменялось в пределах от 1 до 1,5%, в наихудших случаях достигая 2,5%. В итоге были выбраны значения $N = 256$ и $\lambda = 64$. Также после этого был проведён аналогичный эксперимент по уточнению ранее заданных параметров T, R и настроек оператора встряски, в результате которого улучшений не было получено, и первоначальные значения были сохранены.

С указанными настройками алгоритмы запускались на каждой задаче 20 раз. Результаты приведены в табл. 1, где указаны размерности задач в виде $n \times m$, названия, лучшие известные значения рекорда (столбец «Верх. оц.»), взятые из [16], минимальные и средние значения целевой функции для каждого из алгоритмов, а также среднее время работы в секундах. Можно заметить, что в большинстве случаев GWW смог найти лучшие решения, чем другие алгоритмы, или такого же качества. Решения, полученные классическим параллельным ГА, оказались за редким исключением хуже, чем у двух других эвристик, впрочем даже этот вариант превосходит представленный в [16] гибридный алгоритм GA-TS

Таблица 2

Усреднённые значения точности решений задачи Flow Shop, %

$n \times m$	GWW			ГА			MA [4]		
	мин.	сред.	макс.	мин.	сред.	макс.	мин.	сред.	макс.
50×10	0	0,04	0,09	0,09	0,33	0,52	0,03	0,08	0,09
50×20	0,25	0,53	0,88	0,96	1,58	2,34	0,27	0,57	0,9
75×20	0,23	0,5	0,87	1,06	1,68	2,41	0,32	0,67	1,06
100×10	0,02	0,07	0,23	0,16	0,49	0,83	0,03	0,17	0,28
100×20	0,38	0,63	1,01	1,12	1,59	2,16	0,42	0,79	1,16
200×10	0,05	0,07	0,09	0,06	0,23	0,39	0,05	0,07	0,13
200×20	0,5	0,86	1,23	0,9	1,34	1,79	0,56	0,8	1,07

и упоминаемый там же ГА с имитацией отжига, что хорошо иллюстрирует выгоду от применения GPU. Для наглядности сводные результаты, усреднённые по задачам каждой размерности, приведены в табл. 2, где указаны отклонения от лучших известных значений в процентах. Сравнивая средние значения, можно заметить, что только для задач большой размерности серии 200×20 результаты GWW несколько хуже, чем у MA, что говорит о большем разбросе значений. Также были проведены эксперименты с алгоритмом итеративного локального поиска без использования GWW (т. е. без применения шагов 4–5 алгоритма 3). Этот вариант оказался хуже, чем MA_{PLS}, и его результаты здесь не представлены.

В целом, можно сказать, что предложенный подход заметно превосходит известные эволюционные эвристики общего назначения, хотя может несколько уступать методам, где существенно используются специфические свойства задачи (см., например, [18]).

3.2. Эксперименты для задачи PSPTW. В работе [8] алгоритм LNS (large neighborhood search), представляющий собой локальный поиск с окрестностью на основе подхода «сломать-починить», тестировался на реальных данных и случайно сгенерированных примерах размерностью до 210 работ и 68 машин. Точные решения находились пакетом CPLEX, для самых больших задач время вычислений составляло от 3 до 9 часов (CPU Xeon X5570). Алгоритм LNS на случайных данных находил приближённые решения с хорошей точностью за 10 минут.

К сожалению, эти тестовые данные недоступны, поэтому для оценки предлагаемых алгоритмов ГА и GWW случайно сгенерированы новые задачи значительно большей размерности: $n = 500$ и $m = 150$ (доступны по ссылке `github.com/pborisovsky/PSPTW`).

Настройка параметров ГА и GWW проводилась на первых двух тестовых задачах таким же образом, как и для Flow Shop. Как и раньше,

замечено, что для ГА размер популяции слабо влияет на качество, но размер турнира в операторе селекции должен быть значительно увеличен. Окончательно были заданы значения $N = 2048$ и $s = 300$. Также были опробованы варианты с циклическим и PMX кроссинговерами, но они не показали заметного улучшения, поэтому в окончательной версии ГА кроссинговер не используется. В отличие от Flow Shop для вероятностей выбора операторов Swap и Ins в операторе мутации были заданы значения 0,75 и 0,25 соответственно, так как Swap показал лучшую производительность. Для алгоритма GWW были выбраны $N = 256$ и $\lambda = 512$. Стоит отметить, что алгоритм GWW на задаче PSPTW оказался значительно менее чувствительным к выбору параметров. При различных значениях N и s в ГА среднее отклонение от нижней оценки изменялось от 10 до 100%. Для алгоритма GWW при переборе N и λ разброс составлял от 2 до 15%.

Применение указанного раннее правила остановки оказалось проблематичным, так как из-за большого количества значений целевой функции изменения могут происходить на поздних итерациях и время счёта становится очень велико. В связи с этим дополнительно был задан лимит времени в 10 минут для обоих алгоритмов. Для остальных параметров выбраны такие же значения, как и в случае Flow Shop. Кроме того, было замечено, что параллельный итеративный локальный поиск без использования схемы GWW позволяет быстро и с большой вероятностью находить оптимальные решения в смысле первого критерия (оптимизация запаздывания). Алгоритм GWW, напротив, лучше минимизирует второй критерий (стоимость). В связи с этим было решено выполнять шаги 4–5, начиная со второй половины общего бюджета времени. Это позволило объединить преимущества обоих вариантов и устойчиво получать близкие к оптимальным решения.

Результаты представлены в табл. 3. Столбцы 2 и 3 содержат результаты решения (нижнюю и верхнюю оценки) пакетом Gurobi 9.1 с заданным ограничением времени, равным пяти часам, и возможностью использовать до 8 ядер. Расчёты проводились на CPU AMD EPYC 7502. Как видно, Gurobi способен находить решения с очень высокой точностью, хотя для четырёх примеров за пять часов не удалось приблизиться к нижней границе. Для алгоритмов GWW и ГА указаны минимальные, средние и максимальные значения целевой функции за 20 запусков на каждом примере. В отличие от Gurobi, во всех случаях удалось найти решения, близкие по качеству к нижним оценкам, хотя для шести примеров решения Gurobi оказались немного лучше. К сожалению, по этим результатам невозможно сравнить алгоритм GWW с LNS [8], но можно сказать, что на больших задачах за короткое время алгоритм GWW показывает примерно такой же порядок отклонений от оптимума, как и LNS на задачах

Таблица 3

Сравнение алгоритмов на задаче PSPTW

№	Gurobi		GWW			ГА		
	ниж.	верх.	мин.	сред.	макс.	мин.	сред.	макс.
1	121,6	402,1	122,08	122,14	122,29	125,34	132,33	145,37
2	128,79	128,89	128,98	129,00	129,02	129,27	138,74	148,26
3	140,29	480,8	140,71	141,27	142,73	143,90	152,05	162,98
4	84,20	84,28	84,34	84,36	84,38	85,57	91,76	104,59
5	70,49	70,56	71,64	72,30	73,67	71,99	80,25	89,92
6	119,63	119,73	119,83	119,86	120,02	121,17	131,79	142,14
7	124,23	124,26	124,35	124,55	125,38	124,65	127,93	133,66
8	70,40	70,41	70,52	71,47	72,71	73,82	79,86	87,84
9	191,75	520,1	192,16	192,92	195,15	198,45	209,89	227,52
10	68,90	109,4	69,35	70,09	71,38	69,605	73,646	82,63

меньшей размерности. Алгоритм ГА почти на всех примерах проигрывает даже по сравнению с худшими результатами GWW.

Заключение

В данной работе предложен алгоритм случайного локального поиска в сочетании со схемой GWW («иди с победителями») для задач составления расписаний на перестановках. Алгоритм показывает хорошие экспериментальные результаты в сравнении с другими известными CPU и GPU вариантами эволюционных алгоритмов, отличается универсальностью и простотой в реализации. Это особенно важно при решении производственных задач, так как высокая сложность моделей в сочетании со сложностью схемы алгоритма и особенностями программирования графических процессоров может стать серьёзным препятствием для практического применения. Предложенный подход мог бы быть полезен в составе универсальных решателей задач дискретной оптимизации.

Возможным направлением дальнейших исследований является адаптация рассмотренного алгоритма для работы на нескольких GPU, в том числе на суперкомпьютере, состоящем из большого числа узлов. Также представляется перспективным опробовать данный подход в комбинации с другими эвристиками, такими как имитация отжига или поиск с запретами.

Автор выражает благодарность Омскому филиалу Института математики им. С. Л. Соболева СО РАН за возможность проведения экспериментов на суперкомпьютерном кластере Tesla.

ЛИТЕРАТУРА

1. Metaheuristics for production scheduling. London: ISTE; Hoboken, NJ: Wiley, 2013. 528 p.
2. **Сандерс Дж., Кэндрот Э.** Технология CUDA в примерах. Введение в программирование графических процессоров. М.: ДМК Пресс, 2013. 232 с.
3. **Essaid M., Idoumghar L., Lepagnot J., Brévilliers M.** GPU parallelization strategies for metaheuristics: A survey // Int. J. Parallel Emerg. Distrib. Syst. 2019. V. 34, No. 5. P. 497–522.
4. **Borisovsky P., Kovalenko Yu.** A memetic algorithm with parallel local search for flowshop scheduling problems // Bioinspired optimization methods and their applications. Proc. 9th Int. Conf. (Brussels, Belgium, Nov. 19–20, 2020). Cham: Springer, 2020. P. 201–213. (Lect. Notes Comput. Sci.; V. 12438).
5. **Cheng J. R., Gen M.** Accelerating genetic algorithms with GPU computing: A selective overview // Comput. Ind. Eng. 2019. V. 128. P. 514–525.
6. **Luo J., Fujimura S., el Baz D., Plazolles B.** GPU based parallel genetic algorithm for solving an energy efficient dynamic flexible flow shop scheduling problem // J. Parallel Distrib. Comput. 2019. V. 133. P. 244–257.
7. **Борисовский П. А.** Решение задачи составления производственного расписания с помощью параллельного алгоритма локального поиска на GPU // Сб. тр. XVIII Рос. конф. с междунар. участием «Распределённые информационно-вычислительные ресурсы» (Новосибирск, Россия, 5–8 дек. 2022 г.). Новосибирск: ФИЦ ИВТ, 2022. С. 16–19.
8. **Berndorfer J., Parragh S. N.** Modeling and solving a real world machine scheduling problem with due windows and processing set restrictions // Procedia Comput. Sci. 2022. V. 200. P. 1646–1653.
9. **Aldous D., Vazirani U.** «Go with the winners» algorithms // Proc. 35th Annu. Symp. Foundations of Computer Science (Santa Fe, USA, Nov. 20–22, 1994). Los Alamitos, CA: IEEE Comput. Soc., 1994. P. 492–501.
10. **Brizuela C. A., Gutiérrez E.** Multi-objective go with the winners algorithm: A preliminary study // Evolutionary multi-criterion optimization. Proc. 3rd Int. Conf. (Guanajuato, Mexico, Mar. 9–11, 2005). Heidelberg: Springer, 2005. P. 206–220. (Lect. Notes Comput. Sci.; V. 3410).
11. **Ebert T., Goldstein D.** A «Go with the winners» approach to finding frequent patterns // Proc. 20th Annu. ACM Symp. Applied Computing (Santa Fe, USA, Mar. 13–17, 2005). New York: ACM, 2005. P. 498–502.
12. **Peinado M., Lengauer T.** Parallel «Go with the winners» algorithms in the LogP model // Proc. 11th Int. Parallel Processing Symp. (Geneva, Switzerland, Apr. 1–5, 1997). Los Alamitos, CA: IEEE Comput. Soc., 1997. P. 656–664.
13. **Juan A. A., Lourenço H., Mateo M., Luo R., Castella Q.** Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues // Int. Trans. Oper. Res. 2014. V. 21, No. 1. P. 103–126.
14. **Reeves C. R.** A genetic algorithm for flowshop sequencing // Comput. Oper. Res. 1995. V. 22, No. 1. P. 5–13.

15. Taillard E. Some efficient heuristic methods for the flow shop sequencing problem // Eur. J. Oper. Res. 1990. V. 47, No. 1. P. 65–74.
16. Umam M. S., Mustafid M., Suryono S. A hybrid genetic algorithm and tabu search for minimizing makespan in flow shop scheduling problem // J. King Saud Univ. — Comput. Inf. Sci. 2022. V. 34, No. 9. P. 7459–7467.
17. Evolutionary computation 1: Basic algorithms and operators. New York: Taylor & Francis, 2000. 340 p.
18. Grabowski J., Wodecki M. A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion // Comput. Oper. Res. 2004. V. 31, No. 11. P. 1891–1909.

Борисовский Павел Александрович

Статья поступила
12 мая 2023 г.
После доработки —
7 августа 2023 г.
Принята к публикации
20 августа 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
October–December 2023. Vol. 30, No. 4. P. 5–23

UDC 519.8 DOI 10.33048/daio.2023.30.773

A PARALLEL "GO WITH THE WINNERS" ALGORITHM FOR SOME SCHEDULING PROBLEMS

P. A. Borisovsky

Sobolev Institute of Mathematics,
4 Acad. Koptyug Avenue, 630090 Novosibirsk, Russia
E-mail: pborisovsky@ofim.oscsbras.ru

Abstract. We consider an approach to solving permutation scheduling problems using graphics accelerators. A parallel evolutionary algorithm based on the iterated random local search and the "Go with the winners" algorithm is proposed. A computational experiment was carried out on test instances of the classic Flow Shop problem and one applied production scheduling problem with time windows. The results show high computing speed and good accuracy of obtained solutions in comparison with various variants of the genetic algorithm and Gurobi solver. The proposed approach is easy to implement and convenient for adaptation to particular features of graphics computing and can be used to solve practical problems. Tab. 3, bibliogr. 18.

Keywords: Flow Shop problem, production scheduling, metaheuristic, GPU.

REFERENCES

1. *Metaheuristics for Production Scheduling* (ISTE, London; Wiley, Hoboken, NJ, 2013).
2. **J. Sanders** and **E. Kandrot,** *CUDA by Example: An Introduction to General-Purpose GPU Programming* (Addison–Wesley, 2011; DMK Press, Moscow, 2013 [Russian]).
3. **M. Essaid, L. Idoumghar, J. Lepagnot,** and **M. Brévilliers,** GPU parallelization strategies for metaheuristics: A survey, *Int. J. Parallel Emerg. Distrib. Syst.* **34** (5), 497–522 (2019).

This research is supported by the Russian Science Foundation (Project 22–71–10015).

English version: Journal of Applied and Industrial Mathematics **17** (4), 687–697 (2023), DOI 10.1134/S1990478923040014.

4. **P. Borisovsky** and **Y. Kovalenko,** A memetic algorithm with parallel local search for flowshop scheduling problems, in *Bioinspired Optimization Methods and Their Applications* (Proc. 9th Int. Conf., Brussels, Belgium, Nov. 19-20, 2020) (Springer, Cham, 2020), pp. 201–213 (Lect. Notes Comput. Sci., Vol. 12438).
5. **J. R. Cheng** and **M. Gen,** Accelerating genetic algorithms with GPU computing: A selective overview, *Comput. Ind. Eng.* **128**, 514–525 (2019).
6. **J. Luo, S. Fujimura, D. el Baz,** and **B. Plazolles,** GPU based parallel genetic algorithm for solving an energy efficient dynamic flexible flow shop scheduling problem, *J. Parallel Distrib. Comput.* **133**, 244–257 (2019).
7. **P. A. Borisovsky,** Solving one production scheduling problem using parallel local search algorithm on GPU, in *Proc. XVIII Russian Conf. Int. Particip. "Distributed Information and Computational Resources", Novosibirsk, Russia, Dec. 5–8, 2022* (Inst. Vychisl. Technol., Novosibirsk, 2022), pp. 16–19 [Russian].
8. **J. Berndorfer** and **S. N. Parragh,** Modeling and solving a real world machine scheduling problem with due windows and processing set restrictions, *Procedia Comput. Sci.* **200**, 1646–1653 (2022).
9. **D. Aldous** and **U. Vazirani,** "Go with the winners" algorithms, in *Proc. 35th Annu. Symp. Foundations of Computer Science, Santa Fe, USA, Nov. 20-22, 1994* (IEEE Comput. Soc., Los Alamitos, CA, 1994), pp. 492–501.
10. **C. A. Brizuela** and **E. Gutiérrez,** Multi-objective go with the winners algorithm: A preliminary study, in *Evolutionary Multi-Criterion Optimization* (Proc. 3rd Int. Conf., Guanajuato, Mexico, Mar. 9–11, 2005) (Springer, Heidelberg, 2005), pp. 206–220 (Lect. Notes Comput. Sci., Vol. 3410).
11. **T. Ebert** and **D. Goldstein,** A "Go with the winners" approach to finding frequent patterns, in *Proc. 20th Annu. ACM Symp. Applied Computing, Santa Fe, USA, Mar. 13–17, 2005* (ACM, New York, 2005), pp. 498–502.
12. **M. Peinado** and **T. Lengauer,** Parallel "Go with the winners" algorithms in the LogP model, in *Proc. 11th Int. Parallel Processing Symp., Geneva, Switzerland, Apr. 1–5, 1997* (IEEE Comput. Soc., Los Alamitos, CA, 1997), pp. 656–664.
13. **A. A. Juan, H. Lourenço, M. Mateo, R. Luo,** and **Q. Castella,** Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues, *Int. Trans. Oper. Res.* **21** (1), 103–126 (2014).
14. **C. Reeves,** A genetic algorithm for flow-shop sequencing, *Comput. Oper. Res.* **22** (1), 5–13 (1995).
15. **E. Taillard,** Some efficient heuristic methods for the flow shop sequencing problem, *Eur. J. Oper. Res.* **47**, 65–74 (1990).
16. **M. S. Umam, M. Mustafid,** and **S. Suryono,** A hybrid genetic algorithm and tabu search for minimizing makespan in flow shop scheduling problem, *J. King Saud Univ. — Comput. Inf. Sci.* **34** (9), 7459–7467 (2022).
17. *Evolutionary Computation 1: Basic Algorithms and Operators* (Taylor & Francis, New York, 2000).

18. J. Grabowski and **M. Wodecki,** A very fast tabu search algorithm for the permutation flow shop problem with makespan criterion, *Comput. Oper. Res.* **31** (11), 1891–1909 (2004).

Pavel A. Borisovsky

Received May 12, 2023
Revised August 7, 2023
Accepted August 20, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Октябрь–декабрь 2023. Т. 30, № 4. С. 24–34

УДК 519.175.3 DOI 10.33048/daio.2023.30.778

ПЕРЕЧИСЛЕНИЕ ПОМЕЧЕННЫХ ГРАФОВ БИБЛОКОВ

В. А. Воблый

Всероссийский институт научной и технической информации РАН,
ул. Усиевича, 20, 125190 Москва, Россия
E-mail: vitvobl@yandex.ru

Аннотация. Граф библоков — это связный граф, у которого все блоки являются полными двудольными графами. В статье перечислены по числу вершин точно и асимптотически помеченные графы библоков и графы библоков без мостов. Доказано, что почти все помеченные связные графы библоков не имеют мостов. Кроме того, перечислены помеченные планарные графы библоков и найдена асимптотическая оценка для числа таких графов. Табл. 1, библиогр. 12.

Ключевые слова: перечисление, помеченный граф, граф без мостов, полный двудольный граф, планарный граф, блок, граф библоков, случайный граф, асимптотика.

Определение 1. *Точкой сочленения* связного графа называется его вершина, после удаления которой вместе с инцидентными ей рёбрами граф становится несвязным. *Блок* — это связный граф без точек сочленения (2-связный граф), а также максимальный связный нетривиальный подграф, не имеющий точек сочленения.

Определение 2. 2-*Раскрашиваемый граф* — это граф, множество вершин которого можно разбить на два класса эквивалентности (цветных класса) таким образом, что каждое ребро графа соединяет вершины из разных классов.

Определение 3. *Двудольный граф* G — это граф, множество вершин которого можно разбить на два подмножества таким образом, что каждое ребро графа G соединяет вершины из разных подмножеств.

Определение 4. *Граф библоков* — это связный граф, у которого все блоки являются полными двудольными графами [1].

Определение 5. Класс графов называется *блочно-устойчивым*, если граф принадлежит этому классу тогда и только тогда, когда каждый блок графа принадлежит этому классу [2].

Графы библоков исследуются в спектральной теории графов [1, 3, 4]. В данной статье перечислены по числу вершин точно и асимптотически помеченные графы библоков и графы библоков без мостов. Доказывается, что почти все помеченные связные графы библоков не имеют мостов. Кроме того, перечислены планарные графы библоков, и найдена асимптотическая оценка для числа таких графов.

Лемма 1. *Обозначим через $\widetilde{B}_n$ число помеченных n-вершинных полных двудольных графов, являющихся блоками. Тогда при $n \geqslant 3$*

$$\widetilde{B}_n = 2^{n-1} - n - 1. \tag{1}$$

Доказательство. Пусть $K_{p,q}$ — полный двудольный граф с p вершинами в 1-й доле и q вершинами во 2-й доле.

Связный граф будет двудольным тогда и только тогда, когда он 2-раскрашиваемый. Обычно цвета считаются взаимозаменяемыми [5], т. е. $K_{p,q}$ и $K_{q,p}$ — это и один и тот же граф.

Пусть $|\Gamma(G)|$ — порядок группы автоморфизмов графа G. Известно [6, с. 581], что $|\Gamma(K_{p,q})| = p!q!$ при $p > q$, а $|\Gamma(K_{p,p})| = 2(p!)^2$.

Так как граф G с n вершинами и группой автоморфизмов $\Gamma(G)$ можно пометить $l(G) = n!/|\Gamma(G)|$ способами [7, с. 211], число $l(K_{p,q})$ помеченных полных двудольных графов с n вершинами, из которых p вершин в 1-й доле и $q = n - p$ вершин во 2-й доле, равно

$$l(K_{p,q}) = \frac{n!}{p!(n-p)!} = \binom{n}{p} \text{ при } p > q, \quad l(K_{p,p}) = \frac{1}{2}\binom{2p}{p}.$$

Тем самым для числа $\widetilde{B}^*_n$ полных двудольных графов с $n \geqslant 3$ вершинами с учётом взаимозаменяемости долей имеем

1) $n = 2m + 1$, m — целое число, n — нечётное число,

$$\widetilde{B}^*_n = l(K_{1,n-1}) + \ldots + l(K_{m,m+1}) = \binom{n}{1} + \ldots + \binom{n}{m}$$
$$= \frac{1}{2}\sum_{i=1}^{n-1}\binom{n}{i} = \frac{1}{2}\left(\sum_{i=0}^{n}\binom{n}{i} - 2\right) = 2^{n-1} - 1;$$

2) $n = 2m$, m — целое число, n — чётное число,

$$\widetilde{B}^*_n = l(K_{1,n-1}) + \ldots + l(K_{m,m}) = \binom{n}{1} + \ldots + \binom{n}{m-1} + \frac{1}{2}\binom{2m}{m}$$
$$= \frac{1}{2}\sum_{i=1}^{n-1}\binom{n}{i} = \frac{1}{2}\left(\sum_{i=0}^{n}\binom{n}{i} - 2\right) = 2^{n-1} - 1.$$

Однако полные двудольные графы $K_{1,q}$ при $q \geqslant 3$ не являются блоками, и таких графов-звёзд с n вершинами n штук, поэтому $\widetilde{B}_n = \widetilde{B}^*_n - n$. Лемма 1 доказана.

Теорема 1. *Для числа BB_n помеченных графов библоков с n вершинами при $n \geqslant 3$ верна формула*

$$BB_n = \frac{(n-1)!}{n}[z^{n-1}]\exp(n(e^{2z} - (z+2)e^z + 2z + 1)). \qquad (2)$$

Доказательство. Пусть C_n — число помеченных связных графов с n вершинами, а B_n — число помеченных блоков с n вершинами. Введём производящую функцию $B(z) = \sum\limits_{n=3}^{\infty} B_n \frac{z^n}{n!}$.

В [8] получена формула

$$C_n = \frac{(n-1)!}{n}[z^{n-1}]\exp(nB'(z)) = \frac{(n-1)!}{n}[z^{-1}]\exp(nB'(z))z^{-n}, \qquad (3)$$

где $[z^i]$ — коэффициентный оператор и $[z^{-1}]$ — оператор формального вычета [9, с. 11, 25].

Эта формула справедлива не только для всего класса связных графов, но и для его блочно-устойчивого подкласса [10]. Класс графов библоков является блочно-устойчивым классом, так как у графов библоков все блоки принадлежат заданному множеству 2-связных графов [10].

Обозначая экспоненциальную производящую функцию для числа блоков помеченных графов библоков через $\widetilde{B}(z)$, получим

$$BB_n = \frac{(n-1)!}{n}[z^{-1}]\exp(n\widetilde{B}'(z))z^{-n},$$

где

$$\widetilde{B}(z) = \frac{z^2}{2} + \sum_{n=3}^{\infty}(2^{n-1} - n - 1)\frac{z^n}{n!},$$

$$\widetilde{B}'(z) = z + \sum_{n=3}^{\infty}(2^{n-1} - n - 1)\frac{z^{n-1}}{(n-1)!}$$

$$= z + \sum_{n=2}^{\infty}(2^n - n - 2)\frac{z^n}{n!} = z + \sum_{n=2}^{\infty}\frac{(2z)^n}{n!} - \sum_{n=2}^{\infty}\frac{z^n}{(n-1)!} - 2\sum_{n=2}^{\infty}\frac{z^n}{n!}$$

$$= z + e^{2z} - 2z - 1 - z\sum_{n=1}^{\infty}\frac{z^n}{n!} - 2(e^z - z - 1) = e^{2z} - (z+2)e^z + 2z + 1.$$

Теорема 1 доказана.

В [11] Флажоле и Седжвиком доказана

Теорема 2 ([11, теорема VIII.8])**.** *Обозначим*

$$F(N,n) = [z^N]\{a(z)(b(z))^n\} = \frac{1}{2\pi i}\oint a(z)(b(z))^n \frac{dz}{z^{N+1}}.$$

Пусть выполнены следующие условия.

1. *Функции* $a(z) = \sum\limits_{j\geqslant 0} a_j z^j$ *и* $b(z) = \sum\limits_{j\geqslant 0} b_j z^j$ *аналитические в точке* $z = 0$ *и имеют неотрицательные коэффициенты, а кроме того,* $b(0) \neq 0$.
2. НОД$\{j \mid b_j > 0\} = 1$.
3. *Если* $R \leqslant \infty$ *— радиус сходимости* $b(z)$*, то радиус сходимости* $a(z)$ *не меньше* R.

Обозначим $T = \lim\limits_{x\to R-0} \frac{xb'(x)}{b(x)}$*. Пусть* λ *— положительное число такое, что* $0 < \lambda < T$*, и* r *— единственный действительный корень уравнения* $r\frac{b'(r)}{b(r)} = \lambda$*. Обозначим* $\sigma = \frac{d^2}{dr^2}(\ln b(r) - \lambda \ln r)$*. Тогда для целого* $N = \lambda n$ *при* $n \to \infty$ *и* $N \to \infty$ *верно асимптотическое равенство*

$$F(N,n) \sim a(r)\frac{(b(r))^n}{r^{N+1}\sqrt{2\pi n\sigma}}.$$

Теорема 3. *Для числа* BB_n *помеченных графов библоков с* n *вершинами при* $n \to \infty$ *верна асимптотическая формула*

$$BB_n \sim cn^{-5/2}a^n n!, \tag{4}$$

где $c \approx 0{,}14640222263$*,* $a \approx 3{,}604255483$.

Доказательство. Используем теорему Флажоле — Седжвика. В нашем случае в силу формулы (1) имеем

$$BB_n = \frac{(n-1)!}{n}[z^{-1}]\exp(n\overline{B}'(z))z^{-n} = \frac{(n-1)!}{n}F(N,n),$$

где $N = n$, $\lambda = 1$, $a(z) = z$, $b(z) = \exp(e^{2z} - (z+2)e^z + 2z + 1)$.

Ряды для функций $a(z)$ и $b(z)$ имеют бесконечный радиус сходимости, поэтому оператор формального вычета является контурным интегралом. Функции $a(z)$ и $b(z)$ аналитические в точке $z = 0$, и $b(0) = 1$. Функция $b(z)$ имеет положительные коэффициенты, так как $b(z) = \exp(B'(z))$ и $B(z)$ — производящая функция для числа помеченных блоков частного вида. Поскольку $b_2 = 1 > 0$, $b_3 = 1/2 > 0$, имеем НОД$\{j \mid b_j > 0\} = 1$.

Таким образом, условия 1–3 теоремы Флажоле — Седжвика выполнены.

Найдём

$$T = \lim_{x\to+\infty} \frac{xb'(x)}{b(x)} = \lim_{x\to+\infty} x(2e^{2x} - (x+3)e^x + 2) = +\infty.$$

В нашем случае $0 < \lambda < T$ и уравнение $r\frac{b'(r)}{b(r)} = \lambda$ имеет вид

$$r(2e^{2r} - (r+3)e^r + 2) = 1.$$

Решая это уравнение с помощью Maple, видим, что его единственным действительным корнем является число $r \approx 0{,}5450068623$. Вычисляя

$$\sigma = \Big(\frac{b'(r)}{b(r)}\Big)' + \frac{\lambda}{r^2} = 4e^{2r} - (r+4)e^r + \frac{1}{r^2},$$

получим $\sigma = 7{,}425484050$. Также с помощью Maple вычислим

$$c = \frac{a(r)}{r\sqrt{2\pi\sigma}} = \frac{1}{\sqrt{2\pi\sigma}} = 0{,}1464022263, \quad a = \frac{b(r)}{r} = 3{,}604255483.$$

Окончательно при $n \to \infty$ имеем асимптотику

$$BB_n = \frac{(n-1)!}{n}F(N,n) \sim \frac{(n-1)!}{n}\frac{1}{\sqrt{2\pi\sigma}}n^{-1/2}\left(\frac{b(r)}{r}\right)^n \sim n!cn^{-5/2}a^n.$$

Теорема 3 доказана.

Теорема 4. *Для числа BB_n помеченных графов библоков с n вершинами без мостов при $n \geqslant 4$ верна формула*

$$BB_n = \frac{(n-1)!}{n}[z^{n-1}]\exp(n(e^{2z} - (z+2)e^z + z + 1)). \tag{5}$$

Доказательство. Графы без мостов не имеют блоков, состоящих из одного ребра, которым в производящей функции $B(z)$ графов блоков соответствует слагаемое $\frac{1}{2}z^2$, поэтому, вычитая это слагаемое из $B(z)$, как следствие формулы (3) для числа $\overline{BB}_n$ помеченных связных n-вершинных графов библоков без мостов получим формулу (5). Теорема 4 доказана.

Теорема 5. *Для числа $\overline{BB}_n$ помеченных графов библоков с n вершинами без мостов при $n \to \infty$ верна асимптотическая формула*

$$\overline{BB}_n \sim \bar{c}n^{-5/2}\bar{a}^n n!, \tag{6}$$

где $\bar{c} \approx 0{,}1374976277$, $\bar{a} \approx 1{,}957494966$.

Доказательство. Воспользуемся снова теоремой Флажоле — Седжвика. В этом случае имеем

$$\overline{BB}_n = \frac{(n-1)!}{n}F(N,n), \quad N = n, \quad \lambda = 1,$$
$$a(z) = z, \quad b(z) = \exp(e^{2z} - (z+2)e^z + z + 1).$$

Ряд для функции $b(z)$ имеет бесконечный радиус сходимости, $b(0) = 1$. Этот ряд имеет положительные коэффициенты, так как $b(z) = \exp(B'(z))$

и $B(z)$ — производящая функция для числа помеченных блоков частного вида. Поскольку $b_3 = 1/2 > 0$, $b_4 = 5/12 > 0$, имеем НОД$\{j \mid b_j > 0\} = 1$.

Таким образом, условия 1–3 теоремы Флажоле — Седжвика выполнены.

Найдём

$$T = \lim_{x\to+\infty} \frac{xb'(x)}{b(x)} = \lim_{x\to+\infty} x(2e^{2x} - (x+3)e^x + 1) = +\infty.$$

В этом случае $0 < \lambda < T$ и уравнение $r\frac{b'(r)}{b(r)} = \lambda$ имеет вид

$$r(2e^{2r} - (r+3)e^r + 1) = 1.$$

Решая это уравнение с помощью Maple, видим, что его единственным действительным корнем является число $r \approx 0{,}6731556968$. Вычисляя

$$\sigma = \Big(\frac{b'(r)}{b(r)}\Big)' + \frac{\lambda}{r^2} = 4e^2 - (r+4)e^r + \frac{1}{r^2},$$

получим $\sigma \approx 8{,}418403176$. Также с помощью Maple вычислим

$$\bar{c} = \frac{a(r)}{r\sqrt{2\pi\sigma}} = \frac{1}{\sqrt{2\pi\sigma}} \approx 0{,}1374976277, \quad \bar{a} = \frac{b(r)}{r} \approx 1{,}957494966.$$

Окончательно при $n \to \infty$ имеем асимптотику

$$\overline{BB}_n = \frac{(n-1)!}{n} F(N,n) \sim \frac{(n-1)!}{n} \frac{1}{\sqrt{2\pi\sigma}} n^{-1/2} \Big(\frac{b(r)}{r}\Big)^n \sim n!\bar{c}n^{-5/2}\bar{a}^n.$$

Теорема 5 доказана.

Следствие 1. *Почти все помеченные графы библоков имеют мосты.*

Доказательство. С учётом формул (4) и (6) и того, что $\bar{a} < a$, имеем

$$\lim_{n\to\infty} \frac{\overline{BB}_n}{BB_n} = \lim_{n\to\infty} \Big(\frac{\bar{c}}{c}\Big)\Big(\frac{\bar{a}}{a}\Big)^n = 0,$$

т. е. доля n-вершинных графов библоков без мостов среди всех помеченных n-вершинных графов библоков при $n \to \infty$ стремится к нулю, что равносильно утверждению следствия. Следствие 1 доказано.

Отметим, что почти все помеченные связные графы не имеют мостов. Однако в таких классах помеченных графов, как графы блоков, кактусы, полноблочно-кактусные графы и последовательно-параллельные графы, почти все связные графы имеют мосты [12].

Теорема 6. *Для числа PBB_n помеченных планарных графов библоков с n вершинами при $n \geqslant 4$ верна формула*

$$PBB_n = \frac{(n-1)!}{n}[z^{n-1}] \exp\left(n\Big(e^z\Big(z + \frac{z^2}{2}\Big) - \frac{z^3}{2} - \frac{3}{2}z^2\Big)\right). \qquad (7)$$

Доказательство. Граф планарен тогда и только тогда, когда каждый его блок планарен [7, с. 129]. Тем самым класс планарных графов является блочно-устойчивым классом графов, и можно использовать перечислительную формулу (3).

Полный двудольный граф $K_{m,n}$ при $3 \leqslant m \leqslant n$ содержит подграф $K_{3,3}$ и, следовательно, не планарен в силу теоремы Понтрягина — Куратовского [7, с. 133]. Полный двудольный граф $K_{2,n}$ планарен при $n \geqslant 2$, так как его можно изобразить следующим образом: в первой строке будет одна вершина из первой доли, во второй — n вершин из второй доли и в третьей строке — вторая вершина из первой доли. Теперь производящая функция $\overline{B}(z)$ для числа планарных блоков графов библоков имеет вид

$$\overline{B}(z) = \frac{z^2}{2} + \sum_{n=2}^{\infty} l(K_{2,n}) \frac{z^{n+2}}{(n+2)!} = \frac{z^2}{2} + \frac{z^4}{8} + z^2 \sum_{n=3}^{\infty} \binom{n+2}{2} \frac{z^n}{(n+2)!}$$

$$= \frac{z^2}{2} + \frac{z^4}{8} + \frac{z^2}{2} \sum_{n=3}^{\infty} \frac{z^n}{n!} = \frac{z^2}{2} + \frac{z^4}{8} + \frac{z^2}{2}(e^z - \frac{z^2}{2} - z - 1),$$

$$\overline{B}\,'(z) = e^z(z + \frac{z^2}{2}) - \frac{z^3}{2} - \frac{3}{2}z^2.$$

После подстановки $\overline{B}\,'(z)$ в формулу (3) получим выражение (7). Теорема 6 доказана.

Теорема 7. *Для числа PBB_n помеченных планарных n-вершинных графов при $n \to \infty$ верна оценка*

$$PBB_n \leqslant c_1 n^{-2} a_1^n n!, \tag{8}$$

где $c_1 \approx 0{,}5528588554$, $a_1 \approx 3{,}583409309$.

Доказательство. В обозначениях теоремы Флажоле — Седжвика имеем $PBB_n = \frac{(n-1)!}{n} F(N, n)$,

$$N = n, \quad \lambda = 1, \quad a(z) = z, \quad b(z) = \exp\left(e^z\left(z + \frac{z^2}{2}\right) - \frac{z^3}{2} - \frac{3}{2}z^2\right).$$

Ряд для функции $b(z)$ имеет бесконечный радиус сходимости, $b(0) = 1$. У этого ряда положительные коэффициенты, так как $b(z) = \exp(\overline{B}\,'(z))$ и $\overline{B}(z)$ — производящая функция для числа помеченных блоков частного вида. С помощью Maple найдём $b_1 = 1$, $b_2 = 1/2$. Следовательно, НОД$\{j \mid b_j > 0\} = 1$. Таким образом, условия 1–3 теоремы Флажоле — Седжвика выполнены.

Далее найдём

$$\phi(z) = z\frac{b'(z)}{b(z)} = z(\ln(b(z)))' = z\left(e^z\left(1 + 2z + \frac{z^2}{2}\right) - \frac{3}{2}z^2 - 3z\right),$$

$$T = \lim_{z \to +\infty} \phi(z) = +\infty.$$

Так как $0 < \lambda < T$, решая уравнение $\phi(r) = 1$ с помощью Maple, найдём, что оно имеет два действительных корня $r_1 \approx -2{,}1115422291$ и $r_2 \approx 0{,}5528588554$, из которых один положительный.

В этом случае вместо асимптотического равенства для $F(N, n)$ в силу утверждения Флажоле — Седжвика [11, утверждение VIII.7] имеем оценку

$$F(N, n) \leqslant a(r_2)(b(r_2))^n r_2^{-n} = c_1 a_1^n,$$

где $c_1 = a(r_2) = r_2 \approx 0{,}5528588554$, $a_1 = b(r_2)/r_2 \approx 3{,}583409309$.

Поскольку $PBB_n = \frac{(n-1)!}{n} F(N, n)$, получим (8). Теорема 7 доказана.

В табл. 1 представлены числа BB_n, $\overline{BB}_n$, PBB_n, вычисленные с помощью формул (2), (5) и (7) (для $n = 3$ непосредственное вычисление).

Таблица 1

n	3	4	5	6	7	8	9	10
BB_n	3	19	195	2701	46473	956999	23039103	636197161
$\overline{BB}_n$	0	3	10	25	686	8519	69546	12112621
PBB_n	3	19	195	2691	46018	941508	22529601	618584005

ЛИТЕРАТУРА

1. **Hou Y., Sun Y.** Inverse of distance matrix of a bi-block graphs // Linear Multilinear Algebra. 2016. V. 64, No. 8. P. 1509–1517.
2. **McDiarmid C., Scott A.** Random graphs from a block stable class // Eur. J. Comb. 2016. V. 58. P. 96–106.
3. **Singh R.** Permanent, determinant, and rank bi-block graphs // Aequationes Math. 2020. V. 94, No. 1. P. 1–12.
4. **Das J., Mohanty S.** On spectral radius of bi-block graphs with given independence number α // Appl. Math. Comput. 2021. V. 402. Paper ID 125912. 8 p.
5. **Harary F., Robinson R. W.** Labeled bipartite blocks // Can. J. Math. 1979. V. 31, No. 1. P. 60–68.
6. **Gross J. L., Yellen J.** Graph theory and its applications. New York: Chapman and Hall/CRC, 2005. 800 p.
7. **Харари Ф.** Теория графов. М.: Мир, 1973. 299 с.
8. **Воблый В. А.** Об одной формуле для числа помеченных связных графов // Дискрет. анализ и исслед. операций. 2012. Т. 19, № 4. С. 48–59.
9. **Гульден Я., Джексон Д.** Перечислительная комбинаторика. М.: Наука, 1990. 504 с.
10. **Воблый В. А.** Второе соотношение Риддела и следствия из него // Дискрет. анализ и исслед. операций. 2019. Т. 26, № 1. С. 20–32.

11. Flajolet P., Sedgewick R. Analytic combinatorics. Cambridge, UK: Camb. Univ. Press, 2009. 810 p.

12. Воблый В. А. О перечислении помеченных связных графов без мостов // Итоги науки и техники. Сер. Соврем. математика и её прил. Темат. обзоры. Т. 223. М.: ВИНИТИ РАН, 2023. С. 138–147.

Воблый Виталий Антониевич

Статья поступила
12 июля 2023 г.
После доработки —
4 августа 2023 г.
Принята к публикации
20 августа 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
October–December 2023. Vol. 30, No. 4. P. 24–34

UDC 519.175.3 DOI 10.33048/daio.2023.30.778

ENUMERATION OF LABELED BI-BLOCK GRAPHS

V. A. Voblyi

All-Russian Institute for Scientific and Technical Information RAS
20 Usievich Street, 125190 Moscow, Russia
E-mail: vitvobl@yandex.ru

Abstract. A bi-block graph is a connected graph in which all blocks are complete bipartite graphs. Labeled bi-block graphs and bridgeless bi-block graphs are enumerated exactly and asymptotically by the number of vertices. It is proved that almost all labeled connected bi-block graphs have no bridges. In addition, planar bi-block graphs are enumerated, and an asymptotic estimate is found for the number of such graphs. Tab. 1, bibliogr. 12.

Keywords: enumeration, labeled graph, bridgeless graph, complete bipartite graph, planar graph, block, bi-block graph, random graph, asymptotics.

REFERENCES

1. **Y. Hou** and **Y. Sun,** Inverse of distance matrix of a bi-block graphs, *Linear Multilinear Algebra* **64** (8), 1509–1517 (2016).
2. **C. McDiarmid** and **A. Scott,** Random graphs from a block stable class, *Eur. J. Comb.* **58**, 96–106 (2016).
3. **R. Singh,** Permanent, determinant, and rank bi-block graphs, *Aequationes Math.* **94** (1), 1–12 (2020).
4. **J. Das** and **S. Mohanty,** On spectral radius of bi-block graphs with given independence number α, *Appl. Math. Comput.* **402**, Paper ID 125912 (2021).
5. **F. Harary** and **R. W. Robinson,** Labeled bipartite blocks, *Can. J. Math.* **31** (1), 60–68 (1979).
6. **J. L. Gross** and **J. Yellen,** *Graph Theory and Its Applications* (Chapman and Hall/CRC, New York, 2005).
7. **F. Harary,** *Graph Theory* (Addison-Wesley, London, 1969; Mir, Moscow, 1973 [Russian]).

English version: Journal of Applied and Industrial Mathematics **17** (4), 901–907 (2023), DOI 10.1134/S1990478923040178.

8. **V. A. Voblyi,** On a formula for the number of labeled connected graphs, *Diskretn. Anal. Issled. Oper.* **19** (4), 48–59 (2012) [Russian].
9. **I. P. Goulden** and **D. M. Jackson,** *Combinatorial Enumeration* (John Wiley & Sons, New York, 1983; Nauka, Moscow, 1990 [Russian]).
10. **V. A. Voblyi,** The second Riddell relation and its consequences, *Diskretn. Anal. Issled. Oper.* **26** (1), 20–32 (2019) [Russian] [*J. Appl. Ind. Math.* **13** (1), 168–174 (2019)].
11. **P. Flajolet** and **R. Sedgewick,** *Analytic Combinatorics* (Camb. Univ. Press, Cambridge, UK, 2009).
12. **V. A. Voblyi,** On enumeration of labeled connected bridgeless graphs, in *Itogi Nauki Tekh., Ser. Sovrem. Mat. Prilozh., Temat. Obz.,* Vol. 223 (VINITI RAN, Moscow, 2023), pp. 138–147 [Russian].

Vitaly A. Voblyi

Received July 12, 2023
Revised August 4, 2023
Accepted August 20, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Октябрь–декабрь 2023. Т. 30, № 4. С. 35–45

УДК 519.7 DOI 10.33048/daio.2023.30.775

О СООТНОШЕНИЯХ, СВЯЗАННЫХ С ФУНКЦИЕЙ ЭЙЛЕРА

В. К. Леонтьев[a], *Э. Н. Гордеев*[b]

Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН,
ул. Вавилова, 40, 119333 Москва, Россия

E-mail: [a] vkleontiev@yandex.ru, [b] werhorn@yandex.ru

Аннотация. Исследуются свойства множества чисел, меньших и взаимно простых с n, с введённой на нём операцией умножения по модулю n (этот объект иногда называют группой Эйлера). Мощность такого множества — известная функция Эйлера $\varphi(n)$, которая является одной из классических функций теории чисел. Области её применения достаточно широкие и включают, например, различные разделы дискретной математики, а также имеют существенные приложения в криптографии. В работе рассматриваются различные комбинаторные задачи, возникающие при исследовании группы Эйлера и функции Эйлера. Выведены соотношения между теоретико-числовыми параметрами, связанными с группой Эйлера и функцией Эйлера. Полученные в работе комбинаторные соотношения могут быть использованы при решении прикладных комбинаторных проблем и в криптографии. Библиогр. 10.

Ключевые слова: делитель числа, функция Эйлера, группа Эйлера, числа Стирлинга, функция Мёбиуса, производящая функция.

Введение

Функция Эйлера $\varphi(n)$ — число натуральных чисел, меньших и взаимно простых с n — одна из классических функций теории чисел, встречающаяся в различных разделах дискретной математики.

Группой Эйлера называется мультипликативная группа взаимно простых с n вычетов по модулю n (см., например, [1]). Таким образом, группа Эйлера является коммутативной группой порядка $\varphi(n)$.

Функции Эйлера посвящены многочисленные работы известных математиков — Ферма, Эйлера, Гаусса, Лежандра, Якоби и др. (см., например, [2–4]). Группа Эйлера также привлекала внимание многих исследователей (см., например, [1, 4, 5]). Подобные исследования актуальны

в настоящее время как для прикладных проблем в дискретной математике (см., например, [1, 4]), так и в криптографии [3]. Функция Эйлера и её свойства до сих пор привлекают внимание исследователей [1, 6, 7].

Некоторые из приведённых здесь результатов были изложены нами ранее в работе [8].

В гл. 17 тома 3 классической справочной книги [9] сформулирован ряд задач и дан обзор некоторых комбинаторных соотношений в исследуемой нами области (с. 195–200). Таким образом, полученные в работе результаты дополняют и расширяют поднятую ранее проблематику.

1. Основная часть

Пусть n — натуральное число и

$$M_n = \{x \in \mathbb{N} \mid (x, n) = 1,\ x \leqslant n\}.$$

Таким образом, M_n — это множество натуральных чисел, не превосходящих n и взаимно простых с n ((a, b) — как обычно — это наибольший общий делитель чисел a и b). Множество M_n с операцией умножения по модулю n является группой $\mathcal{M}_n$, которая носит имя Эйлера. Число элементов $\varphi(n)$ группы $\mathcal{M}_n$ — это функция Эйлера.

Два классических выражения для функция $\varphi(n)$ хорошо известны:

$$\varphi(n) = n \sum_{d \,|\, n} \frac{\mu(d)}{d},$$

$$\varphi(n) = n \prod_{p \,|\, n} \left(1 - \frac{1}{p}\right).$$

Здесь $\mu(d)$ — это функция Мёбиуса, определяемая следующим образом. Пусть $d = p_1^{\gamma_1} p_2^{\gamma_2} \dots p_k^{\gamma_k}$ — каноническое представление натурального числа в виде произведения степеней простых чисел. Тогда

$$\mu(d) = \begin{cases} 1, & \text{если } d = 1, \\ 0, & \text{если } \gamma_i \geqslant 2 \text{ для некоторого } i, \\ (-1)^{\sum\limits_{i=1}^{k} \gamma_i} & \text{иначе.} \end{cases} \tag{1}$$

Ниже будут приведены доказательства нескольких соотношений, связанных со свойствами группы Эйлера $\mathcal{M}_n$.

Теорема 1. *Справедливо соотношение*

$$\mathbb{F}_n(z) = \sum_{(m,n)=1} z^m = \sum_{d \,|\, n} \frac{\mu(d)}{1 - z^d}. \tag{2}$$

Доказательство. Пусть ξ_p^m — предикат делимости натурального m на простое число p, т. е. $\xi_p^m = 1$, если p делит m, и $\xi_p^m = 0$ в противном случае. Тогда при $n = p_1^{\alpha_1} p_2^{\alpha_2} \dots p_k^{\alpha_k}$ имеем

$$\sum_{(m,n)=1} z^m = \sum_{m=0}^{\infty} z^m \bigl(1 - \xi_{p_1}^m\bigr) \dots \bigl(1 - \xi_{p_k}^m\bigr).$$

Отсюда при $|z| < 1$ последовательно получаем

$$\mathbb{F}_n(z) = \sum_{m=0}^{\infty} z^m - \sum_i \sum_{m=0}^{\infty} z^m \xi_{p_i}^m + \sum_{i<j} \sum_{m=0}^{\infty} z^m \xi_{p_i}^m \xi_{p_j}^m - \dots.$$

Далее заметим, что

$$\sum_{m=0}^{\infty} z^m \xi_{p_i}^m = \sum_{r=0}^{\infty} z^{rp_i} = \frac{1}{1 - z^{p_i}},$$
$$\sum_{m=0}^{\infty} z^m \xi_{p_i}^m \xi_{p_j}^m = \sum_{r=0}^{\infty} z^{rp_ip_j} = \frac{1}{1 - z^{p_ip_j}}$$

и т. д., поэтому выполняется соотношение

$$\sum_{(m,n)=1} z^m = \sum_{m=0}^{\infty} z^m - \sum_{i=1}^{k} \sum_{r=0}^{\infty} z^{rp_i} + \sum_{1 \leqslant i < j \leqslant k} \sum_{r=0}^{\infty} z^{rp_ip_j} - \dots$$
$$= \frac{1}{1-z} - \sum_{p \,|\, n} \frac{1}{1 - z^p} + \sum_{\substack{p<q, \\ p \,|\, n,\, q \,|\, n}} \frac{1}{1 - z^{pq}} - \dots. \quad (3)$$

Выражение (2) можно получить из (3) с использованием свойства (1) функции Мёбиуса. Теорема 1 доказана.

Пусть $M_n = \{1, r_2, \dots, r_N\}$ — элементы группы Эйлера $\mathcal{M}_n$. Здесь $N = \varphi(n)$.

Теорема 2. *Справедлива формула*

$$\varphi_n(z) = \sum_{k=1}^{N} z^{r_k} = \sum_{d \,|\, n} \mu(d) \, \frac{1 - z^n}{1 - z^d}. \quad (4)$$

Доказательство. Пусть $N_n = \{m\}$ — все натуральные числа, взаимно простые с n. Тогда

$$m = x \cdot n + r_i$$

для некоторого натурального x и $r_i \in M_n$. Действительно, делим m на n и отмечаем, что $(n, r_i) = 1$. Отсюда

$$\sum_{(m,n)=1} z^m = \sum_{x,\, r_i} z^{xn+r_i} = \sum_x z^{xn} \sum_{i=1}^{N} z^{r_i} = \frac{1}{1-z^n} \sum_{i=1}^{N} z^{r_i}.$$

Из теоремы 1 получаем

$$\sum_{k=1}^{N} z^{r_k} = \sum_{d\,|\,n} \mu(d) \frac{1-z^n}{1-z^d}.$$

Теорема 2 доказана.

Следствие 1. *Имеет место равенство*

$$N = \varphi(n) = n \sum_{d\,|\,n} \frac{\mu(d)}{d}.$$

Доказательство. Переходя в формуле (4) к пределу при $z \to 1$, получаем

$$N = \varphi(n) = n \sum_{d\,|\,n} \frac{\mu(d)}{d}.$$

Следствие 1 доказано.

Следствие 2. *Имеет место равенство*

$$\varphi(n) = n \prod_{p\,|\,n} \Big(1 - \frac{1}{p}\Big).$$

Доказательство. В силу (1) каждый делитель d в формуле (2) имеет вид $d = p_1^{\gamma_1} p_2^{\gamma_2} \dots p_k^{\gamma_k}$, где $\gamma_i \in \{0, 1\}$, $i = 1, 2, \dots, k$. Отсюда

$$\varphi(n) = n \sum_{\{\gamma_1,\dots,\gamma_k\}} \frac{(-1)^{\gamma_1+\gamma_2+\dots+\gamma_k}}{p_1^{\gamma_1} p_2^{\gamma_2} \dots p_k^{\gamma_k}}$$

$$= n \sum_{\gamma_1=0}^{1} \frac{(-1)^{\gamma_1}}{p_1^{\gamma_1}} \sum_{\gamma_2=0}^{1} \frac{(-1)^{\gamma_2}}{p_2^{\gamma_2}} \cdots \sum_{\gamma_k=0}^{1} \frac{(-1)^{\gamma_k}}{p_k^{\gamma_k}} = n \prod_{p\,|\,n} \Big(1 - \frac{1}{p}\Big).$$

Следствие 2 доказано.

Следствие 1 можно вывести из тождества Гаусса

$$\sum_{d\,|\,n} \varphi(n) = n$$

с помощью формулы обращения Мёбиуса. Вывод следствия 2 также известен. Однако формула (4) позволяет получить гораздо больше информации об элементах группы $\mathcal{M}_n$, чем приведённые следствия из неё. Действительно, с помощью (4) можно в «явном» виде получить любые значения симметрической функции от $(r_1, r_2, \ldots, r_N)$. Имеющиеся утверждения поясняют это замечание.

Следствие 3. *Справедлива формула*

$$\sum_{r\in M_n} r = \frac{n\varphi(n)}{2}. \tag{5}$$

Доказательство. Так как при $n = s \cdot d$ выполняется соотношение

$$\frac{1-z^n}{1-z^d} = 1 + z^d + z^{2d} + \cdots + z^{(s-1)d},$$

из (4) после дифференцирования получаем

$$\varphi_n'(z) = \sum_{k=1}^{N} r_k z^{r_k-1} = \sum_{d\,|\,n} \mu(d) \sum_{r=1}^{\frac{n}{d}-1} rdz^{rd-1}.$$

Отсюда при $z = 1$ имеем цепочку соотношений

$$\sum_{k=1}^{N} r_k = \sum_{d\,|\,n} d\mu(d)\binom{n/d}{2} = \frac{1}{2}\sum_{d\,|\,n} d\mu(d)\Big(\frac{n}{d}-1\Big)\frac{n}{d}$$
$$= \frac{n}{2}\sum_{d\,|\,n}\mu(d)\Big(\frac{n}{d}\Big) - \frac{n}{2}\sum_{d\,|\,n}\mu(d) = \frac{n}{2}\varphi(n).$$

Так как $\sum\limits_{d\,|\,n}\mu(d) = 0$, если $n \neq 1$, окончательно имеем

$$\sum_{r_k\in M_n} r_k = \frac{n}{2}\varphi(n).$$

Следствие 3 доказано.

Замечание. Формулу (5) можно получить без всяких вычислений, если заметить, что из того, что $a \in M_n$, вытекает, что $n - a \in M_n$, и отсюда

$$\sum_{a\in M_n} a = \sum_{a\in M_n} (n-a),$$

что и доказывает (5).

Аналогичные вычисления приводят к следующей формуле [3, 4].

Следствие 4. *Справедливо выражение*

$$\sum_{r\in M_n} r^2 = \frac{n^2}{3}\,\varphi(n) - \frac{\varphi(n)}{6}\,p_1p_2\dots p_k,$$

где $n = p_1^{\alpha_1}p_2^{\alpha_2}\dots p_k^{\alpha_k}$.

В общем случае, если

$$\varphi_n(z) = \sum_{r_m\in M_n} z^{r_m} = \sum_{a\in M_n} z^a,$$

то

$$\varphi_n^{(k)}(z) = \sum_{r_m\in M_n} r_m(r_m-1)\dots(r_m-k+1)z^{r_m-k} = \sum_{a\in M_n} a(a-1)\dots(a-k+1)z^{a-k}.$$

Так как

$$(a)_k = a(a-1)\dots(a-k+1) = \sum_{r=1}^{k} s(k,r)a^r,$$

где $s(k,r)$ — числа Стирлинга I рода (см. [2]), то

$$\varphi_n^{(k)}(z) = \sum_{a\in M_n} z^{a_m-k}\sum_{r=1}^{k} s(k,r)a^r = \sum_{r=1}^{k} s(k,r)\sum_{a\in M_n} a^r z^{a-k}.$$

Теорема 3. *Справедлива формула*

$$\Phi_n^{(k)}(1) = \sum_{r=1}^{k} s(k,r)\sum_{a\in M_n} a^r. \tag{6}$$

Если

$$S_n(r) = \sum_{a\in M_n} a^r,$$

то из (6) получаем формулу

$$\Phi_n^{(k)}(1) = \sum_{r=1}^{k} s(k,r)S_n(r),$$

которая позволяет последовательно вычислять $S_n(r)$ для $r = 0, 1, \dots$, исходя из значений $\Phi_n^{(k)}(1)$.

Пусть σ_k — k-й элементарный симметрический полином. Тогда, выражая элементарные симметрические полиномы через степенные функции $S_n(r)$, в силу классического соотношения (см., например, [10, § 52])

$$\prod_{x\in M_n}(z-x)=\sum_{k=1}^{\varphi(n)}(-1)^k z^k\sigma_k$$

можно «явно» вычислить элементы группы M_n.

Для множества $M=\{a_1,a_2,\ldots,a_r\}$ введём обозначение

$$v\cdot M=\{va_1,va_2,\ldots,va_r\}.$$

Пусть

$$M_n^t=\{a\in\mathbb{N}\mid (a,n)=t,\ a\leqslant n\}.$$

Тогда для M_n^t справедливо следующее представление.

Теорема 4. *Имеет место равенство при $t\mid n$.*

$$M_n^t=t\cdot M_{n/t}^1.$$

Доказательство вытекает из того, что при $(a,b)=t$ выполняется равенство $\left(\frac{a}{t},\frac{b}{t}\right)=1$. Теорема 4 доказана.

Примеры. 1. Если $t=2$ и $n=12$, то

$$M_{12}^2=2\cdot\{1,5,7,11\}=\{2,10,2,10\}=\{2,10\}.$$

Действительно, $\left|M_{12}^2\right|=\left|M_6^1\right|=|\{1,5\}|=2$, т. е. $M_6^1=\{1,5\}$, $M_{12}^2=2\cdot\{1,5\}=\{2,10\}$.

2. Если $t=4$ и $n=8$, то

$$M_8^4=4\cdot\{1,3,5,7\}=\{4,4,4,4\}=\{4\}.$$

Действительно, $M_8^4=4\cdot M_2^1=4\cdot\{1\}=\{4\}$.

В терминах производящих функций все приведённые выше рассуждения выглядят так. Пусть

$$F_n^t(z)=\sum_{(x,n)=t}z^x.$$

Теорема 5. *Справедливо соотношение*

$$F_{n,d}^t(z)=\sum_{d\,|\,\frac{n}{t}}\frac{\mu(d)}{1-z^{td}}.$$

Следствие 5. *Имеет место формула*

$$\sum_{a\in M_n^t}z^a=\sum_{d\,|\,\frac{n}{t}}\mu(d)\frac{1-z^n}{1-z^{td}}.$$

Следствие 6. *При $t\mid n$ справедливо равенство*

$$\left|M_n^t\right|=\varphi\left(\frac{n}{t}\right).$$

Пусть

$$F_n^t(z) = \sum_{x=1}^{n} z^{(n,x)}.$$

Утверждение 1. *Имеет место формула*

$$F_n^t(z) = \sum_{d \mid n} z^d \varphi\Big(\frac{n}{t}\Big).$$

Пусть $\varphi_n(a)$ — число меньших n и взаимно простых с числом a чисел, $\tau_p(n)$ — число простых делителей n, $M_p(n)$ — множество простых делителей числа n. Положим для краткости $k = \tau_p(n)$.

Теорема 6. *Имеет место соотношение*

$$\varphi_n(a) = n - \sum_{i=1}^{k} \Big[\frac{n}{p_i}\Big] + \sum_{1 \leqslant i < j \leqslant k} \Big[\frac{n}{p_i p_j}\Big] - \dots,$$

где $a = p_1^{r_1} p_2^{r_2} \dots p_k^{r_k}$.

При этом заметим, что $\varphi_n(n) = \varphi(n)$ и $\varphi_n(a) = \frac{n}{a}\varphi(a)$, если $M_p(n) = M_p(a)$.

Далее приведём ещё три соотношения.

1) $\varphi(n^r) = n^{r-1}\varphi(n)$.

2) Пусть $M_p(a) \subseteq M_p(mn)$. Тогда

$$\varphi_{mn}(a) = mn - \sum_{p \mid a} \Big[\frac{mn}{p}\Big] + \sum_{pq \mid a} \Big[\frac{mn}{pq}\Big] - \dots$$

$$= mn - \sum_{p \mid a} \frac{mn}{p} + \sum_{pq \mid a} \frac{mn}{pq} - \dots = mn \prod_{p \mid a} \Big(1 - \frac{1}{p}\Big).$$

Отсюда

$$\varphi_{mn}(a) = \frac{mn}{a}\varphi(a).$$

3) Если $M_p(a) = M_p(mn)$, то $\varphi_{mn}(a) = \varphi(mn)$.

ЛИТЕРАТУРА

1. **Арнольд В. И.** Группы Эйлера и арифметика геометрических прогрессий. М.: МЦНМО, 2003. 44 с.
2. **Сачков В. Н.** Введение в комбинаторные методы дискретной математики. М.: МЦНМО, 2004.
3. **Алфёров А. П., Зубов А. Ю., Кузьмин А. С., Черёмушкин А. В.** Основы криптографии. М.: Гелиос АРВ, 2002. 480 с.

4. **Грэхем Р., Кнут Д., Паташник О.** Конкретная математика. Основание информатики. М.: Мир, 1998.
5. **Hardy G. H., Wright E. M.** An introduction to the theory of numbers. Oxford: Clarendon Press, 1979. 426 p.
6. **Shramm W.** The Fourier transform of functions of the greatest common divisor // INTEGERS: Electron. J. Comb. Number Theory. 2008. V. 8. Paper ID A50. 7 p.
7. **Coleman R.** Some remarks on Euler's totient function. Ithaca, NY: Cornell Univ., 2012. (Cornell Univ. Libr. e-Print Archive; arXiv:1207.4446).
8. **Леонтьев В. К.** Комбинаторика и информация. Ч. 1. Комбинаторный анализ. М.: МФТИ, 2015. 174 с.
9. **Бейтмен Г., Эрдейи А.** Высшие трансцендентные функции. Т. 3. М.: Наука, 1967. 296 с.
10. **Курош А. Г.** Курс высшей алгебры. М.: Наука, 1968. 431 с.

Леонтьев Владимир Константинович
Гордеев Эдуард Николаевич

Статья поступила
23 мая 2023 г.
После доработки —
2 августа 2023 г.
Принята к публикации
20 августа 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
October–December 2023. Vol. 30, No. 4. P. 35–45

UDC 519.7 DOI 10.33048/daio.2023.30.775

ON RELATIONS ASSOCIATED WITH THE EULER FUNCTION

V. K. Leontiev[a] and *E. N. Gordeev*[b]

Dorodnitsyn Computing Center RAS,
40 Vavilov Street, 119333 Moscow, Russia
E-mail: [a] vkleontiev@yandex.ru, [b] werhorn@yandex.ru

Abstract. The paper studies the properties of the set of numbers smaller than and coprime to n with the modulo n multiplication operation introduced on it (this object is sometimes called the Euler group). The cardinality of such a set is the well-known Euler function $\varphi(n)$, which is one of the classical functions in the number theory. The fields of its application are quite wide and include, for example, various branches of discrete mathematics, and it also has significant applications in cryptography. The paper considers various combinatorial problems arising in the study of the Euler group and the Euler function. Relations between theoretical and numerical parameters associated with the Euler group and Euler function are derived. The combinatorial relations obtained in the paper can be used when solving applied combinatorial problems and in cryptography. Bibliogr. 10.

Keywords: divisor, Euler function, Euler group, Stirling numbers, Möbius function, generating function.

REFERENCES

1. **V. I. Arnold,** *Euler Groups and Arithmetics of Geometric Progression* (MTsNMO, Moscow, 2003) [Russian].
2. **V. N. Sachkov,** *An Introduction to Combinatorial Methods of Discrete Mathematics* (MTsNMO, Moscow, 2004) [Russian].
3. **A. P. Alfyorov, A. Yu. Zubov, A. S. Kuzmin,** and **A. V. Cheryomushkin,** *Basics of Cryptography* (Gelios ARV, Moscow, 2002) [Russian].
4. **R. L. Graham, D. E. Knuth,** and **O. Patashnik,** *Concrete Mathematics: A Foundation for Computer Science* (Addison-Wesley, Reading, MA, 1994; Mir, Moscow, 1998 [Russian]).

English version: Journal of Applied and Industrial Mathematics **17** (4), 760–766 (2023), DOI 10.1134/S1990478923040075.

5. **G. H. Hardy** and **E. M. Wright,** *An Introduction to the Theory of Numbers* (Clarendon Press, Oxford, 1979).
6. **W. Shramm,** The Fourier transform of functions of the greatest common divisor, *INTEGERS: Electron. J. Comb. Number Theory* **8**, Paper ID A50 (2008).
7. **R. Coleman,** Some remarks on Euler's totient function (Cornell Univ., Ithaca, NY, 2012) (Cornell Univ. Libr. e-Print Archive, arXiv:1207.4446).
8. **V. K. Leontiev,** *Combinatorics and Information. Pt. 1. Combinatorial Analysis* (Mosk. Fiz. Tekh. Inst., Moscow, 2015) [Russian].
9. **H. Bateman** and **A. Erdélyi,** *Higher Transcendental Functions,* Vol. 3 (New York, McGraw-Hill Book Co., 1953; Nauka, Moscow, 1967 [Russian]).
10. **A. G. Kurosh,** *A Course in Higher Algebra* (Nauka, Moscow, 1968) [Russian].

Vladimir K. Leontiev
Eduard N. Gordeev

Received May 23, 2023
Revised August 2, 2023
Accepted August 20, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Октябрь–декабрь 2023. Т. 30, № 4. С. 46–90

УДК 519.7 DOI 10.33048/daio.2023.30.771

ПОСТКВАНТОВЫЕ КРИПТОСИСТЕМЫ: ОТКРЫТЫЕ ВОПРОСЫ И СУЩЕСТВУЮЩИЕ РЕШЕНИЯ. КРИПТОСИСТЕМЫ НА РЕШЁТКАХ

Е. С. Малыгина[1,2,a], *А. В. Куценко*[2,b], *С. А. Новосёлов*[1,c], *Н. С. Колесников*[1,d], *А. О. Бахарев*[2,e], *И. С. Хильчук*[2,f], *А. С. Шапоренко*[2,g], *Н. Н. Токарева*[2,1,h]

[1] Балтийский федеральный университет им. И. Канта,
ул. Александра Невского, 14, 236041 Калининград, Россия

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

E-mail: [a] emalygina@kantiana.ru, [b] alexandrkutsenko@bk.ru, [c] novsem@gmail.com, [d] nikolesnikov100@gmail.com, [e] a.bakharev@g.nsu.ru, [f] irina.khilchuk@gmail.com, [g] shaporenko.alexandr@gmail.com, [h] crypto1127@mail.ru

Аннотация. Постквантовая криптография является актуальной областью теоретических и прикладных исследований, включающей в себя разработку и анализ методов криптографической защиты информации, применяемых в условиях широкого использования квантовых вычислений. В работе приведён обзор основных подходов к построению постквантовых криптографических систем, используемых в настоящее время. Подробно рассмотрено направление, в рамках которого предлагаются криптосистемы, стойкость которых основывается на вычислительной трудности ряда задач из теории решёток, представлен сложностной статус данных задач. Приведено описание и характеристики некоторых известных криптосистем, стойкость которых основана на сложности таких задач, как

Работа первого, третьего и четвёртого авторов выполнена при поддержке Северо-западного центра математических исследований им. С. Ковалевской (БФУ им. И. Канта) в рамках соглашения с Министерством науки и высшего образования России (соглашение № 075–02–2023–934). Работа второго, пятого, шестого, седьмого и восьмого авторов выполнена при поддержке Математического центра в Академгородке в рамках соглашения с Министерством науки и высшего образования России (соглашение № 075–15–2022–282).

задача нахождения кратчайшего вектора, задача обучения с ошибками, а также их вариаций. Разобраны основные подходы к решению задач из теории решёток, лежащие в основе атак на соответствующие криптосистемы. В частности, приведены теоретические оценки времени работы и объёма используемой памяти для известных алгоритмов редукции и просеивания решёток. Табл. 6, ил. 1, библогр. 93.

Ключевые слова: постквантовая криптография, квантовый компьютер, целочисленная решётка.

Введение

Термин «постквантовая криптография» появился в середине 2000-х гг. Он фигурировал в названии международной конференции «PQCrypto 2006: International Workshop on Post-Quantum Cryptography», проведённой в 2006 г. в университете г. Лёвена (Бельгия) и являвшейся первым крупным научным собранием, посвящённым вопросам развития криптографии в условиях существования квантового компьютера. В течение нескольких лет конференция проводилась раз в два года, а с 2016 г. стала ежегодно организуемым мероприятием, собирающим большое число исследователей со всего мира, а также крупнейшей площадкой для обсуждения перспектив и открытых проблем в данной области.

В 2009 г. под редакторством Бернштейна, Бухманна и Дамена был опубликован сборник работ «Post-quantum cryptography». Этот сборник представляется первой попыткой собрать, проанализировать и структурировать информацию касательно нового направления криптографии, предпринятой различными авторами. В частности, в его введении был приведён обзор основных направлений постквантовой криптографии, актуальный и на сегодняшний день [1]. С тех пор интерес к постквантовой криптографии с каждым годом лишь возрастает. Данный рост обусловлен, главным образом, постоянным увеличением числа логических кубитов, с которыми может работать универсальный квантовый компьютер (табл. 1) и, как следствие, увеличением его мощности.

В широком смысле термин «постквантовая криптография» включает в себя разработку и исследование тех криптографических систем и подходов к их построению, которые останутся актуальными и после появления квантового компьютера, располагающего достаточными мощностями для реализации алгоритмов квантового криптоанализа. На данный момент с учётом наличия зашумлённости, возникновения ошибок и необходимости задействовать алгоритмы их исправления при выполнении квантовых вычислений требуемое количество кубитов варьируется от десятков тысяч до нескольких миллионов (см., например, [2]). Вместе с тем, в силу того, что квантовые вычисления позволяют реализовать

Таблица 1

Некоторые известные квантовые процессоры

Название	Разработчик	Число кубит	Год разработки
IBM Osprey	IBM (США)	433	2022 (ноябрь)
Borealis	Xanadu (Канада)	216	2022 (июнь)
IBM Eagle	IBM (США)	127	2021
Jiuzhang	Научно-технический университет Китая (КНР)	76	2020
Sycamore	Google (США)	53	2019
Tangle Lake	Intel (США)	49	2018
Advantage[1)]	D-Wave (Канада)	5640	2020

и классические алгоритмы, речь идёт об устойчивости уже к *квантово-классическому* криптоанализу. На данный момент не известен полиномиальный квантовый алгоритм решения какой-либо NP-полной или NP-трудной задачи, что допускает возможность существования постквантовых криптографических систем. При этом природа квантовых вычислений, а также их потенциальные возможности оставляют большое число открытых вопросов. Это несомненно усложняет исследования и позволяет говорить об устойчивости и актуальности той или иной криптографической системы, лишь исходя из *текущего* состояния развития квантовых вычислений, оставляя возможность ситуации измениться в будущем.

Постквантовая криптография подразумевает развитие именно классической криптографии, без использования квантовых эффектов. При этом квантовые технологии могут быть использованы и для решения криптографических задач: в частности, широкое развитие получило направление, известное под названием *квантовое распределение ключей* (quantum key distribution), которое при наличии дополнительного описания функции шифрования во многих случаях именуется *квантовой криптографией*. Начало данному направлению положила работа Беннетта и Брассара, опубликованная в 1984 г., в которой был описан первый алгоритм квантового распределения ключей, получивший название BB84 (см. [3]). В рамках него с помощью квантово-механических эффектов решается задача распределения секретного ключа между двумя абонентами. Такой подход предполагает защиту от копирования передаваемого сигнала и даёт возможность установить наличие несанкционированного доступа к используемому квантовому каналу связи.

[1)]Квантовый компьютер D-Wave является примером адиабатического квантового вычислителя и предназначен для решения задач оптимизации

В криптографии традиционно выделяют две ветви — симметричную и асимметричную, которые в настоящее время в разной степени подвержены влиянию развития квантовых вычислений. Отметим, что при упоминании постквантовой криптографии, как правило, речь идёт об исследовании асимметричных алгоритмов, а также таких вычислительно сложных задач, сложность которых уже в квантовом окружении не меняется кардинальным образом.

1. Квантовый криптоанализ. Под *квантовым криптоанализом* подразумевается криптоанализ с доступом к квантовому компьютеру достаточной мощности, что в ряде случаев позволяет получить существенное ускорение при решении задач определённого типа.

История развития квантовых вычислений и построения квантового компьютера берёт начало с идеи того, что квантовые компьютеры могут быть более эффективны в задачах моделирования квантово-механических систем, что было впервые отражено в 1980 г. в книге советского математика Ю. И. Манина «Вычислимое и невычислимое» [4] и позднее отмечено в 1982 г. в работе «Simulating physics with computers» американского физика Фейнмана [5]. В 1985 г. Дойчем была описана модель универсального квантового компьютера [6], а в последующие годы был предложен ряд квантовых алгоритмов, в частности, алгоритмы Дойча, Дойча — Йожа [7], Бернштейна — Вазирани [8], Саймона [9] и др.

В основе эффекта, за счёт которого на квантовом компьютере достигается более высокая эффективность при решении определённых задач, лежит понятие *квантового параллелизма*. Он позволяет, располагая регистром из n логических квантовых бит (кубит), за одно обращение к квантовому преобразованию, моделирующему, например, некоторую булеву функцию f от n переменных, получить квантовое состояние, в котором все значения данной функции $\{f(x)\}_{x\in\mathbb{F}_2^n}$ специальным образом определяют суперпозицию соответствующих базисных состояний. Получаемое состояние подвергается дальнейшему воздействию с целью получения такого квантового состояния, результат измерения которого позволяет решить исходную задачу.

В рамках квантовой теории сложности в настоящее время установлена следующая цепочка включений:

$$\mathrm{P} \subseteq \mathrm{BPP} \subseteq \mathrm{BQP} \subseteq \mathrm{PP} \subseteq \mathrm{PSPACE},$$

где через BQP обозначен класс задач разрешимости, для которых существуют полиномиальные квантовые алгоритмы решения с ограниченной величиной ошибки (bounded error quantum polynomial time), тогда как обозначения P, BPP, PP, PSPACE соответствуют хорошо известным классам задач из классической теории сложности. Открытым вопросом

является отношение классов BPP и BQP относительно обратного включения, другими словами, верно ли, что $\mathrm{BQP} \not\subseteq \mathrm{BPP}$, из чего немедленно следует строгое включение $\mathrm{BPP} \subset \mathrm{BQP}$? Достоверная различность данных классов подтвердила бы теоретическое превосходство квантовых вычислений над классическими в смысле временной сложности.

Более подробную информацию про математический аппарат, используемый в квантовых вычислениях, квантовый параллелизм, а также известные классы квантовых алгоритмов можно найти в монографии Нильсена и Чанга [10].

Асимметричная криптография. Для асимметричной криптографии ключевое значение имела работа Шора [11], опубликованная в 1994 г. В ней был предложен *алгоритм Шора* — квантовый алгоритм, позволяющий за полиномиальное от $\log N$ число операций разложить на множители натуральное число N, т. е. за полиномиальное число операций решить задачу факторизации. Также был описан квантовый полиномиальный алгоритм решения задачи дискретного логарифмирования. На сегодняшний день все известные, в том числе вероятностные алгоритмы решения данных задач с использованием только классических вычислений требуют субэкспоненциального числа операций.

Наличие эффективных квантовых алгоритмов решения данных задач имеет ключевое значение в вопросе дальнейшего использования таких алгоритмов, как, например, протокол Эль-Гамаля, криптосистема RSA, протокол обмена ключами Диффи — Хеллмана. Это обусловлено тем, что в силу существования алгоритма Шора данные криптосистемы гарантированно не являются постквантовыми. В 2003 г. было показано, что алгоритм Шора может быть адаптирован для случая постановки задачи дискретного логарифмирования в группе точек эллиптической кривой [12], что, в свою очередь, имеет значение для соответствующих вариантов асимметричных криптосистем.

Симметричная криптография. Для симметричной криптографии ситуация имеет иной характер, так как на данный момент не предложено квантового алгоритма, существенным образом влияющего на стойкость симметричных криптосистем, переводящих их в разряд небезопасных.

Следует отметить предложенный в 1996 г. квантовый алгоритм — алгоритм Гровера [13]. В исходной постановке он позволяет отыскать в неупорядоченном списке из N элементов присутствующий в единственном экземпляре элемент с отличительным свойством за $O\left(\sqrt{N}\right)$ шагов — *итераций Гровера*. Применительно к симметричной криптографии алгоритм Гровера может быть использован для ускорения атаки полным перебором: вместо $O(K)$ опробований ключа, как в классическом случае, предлагается производить $O\left(\sqrt{K}\right)$ опробований, где K — мощность

пространства ключей. В 1998 г. Брассар, Хойер и Тапп предложили использовать алгоритм Гровера для более эффективного поиска коллизий хэш-функций определённого вида, этот алгоритм получил обозначение BHT [14]. Дальнейшее развитие алгоритма Гровера включает поиск в неупорядоченном списке без ограничений на единственность искомого элемента (так называемое усиление амплитуды, amplitude amplification), а также квантовый счёт (quantum counting) [15] — алгоритм оценки числа решений уравнения $f(x) = 1$, где f — булева функция.

Дальнейшее развитие квантового криптоанализа симметричных криптосистем предполагает более детальное рассмотрение конструкций шифров и комбинирование известных квантовых алгоритмов. Так, Кувакадо и Мори в 2012 г. предложили квантовую атаку на схему Эвена — Мансура [16]. Данная атака основана на использовании алгоритма Саймона нахождения периода функции. В [17] рассматриваются квантовые атаки на симметричный шифры, построенные на основе сети Фейстеля. Алгоритм Бернштейна — Вазирани нашёл своё применение в ряде квантовых атак на блочные шифры, что было показано в работе Се и Яна [18]. В работе Леандра и Мэя в 2017 г. предложена композиция алгоритмов Гровера и Саймона для описания возможной квантовой атаки на FX-конструкцию [19]. В ряде работ предлагается построение эффективного квантового алгоритма, позволяющего отличить конкретный шифр или криптографический примитив от случайной подстановки. Данный вопрос был изучен применительно к трёхраундовой сети Фейстеля в 2010 г. Кувакадо и Мори [20], а в отношении 7 и 8 раундов стандарта шифрования KHP SMS4 рассмотрен в работе Ходжича и Кнудсена 2020 г. [21].

Стоит отметить ряд работ, посвящённых исследованию возможности повышения эффективности статистических методов криптоанализа симметричных шифров с помощью квантовых алгоритмов: в частности, те работы, в которых были рассмотрены квантовый разностный и квантовый линейный криптоанализ [22, 23], а также известную работу с анализом обоих методов [24]. Кроме того, в [24] предложены следующие две модели стойкости блочных шифров в зависимости от действий криптоаналитика, использующего квантовый компьютер.

• Стандартная безопасность. Блочный шифр обладает *стандартной стойкостью*, если не известен эффективный квантовый алгоритм, который мог бы отличить блочный шифр от псевдослучайной подстановки (или псевдослучайной функции), выполняя только классические запросы. Обозначается через $Q1$.

• Квантовая безопасность. Блочный шифр обладает *квантовой стойкостью*, если не известен эффективный квантовый алгоритм, который может отличить блочный шифр от псевдослучайной подстановки (или

псевдослучайной функции) даже с помощью квантовых запросов. Обозначается через $Q2$.

2. Конкурс на стандарт постквантовой криптографии и основные подходы к построению. В 2016 г. Национальный институт стандартов и технологий США опубликовал отчёт «NISTIR 8105: Report on Post-Quantum Cryptography» [25], в котором было проанализировано влияние квантовых вычислений на тот момент (табл. 2) и описаны основные подходы к построению постквантовых криптосистем.

Таблица 2

Влияние на некоторые известные криптосистемы, используемые в настоящее время [25]

Название	Тип	Предназначение	Влияние квантовых вычислений
AES	Симметр.	Шифрование	Требуется бóльшая длина ключа
SHA-2, SHA-3	Хэш-функция	Хэширование	Требуется бóльшая длина выходной последовательности
RSA	Асимметр.	Подпись, установление ключа	Не безопасен
ECDSA, ECDH (криптография на эллиптических кривых)	Асимметр.	Подпись, обмен ключами	Не безопасен
DSA (криптография над конечными полями)	Асимметр.	Подпись, обмен ключами	Не безопасен

В конце 2017 г. Национальным институтом стандартов и технологий США был окончен приём заявок на участие в первом раунде конкурса, по итогам которого должен быть выбран стандарт постквантового асимметричного криптографического механизма для решения задач шифрования, установления общего секретного ключа, а также формирования электронной цифровой подписи [26]. Ниже дана краткая характеристика основных предложенных подходов к построению постквантовых криптосистем.

• Криптография на решётках (lattice-based cryptography). Математический аппарат таких криптосистем базируется на работе с целочисленными решётками или с решётками над конечными полями. Стойкость

основывается на вычислительной сложности решения ряда задач, например, нахождения кратчайшего или ближайшего вектора в решётке, так как эффективные атаки, как правило, сводятся к решению данных задач. Преимуществами данного направления являются относительная простота реализации, а также потенциальная возможность построить криптосистему, обладающую другими важными свойствами, например, обеспечивающую полностью гомоморфное шифрование. Немаловажен тот факт, что часть задач, лежащих в основе криптографии на решётках, принадлежит к классу NP-трудных задач.

• Криптография на основе кодов, исправляющих ошибки (code-based cryptography). Классическими криптосистемами, основанными на теории кодирования, являются криптосистемы Мак-Элиса (1978 г.) и Нидеррайтера (1986 г.). В них для процесса шифрования предлагается использовать некоторый линейный код, представленный порождающей или проверочной матрицей, замаскированной с помощью ряда преобразований. Структура получаемого кода является частью секретного ключа. На этапе зашифрования применяется процедура кодирования или вычисления синдрома, а расшифрование задействует декодирование. Сложность атаки обусловлена вычислительной сложностью решения задачи декодирования полного линейного кода — известной NP-трудной задачи, а также задачи восстановления структуры кода. К недостаткам подобных криптосистем можно отнести достаточно большой размер ключа.

• Криптография, основанная на изогениях суперсингулярных эллиптических кривых (isogenies on supersingular elliptic curves based cryptography). Криптосистемы на изогениях представляют собой сравнительно новое направление исследований, которое начало развиваться в 2000-х гг., начиная с работ Ростовцева, Столбунова и Кувейна. В основе таких криптосистем лежит сложность задачи вычисления изогении между двумя заданными эллиптическими кривыми над конечным полем. Главным преимуществом криптосистем на изогениях являются малые размеры ключей по сравнению с другими постквантовыми схемами. В то же время криптосистемы на изогениях обладают низкой скоростью работы, что делает их непрактичными в использовании. Были предприняты попытки оптимизировать работу таких криптосистем, используя дополнительную информацию об изогении — её значения в точках кручения, как в схеме обмена ключами SIDH. Однако, такой подход к оптимизации криптосистем не увенчался успехом — в 2022 г. Кастрик и Декру предложили полиномиальный алгоритм вычисления изогении при известных значениях в точках кручения.

• Криптография на основе хэш-функций (hash-based cryptography). Данный подход может быть использован для реализации электронной

цифровой подписи. Для решения этой задачи применяется одноразовая подпись, например, подпись Лэмпорта (1979 г.) и дерево хэш-значений — дерево Меркла (1979 г.). Стойкость к классическим и квантовым атакам основана на вычислительной сложности поиска коллизий хэш-функции как при использовании классических вычислений, так и квантовых. Недостатки данного направления включают необходимость в ряде случаев хранения всего списка подписанных ранее сообщений, что делает схемы чувствительными к наличию ошибок. Также недостатком является ограниченное число подписей, при этом увеличение этого числа приводит к изменению размера подписи.

• Криптография на основе многочленов от многих переменных (multivariate polynomial cryptography). Математическое описание подобных криптосистем представляет собой композицию нескольких отображений, линейных и нелинейных, заданных в полиномиальной форме и действующих на векторных пространствах над конечными полями. Атаки на них сводятся к решению нелинейных систем уравнений со многими неизвестными. В общем случае задача решения системы булевых уравнений степени 2 и выше NP-трудна, что обуславливает стойкость таких криптосистем. Криптографические алгоритмы такого вида могут быть использованы для реализации механизма электронной цифровой подписи.

В 2022 г. завершился третий раунд конкурса, и в результате для шифрования и установления общего ключа был выбран алгоритм CRYSTALS-Kyber, основанный на теории решёток, а для электронной цифровой подписи — алгоритмы CRYSTALS-Dilithium, FALCON, SPHINCS$^+$, базирующиеся на решётках и использовании хэш-функций. В заявку на чётвертый раунд конкурса вошли альтернативные кандидатуры, среди которых присутствуют криптосистемы, основанные на использовании кодов, исправляющих ошибки.

1. Решётки

1.1. Предварительные сведения. Решётки использовались в математике по меньшей мере с XVIII века, однако с вычислительной точки зрения мало исследовались. Так продолжалось до 1980-х гг., до появления алгоритма редукции базиса решётки LLL [27], предложенного А. Ленстрой, Х. Ленстрой и Ловасом в 1982 г. В том же 1982 г. Шамиром [28] был предложен полиномиальный алгоритм взлома криптосистемы Меркла — Хеллмана, основанной на задаче о рюкзаке. С использованием алгоритма LLL и его усовершенствований [29, 30] были взломаны протокол обмена ключами Блума [31] и криптосистемы, основанные на рациональных числах [32] и на задаче о рюкзаке [33].

В 1990-е гг. теория решёток нашла применение не только для взлома криптографических систем, но и для создания новых. Начало этому

положила работа Айтаи [34], в которой была показана связь между сложностью в среднем и сложностью в худшем случае для некоторых задач из теории решёток, что позволило использовать эти задачи в приложениях криптографии. Позже, в 1997 г. Айтаи и Дворк [35] предложили криптосистему, основанную на задаче SVP, а Голдрайх, Голдвассер и Халеви [36] — основанную на задаче CVP. Обе они были взломаны Нгуеном в 1998 г. [37] и 1999 г. [38] соответственно. В 1996 г. Хоффштейном, Пайфером и Сильверманом была предложена криптосистема с открытым ключом NTRU [39], впоследствии несколько раз модифицированная.

Криптосистемы, основанные на задачах из теории решёток, являются популярными кандидатами на роль постквантовых криптосистем. Во многом потому, что несмотря на значительные усилия, по сей день для задач из теории решёток не было предложено квантового алгоритма, который бы в значительной мере превосходил существующие классические алгоритмы. Также подобные криптосистемы легко реализуются, эффективны и хорошо параллелизуются.

Пусть векторы $v_1, \ldots, v_n \in \mathbb{R}^m$ линейно независимы. *Решёткой*, порождённой векторами $v_1, \ldots, v_n$, называется набор линейных комбинаций векторов $v_1, \ldots, v_n$ с коэффициентами из $\mathbb{Z}$,

$$L = \{a_1 v_1 + a_2 v_2 + \cdots + a_n v_n \mid a_1, a_2, \ldots, a_n \in \mathbb{Z}\}.$$

Базисом для L является любой линейно независимый набор векторов, порождающий L. Любой такой набор векторов имеет одинаковое число элементов. *Ранг* решётки L — число векторов в базисе L, *размерность* решётки L равна m; в случае, когда $n = m$, решётка L называется *решёткой полного ранга*. Любые два базиса L связаны преобразованием с целочисленной матрицей, определитель которой равен ± 1. Несмотря на то, что в качестве базиса может быть выбран любой порождающий решётку и линейно независимый набор векторов, для решения задач удобнее выбирать базис, состоящий из коротких векторов и как можно более близкий к ортогональному. Наиболее известные алгоритмы редукции базиса — алгоритмы LLL и BKZ [29].

Определителем решётки L называется объём параллелепипеда, натянутого на базисные вектора. *Минимальным расстоянием* λ_1 в решётке L называется евклидова норма кратчайшего ненулевого вектора в L: $\lambda_1(L) = \min\limits_{v \in L} \|v\|$, $v \neq 0$. Вообще, $\lambda_i(L)$ — наименьший радиус r такой, что в L существуют i линейно независимых векторов, норма которых не превосходит r. Известны следующие оценки минимального расстояния:

$$\lambda_1(L) \geqslant \min_i \|\widetilde{\mathbf{b}}_i\|,$$

$$\lambda_1(L) \leqslant \sqrt{n}(\det L)^{1/n}.$$

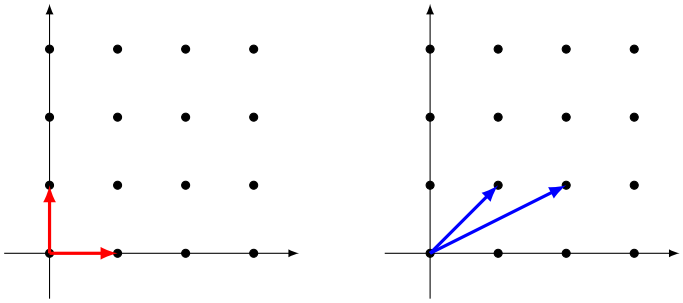


Рис. 1. Примеры двух различных базисов одной решётки $L \subset \mathbb{R}^2$

Здесь $\{\widetilde{\mathbf{b}}_i\}$ — базис, полученный из базиса решётки L ортогонализацией Грама — Шмидта; второе неравенство — следствие из теоремы Минковского.

Для более полного изучения теории решёток рекомендуется ознакомиться, например, с работами [40, 41].

Классические трудно вычислимые задачи в теории решёток — поиск ненулевого вектора наименьшей длины (shortest vector problem, SVP) и поиск вектора в решётке, ближайшего к заданному вектору (closest vector problem, CVP).

Задача о кратчайшем векторе (SVP). Найти ненулевой вектор наименьшей длины в решётке L, т. е. найти ненулевой вектор $v \in L$, для которого выполняется $\|v\| = \lambda_1(L)$.

Задача о ближайшем векторе (CVP). Для фиксированного вектора $w \in \mathbb{R}^m \setminus L$ найти ближайший к w вектор $v \in L$, минимизирующий евклидову норму $\|w - v\|$.

В приложениях криптографии используются приближённые версии указанных выше задач и некоторых других. Точность приближения регулируется параметром $\gamma \geqslant 1$, который, как правило, является некоторой функцией $\gamma(L)$ от решётки L.

Приближённый кратчайший вектор (γ-approximate SVP, SVP_γ). При заданном параметре γ найти ненулевой вектор $v \in L$, для которого выполняется $\|v\| \leqslant \gamma \cdot \lambda_1(L)$.

Булева задача о кратчайшем векторе (decisional SVP, GapSVP_γ). При условии, что для решётки L имеет место либо $\lambda_1(L) \leqslant 1$, либо $\lambda_1(L) \geqslant \gamma$, определить, какой именно случай выполняется.

Приближённый поиск линейно независимых кратчайших векторов (γ-approximate shortest independent vector problem, SIVP_γ). Дана n-мерная решётка L. Найти набор $S = \{s_i\}_{i=1}^n$, содержащий n линейно

независимых векторов из L таких, что

$$\|s_i\| \leqslant \gamma \cdot \lambda_n(L), \quad i = 1, \ldots, n.$$

В табл. 3 приведены сведения о сложности приближённого решения задачи SVP: известные алгоритмы с полиномиальным временем работы (LLL и более поздние алгоритмы на его основе) могут гарантировать точность только чуть меньше экспоненциальной, а известные алгоритмы с полиномиальной или лучшей точностью требуют либо суперэкспоненциального времени работы, либо экспоненциального времени и экспоненциального пространства. Важно, что это относится и к квантовым алгоритмам, в то время как задачи дискретного логарифмирования и факторизации могут быть решены за полиномиальное время с использованием квантового алгоритма Шора. Задачи, используемые в приложениях криптографии, имеют точность $\mathrm{poly}(n)$. Доказательств принадлежности их классу NP нет, также не известно, существует ли полиномиальный алгоритм их решения. Однако, так как лучшее на данный момент приближение с полиномиальным временем работы — $2^{\log\log n/\log n}$ только незначительно улучшает результат LLL, сложность в улучшении приближения позволяет полагать, что приближённо (с фактором, значительно меньшим экспоненциального) решить SVP_γ за полиномиальное время невозможно [42].

Таблица 3

Сложность приближённого решения задачи поиска вектора наименьшей длины

Задача	Оценка точности γ	Сложность
SVP	Точное решение	NP-трудная [43]
SVP_γ	$O(1)$	NP-трудная [44]
	$2^{(\log n)^{1-\varepsilon}}$	NP-трудная [45]
	$\sqrt{n}$	NP $\cap$ coNP [46]
	$\mathrm{poly}(n)$	—
	$2^{O(n)}$	класс P [27]

В [47] показано, что любой приближённый алгоритм решения CVP может быть использован для приближённого решения SVP с тем же приближением и с почти такой же вычислительной сложностью, а значит, CVP не проще (возможно, и более сложна), чем SVP. В [48] показано, что приближённое решение SIVP_γ сводится к приближённому решению CVP с тем же параметром γ.

Большинство современных криптосистем основаны на следующих, более современных задачах из теории решёток: нахождение короткого целочисленного решения, обучение с ошибками, обучение с округлениями.

Параметрами задачи нахождения короткого целочисленного решения (short integer solution, SIS) являются натуральные n и q (определяющие группу $\mathbb{Z}_q^n$), вещественное $\beta > 0$ и натуральное m.

Короткое целочисленное решение ($\mathrm{SIS}_{n,q,\beta,m}$). Даны m случайных равновероятных вектор-столбцов $a_i \in \mathbb{Z}_q^n$, которые образуют матрицу $A \in \mathbb{Z}_q^{n\times m}$, и параметр $\beta \in \mathbb{R}$. Найти ненулевой вектор $e \in \mathbb{Z}^m$ с целочисленными координатами такой, что $Ae \equiv 0 \pmod q$ и $\|e\| \leqslant \beta$.

Задача SIS используется для построения таких криптографических примитивов, как односторонние функции и хэш-функции, устойчивые к коллизиям, но для построения криптосистем с открытым ключом используются задачи обучения с ошибками (learning with errors, LWE) и их вариации над кольцами (R-LWE) и над модулями (M-LWE).

Можно выделить два основных класса задач обучения с ошибками: задачи поиска (search) и задачи различения распределений (decisional). В первом случае по набору произвольного числа выборок (a_i, b_i) из заданного распределения требуется восстановить исходный элемент s, называемый также секретом. Для задач принятия решения необходимо определить, были ли пары (a_i, b_i) получены с помощью заранее заданного распределения вероятностей или же были выбраны случайно равновероятно. Приведём наиболее общий вариант задач обучения с ошибками.

Параметрами задачи M-LWE являются кольцо R многочленов степени не выше n над $\mathbb{Z}$, положительное целое число q (определяющее фактор-кольцо $R_q = R/qR$) и распределение вероятностей χ над R_q. Для $s \in R_q^k$ распределение $A_{s,\chi}$ над $R_q^k \times R_q$ определяется следующим образом: $a \in R_q^k$ выбирается равномерно случайно, ошибка $e \in R_q$ выбирается в соответствии с распределением χ, на выходе — пара (a, b), где $b = (\langle s, a\rangle + e) \bmod q$.

Поиск над модулем при обучении с ошибками (search module LWE, S-M-$\mathrm{LWE}_{q,\chi}$). Даны m независимых пар (a_i, b_i) из $R_q^k \times R_q$, полученных в соответствии с распределением $A_{s,\chi}$ (s случайно равновероятно выбран из R_q^k, но фиксирован для всех пар). Необходимо найти многочлен s.

Решение над модулем при обучении с ошибками (decisional module LWE, D-M-$\mathrm{LWE}_{q,\chi}$). Даны m независимых пар (a_i, b_i) из $R_q^k \times R_q$. Необходимо определить, распределены ли они в соответствии с $A_{s,\chi}$ (s случайно равновероятно выбран из R_q^k, фиксирован для всех пар) или выбраны в соответствии с равномерным распределением вероятностей.

Задачи обучения с ошибками над кольцом (R-LWE) получаются из задач M-LWE при $k = 1$. Задачи обучения с ошибками (LWE) можно получить, если заменить R на кольцо целых чисел.

В работе Регева [49] показано, что ряд вычислительно трудных проблем таких, как GapSVP и SIVP, допускает сведение к соответствующим вариантам задач обучения с ошибками, по крайней мере для худшего случая. В силу того, что данное сведение задействует квантовые вычисления, существование эффективного алгоритма для задачи обучения с ошибками влечёт существование *квантового* алгоритма для решения задач GapSVP и SIVP. Узнать подробнее об использовании задач обучения с ошибками в современной криптографии можно в разд. 2.3 (схемы CRYSTALS-Kyber и FrodoKEM).

Задачи обучения с округлением LWR были представлены в [50]. Они используются для создания псевдослучайных функций [50] и детерминированных систем шифрования [51]. В разд. 2.3 настоящей статьи рассматривается механизм инкапсуляции ключей Saber, основанный на задачах обучения с округлением. В этой вариации обучения с ошибками вместо добавления случайной ошибки e к $\langle a_i, s\rangle \in \mathbb{Z}_q$ значение $\langle a_i, s\rangle$ детерминированно округляется. Пусть $p < q$. Разделим $\mathbb{Z}_q$ на p интервалов примерно по q/p элементов в каждом и определим функцию округления $\lfloor\cdot\rfloor_p \colon \mathbb{Z}_q \to \mathbb{Z}_p$, которая ставит в соответствие каждому $x \in \mathbb{Z}_q$ индекс интервала, которому x принадлежит. В случае, когда p и q являются степенями числа b, результатом функции округления будут $\log_b p$ наиболее значимых битов числа x. Вновь приведём наиболее общую вариацию — модульные задачи обучения с округлением (M-LWR).

Параметрами задачи M-LWR являются кольцо R многочленов степени не выше n над $\mathbb{Z}$, положительное целое число q (определяющее фактор-кольцо $R_q = R/qR$) и распределение вероятностей χ над R_q. Для $s \in R_q^k$, называемого секретом, распределение $A_{s,\chi}$ над $R_q^k \times R_q$ определяется следующим образом: $a \in R_q^k$ выбирается равномерно случайно, ошибка $e \in R_q$ выбирается в соответствии с распределением χ, на выходе — пара (a, b), где $b = \lfloor\langle a_i, s\rangle\rfloor_p$.

Поиск над модулем при обучении с округлением (S-M-LWR$_{q,\chi}$). Даны m независимых пар (a_i, b_i) из $R_q^k \times R_q$, полученных в соответствии с распределением $A_{s,\chi}$ (s случайно равновероятно выбран из R_q^k, но фиксирован для всех пар). Необходимо найти многочлен s.

Решение над модулем при обучении с округлением (D-M-LWR$_{q,\chi}$). Даны m независимых пар (a_i, b_i) из $R_q^k \times R_q$. Необходимо определить, распределены ли они в соответствии с $A_{s,\chi}$ (s случайно равновероятно выбран из R_q^k, фиксирован для всех пар) или выбраны выбраны в соответствии с равномерным распределением вероятностей.

Задачи обучения с округлением над кольцом (R-LWR) получаются из задач M-LWR при равенстве параметра k единице. Задачи обучения с округлением (LWR) можно получить, если заменить R на кольцо целых чисел.

1.2. Базовая схема NTRU. На конференции Crypto'96 и позже в работе [39] была представлена криптосистема NTRU в качестве альтернативы шифрсистеме RSA [52], которая подвержена атаке с использованием квантового алгоритма Шора [11].

Обозначим через R кольцо полиномов $\mathbb{Z}[x]/(x^n-1)$ степени меньше n с коэффициентами из кольца $\mathbb{Z}$. Любой элемент

$$a = \sum_{i=0}^{n-1} a_i x^i \in R, \quad a_i \in \mathbb{Z},$$

можно представить как вектор

$$a = (a_0, \ldots, a_{n-1}).$$

Операция умножения «$*$» в R определяется как результат циклической свёртки:

$$f * g = h, \quad \text{где } f, g, h \in R,$$

и для $k = 0, 1, \ldots, n-1$ выполнено

$$h_k = \sum_{i=0}^{k} f_i g_{k-i} + \sum_{i=k+1}^{n-1} f_i g_{n+k-i} = \sum_{\substack{i,j: \\ i+j \equiv k \pmod n}} f_i g_j.$$

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

Криптосистема NTRU зависит от трёх положительных целочисленных параметров (n, p, q) и четырёх подмножеств $\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_\varphi, \mathcal{L}_m$ множества R. Числа p и q взаимно просты, и q значительно больше p. Множество сообщений $\mathcal{L}_m$ определяется следующим образом:

$$\mathcal{L}_m = \Big\{ m \in R \mid \text{коэффициенты } m \text{ принадлежат множеству } \Big\{ -\Big\lfloor \frac{p-1}{2} \Big\rfloor, \ldots, \Big\lfloor \frac{p}{2} \Big\rfloor \Big\} \Big\}.$$

Для определения $\mathcal{L}_f, \mathcal{L}_g$ и $\mathcal{L}_\varphi$ введём множество

$$\mathcal{L}(d_1, d_2) = \{ f \in R \mid d_1 \text{ коэффициентов } f \text{ равны } 1, \; d_2 \text{ — равны } -1, \text{ остальные равны } 0 \}.$$

Тогда для некоторых положительных чисел d_f, d_g, d

$$\mathcal{L}_f = \mathcal{L}(d_f, d_f - 1), \quad \mathcal{L}_g = \mathcal{L}(d_g, d_g), \quad \mathcal{L}_\varphi = \mathcal{L}(d, d).$$

Для наивысшего уровня стойкости авторами криптосистемы были предложены следующие параметры:

$$(n, p, q) = (503, 3, 256), \quad d_f = 216, \quad d_g = 72, \quad d = 55.$$

При данных параметрах длины секретного и открытого ключей равны 1595 и 4024 бит соответственно.

Генерация ключей. Случайным образом выбираются два многочлена $f \in \mathcal{L}_f$, $g \in \mathcal{L}_g$ так, чтобы полином f был обратим по модулям p и q, т. е. чтобы существовали полиномы f_q и f_p такие, что выполнено

$$f_q * f \equiv 1 \pmod{q}, \quad f_p * f \equiv 1 \pmod{p}.$$

Определим полином h, являющийся открытым ключом:

$$h = f_q * g \bmod q.$$

Секретным ключом являются полиномы f и f_p.

Зашифрование. Для зашифрования сообщения $m \in \mathcal{L}_m$ используется полином h из открытого ключа и выбранный случайным образом полином $\varphi \in \mathcal{L}_\varphi$. Шифртекст c равен

$$c = (p\varphi * h + m) \bmod q.$$

Расшифрование. Для расшифрования шифртекста c используются полиномы f и f_p из секретного ключа. Сначала вычисляется полином

$$a = f * c \bmod q,$$

коэффициенты которого выбираются в интервале от $-\frac{q}{2}$ до $\frac{q}{2}$. Теперь сообщение m' восстанавливается следующим вычислением:

$$m' = f_p * a \bmod p,$$

где $m' \in \mathcal{L}_m$.

Замечание. При соответствующих значениях параметров существует высокая вероятность того, что процедура расшифрования восстановит исходное сообщение. Однако, некоторые варианты параметров могут вызывать периодический сбой в расшифровании, поэтому вероятно следует включать несколько контрольных битов в каждый блок сообщения. Частой причиной сбоя расшифрования будет то, что сообщение неправильно центрировано. В этом случае можно восстановить сообщение, выбрав коэффициенты полинома a в несколько другом интервале, например, от $-\frac{q}{2} + x$ до $\frac{q}{2} + x$ для некоторого небольшого ненулевого x. Если при изменении интервала не удаётся корректно расшифровать сообщение, то говорят, что есть *нарушение целостности* и сообщение не может быть расшифровано напрямую. Для правильно подобранных значений параметров вероятность возникновения данной ситуации пренебрежимо мала и на практике, как правило, игнорируется.

Корректность расшифрования. Пользуясь определениями многочленов h и f_q, получим следующие равенства для полинома a:

$$a = f * c \bmod q = (f * p\varphi * h + f * m) \bmod q$$
$$= (f * p\varphi * f_q * g + f * m) \bmod q = (p\varphi * g + f * m) \bmod q. \quad (1)$$

При соответствующем выборе параметров можно гарантировать, что (почти всегда) все коэффициенты последнего полинома в (1) лежат в интервале от $-\frac{q}{2}$ до $\frac{q}{2}$, так что он не меняется, если его коэффициенты берутся по модулю q. Это означает, что при выборе коэффициентов $f * c$ по модулю q из интервала от $-\frac{q}{2}$ до $\frac{q}{2}$ восстанавливается именно многочлен

$$a = p\varphi * g + f * m \in R.$$

Тогда, используя определение f_p, получим

$$m' = f_p * a \bmod p = (f_p * p\varphi * g + f_p * f * m) \bmod p = (0 + 1 * m) \bmod p = m.$$

1.3. Схемы, актуальные на сегодняшний день. Здесь представлены схемы инкапсуляции ключей и цифровой подписи, которые основаны на решётках. Схемы учавствовали в третьем раунде конкурса NIST по стандартизации постквантовой криптографии [53].

Для анализа и оценки безопасности схем NIST определил пять уровней стойкости [54], которые определяются следующим образом:

- схема обладает 1, 3 или 5 уровнем стойкости, если любая атака, нарушающая соответствующее определение стойкости, должна требовать вычислительные ресурсы, сравнимые или превышающие ресурсы, необходимые для поиска ключа в блочном шифре со 128-битным, 192-битным или 256-битным ключом соответственно;
- схема обладает 2 или 4 уровнем стойкости, если любая атака, нарушающая соответствующее определение стойкости, должна требовать вычислительные ресурсы, сравнимые или превышающие ресурсы, необходимые для поиска коллизий 256-битной или 384-битной хэш-функции соответственно.

Все схемы, представленные в этом пункте, безопасны с точки зрения неразличимости шифртекста. В зависимости от предположений о возможностях злоумышленника разделяют три типа такой безопасности:

- IND-CPA — неразличимость шифртекста для атак на основе подобранного открытого текста;
- IND-CCA — неразличимость шифртекста для атак на основе подобранного шифртекста;
- IND-CCA2 — неразличимость шифртекста для адаптивных атак, основанных на подобранном шифртексте.

CRYSTALS-Kyber. KYBER — это механизм инкапсуляции ключа (key encapsulation mechanism, KEM), основанный на задаче M-LWE [55].

KYBER сначала строится как схема шифрования с открытым ключом (PKE), надёжная в смысле IND-CPA, а затем модифицируется до схемы KEM, надёжной в смысле IND-CCA с помощью преобразования типа Фуджисаки — Окомото [56].

Базовая схема PKE основана на задаче M-LWE. Обозначим через R кольцо $\mathbb{Z}[x]/(x^{256}+1)$. Параметры включают целочисленный модуль $q = 3329$, распределение χ на R_q и открытую матрицу полиномов $A \in R_q^{k\times k}$, псевдослучайно сгенерированную из случайной и равномерной 256-битной строки, где $k = 2, 3, 4$ (соответственно уровням стойкости $1, 3, 5$). Два секретных вектора полиномов $s, e \in R_q^k$ выбираются независимо в соответствии с распределением χ. Вектор e называется вектором ошибки.

Открытый ключ. Пусть $b = As + e$. Тогда открытым ключом PKE является $pk = (A, b)$.

Секретный ключ. Вектор s считается секретным ключом.

Зашифрование. Для зашифрования сообщения m (256-битная строка) выбираются два вектора полиномов $r, e_1 \in R_q^k$, а также полином $e_2 \in R_q$, причём все коэффициенты каждого полинома выбираются независимо в соответствии с распределением χ. Затем зашифрованный текст c формируется следующим образом:

$$c = (c_1, c_2) = \left(rA + e_1,\, rb + e_2 + \left\lceil \frac{q}{2} \right\rceil \cdot m\right) \in R_q^k \times R_q,$$

где $\lceil \frac{q}{2} \rceil \cdot m$ — вектор коэффициентов полинома m, каждый из которых умножен на $\lceil \frac{q}{2} \rceil$.

Расшифрование. Чтобы расшифровать зашифрованный текст c при помощи секретного ключа s, после первого «распаковывания» зашифрованного текста вычисляют промежуточное значение $\nu = c_2 - c_1 s$, затем каждый коэффициент полинома ν берётся по модулю 2, чтобы извлечь переданную битовую строку m.

KYBER был выбран для стандартизации после третьего раунда конкурса NIST по стандартизации постквантовой криптографии [53]. Схема обладает одними из лучших показателей производительности среди схем, представленных на конкурс.

Saber. Saber — это схема KEM, надёжная в смысле IND-CCA2 и основанная на задаче M-LWR [57].

Saber сначала строится как схема PKE, надёжная в смысле IND-CPA, а затем модифицируется до схемы KEM, надёжной в смысле IND-CCA, с помощью преобразования типа Фуджисаки — Окомото [56].

Базовая схема PKE основана на задаче M-LWR. Обозначим через R кольцо $\mathbb{Z}[x]/(x^{256}+1)$. Для каждого набора параметров Saber использует

модули $p = 2^{10}$, $q = 2^{13}$ и $T = 2^3, 2^4, 2^6$ (соответственно уровням стойкости $1, 3, 5$). Saber также использует операцию округления Round_p, которую можно представить как взятие элементов $\mathbb{Z}_q$ и преобразование их в $\mathbb{Z}_p$ путём округления до ближайшего числа, кратного $\frac{q}{p}$, и отбрасывания $\log_2(q) - \log_2(p)$ младших битов для получения $\log_2(p)$-битной величины, которая рассматривается как элемент $\mathbb{Z}_p$. Подобные операции Round_T и Round_2 отображают элементы $\mathbb{Z}_T$ и $\mathbb{Z}_2$ соответственно.

При генерации ключа матрица $A \in R_q^{k \times k}$ выбирается случайно и равномерно, где $k = 2, 3$ или 4 (соответствует уровням стойкости $1, 3, 5$), а вектор $s \in R^k$ выбирается случайным образом в соответствии с центрированным биномиальным распределением.

Открытый ключ. Пусть $b = \mathrm{Round}_p(A^\top s)$. Тогда открытым ключом будет $pk = (A, b)$.

Секретный ключ. Секретным ключом является s.

Зашифрование. Чтобы зашифровать 256-битное сообщение m, сначала в соответствии с центрированным биномиальным распределением выбираются коэффициенты вектора полиномов $s' \in R^k$. Затем шифртекст c формируется следующим образом:

$$c = (c_m, b') = \Big(\mathrm{Round}_T\Big(b^\top s' + \frac{mT}{2}\Big),\ \mathrm{Round}_p(As')\Big).$$

Расшифрование. Чтобы расшифровать зашифрованный текст c, с помощью секретного ключа s вычисляют $m = \mathrm{Round}_2(b'^\top s - c_m)$.

NTRU. Хотя за последние годы было представлено несколько версий этой криптосистемы, основные особенности дизайна остались неизменными и присутствуют в версиях, представленных на конкурсе NIST по стандартизации постквантовой криптографии. В третьем раунде были представлены две версии схемы NTRU — NTRU-HPS и NTRU-HRSS [58].

Введём несколько множеств для определения параметров NTRU-HPS и NTRU-HRSS:

$\mathcal{T}$ — множество ненулевых полиномов степени, не превышающей $n-2$, коэффициенты которых лежат в $\{-1, 0, 1\}$;

$\mathcal{T}(d)$ — подмножество $\mathcal{T}$, состоящее из полиномов, у которых в точности $d/2$ коэффициентов равно 1 и $d/2$ коэффициентов равно -1, где d — положительное целое чётное число;

$\mathcal{T}_+$ — подмножество $\mathcal{T}$, состоящее из полиномов $\sum\limits_{i=0}^{n-1} a_i x^i$ таких, что

$$\sum_{i=1}^{n-2} a_i a_{i+1} \geqslant 0.$$

Напомним, что криптосистема NTRU зависит от трёх целочисленных параметров (n, p, q) и четырёх подмножеств $\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_\varphi, \mathcal{L}_m$.

Версия NTRU-HPS имеет следующие параметры: n — простое число такое, что 2 и 3 имеют порядок $n-1$ в мультипликативной группе кольца вычетов $(\mathbb{Z}/n\mathbb{Z})^\times$,

$$p = 3, \quad q \text{ — степень 2},$$
$$\mathcal{L}_f = \mathcal{L}_\varphi = \mathcal{T}, \quad \mathcal{L}_g = \mathcal{L}_m = \mathcal{T}(q/8 - 2).$$

Версия NTRU-HRSS имеет следующие параметры: n — простое число такое, что 2 и 3 имеют порядок $n-1$ в мультипликативной группе кольца вычетов $(\mathbb{Z}/n\mathbb{Z})^\times$,

$$p = 3, \quad q = 2^{\lceil 7/2 + \log_2(n) \rceil}, \quad \mathcal{L}_f = \mathcal{T}_+,$$
$$\mathcal{L}_g = \{(x-1) \cdot a \mid a \in \mathcal{T}_+\}, \quad \mathcal{L}_\varphi = \mathcal{L}_m = \mathcal{T}.$$

Отметим, что в версии NTRU-HPS при генерации полиномов их коэффициенты, каждый из которых взят из $\{-1, 0, 1\}$, имеют фиксированное общее количество 1 и -1. Для сравнения в версии NTRU-HRSS каждый коэффициент полиномов выбирается случайно и равномерно из набора $\{-1, 0, 1\}$.

Схема NTRU PKE не надёжна в смысле IND-CCA. Для преобразования PKE в схему KEM, надёжную в смысле IND-CCA2, используется версия преобразования Фуджисака — Окамото [59].

FrodoKEM. Это механизм инкапсуляции ключей на основе задачи LWE [60].

Напомним, что в задаче D-LWE$_{n,q,\chi}$ даны независимые пары (a_i, b_i) из $\mathbb{Z}_q^n \times \mathbb{Z}_q$, полученные в соответствии с распределением $A_{s,\chi}$, где $b = (\langle a_i, s\rangle + e) \bmod q$, $s \in \mathbb{Z}_q^n$, и ошибка e выбирается в соответствии с распределением χ. Задача приводит к следующей схеме шифрования с открытым ключом: секретный вектор $s \in \mathbb{Z}_q^n$ является секретным ключом, а набор независимых пар (a_i, b_i), организованных в виде матрицы $\big[A \mid As + e\big]$, является открытым ключом. Чтобы зашифровать бит b, нужно суммировать случайное подмножество выборок, а затем добавить $\big(0, 0, \ldots, 0, b \cdot \frac{q}{2}\big)$. Здесь $q \leqslant 2^{16}$ — целочисленный модуль, выбранный как степень числа 2. Различение зашифрованных текстов тогда сводится к различению «почти истинных» и «далёких от истинных» уравнений $\bmod q$ с неизвестными переменными s. Эта задача столь же трудна, как и задача принятия решения для обучения с ошибками LWE.

FrodoPKE — это схема шифрования с открытым ключом, надёжная в смысле IND-CPA, основанная на оптимизированной версии схемы, которая описана выше.

В реализации FrodoPKE псевдослучайным образом генерируется матрица A с использованием AES-128 или SHAKE128.

Открытый ключ. Открытый ключ — это $(A, B = AS + E)$, где элементы матриц S и E выбраны в соответствии с дискретным гауссовым распределением χ на $\mathbb{Z}_q$.

Секретный ключ. Секретный ключ представляет собой матрицу S.

Зашифрование. Для зашифрования сообщения, закодированного в виде матрицы M над $\mathbb{Z}_q$, отправитель выбирает случайные гауссовы матрицы S', E', E'' и отправляет зашифрованный текст

$$(C_1, C_2) = (S'A + E',\ S'B + E'' + M).$$

Расшифрование. При условии, что исходное кодирование получателя устойчиво к шуму в младших битах M, отправитель может затем восстановить сообщение получателя, вычислив $C_2 - C_1S$.

Авторы FrodoPKE применяют специальное преобразование Фуджисаки — Окамото [61] для получения механизма безопасной инкапсуляции ключей FrodoKEM, надёжного в смысле IND-CCA.

CRYSTALS-Dilithium. Dilithium — это алгоритм цифровой подписи на решётках, основанный на парадигме Фиата Шамира [62]. Dilithium использует кольцо $R_q = \mathbb{Z}_q[x]/(x^{256}+1)$, где q — простое число $2^{23}-2^{13}+1$.

Открытый ключ. Открытый ключ — выборка вида $(A, t = As_1 + s_2)$, где A — матрица над R_q.

Секретный ключ. Векторы s_1 и s_2 секретные, случайно выбираются из R_q.

Одной из отличительных особенностей Dilithium является его распределение ошибок: в то время как алгоритмы подписи на основе решёток обычно используют усечённое распределение Гаусса для вычисления коэффициентов в своих векторах ошибок, Dilithium использует равномерное распределение на $\{-\eta, -\eta+1, \ldots, \eta\}$, где η — небольшое положительное целое число.

Dilithium использует схему идентификации на решётках, которая позволяет доказывающему убедить проверяющего в том, что он владеет секретным ключом (s_1, s_2), не раскрывая его.

Процедура подписи. Процесс начинается с того, что доказывающая сторона вычисляет вектор w, состоящий из старших битов Ay (для случайного y), и отправляет его проверяющей стороне. Затем вычисляется $c \in R_q$, который является хэш-значением сообщения и w. Затем доказывающая сторона отвечает вектором $z = y + cs_1$. Чтобы избежать зависимости z от секретного ключа s_1, дополнительно проверяется выполнение некоторых условий на коэффициенты z и $Az - ct$. Процесс формирования подписи повторяется, пока эти условия не будут выполнены.

Проверка подписи. Проверяющий сначала вычисляет вектор w', состоящий из старших битов $Az - ct$, а затем принимает подпись, если c является хэш-значением сообщения и w' и выполняются некоторые условия на коэффициенты z.

Схема Dilithium включает в себя несколько дополнительных оптимизаций. Например, открытый ключ сжимается как за счёт использования псевдослучайности, так и за счёт отбрасывания некоторых младших битов t. Чтобы компенсировать сжатые биты, подписывающий включает «подсказки» как часть каждой подписи [62]. Эти подсказки, по сути, являются определёнными носителями, которые позволяют проверяющему по-прежнему правильно выполнять проверку, описанную выше.

Dilithium был выбран для стандартизации после третьего раунда конкурса NIST по стандартизации постквантовой криптографии [53].

FALCON. Это схема подписи на основе решёток, использующая парадигму «хэш-и-подпись» [63].

Обозначим через R кольцо полиномов $\mathbb{Z}[x]/(x^n + 1)$, где n является степенью двойки. Пусть полиномы $f, g, F, G \in R$ удовлетворяют уравнению $fG - gF \equiv q$, где $q \in \mathbb{N}^*$.

Открытый ключ. Открытый ключ — полином $h = g \cdot f^{-1} \bmod q$.

Секретный ключ. Полиномы f, g, F, G секретные.

Таблица 4

Размеры ключа и зашифрованного текста для актуальных KEM-схем (в байтах)

Схема	Уровень стойкости	Открытый ключ	Секретный ключ	Шифртекст
KYBER512	1	800	1632	768
KYBER768	3	1184	2400	1088
KYBER1024	5	1568	3168	1568
NTRU-HPS2048677	1	930	1234	930
NTRU-HRSS701	1	1138	1450	1138
NTRU-HPS4096821	3	1230	1590	1230
NTRU-HPS40961229	5	1842	2366	1842
NTRU-HRSS1373	5	2401	2983	2401
Light Saber	1	672	832	736
Saber	3	992	1248	1088
Fire Saber	5	1312	1664	1472
FrodoKEM-640	1	9616	19 888	9720
FrodoKEM-976	3	15 632	31 296	15 744
FrodoKEM-1344	5	21 520	43 088	21 632

Для правильно сгенерированных секретных ключей полином h будет казаться случайным, в то время как основания $\begin{bmatrix} 1 & h \\ 0 & q \end{bmatrix}$ и $\begin{bmatrix} f & g \\ F & G \end{bmatrix}$ генерируют одну и ту же решётку.

Процедура подписи. Сначала генерируется случайная и равномерная 320-битная строка r. Затем конкатенация строк r и m хэшируется полиномом $c \in R_q$ специальным алгоритмом хэширования [63]. Далее вычисляется прообраз t элемента c, который подаётся на вход алгоритма быстрой выборки Фурье. Последний выдаёт два коротких полинома $s_1, s_2 \in R$ таких, что $s_1 + s_2 h \equiv c \pmod{q}$. Подпись состоит из пары (r, s), где s получается из s_2 с помощью специального алгоритма сжатия [63].

Проверка подписи. Для проверки того, что (r, s) — верная подпись сообщения m, конкатенация строк r и m хэшируется полиномом $c \in R_q$. Полином $s_2 \in R$ получается из s специальным алгоритмом «распаковки» [63]. В итоге вычисляется $s_1 = (c - s_2 h) \bmod q$, и подпись принимается, если $\|(s_1, s_2)\|^2 \leqslant \lfloor \beta^2 \rfloor$, где $\lfloor \beta^2 \rfloor$ — граница принятия.

FALCON был выбран для стандартизации после третьего раунда конкурса NIST по стандартизации постквантовой криптографии [53].

Сравнение схем. В табл. 4, 5 представлены размеры ключей, шифртекстов и подписей для схем, описанных выше. Для сравнения в табл. 5 представлена схема цифровой подписи на хэшах SPHINCS$^+$, которая также была выбрана для стандартизации по результатам третьего раунда конкурса NIST [53].

В [53] приведены анализ и сравнение производительности схем KYBER, Saber, NTRU и FrodoKEM с уровнями стойкости 1 и 3. Сравнивались показатели вычислительной производительности, полученные на процессоре x86-64 с расширениями AVX2.

Таблица 5

Размеры ключа и подписи для актуальных схем подписи (в байтах)

Схема	Уровень стойкости	Открытый ключ	Секретный ключ	Подпись
Dilithium	2	1312	2528	2420
	3	1952	4000	3293
	5	2592	4864	4595
FALCON-512	1	897	7553	666
FALCON-1024	5	1793	13 953	1280
SPHINCS$^+$-128s	1	32	64	7856
SPHINCS$^+$-192s	3	48	96	16 224
SPHINCS$^+$-256s	5	64	128	29 792

Схемы KYBER, Saber и NTRU имеют очень высокую скорость инкапсуляции и декапсуляции ключей. Общая стоимость использования Saber ниже, чем у KYBER, за счёт меньших размеров открытых ключей и шифртекстов, однако разница незначительная. В свою очередь, стоимость генерации ключа для NTRU-HPS2048677 или NTRU-HRSS701 примерно в 11 раз больше, чем для KYBER512. В результате общая стоимость NTRU-HPS2048677 примерно на 30% выше, чем у KYBER512. Показатели для схемы FrodoKEM значительно хуже, чем у KYBER, Saber и NTRU.

1.4. Основные методы криптоанализа.

Атака на базовую криптосистему NTRU с помощью решёток. В п. 1.2 мы говорили о базовой криптосистеме NTRU. Рассмотрим атаку с помощью решёток на секретный ключ данной криптосистемы, предложенную авторами криптосистемы NTRU. Пусть α — параметр (будет определён позже) и L — решётка, порождённая строками матрицы

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

По определению $h = f_q * g \bmod q$, тогда решётка L содержит вектор $x = (\alpha f, g)$, с большой вероятностью являющийся кратчайшим ненулевым вектором решётки L. Вектор x состоит из $2n$ координат, первые n определяются коэффициентами многочлена αf, а последующие n — коэффициентами многочлена g. Согласно Гауссовой эвристике ожидаемая длина кратчайшего вектора в случайной решётке размерности d лежит между числами

$$(\det L)^{1/d}\sqrt{\frac{d}{2\pi e}} \quad \text{и} \quad (\det L)^{1/d}\sqrt{\frac{d}{\pi e}}.$$

В нашем случае $d = 2n$ и $\det L = q^n\alpha^n$, тогда ожидаемая длина кратчайшего вектора не меньше чем

$$s = \sqrt{\frac{n\alpha q}{\pi e}}.$$

Алгоритмы нахождения кратчайшего вектора решётки будут иметь наибольшие шансы найти x или другой вектор длины, близкой к длине x,

если параметр α будет выбран так, чтобы максимизировать отношение $s/|x|_2$. Возведя в квадрат это отношение, можно видеть, что нужно выбрать α так, чтобы максимизировать

$$\frac{\alpha}{\alpha^2|f|_2^2 + |g|_2^2} = \left(\alpha|f|_2^2 + \alpha^{-1}|g|_2^2\right)^{-1}.$$

Тогда $\alpha = |g|_2/|f|_2$. (Заметим, что $|g|_2$ и $|f|_2$ являются открытыми параметрами криптосистемы.) Отметим, что на практике для избежания численных ошибок параметр α выбирается ближайшим ненулевым натуральным числом к значению $|g|_2/|f|_2$.

Когда α выбирается таким образом, определим константу c_h из равенства $|x|_2 = c_h s$. Таким образом, c_h — это отношение длины вектора x к длине ожидаемого кратчайшего вектора. Чем меньше значение c_h, тем легче найти вектор x. Подставляя в c_h значение s, получим

$$c_h = \sqrt{\frac{2\pi e|f|_2|g|_2}{nq}}.$$

Для заданной пары (f, g), используемой для создания криптосистемы, можно рассматривать c_h как меру того, насколько сильно решётка L отличается от случайной решётки. Если c_h близко к 1, то L будет напоминать случайную решётку, и будет трудно найти кратчайший вектор этой решётки, в частности, вектор x. По мере уменьшения значения c_h поиск x упрощается.

В следующих разделах будут описаны различные подходы к поиску кратчайшего ненулевого вектора в решётке. Стоит заметить, что согласно работе [64] не только x, но и *любой* короткий вектор из решётки L может быть полезен при атаке.

Алгоритмы редукции решёток. Для решения задач SVP и CVP существует хорошо известный алгоритм LLL [27]. Данный алгоритм разработан А. Ленстрой, Х. Ленстрой и Ловасом в 1982 г. и имеет множество приложений таких, как факторизация полиномов над $\mathbb{Z}$ или $\mathbb{Q}$, нахождение минимальных полиномов для алгебраических чисел, решение задач целочисленного линейного программирования и др. Будем называть *коэффициентом аппроксимации* отношение длины найденного алгоритмом вектора к длине кратчайшего ненулевого вектора в решётке. Алгоритм LLL для SVP работает полиномиальное время, но имеет коэффициент аппроксимации порядка $2^{O(n)}$, где n — размерность решётки. Введём следующие определения: $\lceil\cdot\rfloor$ — округление числа к ближайшему целому, $\langle\cdot,\cdot\rangle$ — скалярное произведение, определяемое следующим образом. Пусть $x = (x_1, \ldots, x_m)$ и $y = (y_1, \ldots, y_m)$ — векторы из пространства $\mathbb{R}^m$. Тогда $\langle x, y\rangle = \sum\limits_{i=1}^{m} x_i y_i$.

Используя данные определения, опишем алгоритм LLL.

Алгоритм 1. LLL

Вход: $\{b_1, \ldots, b_n\}$ — базис решётки L
Выход: Редуцированный базис решётки L

▷ Ортогонализация Грама — Шмидта:

1: $b_1^* = b_1$

2: $b_i^* = b_i - \sum\limits_{j=1}^{i-1} \mu_{ij} b_j^*$, где $\mu_{ij} = \frac{\langle b_i, b_j^* \rangle}{\langle b_j^*, b_j^* \rangle}$, $i = 2, 3, \ldots, n$

▷ Шаг редукции:

3: **for** $i = 2, 3, \ldots, n$ **do**
4: **for** $j = i-1, i-2, \ldots, 1$ **do**
5: $b_i \leftarrow b_i - \lceil \mu_{ij} \rfloor b_j$

▷ Шаг замены:

6: **if** существует i такое, что $\frac{3}{4}\|b_i^*\|^2 > \|\mu_{i+1,i} b_i^* + b_{i+1}^*\|^2$ **then**
7: $b_i \leftrightarrow b_{i+1}$
8: Перейти на ШАГ 1

Теорема 1. *Пусть $\{b_1, \ldots, b_n\}$ — базис решётки L, полученный на выходе алгоритма* 1. *Тогда для любого набора линейно независимых векторов $\{x_1, \ldots, x_t\}$ из решётки L выполнено*

$$\|b_j\| \leqslant 2^{\frac{n-1}{2}} \max\{\|x_1\|, \ldots, \|x_t\|\}, \quad j < t.$$

В частности, для любого $x \in L \setminus \{0\}$ выполнено

$$\|b_1\| \leqslant 2^{\frac{n-1}{2}} \|x\|.$$

В работах [29, 65] Шнорр и Эйхнер представили важное расширение алгоритма LLL, которое было названо блочным алгоритмом Коркина — Золотарёва (block Korkin — Zolotarev algorithm, BKZ). Его основной идеей является применение алгоритмов LLL и перечисления [66–68] для решёток размерности β, где β является параметром алгоритма, выбранным из множества $\{2, 3, \ldots, n\}$. Время работы алгоритма BKZ равно $2^{O(\beta \log \beta)}$, а коэффициент аппроксимации — $\beta^{O(n/\beta)}$. В 2011 г. Чен и Нгуен представили модернизацию BKZ, включающую в себя несколько эвристик и названную BKZ 2.0 [69]. На практике BKZ 2.0 имеет значительное преимущество в производительности, но асимптотические оценки сложности и точности остаются неизменными.

Алгоритмы просеивания. Одной из основополагающих работ, положивших начало активным исследованиям в области алгоритмов просеивания, является работа Айтаи, Кумара и Сивакумара [70]. Основной

идеей, лежащей в основе алгоритмов просеивания, является нахождение в решётке векторов, сумма или разность которых даст меньший по длине вектор. На сегодняшний день существует множество вариаций алгоритмов просеивания, например [71–75]. Данный класс алгоритмов представляет особый интерес в практической криптографии и криптоанализе, так как его временная сложность составляет $2^{O(n)}$. Однако, в отличие от ранее представленных алгоритмов редукции данный класс алгоритмов имеет пространственную сложность $2^{O(n)}$.

Для ознакомления с алгоритмами просеивания рассмотрим алгоритм NV-Sieve [72], предложенный Нгуеном и Видиком в 2008 г. Данный алгоритм является эвристическим вариантом алгоритма AKS-Sieve [70]. Слегка изменённое, но эквивалентное описание этого алгоритма приведено в алгоритме 2.

Алгоритм 2. NV-Sieve

Вход: $\{b_1, \ldots, b_n\}$ — базис решётки L, параметр $\gamma \in \left(\frac{2}{3}, 1\right)$
Выход: Кратчайший ненулевой вектор решётки

Заполнить список L_0 экспоненциально большим числом случайных векторов решётки L, длина которых не превосходит $n = \max\limits_i \|b_i\|$
$m = 0$
repeat
    $R_m = \max\limits_{v \in L_m} \|v\|$
    Инициализировать пустые списки L_{m+1} и C_{m+1}
    for $v \in L_m$ **do**
        if $\|v\| \leqslant \gamma R_m$ **then** добавить v в список L_{m+1}
        while $w \leftarrow$ поиск$\{w \in C_{m+1} \mid \|w \pm v\| \leqslant \|w\|\}$ **do**
            Уменьшить v с помощью w: $v \leftarrow v \pm w$
            Добавить v в список L_{m+1}
            После окончания цикла продолжить самый внешний цикл
        Добавить v в список C_{m+1}
    $m := m + 1$
until $L_m = \varnothing$
return кратчайший вектор из L_{m-1}

Теорема 2. *Пусть* $\gamma \in \left(\frac{2}{3}, 1\right)$ *и*

$$c_h = -\log_2 \gamma - \frac{1}{2} \log_2 \left(1 - \frac{\gamma^2}{4}\right).$$

Тогда алгоритм 2 *возвращает кратчайший ненулевой вектор решётки* L*, используя не более чем* $2^{2c_h n + o(n)}$ *времени и* $2^{c_h n + o(n)}$ *пространства.*

Следствие 1. *Пусть* $\gamma \to 1$. *Тогда алгоритм* 2 *возвращает кратчайший ненулевой вектор решётки* L, *используя не более чем* $2^{0,415n+o(n)}$ *времени и* $2^{0,208n+o(n)}$ *пространства.*

Так как самая трудозатратная часть алгоритмов просеивания — это поиск элементов в неупорядоченном списке (алгоритм 2, шаг 8), такие алгоритмы можно ускорить, применив квантовый алгоритм поиска в неупорядоченном списке (алгоритм Гровера [13]). Для подробного изучения этой темы рекомендуем ознакомиться с работами [76–79].

Другие подходы. Также известен ряд алгоритмов, решающих задачу SVP и не относящихся к алгоритмам редукции базиса и алгоритмам просеивания. Примерами таких алгоритмов являются [80, 81] и [82], в основе которых лежат диаграммы Вороного и дискретная гауссова выборка соответственно. Асимптотическая сложность по времени и пространству данных алгоритмов совпадает с асимптотической сложностью алгоритмов просеивания. Открытым вопросом на сегодняшний день является существование алгоритма, решающего задачу SVP и имеющего сложности $2^{O(n)}$ и $\mathrm{poly}(n)$ для времени и пространства соответственно.

Таблица 6

Алгоритмы точного решения SVP и CVP

Тип	Алгоритм	$\log_2$(время)	$\log_2$(пространство)
Вероятностные и детерминированные	Перечисление [66–68, 83]	$\frac{n}{2e}\log_2 n$	$O(\log_2 n)$
	AKS-Sieve [70]	$3{,}398n$	$1{,}985n$
	ListSieve [73]	$3{,}199n$	1,325
	AKS-Sieve-Birthday [84]	$2{,}571n$	$1{,}407n$
	ListSieve-Birthday [71]	$2{,}465n$	$1{,}233n$
	Диаграммы Вороного [80, 81]	$2n$	n
	Гауссова выборка [82]	n	n
Эвристические	NV-Sieve [72]	$0{,}415n$	$0{,}208n$
	GaussSieve [73, 85]	$0{,}415n$	$0{,}208n$
	Triple sieve [86, 87]	$0{,}396n$	$0{,}189n$
	Overlattice sieve [88]	$0{,}377n$	$0{,}293n$
	Triple sieve with NNS [75]	$0{,}359n$	$0{,}189n$
	Hyperplane LSH [89, 90]	$0{,}337n$	$0{,}337n$
	Spherical LSH [91, 92]	$0{,}297n$	$0{,}297n$
	Spherical LSF [74, 93]	$0{,}292n$	$0{,}292n$

Заключение

В настоящей работе приведён обзор основных подходов к построению постквантовых криптосистем, используемых в настоящее время. Подробно разобрано направление постквантовой криптографии, основанное на использовании вычислительно-трудных задач из теории решёток. Приведено описание и характеристики некоторых известных криптосистем, стойкость которых основана на сложности задачи нахождения кратчайшего вектора, задачи обучения с ошибками. Разобраны основные подходы к решению задач из теории решёток, а также некоторые атаки на конкретные криптосистемы. Приведены теоретические оценки сложности известных алгоритмов редукции и просеивания решёток.

ЛИТЕРАТУРА

1. **Bernstein D. J.** Introduction to post-quantum cryptography // Post-quantum cryptography. Heidelberg: Springer, 2009. P. 1–14.
2. **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.
3. **Bennett C. H., Brassard G.** Quantum cryptography: Public key distribution and coin tossing // Theor. Comput. Sci. 2014. V. 560. P. 7–11.
4. **Манин Ю. И.** Вычислимое и невычислимое. М.: Сов. радио, 1980. 128 с.
5. **Feynman R. P.** Simulating physics with computers // Int. J. Theor. Phys. 1982. V. 21. P. 467–468.
6. **Deutsch D.** Quantum theory, the Church — Turing principle and the universal quantum computer // Proc. R. Soc. Lond. Ser. A. Math. Phys. Sci. 1985. V. 400, No. 1818. P. 97–117.
7. **Deutsch D., Jozsa R.** Rapid solution of problems by quantum computation // Proc. R. Soc. Lond. Ser. A. Math. Phys. Sci. 1992. V. 439, No. 1907. P. 553–558.
8. **Bernstein E., Vazirani U.** Quantum complexity theory // SIAM J. Comput. 1997. V. 26, No. 5. P. 1411–1473.
9. **Simon D. R.** On the power of quantum computation // SIAM J. Comput. 1997. V. 26, No. 5. P. 1474–1483.
10. **Nielsen M. A., Chuang I. L.** Quantum computation and quantum information. Cambridge: Camb. Univ. Press, 2010.
11. **Shor P. W.** Algorithms for quantum computation: Discrete logarithms and factoring // Proc. 35th Annu. Symp. Foundations of Computer Science (Santa Fe, USA, Nov. 20–22, 1994). Los Alamitos, CA: IEEE Comput. Soc., 1994. P. 124–134.
12. **Proos J., Zalka C.** Shor's discrete logarithm quantum algorithm for elliptic curves // Quantum Inf. Comput. 2003. V. 3, No. 4. P. 317–344.
13. **Grover L. K.** A fast quantum mechanical algorithm for database search // Proc. 28th ACM Symp. Theory of Computing (Philadelphia, PA, USA, May 22–24, 1996). New York: ACM, 1996. P. 212–219.

14. **Brassard G., Høyer P., Tapp A.** Quantum cryptanalysis of hash and claw-free functions // LATIN'98: Theoretical informatics. Proc. 3rd Lat. Am. Symp. (Campinas, Brazil, Apr. 20–24, 1998). Heidelberg: Springer, 1998. P. 163–169. (Lect. Notes Comput. Sci.; V. 1380).
15. **Brassard G., Høyer P., Tapp A.** Quantum counting // Automata, languages and programming. Proc. 25th Int. Colloq. (Aalborg, Denmark, July 13–17, 1998). Heidelberg: Springer, 1998. P. 820–831. (Lect. Notes Comput. Sci.; V. 1443).
16. **Kuwakado H., Morii M.** Security on the quantum-type Even — Mansour cipher // Proc. 2012 Int. Symp. Information Theory and Its Applications (Honolulu, HI, USA, Oct. 28–31, 2012). Los Alamitos, CA: IEEE Comput. Soc., 2012. P. 312–316.
17. **Dong X., Dong B., Wang X.** Quantum attacks on some Feistel block ciphers // Des. Codes Cryptogr. 2020. V. 88, No. 6. P. 1179–1203.
18. **Xie H., Yang L.** Using Bernstein — Vazirani algorithm to attack block ciphers // Des. Codes Cryptogr. 2019. V. 87, No. 5. P. 1161–1182.
19. **Leander G., May A.** Grover meets Simon — quantumly attacking the FX-construction // Advances in cryptology — ASIACRYPT 2017. Proc. 23rd Int. Conf. Theory and Applications of Cryptology and Information Security (Hong Kong, China, Dec. 3–7, 2017). Pt. II. Cham: Springer. 2017. P. 161–178. (Lect. Notes Comput. Sci.; V. 10625).
20. **Kuwakado H., Morii M.** Quantum distinguisher between the 3-round Feistel cipher and the random permutation // Proc. 2010 IEEE Int. Symp. Information Theory (Austin, TX, USA, June 13–18, 2010). Los Alamitos, CA: IEEE Comput. Soc., 2010. P. 2682–2685.
21. **Hodžić S., Knudsen L. R.** A quantum distinguisher for 7/8-round SMS4 block cipher // Quantum Inf. Process. 2020. V. 19, No. 11. Paper ID 411. 22 p.
22. **Zhou Q., Lu S., Zhang Z., Sun J.** Quantum differential cryptanalysis // Quantum Inf. Process. 2015. V. 14, No. 6. P. 2101–2109.
23. **Shi R., Xie H., Feng H., Yuan F., Liu B.** Quantum zero correlation linear cryptanalysis // Quantum Inf. Process. 2022. V. 21, No. 8. Paper ID 293. 30 p.
24. **Kaplan M., Leurent G., Leverrier A., Naya-Plasencia M.** Quantum differential and linear cryptanalysis // IACR Trans. Symmetric Cryptol. 2016. V. 2016, No. 1. P. 71–94.
25. **Chen L., Jordan S., Liu Y.-K.** [et al.]. Report on post-quantum cryptography. Nat. Inst. Stand. Technol. interag. intern. rep. NIST IR 8105. Gaithersburg, MD: NIST, 2016. 15 p. Available at `doi.org/10.6028/NIST.IR.8105` (accessed Sept. 13, 2023).
26. Post-quantum cryptography. Gaithersburg, MD: NIST, 2017. Available at `csrc.nist.gov/projects/post-quantum-cryptography` (accessed Sept. 13, 2023).
27. **Lenstra A. K., Lenstra H. W., Lovász L.** Factoring polynomials with rational coefficients // Math. Ann. 1982. V. 261, No. 4. P. 515–534.
28. **Shamir A.** A polynomial-time algorithm for breaking the basic Merkle — Hellman cryptosystem // Advances in cryptology. Proc. Crypto 82 (Santa Barbara, USA, Aug. 23–25, 1982). New York: Plenum Press, 1983. P. 279–288.

29. **Schnor C. P.** A hierarchy of polynomial time lattice basis reduction algorithms // Theor. Comput. Sci. 1987. V. 53, No. 2–3. P. 201–224.
30. **Schnor C. P.** A more efficient algorithm for lattice basis reduction // J. Algorithms. 1988. V. 9, No. 1. P. 47–62.
31. **Frieze A., Håstad J., Kannan R., Lagarias J., Shamir A.** Reconstructing truncated integer variables satisfying linear congruences // SIAM J. Comput. 1988. V. 17, No. 2. 262–280.
32. **Stern J., Toffin P.** Cryptanalysis of a public-key cryptosystem based on approximations by rational numbers // Advances in cryptology — EUROCRYPT'90. Proc. Workshop Theory and Application of Cryptographic Techniques (Aarhus, Denmark, May 21–24, 1990). Heidelberg: Springer, 1991. P. 313–317. (Lect. Notes Comput. Sci.; V. 473).
33. **Joux A., Stern J.** Cryptanalysis of another knapsack cryptosystem // Advances in cryptology — ASIACRYPT'91. Proc. Int. Conf. Theory and Application of Cryptology (Fujiyoshida, Japan, Nov. 11–14, 1991). Heidelberg: Springer, 1993. P. 470–476. (Lect. Notes Comput. Sci.; V. 739).
34. **Ajtai M.** Generating hard instances of lattice problems (extended abstract) // Proc. 28th Annu. ACM Symp. Theory of Computing (Philadelphia, PA, USA, May 22–24, 1996). New York: ACM, 1996. P. 99–108.
35. **Ajtai M., Dwork C.** A public-key cryptosystem with worst-case/average-case equivalence // Proc. 29th Annu. ACM Symp. Theory of Computing (El Paso, TX, USA, May 4–6, 1997). New York: ACM, 1997. P. 284–293.
36. **Goldreich O., Goldwasser S., Halevi S.** Public-key cryptosystems from lattice reduction problems // Advances in cryptology — CRYPTO'97. Proc. 17th Annu. Int. Cryptology Conf. (Santa Barbara, USA, Aug. 17–21, 1997). Heidelberg: Springer, 1997. P. 112–131. (Lect. Notes Comput. Sci.; V. 1294).
37. **Nguyen P., Stern J.** Cryptanalysis of the Ajtai — Dwork cryptosystem // Advances in cryptology — CRYPTO'98. Proc. 18th Annu. Int. Cryptology Conf. (Santa Barbara, USA, Aug. 23–27, 1998). Heidelberg: Springer, 1998. P. 223–242. (Lect. Notes Comput. Sci.; V. 1462).
38. **Nguyen P., Stern J.** Cryptanalysis of the Goldreich — Goldwasser — Halevi Cryptosystem from CRYPTO'97 // Advances in cryptology — CRYPTO'99. Proc. 19th Annu. Int. Cryptology Conf. (Santa Barbara, USA, Aug. 15–19, 1999). Heidelberg: Springer, 1999. P. 288–304. (Lect. Notes Comput. Sci.; V. 1666).
39. **Hoffstein J., Pipher J., Silverman J. H.** NTRU: A ring-based public key cryptosystem // Algorithmic number theory. Proc. 3rd Int. Symp. (Portland, OR, USA, June 21–25, 1998). Heidelberg: Springer, 1998. P. 267–288. (Lect. Notes Comput. Sci.; V. 1423).
40. **Silverman J. H., Pipher J., Hoffstein J.** An introduction to mathematical cryptography. New York: Springer, 2008.
41. **Silverman J. H.** An introduction to lattices, lattice reduction, and lattice-based cryptography // Lect. Notes 30th Annu. PCMI Graduate Summer School (Princeton, USA, July 5–25, 2020). Princeton: Inst. Adv. Study, 2020. 70 p. Available at `ias.edu/sites/default/files/Silverman_PCMI_Note_DistributionVersion_220705.pdf` (accessed Sept. 13, 2023).

42. Peikert C. A decade of lattice cryptography // Found. Trends Theor. Comput. Sci. 2016. V. 10, No. 4. P. 283–424.

43. Ajtai M. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract) // Proc. 30th Annu. ACM Symp. Theory of Computing (Dallas, USA, May 24–26, 1998). New York: ACM, 1998. P. 10–19.

44. Micciancio D. The shortest vector in a lattice is hard to approximate to within some constant // SIAM J. Comput. 2001. V. 30, No. 6. P. 2008–2035.

45. Haviv I., Regev O. Tensor-based hardness of the shortest vector problem to within almost polynomial factors // Proc. 39th Annu. ACM Symp. Theory of Computing (San Diego, CA, USA, June 11–13, 2007). New York: ACM, 2007. P. 469–477.

46. Aharonov D., Regev O. Lattice problems in NP $\cap$ coNP // J. ACM. 2005. V. 52, No. 5. P. 749–765.

47. Goldreich O., Micciancio D., Safra S., Seifert J.-P. Approximating shortest lattice vectors is not harder than approximating closest lattice vectors // Inf. Process. Lett. 1999. V. 71, No. 2. P. 55–61.

48. Micciancio D. Efficient reductions among lattice problems // Proc. 19th Annu. ACM-SIAM Symp. Discrete Algorithms (San Francisco, USA, Jan. 20–22, 2008). Philadelphia, PA: SIAM, 2008. P. 84–93.

49. Regev O. On lattices, learning with errors, random linear codes, and cryptography // J. ACM. 2009. V. 56, No. 6. P. 1–40.

50. Banerjee A., Peikert C., Rosen A. Pseudorandom functions and lattices // Advances in cryptology — EUROCRYPT 2012. Proc. 31st Annu. Int. Conf. Theory and Applications of Cryptographic Techniques (Cambridge, UK, Apr. 15–19, 2012). Heidelberg: Springer, 2012. P. 719–737. (Lect. Notes Comput. Sci.; V. 7237).

51. Alwen J., Krenn S., Pietrzak K., Wichs D. Learning with rounding, revisited // Advances in cryptology — CRYPTO 2013. Proc. 33rd Annu. Cryptology Conf. (Santa Barbara, USA, Aug. 18–22, 2013). Pt. I. Heidelberg: Springer, 2013. P. 57–74. (Lect. Notes Comput. Sci.; V. 8042).

52. 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.

53. 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 Sept. 13, 2023).

54. Submission requirements and evaluation criteria for the post-quantum cryptography standardization process. Gaithersburg, MD: NIST, 2016. Available at `csrc.nist.gov/csrc/media/projects/post-quantum-cryptography/documents/call-for-proposals-final-dec-2016.pdf` (accessed Sept. 13, 2023).

55. Avanzi R., Bos J., Ducas L. [et al.]. CRYSTALS-Kyber. Algorithm specifications and supporting documentation. Gaithersburg, MD: NIST, 2021. Available at `pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf` (accessed Sept. 13, 2023).

56. Fujisaki E., Okamoto T. Secure integration of asymmetric and symmetric encryption schemes // Advances in cryptology — CRYPTO'99. Proc. 19th Annu. Int. Cryptology Conf. (Santa Barbara, USA, Aug. 15–19, 1999). Heidelberg: Springer, 1999. P. 537–554. (Lect. Notes Comput. Sci.; V. 1666).

57. Basso A., Mera J. M. B., D'Anvers J.-P. [et al.]. SABER: mod-LWR based KEM (Round 3 Submission). Leuven: KU Leuven, 2020. Available at `esat.kuleuven.be/cosic/pqcrypto/saber/files/saberspecround3.pdf` (accessed Sept. 13, 2023).

58. Chen C., Danba O., Hoffstein J. [et al.]. NTRU. Algorithm specifications and supporting documentation. Eindhoven: Eindh. Univ. Technol., 2020. Available at `cryptojedi.org/papers/ntrunistr3-20200930.pdf` (accessed Sept. 13, 2023).

59. Targhi E. E., Unruh D. Post-quantum security of the Fujisaki — Okamoto and OAEP transforms // Theory of cryptography. Proc. 14th Int. Conf. (Beijing, China, Oct. 31 — Nov. 3, 2016). Pt. II. Heidelberg: Springer, 2016. P. 192–216. (Lect. Notes Comput. Sci.; V. 9986).

60. Alkim E., Bos J. W., Ducas L. [et al.]. FrodoKEM. Learning with errors key encapsulation: Algorithm specifications and supporting documentation. Gaithersburg, MD: NIST, 2021. Available at `frodokem.org/files/FrodoKEM-specification-20210604.pdf` (accessed Sept. 13, 2023).

61. Fujisaki E., Okamoto T. Secure integration of asymmetric and symmetric encryption schemes // J. Cryptol. 2013. V. 26. P. 80–101.

62. Bai S., Ducas L., Kiltz E. [et al.]. CRYSTALS-Dilithium. Algorithm specifications and supporting documentation. Gaithersburg, MD: NIST, 2021. Available at `pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf` (accessed Sept. 13, 2023).

63. Fouque P.-A., Hoffstein J., Kirchner P. [et al.]. Falcon: Fast-fourier lattice-based compact signatures over NTRU. Gaithersburg, MD: NIST, 2020. Available at `falcon-sign.info/falcon.pdf` (accessed Sept. 13, 2023).

64. Coppersmith D., Shamir A. Lattice attacks on NTRU // Advances in cryptology — EUROCRYPT'97. Proc. Int. Conf. Theory and Applications of Cryptographic Techniques (Konstanz, Germany, May 11–15, 1997). Heidelberg: Springer, 1997. P. 52–61. (Lect. Notes Comput. Sci.; V. 1233).

65. Schnorr C. P., Euchner M. Lattice basis reduction: Improved practical algorithms and solving subset sum problems // Math. Program. 1994. V. 66. P. 181–199.

66. Kannan R. Improved algorithms for integer programming and related lattice problems // Proc. 15th Annu. ACM Symp. Theory of Computing (Boston, MA, USA, Apr. 25–27, 1983). New York: ACM, 1983. P. 193–206.

67. Fincke U., Pohst M. Improved methods for calculating vectors of short length in a lattice, including a complexity analysis // Math. Comput. 1985. V. 44, No. 170. P. 463–471.

68. Gama N., Nguyen P. Q., Regev O. Lattice enumeration using extreme pruning // Advances in cryptology — EUROCRYPT 2010. Proc. 29th Annu. Int. Conf. Theory and Applications of Cryptographic Techniques (French Riviera, France, May 30 — June 3, 2010). Heidelberg: Springer, 2010. P. 257–278. (Lect. Notes Comput. Sci.; V. 6110).

69. Chen Y., Nguyen P. Q. BKZ 2.0: Better lattice security estimates // Advances in cryptology — ASIACRYPT 2011. Proc. 17th Int. Conf. Theory and Application of Cryptology and Information Security (Seoul, South Korea, Dec. 4–8, 2011). Heidelberg: Springer, 2011. P. 1–20. (Lect. Notes Comput. Sci.; V. 7073).

70. Ajtai M., Kumar R., Sivakumar D. A sieve algorithm for the shortest lattice vector problem // Proc. 33rd Annu. ACM Symp. Theory of Computing (Hersonissos, Greece, July 6–8, 2001). New York: ACM, 2001. P. 601–610.

71. Pujol X., Stehlé D. Solving the shortest lattice vector problem in time $2^{2,465n}$. San Diego: Univ. California, 2009. (Cryptology ePrint Archive; Paper ID 2009/605). Available at `eprint.iacr.org/2009/605` (accessed Sept. 13, 2023).

72. 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.

73. 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.

74. 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. P. 10–24.

75. Herold G., Kirshanova E., Laarhoven T. Speed-ups and time-memory trade-offs for tuple lattice sieving // Public-key cryptography — PKC 2018. Proc. 21st IACR Int. Conf. Practice and Theory of Public-Key Cryptography (Rio de Janeiro, Brazil, Mar. 25–29, 2018). Pt. I. Cham: Springer, 2018. P. 407–436. (Lect. Notes Comput. Sci.; V. 10769).

76. Laarhoven T., Mosca M., van de Pol J. Finding shortest lattice vectors faster using quantum search // Des. Codes Cryptogr. 2015. V. 77, No. 2–3. P. 375–400.

77. Laarhoven T. Search problems in cryptography: From fingerprinting to lattice sieving. Eindhoven: Tech. Univ. Eindhoven, 2016. 230 p.

78. 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 and Application of Cryptology and Information Security (Daejeon, South Korea, Dec. 7–11, 2020). Pt. II. Cham: Springer, 2020. P. 583–613. (Lect. Notes Comput. Sci.; V. 12492).

79. Kirshanova E., Mårtensson E., Postlethwaite E. W., Moulik S. R. Quantum algorithms for the approximate k-list problem and their application to lattice sieving // Advances in cryptology — ASIACRYPT 2019. Proc. 25th Int. Conf. Theory and Application of Cryptology and Information Security (Kobe, Japan, Dec. 8–12, 2019). Pt. I. Cham: Springer, 2019. P. 521–551. (Lect. Notes Comput. Sci.; V. 11921).

80. Micciancio D., Voulgaris P. A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations // Proc. 42nd ACM Symp. Theory of Computing (Cambridge, MA, USA, June 5–8, 2010). New York: ACM, 2010. P. 351–358.

81. Doulgerakis E., Laarhoven T., de Weger B. Finding closest lattice vectors using approximate Voronoi cells // Post-quantum cryptography. Rev. Sel. Pap. 10th Int. Conf. (Chongqing, China, May 8–10, 2019). Cham: Springer, 2019. P. 3–22. (Lect. Notes Comput. Sci.; V. 11505).

82. 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 of Computing (Portland, OR, USA, June 14–17, 2015). New York: ACM, 2015. P. 733–742.

83. Hanrot G., Stehlé D. Improved analysis of Kannan's shortest lattice vector algorithm // Advances in cryptology — CRYPTO 2007. Proc. 27th Annu. Cryptology Conf. (Santa Barbara, USA, Aug. 19–23, 2007). Heidelberg: Springer, 2007. P. 170–186. (Lect. Notes Comput. Sci.; V. 4622).

84. Hanrot G., Pujol X., Stehlé D. Algorithms for the shortest and closest lattice vector problems // Coding and cryptology. Proc. 3rd Int. Workshop (Qingdao, China, May 30 — June 3, 2011). Heidelberg: Springer, 2011. P. 159–190. (Lect. Notes Comput. Sci.; V. 6639).

85. Yang S. Y., Kuo P. C., Yang B. Y., Cheng C. M. Gauss sieve algorithm on GPUs // Topics in cryptology — CT-RSA 2017. Cryptographers' Track at the RSA Conf. 2017 (San Francisco, USA, Feb. 14–17, 2017). Cham: Springer, 2017. P. 39–57. (Lect. Notes Comput. Sci.; V. 10159).

86. Bai S., Laarhoven T., Stehlé D. Tuple lattice sieving // LMS J. Comput. Math. 2016. V. 19, No. A. P. 146–162.

87. Herold G., Kirshanova E. Improved algorithms for the approximate k-list problem in Euclidean norm // Public-key cryptography — PKC 2017. Proc. 20th IACR Int. Conf. Practice and Theory of Public-Key Cryptography (Amsterdam, Netherlands, Mar. 28–31, 2017). Pt. I. Heidelberg: Springer, 2017. P. 16–40. (Lect. Notes Comput. Sci.; V. 10174).

88. Becker A., Gama N., Joux A. A sieve algorithm based on overlattices // LMS J. Comput. Math. 2014. V. 17, No. A. P. 49–70.

89. Laarhoven T. Sieving for shortest vectors in lattices using angular locality-sensitive hashing // Advances in cryptology — CRYPTO 2015. Proc. 35th Annu. Cryptology Conf. (Santa Barbara, USA, Aug. 16–20, 2015). Pt. I. Heidelberg: Springer, 2015. P. 3–22. (Lect. Notes Comput. Sci.; V. 9215).

90. Laarhoven T., Mariano A. Progressive lattice sieving // Post-quantum cryptography. Proc. 9th Int. Conf. (Fort Lauderdale, FL, USA, Apr. 9–11, 2018). Cham: Springer, 2018. P. 292–311. (Lect. Notes Comput. Sci.; V. 10786).

91. Andoni A., Indyk P., Nguyen H. L., Razenshteyn I. Beyond locality-sensitive hashing // Proc. 25th Annu. ACM-SIAM Symp. Discrete Algorithms (Portland, Oregon, USA, Jan. 5–7, 2014). Philadelphia, PA: SIAM, 2014. P. 1018–1028.

92. Laarhoven T., de Weger B. Faster sieving for shortest lattice vectors using spherical locality-sensitive hashing // Progress in cryptology — LATINCRYPT 2015. Proc. 4th Int. Conf. Cryptology and Information Security in Latin America (Guadalajara, Mexico, Aug. 23–26, 2015). Cham: Springer, 2015. P. 101–118. (Lect. Notes Comput. Sci.; V. 9230).

93. Ducas L., Stevens M., van Woerden W. Advanced lattice sieving on GPUs, with tensor cores // Advances in cryptology — EUROCRYPT 2021. Proc. 40th Annu. Int. Conf. Theory and Applications of Cryptographic Techniques (Zagreb, Croatia, Oct. 17–21, 2021). Pt. II. Cham: Springer, 2021. P. 249–279. (Lect. Notes Comput. Sci.; V. 12697).

Малыгина Екатерина Сергеевна
Куценко Александр Владимирович
Новосёлов Семён Александрович
Колесников Никита Сергеевич
Бахарев Александр Олегович
Хильчук Ирина Сергеевна
Шапоренко Александр Сергеевич
Токарева Наталья Николаевна

Статья поступила
4 мая 2023 г.

После доработки —
28 июля 2023 г.

Принята к публикации
20 августа 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
October–December 2023. Vol. 30, No. 4. P. 46–90

UDC 519.7 DOI 10.33048/daio.2023.30.771

POST-QUANTUM CRYPTOSYSTEMS: OPEN PROBLEMS AND SOLUTIONS. LATTICE-BASED CRYPTOSYSTEMS

E. S. Malygina[1, 2, a], *A. V. Kutsenko*[2, b], *S. A. Novoselov*[1, c], *N. S. Kolesnikov*[1, d], *A. O. Bakharev*[2, e], *I. S. Khilchuk*[2, f], *A. S. Shaporenko*[2, g], and *N. N. Tokareva*[2, 1, h]

[1] Immanuel Kant Baltic Federal University,
14 Aleksandr Nevskii Street, 236041 Kaliningrad, Russia

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

E-mail: [a] emalygina@kantiana.ru, [b] alexandrkutsenko@bk.ru, [c] novsem@gmail.com, [d] nikolesnikov100@gmail.com, [e] a.bakharev@g.nsu.ru, [f] irina.khilchuk@gmail.com, [g] shaporenko.alexandr@gmail.com, [h] crypto1127@mail.ru

Abstract. The paper provides an overview of the main approaches to the construction of post-quantum cryptographic systems that are currently used. The area of lattice-based cryptography is analyzed in detail. We give the description and characteristics of some known lattice-based cryptosystems whose security is based on the complexity of the shortest vector problem, learning with errors problem, and their variations. The main approaches to solving the problems from lattice theory, on which attacks on the corresponding cryptosystems are based, are analyzed. In particular, some known theoretical estimates of time and memory complexity of lattice basis reduction and lattice sieving algorithms are presented. Tab. 6, illustr. 1, biblogr. 93.

Keywords: post-quantum cryptography, quantum computer, integer lattice.

The work of the first, third, and fourth authors is supported by the Kovalevskaya North-Western Mathematical Center of Immanuel Kant Baltic Federal University under the agreement with the Ministry of Science and Higher Education of Russia (Agreement 075–02–2023–934). The work of the second, fifth, sixth, seventh, and eighth authors is supported by the Mathematical Center in Akademgorodok under 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** (4), 767–790 (2023), DOI 10.1134/S1990478923040087.

REFERENCES

1. **D. J. Bernstein,** Introduction to post-quantum cryptography, in *Post-Quantum Cryptography* (Springer, Heidelberg, 2009), pp. 1–14.
2. **C. Gidney** and **M. Ekerå,** How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, *Quantum* **5**, 433 (2021).
3. **C. H. Bennett** and **G. Brassard,** Quantum cryptography: Public key distribution and coin tossing, *Theor. Comput. Sci.* **560**, 7–11 (2014).
4. **Yu. I. Manin,** *Computable and Incomputable* (Sov. Radio, Moscow, 1980) [Russian].
5. **R. P. Feynman,** Simulating physics with computers, *Int. J. Theor. Phys.* **21**, 467–468 (1982).
6. **D. Deutsch,** Quantum theory, the Church — Turing principle and the universal quantum computer, *Proc. R. Soc. Lond. Ser. A. Math. Phys. Sci.* **400** (1818), 97–117 (1985).
7. **D. Deutsch** and **R. Jozsa,** Rapid solution of problems by quantum computation, *Proc. R. Soc. Lond. Ser. A. Math. Phys. Sci.* **439** (1907), 553–558 (1992).
8. **E. Bernstein** and **U. Vazirani,** Quantum complexity theory, *SIAM J. Comput.* **26** (5), 1411–1473 (1997).
9. **D. R. Simon,** On the power of quantum computation, *SIAM J. Comput.* **26** (5), 1474–1483 (1997).
10. **M. A. Nielsen** and **I. L. Chuang,** *Quantum Computation and Quantum Information* (Camb. Univ. Press, Cambridge, 2010).
11. **P. W. Shor,** Algorithms for quantum computation: Discrete logarithms and factoring, in *Proc. 35th Annu. Symp. Foundations of Computer Science, Santa Fe, USA, Nov. 20–22, 1994* (IEEE Comput. Soc., Los Alamitos, CA, 1994), pp. 124–134.
12. **J. Proos** and **C. Zalka,** Shor's discrete logarithm quantum algorithm for elliptic curves, *Quantum Inf. Comput.* **3** (4), 317–344 (2003).
13. **L. K. Grover,** A fast quantum mechanical algorithm for database search, in *Proc. 28th ACM Symp. Theory of Computing, Philadelphia, PA, USA, May 22–24, 1996* (ACM, New York, 1996), pp. 212–219.
14. **G. Brassard, P. Høyer** and **A. Tapp,** Quantum cryptanalysis of hash and claw-free functions, in *LATIN'98: Theoretical Informatics* (Proc. 3rd Lat. Am. Symp., Campinas, Brazil, Apr. 20–24, 1998) (Springer, Heidelberg, 1998), pp. 163–169 (Lect. Notes Comput. Sci., Vol. 1380).
15. **G. Brassard, P. Høyer** and **A. Tapp,** Quantum counting, in *Automata, Languages and Programming* (Proc. 25th Int. Colloq., Aalborg, Denmark, July 13–17, 1998) (Springer, Heidelberg, 1998), pp. 820–831 (Lect. Notes Comput. Sci., Vol. 1443).
16. **H. Kuwakado** and **M. Morii,** Security on the quantum-type Even — Mansour cipher, in *Proc. 2012 Int. Symp. Information Theory and Its Applications, Honolulu, HI, USA, Oct. 28–31, 2012* (IEEE Comput. Soc., Los Alamitos, CA, 2012), pp. 312–316.
17. **X. Dong, B. Dong,** and **X. Wang,** Quantum attacks on some Feistel block ciphers, *Des. Codes Cryptogr.* **88** (6), 1179–1203 (2020).

18. **H. Xie** and **L. Yang,** Using Bernstein — Vazirani algorithm to attack block ciphers, *Des. Codes Cryptogr.* **87** (5), 1161–1182 (2019).
19. **G. Leander** and **A. May,** Grover meets Simon — quantumly attacking the FX-construction, in *Advances in Cryptology — ASIACRYPT 2017* (Proc. 23rd Int. Conf. Theory and Applications of Cryptology and Information Security, Hong Kong, China, Dec. 3–7, 2017), Pt. II (Springer, Cham, 2017), pp. 161–178 (Lect. Notes Comput. Sci., Vol. 10625).
20. **H. Kuwakado** and **M. Morii,** Quantum distinguisher between the 3-round Feistel cipher and the random permutation, in *Proc. 2010 IEEE Int. Symp. Information Theory, Austin, TX, USA, June 13–18, 2010* (IEEE Comput. Soc., Los Alamitos, CA, 2010), pp. 2682–2685.
21. **S. Hodžić** and **L. R. Knudsen,** A quantum distinguisher for 7/8-round SMS4 block cipher, *Quantum Inf. Process.* **19** (11), Paper ID 411 (2020).
22. **Q. Zhou, S. Lu, Z. Zhang,** and **J. Sun,** Quantum differential cryptanalysis, *Quantum Inf. Process.* **14** (6), 2101–2109 (2015).
23. **R. Shi, H. Xie, H. Feng, F. Yuan,** and **B. Liu,** Quantum zero correlation linear cryptanalysis, *Quantum Inf. Process.* **21** (8), Paper ID 293 (2022).
24. **M. Kaplan, G. Leurent, A. Leverrier,** and **M. Naya-Plasencia,** Quantum differential and linear cryptanalysis, *IACR Trans. Symmetric Cryptol.* **2016** (1), 71–94 (2016).
25. **L. Chen, S. Jordan, Y.-K. Liu,** [et al.], Report on post-quantum cryptography, *Nat. Inst. Stand. Technol. Interag. Intern. Rep. NIST IR 8105* (NIST, Gaithersburg, MD, 2016). Available at `doi.org/10.6028/NIST.IR.8105` (accessed Sept. 13, 2023).
26. *Post-Quantum Cryptography* (NIST, Gaithersburg, MD, 2017). Available at `csrc.nist.gov/projects/post-quantum-cryptography` (accessed Sept. 13, 2023).
27. **A. K. Lenstra, H. W. Lenstra,** and **L. Lovász,** Factoring polynomials with rational coefficients, *Math. Ann.* **261** (4), 515–534 (1982).
28. **A. Shamir,** A polynomial-time algorithm for breaking the basic Merkle — Hellman cryptosystem, in *Advances in Cryptology* (Proc. Crypto 82, Santa Barbara, USA, Aug. 23–25, 1982) (Plenum Press, New York, 1983), pp. 279–288.
29. **C. P. Schnorr,** A hierarchy of polynomial time lattice basis reduction algorithms, *Theor. Comput. Sci.* **53** (2–3), 201–224 (1987).
30. **C. P. Schnorr,** A more efficient algorithm for lattice basis reduction, *J. Algorithms* **9** (1), 47–62 (1988).
31. **A. Frieze, J. Håstad, R. Kannan, J. Lagarias,** and **A. Shamir,** Reconstructing truncated integer variables satisfying linear congruences, *SIAM J. Comput.* **17** (2), 262–280 (1988).
32. **J. Stern** and **P. Toffin,** Cryptanalysis of a public-key cryptosystem based on approximations by rational numbers, in *Advances in Cryptology — EUROCRYPT'90* (Proc. Workshop Theory and Application of Cryptographic Techniques, Aarhus, Denmark, May 21–24, 1990) (Springer, Heidelberg, 1991), pp. 313–317 (Lect. Notes Comput. Sci., Vol. 473).

33. A. Joux and **J. Stern,** Cryptanalysis of another knapsack cryptosystem, in *Advances in Cryptology — ASIACRYPT'91* (Proc. Int. Conf. Theory and Application of Cryptology, Fujiyoshida, Japan, Nov. 11–14, 1991) (Springer, Heidelberg, 1993), pp. 470–476 (Lect. Notes Comput. Sci., Vol. 739).

34. M. Ajtai, Generating hard instances of lattice problems (extended abstract), in *Proc. 28th Annu. ACM Symp. Theory of Computing, Philadelphia, PA, USA, May 22–24, 1996* (ACM, New York, 1996), pp. 99–108.

35. M. Ajtai and **C. Dwork,** A public-key cryptosystem with worst-case/average-case equivalence, in *Proc. 29th Annu. ACM Symp. Theory of Computing, El Paso, TX, USA, May 4–6, 1997* (ACM, New York, 1997), pp. 284–293.

36. O. Goldreich, S. Goldwasser, and **S. Halevi,** Public-key cryptosystems from lattice reduction problems, in *Advances in Cryptology — CRYPTO'97* (Proc. 17th Annu. Int. Cryptology Conf., Santa Barbara, USA, Aug. 17–21, 1997) (Springer, Heidelberg, 1997), pp. 112–131 (Lect. Notes Comput. Sci., Vol. 1294).

37. P. Nguyen and **J. Stern,** Cryptanalysis of the Ajtai — Dwork cryptosystem, in *Advances in Cryptology — CRYPTO'98* (Proc. 18th Annu. Int. Cryptology Conf., Santa Barbara, USA, Aug. 23–27, 1998) (Springer, Heidelberg, 1998), pp. 223–242 (Lect. Notes Comput. Sci., Vol. 1462).

38. P. Nguyen and **J. Stern,** Cryptanalysis of the Goldreich — Goldwasser — Halevi Cryptosystem from CRYPTO'97, in *Advances in Cryptology — CRYPTO'99* (Proc. 19th Annu. Int. Cryptology Conf., Santa Barbara, USA, Aug. 15–19, 1999) (Springer, Heidelberg, 1999), pp. 288–304 (Lect. Notes Comput. Sci., Vol. 1666).

39. J. Hoffstein, J. Pipher, and **J. H. Silverman,** NTRU: A ring-based public key cryptosystem, in *Algorithmic Number Theory* (Proc. 3rd Int. Symp., Portland, OR, USA, June 21–25, 1998) (Springer, Heidelberg, 1998), pp. 267–288 (Lect. Notes Comput. Sci., Vol. 1423).

40. J. H. Silverman, J. Pipher, and **J. Hoffstein,** *An Introduction to Mathematical Cryptography* (Springer, New York, 2008).

41. J. H. Silverman, An introduction to lattices, lattice reduction, and lattice-based cryptography, in *Lect. Notes 30th Annu. PCMI Graduate Summer School, Princeton, USA, July 5–25, 2020* (Inst. Adv. Study, Princeton, 2020). Available at `ias.edu/sites/default/files/Silverman_PCMI_Note_DistributionVersion_220705.pdf` (accessed Sept. 13, 2023).

42. C. Peikert, A decade of lattice cryptography, *Found. Trends Theor. Comput. Sci.* **10** (4), 283–424 (2016).

43. M. Ajtai, The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract), in *Proc. 30th Annu. ACM Symp. Theory of Computing, Dallas, USA, May 24–26, 1998* (ACM, New York, 1998), pp. 10–19.

44. D. Micciancio, The shortest vector in a lattice is hard to approximate to within some constant, *SIAM J. Comput.* **30** (6), 2008–2035 (2001).

45. I. Haviv and **O. Regev,** Tensor-based hardness of the shortest vector problem to within almost polynomial factors, in *Proc. 39th Annu. ACM Symp. Theory of Computing, San Diego, CA, USA, June 11–13, 2007* (ACM, New York, 2007), pp. 469–477.

46. D. Aharonov and **O. Regev,** Lattice problems in NP∩coNP, *J. ACM* **52** (5), 749–765 (2005).

47. O. Goldreich, D. Micciancio, S. Safra, and **J.-P. Seifert,** Approximating shortest lattice vectors is not harder than approximating closest lattice vectors, *Inf. Process. Lett.* **71** (2), 55–61 (1999).

48. D. Micciancio, Efficient reductions among lattice problems, in *Proc. 19th Annu. ACM-SIAM Symp. Discrete Algorithms, San Francisco, USA, Jan. 20–22, 2008* (SIAM, Philadelphia, PA, 2008), pp. 84–93.

49. O. Regev, On lattices, learning with errors, random linear codes, and cryptography, *J. ACM* **56** (6), 1–40 (2009).

50. A. Banerjee, C. Peikert, and **A. Rosen,** Pseudorandom functions and lattices, in *Advances in Cryptology — EUROCRYPT 2012* (Proc. 31st Annu. Int. Conf. Theory and Applications of Cryptographic Techniques, Cambridge, UK, Apr. 15–19, 2012) (Springer, Heidelberg, 2012), pp. 719–737 (Lect. Notes Comput. Sci., Vol. 7237).

51. J. Alwen, S. Krenn, K. Pietrzak, and **D. Wichs,** Learning with rounding, revisited, in *Advances in Cryptology — CRYPTO 2013* (Proc. 33rd Annu. Cryptology Conf., Santa Barbara, USA, Aug. 18–22, 2013), Pt. I (Springer, Heidelberg, 2013), pp. 57–74 (Lect. Notes Comput. Sci., Vol. 8042).

52. 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).

53. 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 Sept. 13, 2023).

54. Submission requirements and evaluation criteria for the post-quantum cryptography standardization process (NIST, Gaithersburg, MD, 2016). Available at `csrc.nist.gov/csrc/media/projects/post-quantum-cryptography/documents/call-for-proposals-final-dec-2016.pdf` (accessed Sept. 13, 2023).

55. R. Avanzi, J. Bos, L. Ducas, [et al.], CRYSTALS-Kyber. Algorithm specifications and supporting documentation (NIST, Gaithersburg, MD, 2021). Available at `pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf` (accessed Sept. 13, 2023).

56. E. Fujisaki and **T. Okamoto,** Secure integration of asymmetric and symmetric encryption schemes, in *Advances in Cryptology — CRYPTO'99* (Proc. 19th Annu. Int. Cryptology Conf., Santa Barbara, USA, Aug. 15–19, 1999) (Springer, Heidelberg, 1999), pp. 537–554 (Lect. Notes Comput. Sci., Vol. 1666).

57. A. Basso, J. M. B. Mera, J.-P. D'Anvers, [et al.], SABER: mod-LWR based KEM (Round 3 Submission) (KU Leuven, Leuven, 2020). Available at `esat.kuleuven.be/cosic/pqcrypto/saber/files/saberspecround3.pdf` (accessed Sept. 13, 2023).

58. C. Chen, O. Danba, J. Hoffstein, [et al.], NTRU. Algorithm specifications and supporting documentation (Eindh. Univ. Technol., Eindhoven, 2020). Available at `cryptojedi.org/papers/ntrunistr3-20200930.pdf` (accessed Sept. 13, 2023).

59. E. E. Targhi and **D. Unruh,** Post-quantum security of the Fujisaki — Okamoto and OAEP transforms, in *Theory of Cryptography* (Proc. 14th Int. Conf., Beijing, China, Oct. 31 — Nov. 3, 2016), Pt. II (Springer, Heidelberg, 2016), pp. 192–216 (Lect. Notes Comput. Sci., Vol. 9986).

60. E. Alkim, J. W. Bos, L. Ducas, [et al.],. FrodoKEM. Learning with errors key encapsulation: Algorithm specifications and supporting documentation (NIST, Gaithersburg, MD, 2021). Available at `frodokem.org/files/FrodoKEM-specification-20210604.pdf` (accessed Sept. 13, 2023).

61. E. Fujisaki and **T. Okamoto,** Secure integration of asymmetric and symmetric encryption schemes, *J. Cryptol.* **26**, 80–101 (2013).

62. S. Bai, L. Ducas, E. Kiltz, [et al.], CRYSTALS-Dilithium. Algorithm specifications and supporting documentation (NIST, Gaithersburg, MD, 2021). Available at `pq-crystals.org/dilithium/data/dilithium-specification-round3-20210208.pdf` (accessed Sept. 13, 2023).

63. P.-A. Fouque, J. Hoffstein, P. Kirchner, [et al.], Falcon: Fast-fourier lattice-based compact signatures over NTRU (NIST, Gaithersburg, MD, 2020). Available at `falcon-sign.info/falcon.pdf` (accessed Sept. 13, 2023).

64. D. Coppersmith and **A. Shamir,** Lattice attacks on NTRU, in *Advances in Cryptology — EUROCRYPT'97* (Proc. Int. Conf. Theory and Applications of Cryptographic Techniques, Konstanz, Germany, May 11–15, 1997) (Springer, Heidelberg, 1997), pp. 52–61 (Lect. Notes Comput. Sci., Vol. 1233).

65. C. P. Schnorr and **M. Euchner,** Lattice basis reduction: Improved practical algorithms and solving subset sum problems, *Math. Program.* **66**, 181–199 (1994).

66. R. Kannan, Improved algorithms for integer programming and related lattice problems, in *Proc. 15th Annu. ACM Symp. Theory of Computing, Boston, MA, USA, Apr. 25–27, 1983* (ACM, New York, 1983), pp. 193–206.

67. U. Fincke and **M. Pohst,** Improved methods for calculating vectors of short length in a lattice, including a complexity analysis, *Math. Comput.* **44** (170), 463–471 (1985).

68. N. Gama, P. Q. Nguyen, and **O. Regev,** Lattice enumeration using extreme pruning, in *Advances in Cryptology — EUROCRYPT 2010* (Proc. 29th Annu. Int. Conf. Theory and Applications of Cryptographic Techniques, French Riviera, France, May 30 — June 3, 2010) (Springer, Heidelberg, 2010), pp. 257–278 (Lect. Notes Comput. Sci., Vol. 6110).

69. Y. Chen and **P. Q. Nguyen,** BKZ 2.0: Better lattice security estimates, in *Advances in Cryptology — ASIACRYPT 2011* (Proc. 17th Int. Conf. Theory and Application of Cryptology and Information Security, Seoul, South Korea, Dec. 4–8, 2011) (Springer, Heidelberg, 2011), pp. 1–20 (Lect. Notes Comput. Sci., Vol. 7073).

70. M. Ajtai, R. Kumar, D. Sivakumar A sieve algorithm for the shortest lattice vector problem, in *Proc. 33rd Annu. ACM Symp. Theory of Computing* (Hersonissos, Greece, July 6–8, 2001) (ACM, New York, 2001), pp. 601–610.

71. X. Pujol and **D. Stehlé,** Solving the shortest lattice vector problem in time $2^{2,465n}$ (Univ. California, San Diego, 2009) (Cryptology ePrint Archive, Paper ID 2009/605). Available at `eprint.iacr.org/2009/605` (accessed Sept. 13, 2023).

72. P. Q. Nguyen and **T. Vidick,** Sieve algorithms for the shortest vector problem are practical, *J. Math. Cryptol.* **2** (2), 181–207 (2008).

73. 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.

74. 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.

75. G. Herold, E. Kirshanova, T. Laarhoven Speed-ups and time-memory trade-offs for tuple lattice sieving, in *Public-Key Cryptography — PKC 2018* (Proc. 21st IACR Int. Conf. Practice and Theory of Public-Key Cryptography, Rio de Janeiro, Brazil, Mar. 25–29, 2018), Pt. I (Springer, Cham, 2018), pp. 407–436 (Lect. Notes Comput. Sci., Vol. 10769).

76. T. Laarhoven, M. Mosca, and **J. van de Pol,** Finding shortest lattice vectors faster using quantum search, *Des. Codes Cryptogr.* **77** (2–3), 375–400 (2015).

77. T. Laarhoven, *Search Problems in Cryptography: From Fingerprinting to Lattice Sieving* (Tech. Univ. Eindhoven, Eindhoven, 2016).

78. 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 and Application of Cryptology and Information Security, Daejeon, South Korea, Dec. 7–11, 2020), Pt. II (Springer, Cham, 2020), pp. 583–613 (Lect. Notes Comput. Sci., Vol. 12492).

79. E. Kirshanova, E. Mårtensson, E. W. Postlethwaite, and **S. R. Moulik,** Quantum algorithms for the approximate k-list problem and their application to lattice sieving, in *Advances in Cryptology — ASIACRYPT 2019* (Proc. 25th Int. Conf. Theory and Application of Cryptology and Information Security, Kobe, Japan, Dec. 8–12, 2019), Pt. I (Springer, Cham, 2019), pp. 521–551 (Lect. Notes Comput. Sci., Vol. 11921).

80. D. Micciancio, P. Voulgaris A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations, in *Proc. 42nd ACM Symp. Theory of Computing, Cambridge, MA, USA, June 5–8, 2010* (ACM, New York, 2010), pp. 351–358.

81. E. Doulgerakis, T. Laarhoven, and **B. de Weger,** Finding closest lattice vectors using approximate Voronoi cells, in *Post-Quantum Cryptography* (Rev. Sel. Pap. 10th Int. Conf., Chongqing, China, May 8–10, 2019) (Springer, Cham, 2019), pp. 3–22 (Lect. Notes Comput. Sci., Vol. 11505).

82. 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 of Computing, Portland, OR, USA, June 14–17, 2015* (ACM, New York, 2015), pp. 733–742.

83. G. Hanrot and **D. Stehlé,** Improved analysis of Kannan's shortest lattice vector algorithm, in *Advances in Cryptology — CRYPTO 2007* (Proc. 27th Annu. Cryptology Conf., Santa Barbara, USA, Aug. 19–23, 2007) (Springer, Heidelberg, 2007), pp. 170–186 (Lect. Notes Comput. Sci., Vol. 4622).

84. G. Hanrot, X. Pujol, and **D. Stehlé,** Algorithms for the shortest and closest lattice vector problems, in *Coding and Cryptology* (Proc. 3rd Int. Workshop, Qingdao, China, May 30 — June 3, 2011) (Springer, Heidelberg, 2011), pp. 159–190 (Lect. Notes Comput. Sci., Vol. 6639).

85. S. Y. Yang, P. C. Kuo, B. Y. Yang, and **C. M. Cheng,** Gauss sieve algorithm on GPUs, in *Topics in Cryptology — CT-RSA 2017* (Cryptographers' Track at the RSA Conf. 2017, San Francisco, USA, Feb. 14–17, 2017) (Springer, Cham, 2017), pp. 39–57 (Lect. Notes Comput. Sci., Vol. 10159).

86. S. Bai, T. Laarhoven, and **D. Stehlé,** Tuple lattice sieving, *LMS J. Comput. Math.* **19** (A), 146–162 (2016).

87. G. Herold and **E. Kirshanova,** Improved algorithms for the approximate k-list problem in Euclidean norm, in *Public-Key Cryptography — PKC 2017* (Proc. 20th IACR Int. Conf. Practice and Theory of Public-Key Cryptography, Amsterdam, Netherlands, Mar. 28–31, 2017), Pt. I (Springer, Heidelberg, 2017), pp. 16–40 (Lect. Notes Comput. Sci., Vol. 10174).

88. A. Becker, N. Gama, and **A. Joux,** A sieve algorithm based on overlattices, *LMS J. Comput. Math.* **17** (A), 49–70 (2014).

89. T. Laarhoven, Sieving for shortest vectors in lattices using angular locality-sensitive hashing, in *Advances in Cryptology — CRYPTO 2015* (Proc. 35th Annu. Cryptology Conf., Santa Barbara, USA, Aug. 16–20, 2015), Pt. I (Springer, Heidelberg, 2015), pp. 3–22 (Lect. Notes Comput. Sci., Vol. 9215).

90. T. Laarhoven and **A. Mariano,** Progressive lattice sieving, in *Post-Quantum Cryptography* (Proc. 9th Int. Conf., Fort Lauderdale, FL, USA, Apr. 9–11, 2018) (Springer, Cham, 2018), pp. 292–311 (Lect. Notes Comput. Sci., Vol. 10786).

91. A. Andoni, P. Indyk, H. L. Nguyen, and **I. Razenshteyn,** Beyond locality-sensitive hashing, in *Proc. 25th Annu. ACM-SIAM Symp. Discrete Algorithms, Portland, Oregon, USA, Jan. 5–7, 2014* (SIAM, Philadelphia, PA, 2014), pp. 1018–1028.

92. T. Laarhoven and **B. de Weger,** Faster sieving for shortest lattice vectors using spherical locality-sensitive hashing, in *Progress in Cryptology — LATINCRYPT 2015* (Proc. 4th Int. Conf. Cryptology and Information Security in Latin America, Guadalajara, Mexico, Aug. 23–26, 2015) (Springer, Cham, 2015), pp. 101–118 (Lect. Notes Comput. Sci., Vol. 9230).

93. L. Ducas, M. Stevens, and **W. van Woerden,** Advanced lattice sieving on GPUs, with tensor cores, in *Advances in Cryptology — EUROCRYPT 2021* (Proc. 40th Annu. Int. Conf. Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, Oct. 17–21, 2021), Pt. II (Springer, Cham, 2021), pp. 249–279 (Lect. Notes Comput. Sci., Vol. 12697).

Ekaterina S. Malygina
Aleksandr V. Kutsenko
Semyon A. Novoselov
Nikita S. Kolesnikov
Aleksandr O. Bakharev
Irina S. Khilchuk
Aleksandr S. Shaporenko
Natalia N. Tokareva

Received May 4, 2023
Revised July 28, 2023
Accepted August 20, 2023

ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Октябрь–декабрь 2023. Т. 30, № 4. С. 91–109

УДК 519.17 DOI 10.33048/daio.2023.30.758

ПОЛНАЯ СЛОЖНОСТНАЯ ДИХОТОМИЯ ЗАДАЧИ О РЁБЕРНОЙ РАСКРАСКЕ ДЛЯ ВСЕХ МНОЖЕСТВ 8-РЁБЕРНЫХ ЗАПРЕЩЁННЫХ ПОДГРАФОВ

Д. С. Малышев[1,2,a], *О. И. Дугинов*[3,b]

[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

[3] Белорусский гос. университет, пр. Независимости, 4, 220030 Минск, Беларусь

E-mail: [a] dsmalyshev@rambler.ru, [b] oduginov@gmail.com

Аннотация. Задача о рёберной раскраске для заданного графа состоит в том, чтобы минимизировать количество цветов, достаточное для окрашивания его рёбер так, чтобы соседние рёбра были окрашены в разные цвета. Для всех классов графов, определяемых множествами запрещённых подграфов с 7 рёбрами каждый, известен сложностной статус данной задачи. В данной работе получен аналогичный результат для всех множеств 8-рёберных запретов. Ил. 2, библиогр. 38.

Ключевые слова: задача о рёберной раскраске, вычислительная сложность, монотонный класс.

Введение

В работе рассматриваются *обыкновенные* графы, т. е. неориентированные графы без петель и кратных рёбер. *Наследственный* класс графов — множество графов, замкнутое относительно изоморфизма и удаления вершин. Каждый наследственный класс $\mathcal{X}$ (и только наследственный класс) может быть задан множеством своих запрещённых порождённых подграфов $\mathcal{Y}$, при этом принята запись $\mathcal{X} = \mathrm{Free}(\mathcal{Y})$. *Монотонный* класс

Разделы 2 и 3 выполнены при финансовой поддержке Российского фонда фундаментальных исследований и Белорусского республиканского фонда фундаментальных исследований (проект № 20–51–04001 (Ф21РМ-001)). Разделы 4 и 5 выполнены при финансовой поддержке Российского научного фонда (проект № 21–11–00194).

графов — наследственный класс, замкнутый ещё и относительно удаления рёбер. Каждый монотонный класс (и только монотонный класс) $\mathcal{X}$ может быть задан множеством своих запрещённых подграфов $\mathcal{Y}$, при этом пишем $\mathcal{X} = \mathrm{Free}_s(\mathcal{Y})$.

Пусть $G = (V, E)$ — граф. Любое отображение $c\colon E \to \{1, 2, \ldots, k\}$ такое, что $c(e_1) \neq c(e_2)$ для всех смежных рёбер e_1 и e_2, называется *рёберной k-раскраской* графа G. *Хроматический индекс* G — наименьшее число k, для которого существует k-раскраска рёбер G. Он обозначается через $\chi'(G)$.

Задача о рёберной k-раскраске (*задача* k-PP) состоит для заданного графа G в том, чтобы распознать, выполняется ли неравенство $\chi'(G) \leqslant k$. *Задача о рёберной раскраске* (*задача* PP) для заданных графа G и числа k состоит в том, чтобы распознать, выполняется ли неравенство $\chi'(G) \leqslant k$. Хорошо известно, что задача 3-PP (а следовательно, и задача PP) NP-полна [1].

По известному результату В. Г. Визинга [2] справедливо неравенство $\Delta(G) \leqslant \chi'(G) \leqslant \Delta(G) + 1$, где $\Delta(G)$ — максимальная из степеней вершин G. Тем самым задача PP для графа G эквивалентна распознаванию того, верно равенство $\chi'(G) = \Delta(G)$ или нет.

В работе [3] при любом k представлена полная классификация сложности задачи k-PP для всех наследственных классов, задаваемых одним запрещённым порождённым подграфом. В [4] получена полная дихотомия сложности задачи 3-PP для пар 6-вершинных запрещённых порождённых фрагментов, а в [5, 6] — аналогичный результат для задачи PP и семейств монотонных классов, задаваемых запрещением подграфов с не более чем 7 вершинами или 7 рёбрами каждый.

Некоторые результаты для вершинных аналогов задач k-PP и PP представлены в работах [8–36].

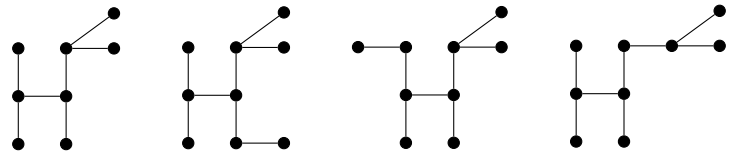


Рис. 1. Графы B_1^*, B_{1+}^*, ${}^+B_1^*$, B_1^{+*}

Через $G_1 + G_2$ обозначается дизъюнктное объединение графов G_1 и G_2 с непересекающимися множествами вершин, а через P_n и O_n — простой путь и пустой граф с n вершинами. В работе [7] рассматривались деревья B_1^*, B_{1+}^*, ${}^+B_1^*$, B_1^{+*} (рис. 1) и была доказана

Теорема 1. *Пусть F — произвольный 8-рёберный лес, не принадлежащий множеству*

$$\{B_1^* + P_2 + O_n,\ {}^+B_1^* + O_n,\ B_1^{+*} + O_n,\ B_{1+}^* + O_n \mid n \geqslant 0\}.$$

Тогда задача PP *полиномиально разрешима в классе* $\mathrm{Free}_s(\{F\})$. *Если F принадлежит данному множеству, то задача* PP *полиномиально разрешима в классе* $\{G \in \mathrm{Free}_s(\{F\}) \mid \Delta(G) \geqslant 4\}$.

В настоящей работе улучшаются результаты из [6, 7]. А именно, получена полная классификация сложности задачи PP для всех множеств 8-рёберных запретов.

1. Некоторые определения, обозначения и факты

Обхватом графа называется длина кратчайшего цикла, содержащегося в данном графе. Если граф ациклический, то его обхват полагается равным бесконечности. Для графа $G = (V, E)$ операция *стягивания* его (связного) подграфа $H = (V', E')$ в вершину состоит в удалении из G всех вершин подграфа H, последующего добавления новой вершины v и всех рёбер вида vu таких, что $u \in V \setminus V'$ и существует ребро $wu \in E$, где $w \in V'$.

Пусть G — некоторый граф, а x — вершина G. Окрестность x обозначается через $N(x)$. Через $\deg(x)$ обозначается степень x, а через $\Delta(G)$ — максимальная из степеней вершин G. Если $\Delta(G) \leqslant 3$, то G называется *субкубическим*. Если степени всех вершин графа равны 3, то он называется *кубическим*.

В разд. 28.1 монографии [37] (см. доказательство теоремы 28.1) доказана

Лемма 1. *Для любого графа G, содержащего вершину x, для которой* $|\{y \in N(x) \mid \deg(y) = \Delta(G)\}| \leqslant 1$, *выполнено*

$$\chi'(G) = \Delta(G) \Leftrightarrow \chi'(G \setminus \{x\}) \leqslant \Delta(G).$$

Шарниром называется вершина графа, удаление которой увеличивает количество его компонент связности. Связный граф G без шарниров назовём *несжимаемым*, если любая вершина G имеет не менее двух соседей степени $\Delta(G)$. В [7, разд. 2] отмечено, что справедлива

Лемма 2. *Задача* PP *для графов из произвольного монотонного класса полиномиально сводится к той же задаче для несжимаемых графов из этого монотонного класса.*

Пусть G — граф и $V' \subseteq V(G)$. Тогда $G[V']$ — подграф графа G, порождённый V', а $G \setminus V'$ — результат удаления из G всех элементов V'.

Пусть G_1 и G_2 — графы. Запись $G_1 \cong G_2$ означает, что графы G_1 и G_2 изоморфны. Если $V(G_1) \cap V(G_2) = \varnothing$, то граф $(V(G_1) \cup V(G_2)$,

$E(G_1) \cup E(G_2)$) обозначим через $G_1 + G_2$. Для графа G и числа k положим $kG = \underbrace{G + G + \cdots + G}_{k \text{ раз}}$.

Пусть G, H_1, H_2, …, H_k — графы. Тогда запись $\langle G; H_1, H_2, \ldots, H_k \rangle$ означает, что G содержит каждый из графов $H_1, H_2, \ldots, H_k$ в качестве подграфа.

Как обычно, через O_n, K_n, P_n и C_n обозначаются пустой граф, полный граф, простой путь и простой цикл на n вершинах. Полный двудольный граф с p вершинами в одной доле и q вершинами в другой обозначается через $K_{p,q}$. Через $K_4 - e$ и $K_{3,3} - e$ обозначаются результаты удаления ребра из K_4 и $K_{3,3}$ соответственно.

Через $T_{i,j,k}$, $i, j, k \geqslant 0$, обозначается дерево, называемое *триодом*, полученное отождествлением по вершине v концов трёх простых путей $(v = x_0, x_1, \ldots, x_i)$, $(v = y_0, y_1, \ldots, y_j)$, $(v = z_0, z_1, \ldots, z_k)$ (рис. 2).

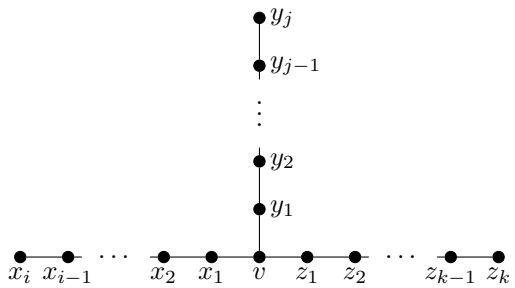


Рис. 2. Граф $T_{i,j,k}$

Далее в доказательствах для вершин графа $T_{i,j,k}$ будут использоваться обозначения, введённые при его определении. Класс всех лесов, каждая компонента связности которых является триодом, обозначается через $\mathcal{T}$. С ним связана

Лемма 3 [7, лемма 3]**.** *Пусть $H' \in \mathcal{T}$ и $\mathcal{X}$ — класс графов, причём для некоторого графа H выполнено $\mathcal{X} \subseteq \mathrm{Free}_s(\{H + H'\})$. Тогда задача* PP *в классе $\mathcal{X}$ полиномиально сводится к той же задаче в классе $\mathcal{X} \cap \mathrm{Free}_s(\{H\})$.*

Монотонное замыкание класса графов $\mathcal{X}$ — множество всех графов, являющихся подграфами графов из $\mathcal{X}$. Оно обозначается через $[\mathcal{X}]_s$. Множество попарно не смежных вершин графа называется *независимым*.

2. NP-полнота задачи PP для некоторых классов субкубических графов

Преобразования, называемые *заменой вершины треугольником* и *заменой вершины* (2,3)-*бикликой*, хорошо известны. Их применяют к вершине x графа, окрестность которой состоит в точности из вершин x_1, x_2, x_3, и определяются следующим образом. В первом из них удаляется x, добавляются вершины x'_1, x'_2, x'_3 и рёбра x'_1x_1, x'_2x_2, x'_3x_3, $x'_1x'_2$, $x'_2x'_3$, $x'_1x'_3$. Во втором удаляется x, добавляются вершины y_1, y_2, z_1, z_2, z_3 и рёбра y_1z_1, y_1z_2, y_1z_3, y_2z_1, y_2z_2, y_2z_3, x_1z_1, x_2z_2, x_3z_3. Нетрудно видеть, что 3-раскраска рёбер исходного графа существует тогда и только тогда, когда она существует для полученного графа.

Через $\mathcal{Z}_k$ обозначим множество кубических графов обхвата не менее чем $k+1$, т. е. не содержащих циклов длины до k включительно. Ясно, что $\mathcal{Z}_1$ и $\mathcal{Z}_2$ совпадают с множеством всех кубических графов. Обозначим через $\mathcal{Z}_k^*$ множество графов, которые получаются из графов класса $\mathcal{Z}_k$ последовательной заменой всех их вершин треугольниками, причём повторная замена вершин во вновь добавленных треугольниках не допускается. Через $\mathcal{Z}_k^{**}$ обозначим множество графов, которые получаются из графов класса $\mathcal{Z}_k$ последовательной заменой всех их вершин (2,3)-бикликами, причём повторная замена вершин во вновь добавленных (2,3)-бикликах не допускается.

Следующее утверждение в части класса $\mathcal{Z}_k^*$ является леммой 9 работы [6], а в части класса $\mathcal{Z}_k^{**}$ может быть доказано по аналогии с ней.

Лемма 4. *Для любого k задача* PP *NP-полна для графов из классов $\mathcal{Z}_k^*$ и $\mathcal{Z}_k^{**}$.*

Доказательство основано на NP-полноте при любом k задачи 3-PP в множестве субкубических графов обхвата не менее k (см. [38]), а также эквивалентности задач 3-PP для графа до замены вершины (2,3)-бикликой и после неё.

Заметим, что каждый 8-рёберный граф из $[\mathcal{Z}_4^*]_s$ не содержит циклов длины, отличной от 3. Заметим также, что каждый 8-рёберный граф из $[\mathcal{Z}_4^{**}]_s$ не содержит циклов длины, отличной от 4. Понятно, что выполнено $[\mathcal{Z}_4^*]_s \subseteq \mathrm{Free}_s(\{B_1^*\})$ и $[\mathcal{Z}_4^{**}]_s \subseteq \mathrm{Free}_s(\{B_{1+}^*\})$.

3. Полиномиальная разрешимость задачи PP для некоторых классов субкубических графов без подграфа B_{1+}^*

Через Belt_k, $k \geqslant 2$, обозначим граф такой, что

$$V(\mathrm{Belt}_k) = \{v_1, v_2, \ldots, v_k, u_1, u_2, \ldots, u_k\},$$

$$E(\mathrm{Belt}_k) = \{v_iu_i \mid 1 \leqslant i \leqslant k\} \cup \{v_iv_{i+1},\, u_iu_{i+1} \mid 1 \leqslant i \leqslant k-1\}.$$

Иными словами, Belt_k получается из C_{2k} добавлением $k-2$ параллельных хорд.

Лемма 5. *Пусть $G = (V, E) \in \mathrm{Free}_s(\{B^*_{1+}\})$ — несжимаемый субкубический граф и Belt_k — максимальный по включению подграф, содержащийся в G в качестве подграфа (не обязательно порождённого), но не содержащийся в каждом из графов K_4–e, $K_{2,3}$, $K_{3,3}$–e, причём каждое из рёбер v_1u_1 и v_ku_k не входит ни в один треугольник графа G. Тогда*

$$\chi'(G) \leqslant 3 \Leftrightarrow \chi'(G \setminus V(\mathrm{Belt}_k)) \leqslant 3,$$

если не выполнено ни одно из следующий условий:

1) $k \in \{3, 4\}$, $v_1v_k \in E \vee u_1u_k \in E$, *найдётся* $w \in V$ *такая, что*

$$wu_1, wu_k \in E \vee wv_1, wv_k \in E;$$

2) $k = 3$, *найдутся вершины* $w_1, w_2, w_3 \in V$ *такие, что*

$$w_1v_1,\ w_1v_3,\ w_2u_1,\ w_2u_3,\ w_1w_3,\ w_2w_3 \in E.$$

В этих случаях выполнено $\chi'(G) = 4$.

Доказательство. Поскольку граф G несжимаем, он не содержит висячих вершин. Так как рассматриваемый подграф Belt_k не содержится ни в одном из графов K_4–$e, K_{3,3} - e$, то $v_1u_2,\ v_2u_1,\ v_1u_3,\ v_3u_1 \notin E$.

Предположим, что $v_1u_k \in E$. Тогда $k \geqslant 4$. Если u_1 или v_k имеет соседа вне $V(\mathrm{Belt}_k)$, то $\langle G; B^*_{1+}\rangle$. Если ни u_1, ни v_k не имеет соседа вне $V(\mathrm{Belt}_k)$, то $\chi'(G) = 3$. Всюду далее будем считать, что $v_1u_k,\ u_1v_k \notin E$.

Предположим, что $v_1v_k \in E$ при $k \geqslant 3$. Если $\deg(u_1) = \deg(u_k) = 2$ или $u_1u_k \in E$, то $\chi'(G) = 3$. Поскольку G несжимаем (следовательно, не содержит мостов), имеем

$$N(u_1) = \{z', v_1, u_2\}, \quad N(u_k) = \{z'', v_k, u_{k-1}\}.$$

Так как $\neg\langle G; B^*_{1+}\rangle$, то $k \leqslant 4$. Если $z' = z''$, то $\chi'(G) = 4$. Если $z' \neq z''$, то $\deg(z') = \deg(z'') = 2$, поскольку $\neg\langle G; B^*_{1+}\rangle$. Нетрудно видеть, что

$$\chi'(G) = 3 \Leftrightarrow \chi'(G \setminus V(\mathrm{Belt}_k)) \leqslant 3,$$

в чём можно убедиться, взяв 3-раскраску рёбер графа $G\setminus V(\mathrm{Belt}_k)$ и окрасив $z'u_1,\ z''u_k,\ v_1v_k,\ v_2u_2,\ \ldots,\ v_{k-1}u_{k-1}$ в один цвет с дальнейшим окрашиванием остальных рёбер Belt_k в два цвета. Всюду далее будем считать, что $v_1v_k,\ u_1u_k \notin E$ при $k \geqslant 3$.

Положим

$$N' = (N(v_1) \cup N(u_1) \cup N(v_k) \cup N(u_k)) \setminus V(\mathrm{Belt}_k).$$

Ввиду несжимаемости G множество N' содержит вершину, смежную с некоторой из $\{v_1, u_1\}$, а также вершину, смежную с некоторой из $\{v_k, u_k\}$.

Предположим, что для каждой пары вершин $\{v_1, u_1\}$ и $\{v_k, u_k\}$ либо хотя бы одна из этих вершин имеет степень 2, либо хотя бы одна из них смежна с вершиной степени 2 из N'. Тогда

$$\chi'(G) = 3 \Leftrightarrow \chi'(G \setminus V(\mathrm{Belt}_k)) \leqslant 3.$$

Действительно, достаточно рассмотреть 3-раскраску рёбер $G \setminus V(\mathrm{Belt}_k)$ и в G покрасить рёбра между $(N(v_1) \cup N(u_1)) \setminus V(\mathrm{Belt}_k)$ и $\{v_1, u_1\}$ в один цвет и покрасить рёбра между $(N(v_k) \cup N(u_k)) \setminus V(\mathrm{Belt}_k)$ и $\{v_k, u_k\}$ тоже в один цвет. Тем самым всюду далее считаем, что $\deg(u_1) = \deg(v_1) = 3$ и обе эти вершины смежны с разными вершинами степени 3 из N'.

Предположим, что

$$N(v_1) = \{z_1, v_2, u_1\}, \quad N(u_1) = \{z_2, v_1, u_2\}.$$

Тогда справедливо каждое из следующих утверждений:

- $\deg(z_1) = \deg(z_2) = 3$;
- $z_2 \notin N(z_1)$ ввиду максимальности подграфа Belt_k;
- при $k = 2$ выполнено $z_2 \notin N(v_2)$, так как иначе рассматриваемый подграф Belt_2 вложен в подграф $K_{3,3} - e$;
- при $k \geqslant 3$ выполнено $z_1 v_k$, $z_2 u_k \notin E$, так как иначе $\langle G; B^*_{1+} \rangle$.

Если $k = 3$, то $z_2 \neq v_3$ и $z_1 \neq u_3$ ввиду отсутствия мостов в G. Рассмотрим ситуацию, когда $z_2 u_3 \notin E$. Поскольку $G \in \mathrm{Free}_s(\{B^*_{1+}\})$, то $\deg(z_2) = 2$, а этот случай был разобран ранее. Аналогично рассматривается ситуация, когда $z_1 v_3 \notin E$.

Рассмотрим случай, когда $N(z_1) = \{v_1, v_3, z_3\}$, $N(z_2) = \{u_1, u_3, z_4\}$. Поскольку $G \in \mathrm{Free}_s(\{B^*_{1+}\})$, то $\deg(z_3) = \deg(z_4) = 2$. Если $z_3 \neq z_4$, то

$$\chi'(G) = 3 \Leftrightarrow \chi'(G \setminus (V(\mathrm{Belt}_3) \cup \{z_1, z_2\})) \leqslant 3,$$

так как достаточно рассмотреть 3-раскраску рёбер $G \setminus (V(\mathrm{Belt}_3) \cup \{z_1, z_2\})$ и в G покрасить $z_1 z_3$, $z_2 z_4$, $v_1 u_1$, $v_2 u_2$, $v_3 u_3$ в один цвет. Если $z_3 = z_4$, то $\chi(G) = 4$.

Если $k = 2$, то ввиду несжимаемости G и условий леммы выполнено

$$\max(\deg(v_2), \deg(u_2)) = 3, \quad z_1 \notin N(u_2), \quad N(z_1) \cap N(v_2) = \{v_1\}.$$

Граф G содержит подграф B^*_{1+}, если $\deg(v_2) = 3$. Если $\deg(v_2) = 2$, то $N(u_2) = \{v_2, u_1, z_5\}$, тогда $z_2 z_5 \notin E$ ввиду максимальности Belt_2. Нетрудно видеть, что $\deg(z_2) = 2$, так как $G \in \mathrm{Free}_s(\{B^*_{1+}\})$. Этот случай разобран ранее. Лемма 5 доказана.

Лемма 6. *Пусть H^* и H^{**} — 8-рёберные графы, принадлежащие $[\mathcal{Z}^*_4]_s$ и $[\mathcal{Z}^{**}_4]_s$ соответственно. Тогда задача* PP *полиномиально разрешима для субкубических графов из класса* $\mathrm{Free}_s(\{B^*_{1+}, H^*, H^{**}\})$.

Доказательство. По лемме 3 будем предполагать, что каждая компонента связности графов H^* и H^{**} не принадлежит $\mathcal{T}$. По теореме 1 считаем, что либо $H^* = B_1^{+*}$, либо H^* не является лесом, а также, что либо $H^{**} \in \{{}^+B_1^*, B_1^{+*}\}$, либо H^{**} не является лесом. По лемме 2 будем рассматривать только несжимаемые графы из класса $\mathrm{Free}_s(\{B_{1+}^*, H^*, H^{**}\})$. Пусть $G = (V, E)$ — произвольный такой граф.

Если $N(x) = \{x_1, x_2, x_3\}$ для некоторой вершины $x \in V$, то x принадлежит либо треугольнику, либо порождённому C_4-циклу графа G. Действительно, предположим, что $\{x_1, x_2, x_3\}$ — независимое множество. Среди его элементов не менее двух (скажем, x_1 и x_2) имеют в G степень 3, а вершина x_3 имеет степень не менее чем 2. Можно считать, что

$$N(x_1) \cap N(x_2) = N(x_2) \cap N(x_3) = N(x_1) \cap N(x_3) = \{x\},$$

иначе x принадлежит порождённому C_4-циклу графа G, но тогда G содержит подграф B_{1+}^*.

В графе G множество $\mathfrak{B}$ всех его максимальных подграфов вида Belt_k, одновременно не содержащихся в подграфах $K_4 - e, K_{2,3}, K_{3,3} - e$, может быть найдено за полиномиальное время. Если это множество не пусто, то рассмотрим произвольный такой подграф Belt_k. По доказательству леммы 5 можно предполагать, что v_1v_k, $u_1u_k \notin E$ и случай (2) не реализуется. По лемме 5 можно считать, что существует вершина $w \in V$, образующая треугольник с v_k и u_k. Пусть wv_1, $wu_1 \notin E$, иначе $\chi'(G) = 4$. Если $k = 2$, то

$$N(w) \cap (N(v_1) \cup N(u_1)) = \{v_2, u_2\}$$

ввиду максимальности подграфа Belt_2. Так как $\neg\langle G; B_{1+}^*\rangle$ и G несжимаем, либо v_1u_1 включено в треугольник, либо v_1 смежна с вершиной степени 3 и $\deg(u_1) = 2$, либо каждая из вершин v_1 и u_1 смежна со своей вершиной степени два, причём эти две вершины не смежны. В последнем случае стянем $G[V(\mathrm{Belt}_k) \cup \{w\}]$ в вершину w' и получим граф G', который будет обыкновенным (так как G несжимаем), причём

$$\chi'(G) = 3 \Leftrightarrow \chi'(G') \leqslant 3.$$

По лемме 1

$$\chi'(G') \leqslant 3 \Leftrightarrow \chi'(G' \setminus \{w'\}) \leqslant 3,$$

причём $G' \setminus \{w'\} \cong G \setminus (V(\mathrm{Belt}_k) \cup \{w\})$. Тем самым далее считаем, что либо v_1u_1 включено в треугольник (v_1, u_1, w''), либо v_1 смежна с вершиной степени 3 и $\deg(u_1) = 2$.

Заметим, что

$$\chi'(G) = 3 \Leftrightarrow \chi'(G \setminus V(\mathrm{Belt}_k)) \leqslant 3,$$

если $\deg(w) = 2$, найдётся вершина $w_1 \notin \{v_k, u_k\}$, $\deg(w_1) = 2$, такая, что $ww_1 \in E$, и v_1u_1 не лежит в треугольнике (v_1, u_1, w''), где w'' имеет соседа степени 3 вне $V(\mathrm{Belt}_k)$, поэтому далее считаем, что $\deg(w_1) = 3$.

В дополнение предположим, что (w_1, w_2, w_3) — треугольник графа G. Очевидно, что

$$\{v_1, u_1\} \cap \{w_2, w_3\} = \varnothing \vee \{v_1, u_1\} = \{w_2, w_3\},$$

причём в последнем случае $\chi'(G) = 3$. Ввиду несжимаемости графа G одна из вершин w_2 и w_3 имеет степень 3. Заметим, что если $v_1w_2 \in E$, то $N(u_1) \subseteq \{v_1, u_2, w_3\}$, так как $\langle G; B^*_{1+}\rangle$, поэтому $|V(G)| = 2k + 4$, иначе G содержит шарнир. Из наших рассуждений следует, что $\langle G; H^*\rangle$, так как G содержит все 8-рёберные графы из $[\mathcal{Z}^*_4]_s$, не являющие лесами, каждая компонента связности которых не принадлежит $\mathcal{T}$. Если w_1 является вершиной степени 2 подграфа $K_{2,3}$, то одна из двух других вершин степени 2 этого подграфа имеет в G степень 3. Нетрудно проверить, что $\langle G; H^{**}\rangle$.

Тем самым w_1 является вершиной степени 2 некоторого элемента $\mathrm{Belt}_{k'} \in \mathfrak{B}$. По лемме 5 можно считать, что существует вершина, образующая треугольник с двумя вершинами $\mathrm{Belt}_{k'}$, но тогда

$$\chi'(G) = 3 \Leftrightarrow \chi'(G \setminus V(\mathrm{Belt}_k)) \leqslant 3.$$

Значит, согласно нашим рассуждениям можно полагать, что каждый порождённый C_4-цикл графа G включён в некоторый подграф $K_{2,3}$, который можно считать порождённым по несжимаемости G. По лемме 3 пусть G содержит $2T_{5,5,5}$ в качестве подграфа. Рассмотрим произвольную из компонент связности этого $2T_{5,5,5}$.

Так как G несжимаем и $\neg\langle G; B^*_{1+}\rangle$, из соображений симметрии можно считать, что либо $x_1y_1 \in E$, либо $x_1y_2 \in E$, $y_1x_2 \in E$, либо найдётся вершина $t \notin V(T_{5,5,5})$ такая, что x_1t, y_1t, $z_1t \in E$. В каждом из этих случаев, стянем в вершину v^* либо $G[\{v, x_1, y_1\}]$, либо $G[\{v, x_1, y_1, x_2, y_2\}]$, либо $G[\{v, x_1, y_1, z_1, t\}]$ и получим граф G^*, для которого

$$\chi'(G) = 3 \Leftrightarrow \chi'(G^*) \leqslant 3.$$

Вместе с тем, если в G^* вершина v^* имеет не более одного соседа степени 3, то по лемме 1

$$\chi'(G^*) = 3 \Leftrightarrow \chi'(G^* \setminus \{v^*\}) \leqslant 3.$$

Поскольку $G^* \setminus \{v^*\}$ — порождённый подграф графа G, предполагаем, что данный случай не реализуется.

Если $x_1y_1 \in E$, то можно считать, что вершины x_2 и y_2 имеют в G степень 3. Так как $\neg\langle G; B^*_{1+}, H^*\rangle$, то x_2 и y_2 содержатся в порождённых подграфах, каждый из которых изоморфен $K_{2,3}$. Тогда $\langle G; H^{**}\rangle$, поэтому можно считать, что первый случай не реализуется ни для одной из компонент $2T_{5,5,5}$. Во втором случае можно считать, что $\deg(x_3) = 3$. Если x_3 содержится в порождённой копии $K_{2,3}$, то $\langle G; H^{**}\rangle$. Если x_3 содержится

в треугольнике, то тоже $\langle G; H^{**}\rangle$ (заметим, что подграф $2C_4$ получается от двух разных компонент $2T_{5,5,5}$). Третий случай аналогичен второму.

Из леммы 3 следует справедливость утверждения данной леммы. Лемма 6 доказана.

4. Полиномиальная разрешимость задачи PP для некоторых классов субкубических графов без подграфов $B_1^* + P_2$, ${}^+B_1^*$, B_1^{+*}

Лемма 7. *Пусть $H \in \{B_1^* + P_2, {}^+B_1^*, B_1^{+*}\}$ и H^* — 8-рёберный граф, принадлежащий $[\mathcal{Z}_4^*]_s$. Тогда задача* PP *полиномиально разрешима для субкубических графов из* $\mathrm{Free}_s(\{H, H^*\})$.

Доказательство. Покажем, что задача PP в классе $\mathrm{Free}_s(\{H, H^*\})$ полиномиально сводится к той же задаче для графов из $\mathrm{Free}_s(\{B_1^*, H, H^*, T_{5,5,5}\})$. С учётом этого из леммы 3 будет следовать справедливость утверждения леммы 7. По теореме 1 и лемме 3 будем предполагать, что $H \in \{{}^+B_1^*, B_1^{+*}\}$, H^* не является лесом при $H^* \neq B_1^{+*}$ и каждая компонента связности H^* не принадлежит $\mathcal{T}$.

По лемме 2 достаточно рассмотреть несжимаемый граф $G = (V, E) \in \mathrm{Free}_s(\{H, H^*\})$. Предположим, что G содержит подграф B_1^* с множеством вершин $\{a_1, a_2, a_3, b_1, b_2, b_3, c_1, c_2\}$ и множеством рёбер $\{a_1a_2, a_2a_3, b_1b_2, b_2b_3, a_2c_1, b_2c_1, c_1c_2\}$. При $H = {}^+B_1^*$ ни одна из вершин a_1, a_3, b_1, b_3 не имеет соседа вне $V(B_1^*)$, поэтому либо $|V(G)| = 8$, либо c_2 — шарнир графа G.

Рассмотрим случай $H = B_1^{+*}$. Нетрудно видеть, что либо $|V(G)| \leqslant 16$, либо c_2 — шарнир графа G, либо имеется путь (u, u_1, u_2), в котором $u \in \{a_1, a_3, b_1, b_3\}$, $u_1, u_2 \notin V(B_1^*)$. Можно считать, что $u = b_1$. Так как $\neg\langle G; B_1^{+*}\rangle$ и G несжимаем, то u_1c_2, u_1a_1, $u_1a_3 \notin E$, и если $\deg(b_1) = 2$, то $\deg(u_1) = 3$, $u_1b_3 \in E$, $\deg(b_3) = 2$, что невозможно ввиду несжимаемости G. По тем же причинам, если $\deg(b_1) = 3$, то

$$b_1a_1 \in E \vee b_1a_3 \in E, \quad \deg(u_1) = 2, \quad \deg(u_2) = 3,$$

что означает $\langle G; B_1^{+*}\rangle$.

Таким образом, можно считать, что G лежит в $\mathrm{Free}_s(\{B_1^*\})$ и содержит подграф $T_{5,5,5}$ по лемме 3. Рассмотрим два варианта: 1) множество $\{x_1, y_1, z_1\}$ независимое, 2) множество $\{x_1, y_1, z_1\}$ не независимое.

1) Ввиду несжимаемости графа G можно считать, что $\deg(x_1) = \deg(y_1) = 3$. Пусть $N(x_1) = \{x_1', v, x_2\}$ и $N(y_1) = \{y_1', v, y_2\}$. Очевидно, что либо $x_1' = y_1'$, либо $x_1' = y_2$, либо $y_1' = x_2$, иначе $\langle G; B_1^*\rangle$.

Предположим, что $x_1' = y_1'$. Тогда поскольку $\neg\langle G; B_1^*\rangle$, для каждой вершины $u \in \{x_2, y_2, z_1\}$ либо $\deg(u) = 2$, либо $ux_1' \in E$, причём если $x_1'z_1 \in E$, то $\deg(z_2) = 2$, а если $x_1'x_2$, $x_1'y_2$, $x_1'z_1 \notin E$, то либо $\deg(x_1') = 2$,

либо x'_1 смежна с вершиной x''_1 степени 2. Нетрудно видеть, что если x'_1x_2, $x'_1y_2 \notin E$, то

$$\chi'(G) = 3 \Leftrightarrow \chi'(G \setminus \{v, x_1, y_1, x'_1\}) \leqslant 3,$$

так как достаточно покрасить y_1y_2 и z_1v в один цвет, а x_1x_2 и $x'_1x''_1$ (если такая вершина x''_1 существует) — также в один цвет.

Предположим, что $x'_1x_2 \in E$, $x'_1y_2 \notin E$ (случай $x'_1x_2 \notin E$, $x'_1y_2 \in E$ рассматривается аналогично). Тогда $\deg(y_2) = \deg(z_1) = 2$. Стянем подграф $G[\{v, x_1, y_1, x_2, x'_1\}]$ в вершину u_1 и получим граф G_1, для которого в силу леммы 1 выполнено

$$\chi'(G) = 3 \Leftrightarrow \chi'(G_1) \Leftrightarrow \chi'(G_1 \setminus \{u_1\}) \leqslant 3,$$
$$G_1 \setminus \{u_1\} \cong G \setminus \{v, x_1, y_1, x_2, x'_1\}.$$

Предположим, что $x'_1z_1 \in E$. Тогда $\deg(x_2) = \deg(y_2) = \deg(z_2) = 2$. Стянем $G[\{v, x_1, y_1, z_1, x'_1\}]$ в вершину u_2 и получим граф G_2, для которого по лемме 1 выполнено

$$\chi'(G) = 3 \Leftrightarrow \chi'(G_2) \leqslant 3 \Leftrightarrow \chi'(G_2 \setminus \{u_2\}) \leqslant 3,$$
$$G_2 \setminus \{u_2\} \cong G \setminus \{v, x_1, y_1, z_1, x'_1\}.$$

Случаи, когда $x'_1 = y_2$ или $y'_1 = x_2$, рассматриваются аналогично. Для этого достаточно рассмотреть триод с центром в вершине x_1 или в вершине y_1 и для него повторить представленное доказательство.

2) Из соображений симметрии можно считать, что $x_1y_1 \in E$. Стянем треугольник (v, x_1, y_1) в вершину u_3 и получим граф G_3. Понятно, что $\chi'(G) = 3 \Leftrightarrow \chi'(G_3) \leqslant 3$. Если среди x_2, y_2, z_1 не более одной вершины имеет степень 3, то по лемме 1 выполнено

$$\chi'(G_3) \leqslant 3 \Leftrightarrow \chi'(G_3 \setminus \{u_3\}) \leqslant 3,$$
$$G_3 \setminus \{u_3\} \cong G \setminus \{v, x_1, y_1\}.$$

С учётом этого в силу симметрии можно считать, что $N(x_2) = \{x'_2, x_1, x_3\}$ и $N(y_2) = \{y'_2, y_1, y_3\}$.

Предположим, что $x'_2 = y_2$. Стянем подграф $G[\{v, x_1, y_1, x_2, y_2\}]$ в вершину u_4 и получим граф G_4 такой, что $\chi'(G) = 3 \Leftrightarrow \chi'(G_4) \leqslant 3$. Можно считать, что $\deg(x_3) = 3$ или $\deg(y_3) = 3$, иначе по лемме 1

$$\chi'(G_4) \leqslant 3 \Leftrightarrow \chi'(G_4 \setminus \{u_4\}) \leqslant 3,$$
$$G_4 \setminus \{u_4\} \cong G \setminus \{v, x_1, y_1, x_2, y_2\},$$

поэтому $\langle G; B_1^* \rangle$, и далее предполагаем, что $x'_2 \neq y_2$.

Если $x'_2x_3 \notin E$, то ввиду несжимаемости G либо $\deg(x_3) = 3$, либо $\deg(x'_2) = 3$, поэтому $\langle G; B_1^* \rangle$. Если x'_2x_3, $x'_2y_2 \in E$, то $\langle G; B_1^* \rangle$. Если $x'_2x_3 \in E$ и $x'_2y_2 \notin E$, то G содержит все 8-рёберные графы из $[\mathcal{Z}_4^*]_s$,

не являющиеся лесами, каждая компонента связности которых не принадлежит $\mathcal{T}$, поэтому $\langle G; H^* \rangle$.

Из леммы 3 следует справедливость утверждения данной леммы. Лемма 7 доказана.

5. Основной результат работы

Основным результатом данной работы является

Теорема 2. *Пусть $\mathcal{Y}$ — множество графов, каждый из которых имеет не более 8 рёбер. Тогда задача* PP *полиномиально разрешима для графов из $\mathcal{X} = \mathrm{Free}_s(\mathcal{Y})$, если*

1) *либо $\mathcal{Y}$ содержит субкубический лес, не принадлежащий множеству*

$$[\{B_1^* + P_2 + O_n,\ {}^+B_1^* + O_n,\ B_{1+}^* + O_n \mid n \geqslant 0\}]_s,$$

2) *либо $\mathcal{Y}$ одновременно содержит граф из $[\mathcal{Z}_4^*]_s$ и граф из*

$$[\{B_1^* + P_2 + O_n,\ {}^+B_1^* + O_n \mid n \geqslant 0\}]_s,$$

3) *либо $\mathcal{Y}$ одновременно содержит граф из $[\{B_{1+}^* + O_n \mid n \geqslant 0\}]_s$ и графы из $[\mathcal{Z}_4^*]_s$ и $[\mathcal{Z}_4^{**}]_s$.*

Во всех остальных случаях она NP*-полна для графов из $\mathcal{X}$.*

Доказательство. Напомним, что задача PP NP-полна в классе $\mathcal{Z}_k$ при любом k (см. [38]). Следовательно, можно считать, что $\mathcal{Z}_k \nsubseteq \mathcal{X}$ для любого k. Заметим, что $\mathcal{Y}$ конечно (с точностью до добавления изолированных вершин) и для любого графа G^*, не являющегося субкубическим лесом, существует такое k^* (которое можно положить равным обхвату графа G^*), что $\mathcal{Z}_{k^*+1} \subseteq \mathrm{Free}_s(\{G^*\})$. Следовательно, $\mathcal{Y}$ содержит субкубический лес.

По лемме 3 можно считать, что каждая компонента связности каждого графа из $\mathcal{Y}$ не принадлежит $\mathcal{T}$. Пусть $F \in \mathcal{Y}$ — субкубический лес. Заметим, что $[\{B_1^{+*} + O_n \mid n \geqslant 0\}]_s \subseteq [\mathcal{Z}_4^*]_s$. По теореме 1 и лемме 7 задача PP полиномиально разрешима в классе $\mathrm{Free}_s(\{F\})$, если F не принадлежит множеству

$$[\{B_1^* + P_2 + O_n,\ {}^+B_1^* + O_n,\ B_{1+}^* + O_n \mid n \geqslant 0\}]_s.$$

Если F является графом из

$$[\{B_1^* + P_2 + O_n,\ {}^+B_1^* + O_n \mid n \geqslant 0\}]_s$$

и $\mathcal{Y} \cap [\mathcal{Z}_4^*]_s = \varnothing$, то $\mathcal{Z}_4^* \subseteq \mathcal{X}$. Задача PP NP-полна для графов из $\mathcal{X}$ по лемме 4. Если же $\mathcal{Y} \cap [\mathcal{Z}_4^*]_s \neq \varnothing$, то задача PP полиномиально разрешима для графов из $\mathcal{X}$ по лемме 7.

Предположим, что $F \in [\{B_{1+}^* + O_n \mid n \geqslant 0\}]_s$. Если

$$\mathcal{Y} \cap [\mathcal{Z}_4^*]_s = \varnothing \vee \mathcal{Y} \cap [\mathcal{Z}_4^{**}]_s = \varnothing,$$

то $\mathcal{Z}_4^* \subseteq \mathcal{X} \vee \mathcal{Z}_4^{**} \subseteq \mathcal{X}$ и задача PP NP-полна для графов из $\mathcal{X}$ по лемме 4. Если же $\mathcal{Y} \cap [\mathcal{Z}_4^*]_s \neq \varnothing$ и $\mathcal{Y} \cap [\mathcal{Z}_4^{**}]_s \neq \varnothing$, то задача PP полиномиально разрешима для графов из $\mathcal{X}$ по лемме 6. Теорема 2 доказана.

Заключение

В настоящей работе была получена полная классификация вычислительной сложности задачи о рёберной раскраске для всех классов графов, определяемых множествами запрещённых подграфов, в каждом из которых не более 8 рёбер. А именно, для каждого такого множества запретов установлено, что для определяемого им класса графов задача о рёберной раскраске либо полиномиально разрешима, либо NP-полна. Эта классификация завершает развитие результатов из работы [7].

Следующий естественный шаг — рассмотрение 9-рёберных запретов. Получение полной сложностной дихотомии для них и задачи о рёберной раскраске является интересной целью для будущих исследований. По-видимому, эта задача существенно более сложна, чем для 8-рёберного случая.

ЛИТЕРАТУРА

1. **Holyer I.** The NP-completeness of edge-coloring // SIAM J. Computing. 1981. V. 10, No. 4. P. 718–720.
2. **Визинг В. Г.** Об оценке хроматического класса p-графа // Дискретный анализ. Т. 3. Новосибирск: Ин-т математики СО АН СССР, 1964. С. 25–30.
3. **Galby E., Lima P. T., Paulusma D., Ries B.** Classifying k-edge colouring for H-free graphs // Inf. Process. Lett. 2019. V. 146. P. 39–43.
4. **Malyshev D. S.** The complexity of the edge 3-colorability problem for graphs without two induced fragments each on at most six vertices // Сиб. электрон. мат. изв. 2014. Т. 11. С. 811–822.
5. **Малышев Д. С.** Классификация сложности задачи о рёберной раскраске для некоторого семейства классов графов // Дискрет. математика. 2016. Т. 28, вып. 2. С. 44–50.
6. **Малышев Д. С.** Полная сложностная дихотомия для запрещённых подграфов с 7 рёбрами в задаче о хроматическом индексе // Дискрет. анализ и исслед. операций. 2020. Т. 27, № 4. С. 104–130.
7. **Малышев Д. С., Дугинов О. И.** О случаях полиномиальной разрешимости задачи о рёберной раскраске, порождаемых запрещёнными 8-рёберными субкубическими лесами // Дискрет. анализ и исслед. операций. 2022. Т. 29, № 2. С. 38–61.
8. **Barnaby M., Paulusma D., Siani S.** Colouring graphs of bounded diameter in the absence of small cycles // Discrete Appl. Math. 2022. V. 314, No. 15. P. 150–161.
9. **Bonomo F., Chudnovsky M., Maceli P., Schaudt O., Stein M., Zhong M.** Three-coloring and list three-coloring of graphs without induced paths on seven vertices // Combinatorica. 2018. V. 38, No. 4. P. 779–801.

10. **Broersma H. J., Golovach P. A., Paulusma D., Song J.** Updating the complexity status of coloring graphs without a fixed induced linear forest // Theor. Comput. Sci. 2012. V. 414, No. 1. P. 9–19.
11. **Cameron K., Huang S., Penev I., Sivaraman V.** The class of (P_7, C_4, C_5)-free graphs: Decomposition, algorithms, and χ-boundedness // J. Graph Theory. 2019. V. 93, No. 4. P. 503–552.
12. **Cameron K., da Silva M., Huang S., Vuskovic K.** Structure and algorithms for (cap, even hole)-free graphs // Discrete Math. 2018. V. 341. P. 463–473.
13. **Dai Y., Foley A., Hoàng C.** On coloring a class of claw-free graphs: To the memory of Frédéric Maffray // Electron. Notes Theor. Comput. Sci. 2019. V. 346. P. 369–377.
14. **Fraser D., Hamela A., Hoàng C., Holmes K., LaMantia T.** Characterizations of $(4K_1, C_4, C_5)$-free graphs // Discrete Appl. Math. 2017. V. 231. P. 166–174.
15. **Golovach P., Johnson M., Paulusma D., Song J.** A survey on the computational complexity of coloring graphs with forbidden subgraphs // J. Graph Theory. 2017. V. 84. P. 331–363.
16. **Golovach P. A., Paulusma D., Song J.** 4-Coloring H-free graphs when H is small // Discrete Appl. Math. 2013. V. 161, No. 1–2. P. 140–150.
17. **Hoàng C., Kamiński M., Lozin V. V., Sawada J., Shu X.** Deciding k-colorability of P_5-free graphs in polynomial time // Algorithmica. 2010. V. 57, No. 1. P. 74–81.
18. **Hoàng C., Lazzarato D.** Polynomial-time algorithms for minimum weighted colorings of $(P_5, \overline{P}_5)$-free graphs and similar graph classes // Discrete Appl. Math. 2015. V. 186. P. 105–111.
19. **Huang S.** Improved complexity results on k-coloring P_t-free graphs // Eur. J. Comb. 2016. V. 51. P. 336–346.
20. **Karthick T., Maffray F., Pastor L.** Polynomial cases for the vertex coloring problem // Algorithmica. 2017. V. 81, No. 3. P. 1053–1074.
21. **Klimošová T., Malík J., Masařík T., Novotná J., Paulusma D., Slívová V.** Colouring (P_r+P_s)-free graphs // Algorithmica. 2020. V. 82, No. 7. P. 1833–1858.
22. **Král' D., Kratochvíl J., Tuza Z., Woeginger G.** Complexity of coloring graphs without forbidden induced subgraphs // Proc. 27th Int. Workshop Graph-Theoretic Concepts in Computer Science (Boltenhagen, Germany, June 14–16, 2001). Heidelberg: Springer, 2001. P. 254–262. (Lect. Notes Comput. Sci.; V. 2204).
23. **Lozin V. V., Malyshev D. S.** Vertex coloring of graphs with few obstructions // Discrete Appl. Math. 2017. V. 216. P. 273–280.
24. **Malyshev D. S.** The coloring problem for classes with two small obstructions // Optim. Lett. 2014. V. 8, No. 8. P. 2261–2270.
25. **Malyshev D. S.** The complexity of the 3-colorability problem in the absence of a pair of small forbidden induced subgraphs // Discrete Math. 2015. V. 338, No. 11. P. 1860–1865.

26. Malyshev D. S. Two cases of polynomial-time solvability for the coloring problem // J. Comb. Optim. 2016. V. 31, No. 2. P. 833–845.
27. Malyshev D. S. The complexity of the vertex 3-colorability problem for some hereditary classes defined by 5-vertex forbidden induced subgraphs // Graphs Comb. 2017. V. 33, No. 4. P. 1009–1022.
28. Malyshev D. S. Polynomial-time approximation algorithms for the coloring problem in some cases // J. Comb. Optim. 2017. V. 33. P. 809–813.
29. Malyshev D. S. The weighted coloring problem for two graph classes characterized by small forbidden induced structures // Discrete Appl. Math. 2018. V. 47. P. 423–432.
30. Malyshev D. S. The vertex colourability problem for {claw, butterfly}-free graphs is polynomial-time solvable // Optim. Lett. 2021. V. 15, No. 2. P. 311–326.
31. Malyshev D. S., Lobanova O. O. Two complexity results for the vertex coloring problem // Discrete Appl. Math. 2017. V. 219. P. 158–166.
32. Malyshev D. S., Pristavchenko O. V. An intractability result for the vertex 3-colourability problem // Optim. Lett. 2022. V. 16. P. 1403–1409.
33. Malyshev D. S., Razvenskaya O. O., Pardalos P. M. The computational complexity of weighted vertex coloring for $\{P_5, K_{2,3}, K^+_{2,3}\}$-free graphs // Optim. Lett. 2021. V. 15, No. 1. P. 137–152.
34. Развенская О. О., Малышев Д. С. Эффективная разрешимость задачи о взвешенной вершинной раскраске для некоторых двух наследственных классов графов // Дискрет. анализ и исслед. операций. 2021. Т. 28, № 1. С. 15–47.
35. Сироткин Д. В., Малышев Д. С. О сложности задачи вершинной 3-раскраски для наследственных классов графов, определённых запретами небольшого размера // Дискрет. анализ и исслед. операций. 2018. Т. 25, № 4. С. 112–130.
36. Spirkl S., Chudnovsky M., Zhong M. Four-coloring P_6-free graphs // Proc. 30th Annu. ACM-SIAM Symp. Discrete Algorithms (San Diego, CA, USA, Jan. 6–9, 2019). Philadelphia, PA: SIAM, 2019. P. 1239–1256.
37. Schrijver A. Combinatorial optimization: Polyhedra and efficiency. Heidelberg: Springer, 2003. 1882 p. (Algorithms Comb.; V. 24).
38. Kamiński M., Lozin V. V. Coloring edges and vertices of graphs without short or long cycles // Contrib. Discrete Math. 2007. V. 2, No. 1. P. 61–66.

Малышев Дмитрий Сергеевич
Дугинов Олег Иванович

Статья поступила
24 ноября 2022 г.
После доработки —
10 июня 2023 г.
Принята к публикации
22 июня 2023 г.

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/
October–December 2023. Vol. 30, No. 4. P. 91–109

UDC 519.17 DOI 10.33048/daio.2023.30.758

A COMPLETE COMPLEXITY DICHOTOMY OF THE EDGE-COLORING PROBLEM FOR ALL SETS OF 8-EDGE FORBIDDEN SUBGRAPHS

D. S. Malyshev[1, 2, a] and *O. I. Duginov*[3, b]

[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
[3] Belarusian State University,
4 Nezavisimost Avenue, 220030 Minsk, Belarus
E-mail: [a] dsmalyshev@rambler.ru, [b] oduginov@gmail.com

Abstract. For a given graph, the edge-coloring problem is to minimize the number of colors sufficient to color all the graph edges so that any adjacent edges receive different colors. For all classes defined by sets of forbidden subgraphs, each with 7 edges, the complexity status of this problem is known. In this paper, we obtain a similar result for all sets of 8-edge prohibitions. Illustr. 2, bibliogr. 38.

Keywords: edge-coloring problem, computational complexity, monotone class.

REFERENCES

1. **I. Holyer,** The NP-completeness of edge-coloring, *SIAM J. Computing* **10** (4), 718–720 (1981).
2. **V. G. Vizing,** On an estimate of the chromatic index of a p-graph, *Discrete Analysis,* Vol. 3 (Inst. Mat. SO AN SSSR, Novosibirsk, 1964), pp. 25–30 [Russian].
3. **E. Galby, P. T. Lima, D. Paulusma,** and **B. Ries,** Classifying k-edge colouring for H-free graphs, *Inf. Process. Lett.* **146**, 39–43 (2019).

Sections 2 and 3 have been obtained with the support of the Russian Foundation for Basic Research and the Belarusian Republic Foundation for Basic Research (Project 20–51–04001 (F21RM-001)). Sections 4 and 5 have been obtained with the support of the Russian Science Foundation (Project 21–11–00194).

English version: Journal of Applied and Industrial Mathematics **17** (4), 791–801 (2023), DOI 10.1134/S1990478923040099.

4. **D. S. Malyshev,** The complexity of the edge 3-colorability problem for graphs without two induced fragments each on at most six vertices, *Sib. Elektron. Mat. Izv.* **11**, 811–822 (2014).
5. **D. S. Malyshev,** Complexity classification of the edge coloring problem for a family of graph classes, *Diskretn. Mat.* **28** (2), 44–50 (2016) [Russian] [*Discrete Math. Appl.* **27** (2), 97–101 (2017)].
6. **D. S. Malyshev,** Complete complexity dichotomy for 7-edge forbidden subgraphs in the edge coloring problem, *Diskretn. Anal. Issled. Oper.* **27** (4), 104–130 (2020) [Russian] [*J. Appl. Ind. Math.* **14** (4), 706–721 (2020)].
7. **D. S. Malyshev** and **O. I. Duginov,** Some cases of polynomial solvability for the edge colorability problem generated by forbidden 8-edge subcubic forests, *Diskretn. Anal. Issled. Oper.* **29** (2), 38–61 (2022) [Russian] [*J. Appl. Ind. Math.* **16** (2), 276–291 (2022)].
8. **M. Barnaby, D. Paulusma,** and **S. Siani,** Colouring graphs of bounded diameter in the absence of small cycles, *Discrete Appl. Math.* **314** (15), 150–161 (2022).
9. **F. Bonomo, M. Chudnovsky, P. Maceli, O. Schaudt, M. Stein,** and **M. Zhong,** Three-coloring and list three-coloring of graphs without induced paths on seven vertices, *Combinatorica* **38** (4), 779–801 (2018).
10. **H. J. Broersma, P. A. Golovach, D. Paulusma,** and **J. Song,** Updating the complexity status of coloring graphs without a fixed induced linear forest, *Theor. Comput. Sci.* **414** (1), 9–19 (2012).
11. **K. Cameron, S. Huang, I. Penev,** and **V. Sivaraman,** The class of (P_7, C_4, C_5)-free graphs: Decomposition, algorithms, and χ-boundedness, *J. Graph Theory* **93** (4), 503–552 (2019).
12. **K. Cameron, M. da Silva, S. Huang,** and **K. Vuskovic,** Structure and algorithms for (cap, even hole)-free graphs, *Discrete Math.* **341**, 463–473 (2018).
13. **Y. Dai, A. Foley,** and **C. Hoàng,** On coloring a class of claw-free graphs: To the memory of Frédéric Maffray, *Electron. Notes Theor. Comput. Sci.* **346**, 369–377 (2019).
14. **D. Fraser, A. Hamela, C. Hoàng, K. Holmes,** and **T. LaMantia,** Characterizations of ($4K_1, C_4, C_5$)-free graphs, *Discrete Appl. Math.* **231**, 166–174 (2017).
15. **P. Golovach, M. Johnson, D. Paulusma,** and **J. Song,** A survey on the computational complexity of coloring graphs with forbidden subgraphs, *J. Graph Theory* **84**, 331–363 (2017).
16. **P. A. Golovach, D. Paulusma,** and **J. Song,** 4-Coloring H-free graphs when H is small, *Discrete Appl. Math.* **161** (1–2), 140–150 (2013).
17. **C. Hoàng, M. Kamiński, V. V. Lozin, J. Sawada,** and **X. Shu,** Deciding k-colorability of P_5-free graphs in polynomial time, *Algorithmica* **57** (1), 74–81 (2010).
18. **C. Hoàng** and **D. Lazzarato,** Polynomial-time algorithms for minimum weighted colorings of ($P_5, \overline{P}_5$)-free graphs and similar graph classes, *Discrete Appl. Math.* **186**, 105–111 (2015).
19. **S. Huang,** Improved complexity results on k-coloring P_t-free graphs, *Eur. J. Comb.* **51**, 336–346 (2016).

20. T. Karthick, F. Maffray, and **L. Pastor,** Polynomial cases for the vertex coloring problem, *Algorithmica* **81** (3), 1053–1074 (2017).
21. T. Klimošová, J. Malík, T. Masařík, J. Novotná, D. Paulusma, and **V. Slívová,** Colouring $(P_r + P_s)$-free graphs, *Algorithmica* **82** (7), 1833–1858 (2020).
22. D. Král', J. Kratochvíl, Z. Tuza, and **G. Woeginger,** Complexity of coloring graphs without forbidden induced subgraphs, in *Proc. 27th Int. Workshop Graph-Theoretic Concepts in Computer Science, Boltenhagen, Germany, June 14–16, 2001* (Springer, Heidelberg, 2001), pp. 254–262 (Lect. Notes Comput. Sci., Vol. 2204).
23. V. V. Lozin and **D. S. Malyshev,** Vertex coloring of graphs with few obstructions, *Discrete Appl. Math.* **216**, 273–280 (2017).
24. D. S. Malyshev, The coloring problem for classes with two small obstructions, *Optim. Lett.* **8** (8), 2261–2270 (2014).
25. D. S. Malyshev, The complexity of the 3-colorability problem in the absence of a pair of small forbidden induced subgraphs, *Discrete Math.* **338** (11), 1860–1865 (2015).
26. D. S. Malyshev, Two cases of polynomial-time solvability for the coloring problem, *J. Comb. Optim.* **31** (2), 833–845 (2016).
27. D. S. Malyshev, The complexity of the vertex 3-colorability problem for some hereditary classes defined by 5-vertex forbidden induced subgraphs, *Graphs Comb.* **33** (4), 1009–1022 (2017).
28. D. S. Malyshev, Polynomial-time approximation algorithms for the coloring problem in some cases, *J. Comb. Optim.* **33**, 809–813 (2017).
29. D. S. Malyshev, The weighted coloring problem for two graph classes characterized by small forbidden induced structures, *Discrete Appl. Math.* **47**, 423–432 (2018).
30. D. S. Malyshev, The vertex colourability problem for {claw, butterfly}-free graphs is polynomial-time solvable, *Optim. Lett.* **15** (2), 311–326 (2021).
31. D. S. Malyshev and **O. O. Lobanova,** Two complexity results for the vertex coloring problem, *Discrete Appl. Math.* **219**, 158–166 (2017).
32. D. S. Malyshev and **O. V. Pristavchenko,** An intractability result for the vertex 3-colourability problem, *Optim. Lett.* **16**, 1403–1409 (2022).
33. D. S. Malyshev, O. O. Razvenskaya, and **P. M. Pardalos,** The computational complexity of weighted vertex coloring for $\{P_5, K_{2,3}, K^+_{2,3}\}$-free graphs, *Optim. Lett.* **15** (1), 137–152 (2021).
34. O. O. Razvenskaya and **D. S. Malyshev,** Efficient solvability of the weighted vertex coloring problem for some two hereditary graph classes, *Diskretn. Anal. Issled. Oper.* **28** (1), 15–47 (2021) [Russian] [*J. Appl. Ind. Math.* **15** (1), 97–117 (2021)].
35. D. V. Sirotkin and **D. S. Malyshev,** On the complexity of the vertex 3-coloring problem for the hereditary graph classes with forbidden subgraphs of small size, *Diskretn. Anal. Issled. Oper.* **25** (4), 112–130 (2018) [Russian] [*J. Appl. Ind. Math.* **12** (4), 759–769 (2018)].

36. S. Spirkl, M. Chudnovsky, and **M. Zhong,** Four-coloring P_6-free graphs, in *Proc. 30th Annu. ACM-SIAM Symp. Discrete Algorithms, San Diego, CA, USA, Jan. 6–9, 2019* (SIAM, Philadelphia, PA, 2019), pp. 1239–1256.

37. A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency* (Springer, Heidelberg, 2003) (Algorithms Comb., Vol. 24).

38. M. Kamiński and **V. V. Lozin,** Coloring edges and vertices of graphs without short or long cycles, *Contrib. Discrete Math.* **2** (1), 61–66 (2007).

Dmitry S. Malyshev
Oleg I. Duginov

Received November 24, 2022
Revised June 10, 2023
Accepted June 22, 2023

СОДЕРЖАНИЕ ТОМА 30

№ 1

№ 2

№ 3

№ 4

DISKRETNYI ANALIZ I ISSLEDOVANIE OPERATSII
/DISCRETE ANALYSIS AND OPERATIONS RESEARCH/

CONTENTS OF VOLUME 30

No. 1

No. 2

No. 3

No. 4

ДИСКРЕТНЫЙ АНАЛИЗ
И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

2023. Том 30, № 4

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 19.01.2024 г.
Формат $70 \times 100\,1/16$. Усл. печ. л. 9,3. Объём 0,93 МБ.

Издательство Института математики,
пр. Академика Коптюга, 4, 630090 Новосибирск, Россия