МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА

Научный журнал

2022 **№ 57**

Зарегистрирован в Федеральной службе по надзору
в сфере связи и массовых коммуникаций

Свидетельство о регистрации ПИ № ФС 77-33762 от 16 октября 2008 г.

Подписной индекс в объединённом каталоге «Пресса России» 38696

УЧРЕДИТЕЛЬ
Томский государственный университет

Адрес редакции и издателя: 634050, г. Томск, пр. Ленина, 36
E-mail: pank@mail.tsu.ru

В журнале публикуются результаты фундаментальных и прикладных научных исследований отечественных и зарубежных ученых, включая студентов и аспирантов, в области дискретной математики и её приложений в криптографии, компьютерной безопасности, кибернетике, информатике, программировании, теории надёжности, интеллектуальных системах.

Периодичность выхода журнала: 4 номера в год.

Редактор *Н. И. Шидловская*
Верстка *И. А. Панкратовой*

Подписано к печати 20.09.2022. Формат $60 \times 84\frac{1}{8}$. Усл. п. л. 14,8. Тираж 300 экз.
Заказ № 5152. Цена свободная. Дата выхода в свет 27.09.2022.

Отпечатано на оборудовании
Издательства Томского государственного университета
634050, г. Томск, пр. Ленина, 36
Тел.: 8(3822)53-15-28, 52-98-49

СОДЕРЖАНИЕ

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

Коврижных М. А., Фомин Д. Б. Об эвристическом алгоритме построения подстановок с заданными криптографическими характеристиками с использованием обобщённой конструкции 5

Косолапов Ю. В., Лелюк Е. А. О структурной стойкости криптосистемы типа Мак-Элиса на сумме тензорных произведений бинарных кодов Рида — Маллера 22

Полищук П. А., Черемушкин А. В. Рандомизированный аналог неоспоримой подписи Чаума — ван-Антверпена..... 40

Трифонов Д. И. Криптографические слабости алгоритмов типа «гиперкуб» 52

Vysotskaya V. V., Chizhov I. V. The security of the code-based signature scheme based on the Stern identification protocol..... 67

МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

Рыбалов А. Н. О генерической сложности ограниченной проблемы кластеризации графов 91

ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ

Нефедов В. Н. Медиана для нечётного числа отношений линейного порядка и её использование в задачах группового выбора 98

СВЕДЕНИЯ ОБ АВТОРАХ 128

CONTENTS

MATHEMATICAL METHODS OF CRYPTOGRAPHY

Kovrizhnykh M. A., Fomin D. B. Heuristic algorithm for obtaining permutations with given cryptographic properties using a generalized construction 5
Kosolapov Yu. V., Lelyuk E. A. On the structural security of a McEliece-type cryptosystem based on the sum of tensor products of binary Reed — Muller codes 22
Polyschuk P. A., Cheremushkin A. V. A randomized analog of Chaum — van Antwerpen undeniable signature 40
Trifonov D. I. Flaws of hypercube-like ciphers 52
Vysotskaya V. V., Chizhov I. V. The security of the code-based signature scheme based on the Stern identification protocol 67

MATHEMATICAL BACKGROUNDS OF INFORMATICS AND PROGRAMMING

Rybalov A. N. The generic complexity of the bounded problem of graphs clustering 91

COMPUTATIONAL METHODS IN DISCRETE MATHEMATICS

Nefedov V. N. Median for an odd number of linear order relations and its use in group choice problems 98

BRIEF INFORMATION ABOUT THE AUTHORS 128

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 519.719.2 DOI 10.17223/20710410/57/1

ОБ ЭВРИСТИЧЕСКОМ АЛГОРИТМЕ ПОСТРОЕНИЯ ПОДСТАНОВОК С ЗАДАННЫМИ КРИПТОГРАФИЧЕСКИМИ ХАРАКТЕРИСТИКАМИ С ИСПОЛЬЗОВАНИЕМ ОБОБЩЁННОЙ КОНСТРУКЦИИ

М. А. Коврижных, Д. Б. Фомин

Национальный исследовательский университет «Высшая школа экономики», г. Москва, Россия

E-mail: makovrizhnykh@gmail.com, dfomin@hse.ru

Исследована возможность построения с помощью обобщённой конструкции подстановок с заданными криптографическими характеристиками, обеспечивающими стойкость алгоритмов шифрования к линейному и разностному методам криптоанализа. Предложен эвристический алгоритм поиска параметров обобщённой конструкции, полученных посредством умножения на транспозиции. Используются идеи генетического алгоритма, спектрально-линейного и спектрально-разностного методов. Изучены вопросы оптимизации вычисления криптографических характеристик на каждой итерации алгоритма. Экспериментальные исследования наиболее интересных с практической точки зрения 8-битовых подстановок показали, что можно построить 6-равномерные подстановки с нелинейностью 108.

Ключевые слова: *векторная булева функция, подстановка, дифференциальная δ-равномерность, нелинейность.*

HEURISTIC ALGORITHM FOR OBTAINING PERMUTATIONS WITH GIVEN CRYPTOGRAPHIC PROPERTIES USING A GENERALIZED CONSTRUCTION

M. A. Kovrizhnykh, D. B. Fomin

National Research University Higher School of Economics, Moscow, Russia

In this paper, we study a generalized construction of $(2m, 2m)$-functions using monomial and arbitrary m-bit permutations as constituent elements. We investigate the possibility of constructing bijective vectorial Boolean functions (permutations) with specified cryptographic properties that ensure the resistance of encryption algorithms to linear and differential methods of cryptographic analysis. We propose a heuristic algorithm for obtaining permutations with the given nonlinearity and differential uniformity based on the generalized construction. For this purpose, we look for auxiliary permutations of a lower dimension using the ideas of the genetic algorithm, spectral-linear, and spectral-difference methods. In the case of $m = 4$, the proposed algorithm consists of iterative multiplication of the initial randomly generated 4-bit

permutations by transposition, selecting the best ones in nonlinearity, the differential uniformity, and the corresponding values in the linear and differential spectra among the obtained 8-bit permutations. We show how to optimize the calculation of cryptographic properties at each iteration of the algorithm. Experimental studies of the most interesting, from a practical point of view, 8-bit permutations have shown that it is possible to construct 6-uniform permutations with nonlinearity 108.

Keywords: *vectorial Boolean function, permutation, differential uniformity, nonlinearity.*

Введение

Конструирование алгоритмов шифрования, стойких к известным методам анализа, представляет собой один из главных аспектов защиты информации. Среди наиболее исследуемых методов криптографического анализа выделим линейный и разностный методы.

Разностный криптоанализ был представлен научной общественности в 1990 г. Э. Бихамом и А. Шамиром [1, 2]. Этот метод использует наличие высоковероятного разностного соотношения на определённых раундах шифрования, вероятность выполнения которого не зависит от ключа. Под разностью будем понимать результат побитового сложения n-мерных векторов. Знание наиболее вероятных разностных соотношений даёт возможность использовать материал для восстановления битов раундового ключа.

Линейный криптоанализ впервые описан в 1993 г. М. Мацуи [3]. Метод применим при наличии высоковероятного линейного статистического аналога для определённых раундов шифрования. Аналогично разностному методу, некоторые раунды шифрования можно заменить их линейным аналогом, который, независимо от ключа, выполняется с вероятностью p для случайно заданного открытого текста и соответствующего ему шифртекста выделенной части шифра. Наибольшее отклонение p от $1/2$ определяет эффективность найденного линейного выражения. С использованием построенного эффективного линейного статистического аналога производится отбраковывание ложных ключей.

Векторные булевы функции — одни из основных примитивов современных симметричных шифров, обеспечивающих свойство перемешивания [4], — должны иметь криптографические характеристики, гарантирующие неосуществимость применения разностного и линейного методов криптографического анализа. Так, векторные булевы функции с высокой нелинейностью позволяют гарантировать стойкость к линейному анализу, поскольку их не удастся эффективно заменить линейным аналогом той же размерности. Для конструирования криптографических алгоритмов, стойких к разностному анализу, используют функции с минимально возможным показателем дифференциальной δ-равномерности.

Дополнительно, векторные булевы функции, используемые в блочных шифрах, как правило, должны быть биективными для обеспечения однозначного расшифрования, т. е. подстановками. В современных криптографических алгоритмах ГОСТ 34.12-2018 («Кузнечик»), AES, ARIA, Khazad и других 8-битовые подстановки являются единственными нелинейными элементами конструкции. Полный перебор $(8, 8)$-функций в настоящее время нереализуем. Таким образом, получение подстановок размерности $n \geqslant 8$ бит с заданными криптографическими свойствами, лучшими, чем у псевдослучайно сгенерированных, и допускающих эффективную аппаратную реализацию,

является сложной и актуальной задачей, что подтверждается большим количеством новейших научных публикаций и докладов на всероссийских и международных конференциях, посвященных данной тематике [5 – 15].

Известные подходы к построению подстановок могут быть разделены на явные алгебраические методы, псевдослучайное генерирование и эвристические подходы (см., например, обзор в [5]). Перспективной представляется идея комбинации этих подходов, в частности использование функциональных схем для получения подстановок с помощью функций меньшей размерности (см. обзор в [11]). При этом в таких схемах обычно есть некоторые параметры, подходящим выбором которых можно улучшить криптографические характеристики конструируемых подстановок.

Так, в работе [6] описана конструкция, содержащая инверсию в поле $\mathbb{F}_{2^4}$ и две произвольные подстановки пространства V_4, для построения 8-битовых подстановок со значениями нелинейности до 108, показателя дифференциальной δ-равномерности 6 или 8, алгебраической степени 7, алгебраической иммунности 3.

В работах [7, 8] представлены схемы, основанные на известных структурах Фейстеля и Лея — Месси, для генерации подстановок размерности $n = 2k$, $k > 2$. Предложенные конструкции используют инверсию в поле $\mathbb{F}_{2^k}$, произвольную k-битовую небиективную функцию (которая не имеет прообраза для 0) и любую k-битовую подстановку. Комбинируя эти компоненты с умножением в конечном поле, автор представил новые 8-битовые подстановки без фиксированных точек, обладающие таким же сочетанием криптографических свойств, как и в [6]. В [8] показано, что такой подход может быть использован для построения инволюций и ортоморфизмов с хорошими криптографическими характеристиками. Однако в этих работах не приведено теоретического обоснования выбора предложенных конструкций.

В работе [10] предложены новые классы 8-битовых подстановок, основанные на конструкции типа «бабочка». Показано, что существует не менее 36 новых конструкций подстановок, которые имеют нелинейность 108, показатель дифференциальной δ-равномерности 6, минимальную степень 7 и значение алгебраической иммунности 3 и могут быть эффективно реализованы как программно, так и аппаратно.

Работы [9, 11, 12] распространяют способы построения подстановок из [10] на случай произвольного векторного пространства V_{2m} и теоретически обосновывают полученные в них экспериментальные результаты. В качестве функциональной схемы используется TU-представление, описанное в [16, 17]. Доказаны необходимые, а в некоторых случаях и достаточные условия для того, чтобы результирующая подстановка обладала заданными значениями нелинейности, алгебраической степени и показателя дифференциальной δ-равномерности. Описана новая обобщённая конструкция векторных функций, использующая мономиальные подстановки в качестве основных составляющих элементов. В случае $m = 4$ экспериментально найдены 768 наборов параметров обобщённой конструкции, с использованием которых при правильном выборе вспомогательных 4-битовых подстановок получены 8-битовые 6-равномерные подстановки, имеющие нелинейность 108 и минимальную степень 7.

Остановимся кратко на методах, относящихся к эвристическому подходу и основанных на итеративном улучшении характеристик подстановок.

В [18] предложен усовершенствованный метод градиентного спуска для построения векторных булевых функций с хорошим сочетанием криптографических свойств. Так, 8-битовые подстановки, полученные этим методом, имеют нелинейность 104, показатель дифференциальной δ-равномерности 8, минимальную степень 7.

Используя обратный генетический алгоритм для генерации подстановок размерности от 8×8 до 16×16, авторы работы [15] для случая 8-битовых подстановок достигли значений нелинейности от 106 до 112 при показателе дифференциальной δ-равномерности 6.

В работе [5] представлены методы итеративного улучшения криптографических характеристик подстановки посредством её умножения на транспозиции — спектрально-линейный и спектрально-разностный методы, основанные на использовании линейного и разностного спектров. В результате применения этих методов удалось получить большое количество 8-битовых подстановок со следующими криптографическими свойствами: нелинейность 104, показатель дифференциальной δ-равномерности 6, обобщённая минимальная степень 7.

Целью настоящей работы является исследование возможности построения подстановок с заданными криптографическими характеристиками — дифференциальной δ-равномерностью и нелинейностью — посредством умножения на транспозиции вспомогательных подстановок в обобщённой конструкции.

В п. 1 приводятся основные определения и обозначения, используемые в работе, в п. 2 рассмотрена обобщённая конструкция $(2m, 2m)$-функции. В п. 3 идеи генетического алгоритма, спектрально-линейного и спектрально-разностного методов применены для выбора подстановок меньшей размерности в обобщённой конструкции. Предложен алгоритм, в результате реализации которого построены 6-равномерные 8-битовые подстановки с нелинейностью 108 и минимальной степенью 7. Обоснованы утверждения, в которых исследуются изменения в DDT и LAT подстановок, задаваемых обобщённой конструкцией, при умножении вспомогательных подстановок на транспозиции.

1. Основные определения и обозначения

Обозначим через V_n, $n \in \mathbb{N}$, n-мерное векторное пространство над полем из двух элементов $\mathbb{F}_2$, $V_n^\times = V_n \setminus \{0\}$. Конечное поле из 2^n элементов обозначим через $\mathbb{F}_{2^n}$. Здесь $\mathbb{F}_{2^n} = \mathbb{F}_2[x]/g(x)$, где $g(x)$ — некоторый неприводимый многочлен степени n над полем $\mathbb{F}_2$; $\mathbb{Z}/2^n$ — кольцо целых чисел по модулю 2^n. Существуют взаимно-однозначное отображение $\mathbb{Z}/2^n \to V_n$, ставящее в соответствие элементу кольца $\mathbb{Z}/2^n$ его двоичное представление, и взаимно-однозначное отображение $V_n \to \mathbb{F}_{2^n}$, ставящее в соответствие двоичной строке элемент поля $\mathbb{F}_{2^n}$, определённые следующим образом:

$$a_{n-1}2^{n-1} + \ldots + a_0 \;\leftrightarrow\; (a_{n-1}, \ldots, a_0) \;\leftrightarrow\; a_{n-1}x^{n-1} + \ldots + a_0.$$

Операции сложения и умножения в поле $\mathbb{F}_{2^n}$ будем обозначать знаками «$+$» и «$\cdot$» соответственно.

Пусть $a \in V_n$, $b \in V_m$. *Конкатенацию* двух векторов обозначим через $a\|b \in V_{n+m}$. *Скалярным произведением* двух векторов $a, b \in V_n$ называется элемент поля $\mathbb{F}_2$, вычисляемый по формуле $\langle a, b\rangle = a_{n-1}b_{n-1} + \ldots + a_0b_0$, где сложение и умножение проводятся в поле $\mathbb{F}_2$. Заметим, что прямое произведение векторных пространств $V_m \times V_m$ можно ассоциировать с V_{2m}.

Определение 1. Пусть n и m — натуральные числа. *Векторной булевой (n, m)-функцией* F называется отображение $F : V_n \to V_m$. При $m = 1$ говорят, что F — *булева функция* от n переменных.

Каждая булева функция f от n переменных единственным образом представляется в виде *полинома Жегалкина* — многочлена от n переменных над полем $\mathbb{F}_2$, в котором

каждая переменная входит в каждый моном в степени 0 или 1. При этом под *алгебраической степенью* (она обозначается как $\deg(f)$) булевой функции f понимается максимальная степень монома с коэффициентом, отличным от 0.

Векторная булева (n, m)-функция F однозначно задается набором своих *координатных* булевых функций от n переменных: $F = (f_{m-1}, \ldots, f_0)$; *компонентной* функцией называется любая ненулевая линейная комбинация координатных функций, т. е. $\langle a, F(x)\rangle$, где $a \in V_m^\times$.

Определение 2. *Подстановкой пространства* V_n называется биективная (n, n)-функция.

Симметрическую группу всех подстановок пространства V_n обозначим $S(V_n)$. Одним из способов представления подстановок является таблица из двух строк: первая строка содержит все числа от 0 до $2^n - 1$, а вторая — образы соответствующих элементов первой строки.

Определение 3. *Мономиальные* подстановки поля $\mathbb{F}_{2^m}$ — это подстановки вида x^d, где d — такое положительное целое число, что $(d, 2^m - 1) = 1$.

При этом можно рассматривать только значения $d < 2^m - 1$. В частности, при $m = 4$ мономиальные подстановки получаются при $d \in \{1, 2, 4, 7, 8, 11, 13, 14\}$. При этом линейными мономиальными подстановками поля $\mathbb{F}_{2^4}$ являются x^d при $d \in \{1, 2, 4, 8\}$ [19, с. 72, 74].

Определение 4. *Циклической подстановкой* (*циклом*) называется подстановка, переводящая j_1 в j_2, j_2 — в j_3, …, j_{k-1} — в j_k и j_k — в j_1. Такой цикл кратко записывается в виде $(j_1, j_2, \ldots, j_k)$. *Транспозиция* — это цикл длины 2, т. е. подстановка, оставляющая неподвижными все элементы, за исключением двух, которые меняются местами.

Умножение подстановки F на транспозицию слева $(j_1, j_2) \circ F$ приводит к транспозиции элементов j_1 и j_2 в нижней строке подстановки F [20, с. 51]:

$$(j_1, j_2) \circ \begin{pmatrix} 0 & 1 & i_1 & \ldots & i_2 & \ldots & 2^n - 1 \\ F(0) & F(1) & F(i_1) = j_1 & \ldots & F(i_2) = j_2 & \ldots & F(2^n - 1) \end{pmatrix} =$$
$$= \begin{pmatrix} 0 & 1 & i_1 & \ldots & i_2 & \ldots & 2^n - 1 \\ F(0) & F(1) & j_2 = F(i_2) & \ldots & j_1 = F(i_1) & \ldots & F(2^n - 1) \end{pmatrix}.$$

Умножение подстановки F на транспозицию справа $F \circ (i_1, i_2)$ приводит к транспозиции элементов i_1 и i_2 в верхней строке подстановки F, другими словами, в нижней строке подстановки F меняются местами образы элементов i_1 и i_2.

Определение 5. *Индикаторной функцией* $I_b(x)$ для $b, x \in V_n$ называется функция

$$I_b(x) = \begin{cases} 1, & b = x, \\ 0, & b \neq x. \end{cases}$$

Приведём определения некоторых криптографических характеристик булевых (n, n)-функций.

Определение 6. Говорят, что (n, n)-функция F является *дифференциально* δ_F*-равномерной*, если

$$\delta_F = \max_{a \in V_n^\times, b \in V_n} \delta_F(a, b),$$

где

$$\delta_F(a, b) = |\{x \in V_n : F(x + a) + F(x) = b\}|.$$

Значение δ_F называется *показателем дифференциальной равномерности* функции F.

Использование функций с малым показателем дифференциальной равномерности при синтезе криптографических алгоритмов позволяет повысить стойкость к разностному методу криптоанализа. Наименьшее возможное значение δ_F равно 2, при этом известен только один пример (с точностью до CCZ-эквивалентности) взаимно-однозначной 2-равномерной (n,n)-функции для чётных n — 6-битная подстановка Диллона [21]. Для случая $n=8$ большим значением показателя дифференциальной δ-равномерности будем считать $\delta_F > 8$, поскольку 8-битовую подстановку с $\delta_F = 8$ можно получить псевдослучайным поиском [5, 22, 23].

Определение 7. *Таблицей распределения разностей* (Difference Distribution Table — DDT) (n,n)-функции F называется такая $2^n \times 2^n$ таблица DDT_F, что

$$\mathrm{DDT}_F[a,b] = \delta_F(a,b).$$

Для всех элементов $\delta \in \{0,2,\ldots,2^n\}$ определим множества

$$D_F(\delta) = \{(a,b) \in V_n^\times \times V_n : \delta_F(a,b) = \delta\}.$$

Определение 8. *Разностным спектром* (n,n)-функции F называется множество пар

$$D_F = \{(\delta, |D_F(\delta)|)\}.$$

Определение 9. *Преобразованием Уолша — Адамара* $W_F(a,b)$ (n,n)-функции F называется отображение $W_F : V_n \times V_n \to \mathbb{Z}$, заданное равенством

$$W_F(a,b) = \sum_{x \in V_n} (-1)^{\langle a,x\rangle + \langle b,F(x)\rangle} \quad \text{для любых} \quad a,b \in V_n.$$

Определение 10. *Линейность* ℓ_F (n,n)-функции F определяется следующим образом:

$$\ell_F = \max_{a \in V_n,\, b \in V_n^\times} |W_F(a,b)|.$$

Нелинейностью N_F (n,n)-функции F называется величина, вычисляемая по формуле

$$N_F = 2^{n-1} - \frac{1}{2}\ell_F.$$

Использование функций с большей нелинейностью при синтезе криптографических алгоритмов позволяет повысить стойкость к линейному методу криптографического анализа.

Определение 11. *Таблицей линейных приближений* (Linear Approximation Table — LAT) [24] (n,n)-функции F называется такая $2^n \times 2^n$ таблица LAT_F, что

$$\mathrm{LAT}_F[a,b] = \ell_F(a,b),$$

где

$$\ell_F(a,b) = \big|\{x \in V_n : \langle a,x\rangle = \langle b,F(x)\rangle\}\big| - 2^{n-1} = \frac{1}{2}W_F(a,b).$$

Для всех элементов $\ell \in \{0,2,\ldots,2^{n-1}\}$ определим множества

$$L_F(\ell) = \{(a,b) \in V_n \times V_n^\times : \big|\ell_F(a,b)\big| = \ell\}.$$

Определение 12. *Линейным спектром* (n,n)-функции F называется множество пар

$$L_F = \{(\ell, |L_F(\ell)|)\}.$$

Определение 13. *Минимальной степенью* (n,n)-функции F называется минимальная из алгебраических степеней всех компонентных функций [25, 2.2.1], т. е.

$$\deg_{\min}(F) = \min_{a \in V_n^\times} \deg(\langle a, F(x)\rangle).$$

Подстановки, используемые в блочных шифрах, должны иметь высокую минимальную степень, чтобы обеспечивать стойкость, например, к разностной атаке высоких порядков. Для подстановок $F \in S(V_n)$ максимально возможная минимальная степень равна $n-1$ [25].

2. Обобщённая конструкция $(2m,2m)$-функции

Пусть $(2m,2m)$-функция $F(x_1,x_2) = y_1 \| y_2$, где $x_1, x_2, y_1, y_2 \in V_m$, задаётся следующей *обобщённой* конструкцией, впервые введённой в работе [9]:

$$\begin{aligned} y_1 = F_1(x_1,x_2) &= \begin{cases} x_1^\alpha \cdot x_2^\beta, & x_2 \neq 0, \\ \widehat{\pi}_1(x_1), & x_2 = 0, \end{cases} \\ y_2 = F_2(x_1,x_2) &= \begin{cases} x_1^\gamma \cdot x_2^\delta, & x_1 \neq 0, \\ \widehat{\pi}_2(x_2), & x_1 = 0. \end{cases} \end{aligned} \tag{1}$$

Здесь следует перейти от m-мерных векторов к соответствующим элементам поля $\mathbb{F}_{2^m}$ и выполнить возведение в степень и умножение в поле $\mathbb{F}_{2^m}$. Кроме того, в (1) $\widehat{\pi}_1$, $\widehat{\pi}_2$ — подстановки пространства V_m. Без ограничения общности будем предполагать, что

$$\widehat{\pi}_1(0) = 0, \quad \widehat{\pi}_2(0) = 0. \tag{2}$$

Параметрами функции (1) являются набор показателей степеней $(\alpha,\beta,\gamma,\delta)$ мономиальных подстановок и значения подстановок $\widehat{\pi}_1, \widehat{\pi}_2 \in S(V_m)$.

Отметим, что конструкция (1) основана на структуре типа «бабочка», введённой в [16] и полученной при изучении декомпозиции APN-подстановки Диллона [21], и допускает TU-представление [17].

В работе [14] для $m = 4$ на множестве всех 4096 наборов показателей степеней $(\alpha,\beta,\gamma,\delta)$ мономиальных подстановок введено отношение эквивалентности и получено разбиение этого множества на непересекающиеся классы. Обоснованы утверждения, позволяющие по одному представителю класса эквивалентности отбраковать все функции, определяемые наборами из данного класса, либо по высокому показателю дифференциальной δ-равномерности ($\delta_F \geqslant 14$), либо вследствие того, что они не являются подстановками ни при каких значениях вспомогательных подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$. В результате остались неотбракованными 768 наборов параметров $(\alpha,\beta,\gamma,\delta)$, перспективных для использования в конструкции (1) с целью получения 8-битовых подстановок с достаточно низким показателем дифференциальной δ-равномерности.

3. О подборе подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ в обобщённой конструкции

Далее будем рассматривать обобщённую конструкцию (1) в случае $m = 4$ с одним из 768 наборов параметров $(\alpha,\beta,\gamma,\delta)$, описанных в [9, 12, 14]. Отметим, что вспомогательные подстановки $\widehat{\pi}_1, \widehat{\pi}_2$ в (1) выбираются независимо от параметров $(\alpha,\beta,\gamma,\delta)$.

Дадим эвристический алгоритм поиска таких вспомогательных 4-битовых подстановок, чтобы итоговая 8-битовая подстановка обладала заданными криптографическими характеристиками $N_F = 108$, $\delta_F = 6$; при этом используем идеи генетического алгоритма, спектрально-линейного и спектрально-разностного методов [5].

3.1. О подборе вспомогательных 4-битовых подстановок

Кратко суть предлагаемого алгоритма 1 заключается в последовательном умножении 4-битовых подстановок $\widehat{\pi}_1$ или $\widehat{\pi}_2$ на транспозиции и отборе среди полученных по формулам (1) 8-битовых подстановок лучших по нелинейности, показателю дифференциальной равномерности и соответствующим значениям в линейном и разностном спектрах. Таким образом, текущее поколение из Num_Best пар подстановок $(\widehat{\pi}_1, \widehat{\pi}_2)$ порождает $Num\_Best \cdot Num\_Trans$ новых пар, из них «выживают» только Num_Best лучших. При этом скрещивания не производится, только «случайные мутации» внутри $\widehat{\pi}_1$ (или $\widehat{\pi}_2$).

Алгоритм 1.

Вход: Подстановка $F \in S(V_8)$, построенная по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2), с криптографическими характеристиками $\ell_F > 40$ или $\delta_F > 6$.

Параметры: Num_Trans — количество умножений на транспозиции; Num_Best — количество отбираемых пар $(\widehat{\pi}_1, \widehat{\pi}_2)$ на каждой итерации алгоритма.

1: Сформировать список $Best$ из одной пары подстановок $(\widehat{\pi}_1, \widehat{\pi}_2)$.
2: **Для всех** пар подстановок $(\widehat{\pi}_1, \widehat{\pi}_2)$ из списка $Best$:
3: запомнить пару $(\widehat{\pi}_1, \widehat{\pi}_2)$ как просмотренную;
4: псевдослучайно выбрать номер $t \in \{1, 2\}$.
5: **Для** $i = 1, \ldots, Num\_Trans$
6: псевдослучайно выбрать $x, y \in V_4^{\times}$, $x \neq y$, получить подстановку $\widehat{\pi}_t = \widehat{\pi}_t \circ (x, y)$.
7: **Если** пара $(\widehat{\pi}_1, \widehat{\pi}_2)$ ещё не просмотрена, **то**
8: встроить $\widehat{\pi}_t$ в F;
9: вычислить набор характеристик подстановки (ℓ_F, δ_F, $|L_F(\ell_F/2)|$, $|D_F(\delta_F)|$);
10: добавить пару $(\widehat{\pi}_1, \widehat{\pi}_2)$ в список $Best$.
11: Отобрать (по принципу многоуровневой сортировки по возрастанию) Num_Best лучших (т. е. с меньшими значениями с учётом приоритетов) из всех наборов характеристик подстановок F, порождённых парами $(\widehat{\pi}_1, \widehat{\pi}_2)$ из текущего списка $Best$, считая, что в наборе приоритет убывает от ℓ_F к $|D_F(\delta_F)|$.
12: **Если** в наилучшем наборе значения $\ell_F = 40$ и $\delta_F = 6$, **то**
вывести подстановки $\widehat{\pi}_1, \widehat{\pi}_2$, порождающие подстановку F,
13: **иначе**
14: сформировать новый список $Best$ из Num_Best пар подстановок $(\widehat{\pi}_1, \widehat{\pi}_2)$, соответствующих лучшим наборам, отобранным на шаге 11.
15: **Перейти** к шагу 2.

Выход: Подстановка $F \in S(V_8)$, отличающаяся от исходной только значениями подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$, такая, что

$$\ell_F = 40 \quad \big(N_F = 108\big), \qquad \delta_F = 6. \tag{3}$$

Значения Num_Trans, Num_Best являются входными данными алгоритма 1. Вычислительные эксперименты показали, что при $Num\_Trans = 500$ на первой итерации (размер начальной популяции), $Num\_Trans = 100$ на последующих, и при $Num\_Best = 10$ за приемлемое число итераций получаются 8-битовые подстановки с характеристиками (3) и минимальной степенью 7. Вопрос о возможности получения с использованием конструкции (1) подстановок с $N_F > 108$, $\delta_F \leqslant 6$ требует дополнительного исследования.

3.2. Об изменениях некоторых криптографических характеристик при умножении на транспозицию вспомогательных 4-битовых подстановок

В [26] доказано, что при умножении векторной булевой функции на транспозицию показатель её дифференциальной равномерности и нелинейность изменяются незначительно, в частности обоснованы следующие оценки: для $h = g \circ (x, y)$, таких, что $g, h : V_n \to V_n$, выполняется

$$\delta_h - 4 \leqslant \delta_g \leqslant \delta_h + 4; \tag{4}$$

$$N_h - 2 \leqslant N_g \leqslant N_h + 2. \tag{5}$$

В работе [13] обоснованы подходы к сокращению трудоёмкости вычисления разностного и линейного спектров для подстановки, полученной умножением исходной на одну транспозицию, в частности показано, что изменения в DDT появляются в относительно небольшом числе ячеек, а именно: в каждой строке таблицы, начиная с первой, возможны изменения не более чем в четырёх ячейках. Тем самым предложенный в [13] алгоритм пересчёта разностного спектра и, как следствие, показателя дифференциальной равномерности для $h = g \circ (x, y)$, $g, h \in S(V_n)$ примерно в 2^n раз быстрее по сравнению с алгоритмом его вычисления для произвольной подстановки. Трудоёмкость алгоритма вычисления линейности ℓ_h из работы [13] примерно в n раз меньше трудоёмкости её нахождения для произвольной подстановки.

Наиболее затратным этапом алгоритма 1 является вычисление ℓ_F, δ_F, линейного и разностного спектров. С целью оптимизации этих вычислений применим теорию из [13] для определения ячеек в DDT и LAT, в которых возникают изменения значений при умножении на транспозицию только 4-битовой подстановки $\widehat{\pi}_1$ или $\widehat{\pi}_2$ в обобщённой конструкции (1).

Утверждение 1. Пусть 8-битовая подстановка $G(z) = G(x_1, x_2) = y_1 \| y_2$, $z = x_1 \| x_2$, построена по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2), а подстановка H получена из G одной транспозицией подстановки $\widehat{\pi}_2$, т. е.

$$H = G \circ (0 \| x,\ 0 \| y), \quad x, y \in V_4^{\times}, \quad x \neq y. \tag{6}$$

Пусть $a \in V_8^{\times}$, $b \in V_8$ — произвольные, при этом $a = a_1 \| a_2$, $b = b_1 \| b_2$, $a_i, b_i \in V_4$, $i = 1, 2$. Тогда выполняются соотношения

$$\delta_H(a, b) - \delta_G(a, b) = \begin{cases} 0, & a_1 = 0,\ b_1 \neq 0, \\ 0, & a_1 = 0,\ b_1 = 0,\ a_2 = x + y, \\ 2(I_1 + I_3 - I_0 - I_2), & a_1 = 0,\ b_1 = 0,\ a_2 \neq x + y, \\ 2(\widetilde{I}_1 + \widetilde{I}_3 - \widetilde{I}_0 - \widetilde{I}_2), & a_1 \neq 0, \end{cases}$$

где

$$\begin{aligned} I_1 &= I_{b_2}(\widehat{\pi}_2(x+a_2)+\widehat{\pi}_2(y)), & I_3 &= I_{b_2}(\widehat{\pi}_2(y+a_2)+\widehat{\pi}_2(x)),\\ I_0 &= I_{b_2}(\widehat{\pi}_2(x+a_2)+\widehat{\pi}_2(x)), & I_2 &= I_{b_2}(\widehat{\pi}_2(y+a_2)+\widehat{\pi}_2(y)),\\ \widetilde{I}_1 &= I_b(g_{1,x}\|(g_{2,x}+\widehat{\pi}_2(y))), & \widetilde{I}_3 &= I_b(g_{1,y}\|(g_{2,y}+\widehat{\pi}_2(x))),\\ \widetilde{I}_0 &= I_b(g_{1,x}\|(g_{2,x}+\widehat{\pi}_2(x))), & \widetilde{I}_2 &= I_b(g_{1,y}\|(g_{2,y}+\widehat{\pi}_2(y))),\\ (g_{1,x}\|g_{2,x}) &= G(a_1,x+a_2), & (g_{1,y}\|g_{2,y}) &= G(a_1,y+a_2). \end{aligned} \tag{7}$$

Доказательство. По условию для 8-битовых подстановок H и G имеем

$$\begin{gathered} G(x_1,x_2)=H(x_1,x_2)\quad \forall x_1,x_2\,\big((x_1\neq 0)\text{ или }(x_1=0,\,x_2\neq x,\,x_2\neq y)\big),\\ G(0,x)=0\|\widehat{\pi}_2(x)=H(0,y),\quad G(0,y)=0\|\widehat{\pi}_2(y)=H(0,x). \end{gathered} \tag{8}$$

Далее, в силу определения 6 выпишем следующую цепочку соотношений:

$$\begin{gathered} \delta_H(a,b)-\delta_G(a,b)=\\ =|\{z\in V_8: H(z+a)+H(z)=b\}|-|\{z\in V_8\,|\,G(z+a)+G(z)=b\}|=\\ =\sum_{z\in V_8} I_b(H(z+a)+H(z))-\sum_{z\in V_8} I_b(G(z+a)+G(z))=\\ =2\sum_{z\in\{(0\|x),\,(0\|y)\}}(I_b(H(z+a)+H(z))-I_b(G(z+a)+G(z))). \end{gathered} \tag{9}$$

I. Пусть $a_1=0$, т. е. $a=0\|a_2$. При $b=b_1\|b_2$ и $b_1\neq 0$ из (9) с учётом вида функции (1) имеем $\delta_H(a,b)-\delta_G(a,b)=0$. Рассмотрим теперь случай $b=0\|b_2$.

I.1. Пусть $a_2=x+y$. В этом случае из равенств (9) непосредственно следует, что $\delta_H(a,b)-\delta_G(a,b)=0$, поскольку в силу (8) имеем $G(0,y)+G(0,x)=H(0,x)+H(0,y)$.

I.2. Пусть $a_2\neq x+y$, тогда соотношения (9) преобразуются к виду

$$\begin{gathered} \delta_H(a,b)-\delta_G(a,b)=2\big(I_{b_2}(\widehat{\pi}_2(x+a_2)+\widehat{\pi}_2(y))+I_{b_2}(\widehat{\pi}_2(y+a_2)+\widehat{\pi}_2(x))-\\ -I_{b_2}(\widehat{\pi}_2(x+a_2)+\widehat{\pi}_2(x))-I_{b_2}(\widehat{\pi}_2(y+a_2)+\widehat{\pi}_2(y))\big)=2(I_1+I_3-I_0-I_2). \end{gathered}$$

II. Пусть $a=a_1\|a_2$ и $a_1\neq 0$. В этом случае справедливы соотношения

$$\begin{aligned} G(a_1,x+a_2)+G(0,x)&=(g_{1,x}\|g_{2,x})+(0\|\widehat{\pi}_2(x))=g_{1,x}\|(g_{2,x}+\widehat{\pi}_2(x)),\\ G(a_1,y+a_2)+G(0,y)&=(g_{1,y}\|g_{2,y})+(0\|\widehat{\pi}_2(y))=g_{1,y}\|(g_{2,y}+\widehat{\pi}_2(y)),\\ H(a_1,x+a_2)+H(0,x)&=(g_{1,x}\|g_{2,x})+(0\|\widehat{\pi}_2(y))=g_{1,x}\|(g_{2,x}+\widehat{\pi}_2(y)),\\ H(a_1,y+a_2)+H(0,y)&=(g_{1,y}\|g_{2,y})+(0\|\widehat{\pi}_2(x))=g_{1,y}\|(g_{2,y}+\widehat{\pi}_2(x)). \end{aligned}$$

Тогда

$$\begin{gathered} \delta_H(a,b)-\delta_G(a,b)=2\Big[I_b(g_{1,x}\|(g_{2,x}+\widehat{\pi}_2(y)))+I_b(g_{1,y}\|(g_{2,y}+\widehat{\pi}_2(x)))-\\ -I_b(g_{1,x}\|(g_{2,x}+\widehat{\pi}_2(x)))-I_b(g_{1,y}\|(g_{2,y}+\widehat{\pi}_2(y)))\Big]=2(\widetilde{I}_1+\widetilde{I}_3-\widetilde{I}_0-\widetilde{I}_2). \end{gathered}$$

Утверждение 1 доказано. ■

Аналогично доказывается соответствующее утверждение относительно транспозиции подстановки $\widehat{\pi}_1$:

Утверждение 2. Пусть 8-битовая подстановка $G = G(x_1, x_2) = y_1 \| y_2$ построена по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2), а подстановка H получена из G одной транспозицией подстановки $\widehat{\pi}_1$, т. е.

$$H = G \circ (x\|0,\, y\|0), \quad x, y \in V_4^{\times}, \quad x \neq y.$$

Пусть $a \in V_8^{\times}$, $b \in V_8$ — произвольные, при этом $a = a_1\|a_2, b = b_1\|b_2$, тогда выполняются соотношения

$$\delta_H(a,b) - \delta_G(a,b) = \begin{cases} 0, & a_2 = 0,\ b_2 \neq 0, \\ 0, & a_2 = 0,\ b_2 = 0,\ a_1 = x + y, \\ 2(J_1 + J_3 - J_0 - J_2), & a_2 = 0,\ b_2 = 0,\ a_1 \neq x + y, \\ 2(\widetilde{J}_1 + \widetilde{J}_3 - \widetilde{J}_0 - \widetilde{J}_2), & a_2 \neq 0, \end{cases}$$

где

$$\begin{aligned} J_1 &= I_{b_1}(\widehat{\pi}_1(x + a_1) + \widehat{\pi}_1(y)), & J_3 &= I_{b_1}(\widehat{\pi}_1(y + a_1) + \widehat{\pi}_1(x)), \\ J_0 &= I_{b_1}(\widehat{\pi}_1(x + a_1) + \widehat{\pi}_1(x)), & J_2 &= I_{b_1}(\widehat{\pi}_1(y + a_1) + \widehat{\pi}_1(y)), \\ \widetilde{J}_1 &= I_b((\widetilde{g}_{1,x} + \widehat{\pi}_1(y))\|\widetilde{g}_{2,x}), & \widetilde{J}_3 &= I_b((\widetilde{g}_{1,y} + \widehat{\pi}_1(x))\|\widetilde{g}_{2,y}), \\ \widetilde{J}_0 &= I_b((\widetilde{g}_{1,x} + \widehat{\pi}_1(x))\|\widetilde{g}_{2,x}), & \widetilde{J}_2 &= I_b((\widetilde{g}_{1,y} + \widehat{\pi}_1(y))\|\widetilde{g}_{2,y}), \\ (\widetilde{g}_{1,x}\|\widetilde{g}_{2,x}) &= G(x + a_1, a_2), & (\widetilde{g}_{1,y}\|\widetilde{g}_{2,y}) &= G(y + a_1, a_2). \end{aligned} \tag{10}$$

Принимая во внимание определение 6 и утверждения 1 и 2, приходим к справедливости следующего следствия:

Следствие 1. Для подстановок H и G, рассматриваемых в утверждениях 1 и 2, выполняются неравенства (4).

Утверждения 1 и 2 позволяют, с учётом особенности обобщённой конструкции, конкретизировать индексы ячеек таблицы DDT, в которых появляются изменения при одной транспозиции во вспомогательной 4-битовой подстановке ($\widehat{\pi}_1$ или $\widehat{\pi}_2$). Так, можно указать множество ячеек DDT, значения в которых никогда не меняются при последовательном умножении на любые транспозиции только подстановки $\widehat{\pi}_1$ (или только подстановки $\widehat{\pi}_2$). Например, при последовательном умножении $\widehat{\pi}_2$ на транспозиции неизменными остаются ячейки DDT с индексами из прямоугольного диапазона $\{(a_1\|a_2, b_1\|b_2) : a_1 = 0, b_1 \neq 0\}$. Тем самым эту часть DDT можно не хранить в памяти. Утверждения 1 и 2 можно использовать в алгоритме пересчёта разностного спектра для новой подстановки, предложенном в работе [13, Algorithm 2, p. 117], при этом в силу формул (7), (10) операции побитового XOR будут осуществляться для 4-битовых векторов, а не для 8-битовых (см. далее алгоритм 2 для $\widehat{\pi}_2$). Отметим, что утверждения 1 и 2 могут быть сформулированы и доказаны аналогичным образом для обобщённой конструкции (1) при произвольном m. При этом асимптотическая оценка трудоёмкости нахождения дифференциальной δ-равномерности совпадет с приведённой в [13].

Алгоритм 2. Модификация [13, Algorithm 2, p. 117] при умножении на транспозицию подстановки $\widehat{\pi}_2$

Вход: 8-битовая подстановка $G = G(x_1, x_2) = y_1 \| y_2$, построенная по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2); $x, y \in V_4^{\times}$, $x \neq y$; DDT_G; разностный спектр D_G.

Для всех элементов $a_2 = 1, \ldots, 2^4 - 1$, таких, что $a_2 \neq x + y$:
  вычислить элементы $B_0 = \widehat{\pi}_2(x) + \widehat{\pi}_2(x + a_2)$ и $B_2 = \widehat{\pi}_2(y) + \widehat{\pi}_2(y + a_2)$.
  Если $B_0 = B_2$, **то**
    вычислить элемент $B_1 = \widehat{\pi}_2(y) + \widehat{\pi}_2(x + a_2)$.
    Для $i = 0, 1$ изменить значения:
      $\big|D_G(\delta_G(0\|a_2, 0\|B_i))\big| := \big|D_G(\delta_G(0\|a_2, 0\|B_i))\big| - 1$;
      $\delta_G(0\|a_2, 0\|B_i) := \delta_G(0\|a_2, 0\|B_i) + 4(-1)^{i+1}$;
      $\big|D_G\big(\delta_G(0\|a_2, 0\|B_i)\big)\big| := \big|D_G\big(\delta_G(0\|a_2, 0\|B_i)\big)\big| + 1$;
  иначе
    вычислить элементы $B_1 = \widehat{\pi}_2(y) + \widehat{\pi}_2(x + a_2)$ и $B_3 = \widehat{\pi}_2(x) + \widehat{\pi}_2(y + a_2)$.
    Для всех $i = 0, \ldots, 3$ изменить значения:
      $\big|D_G(\delta_G(0\|a_2, 0\|B_i))\big| := \big|D_G(\delta_G(0\|a_2, 0\|B_i))\big| - 1$;
      $\delta_G(0\|a_2, 0\|B_i) := \delta_G(0\|a_2, 0\|B_i) + 2(-1)^{i+1}$;
      $\big|D_G\big(\delta_G(0\|a_2, 0\|B_i)\big)\big| := \big|D_G\big(\delta_G(0\|a_2, 0\|B_i)\big)\big| + 1$.
Для всех элементов $a_1 \in V_4^{\times}$, $a_2 \in V_4$:
  вычислить $G(a_1, x + a_2) = (g_{1,x} \| g_{2,x})$, $G(a_1, y + a_2) = (g_{1,y} \| g_{2,y})$;
  вычислить элементы $B_0 = g_{1,x} \| (g_{2,x} + \widehat{\pi}_2(x))$ и $B_2 = g_{1,y} \| (g_{2,y} + \widehat{\pi}_2(y))$.
  Если $B_0 = B_2$, **то**
    вычислить элемент $B_1 = g_{1,x} \| (g_{2,x} + \widehat{\pi}_2(y))$.
    Для $i = 0, 1$ изменить значения:
      $\big|D_G(\delta_G(a_1\|a_2, B_i))\big| := \big|D_G(\delta_G(a_1\|a_2, B_i))\big| - 1$;
      $\delta_G(a_1\|a_2, B_i) := \delta_G(a_1\|a_2, B_i) + 4(-1)^{i+1}$;
      $\big|D_G\big(\delta_G(a_1\|a_2, B_i)\big)\big| := \big|D_G\big(\delta_G(a_1\|a_2, B_i)\big)\big| + 1$;
  иначе
    вычислить элементы $B_1 = g_{1,x} \| (g_{2,x} + \widehat{\pi}_2(y))$ и $B_3 = g_{1,y} \| (g_{2,y} + \widehat{\pi}_2(x))$.
    Для всех $i = 0, \ldots, 3$ изменить значения:
      $\big|D_G(\delta_G(a_1\|a_2, B_i))\big| := \big|D_G(\delta_G(a_1\|a_2, B_i))\big| - 1$;
      $\delta_G(a_1\|a_2, B_i) := \delta_G(a_1\|a_2, B_i) + 2(-1)^{i+1}$;
      $\big|D_G\big(\delta_G(a_1\|a_2, B_i)\big)\big| := \big|D_G\big(\delta_G(a_1\|a_2, B_i)\big)\big| + 1$.
Алгоритм останавливается после вычисления $H = G \circ (0\|x, 0\|y)$ и $D_H = D_G$.

Выход: Подстановка $H = G \circ (0\|x, 0\|y)$; DDT_H; разностный спектр D_H.

Теперь, учитывая особенности обобщённой конструкции (1) и тот факт, что на транспозицию умножается только одна из 4-битовых подстановок ($\widehat{\pi}_1$ или $\widehat{\pi}_2$), найдём индексы ячеек LAT, в которых происходят изменения в результате умножения на транспозицию. Приведём два прямых следствия из соответствующего предложения [13].

Утверждение 3. Пусть 8-битовая подстановка $G = G(x_1, x_2) = y_1 \| y_2$ построена по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2), а подстановка H получена из G одной транспозицией подстановки $\widehat{\pi}_2$, т. е.

$$H = G \circ (0\|x,\ 0\|y), \quad x, y \in V_4^{\times}, \quad x \neq y.$$

Пусть $a \in V_8$, $b \in V_8^\times$ — произвольные, при этом $a = a_1 \| a_2, b = b_1 \| b_2$, тогда выполняются соотношения

$$\ell_H(a,b) - \ell_G(a,b) = \begin{cases} 0, & \langle a_2,\, x+y\rangle = 0 \text{ или } \langle b_2, \widehat{\pi}_2(x) + \widehat{\pi}_2(y)\rangle = 0, \\ (-1)^{\langle a_2,\, x\rangle + \langle b_2, \widehat{\pi}_2(x)\rangle + 1} \cdot 2 & \text{в противном случае.} \end{cases} \tag{11}$$

Утверждение 4. Пусть 8-битовая подстановка $G = G(x_1, x_2) = y_1 \| y_2$ построена по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2), а подстановка H получена из G одной транспозицией подстановки $\widehat{\pi}_1$, т. е.

$$H = G \circ (x\|0,\, y\|0), \quad x, y \in V_4^\times, \quad x \neq y.$$

Пусть $a \in V_8$, $b \in V_8^\times$ — произвольные, при этом $a = a_1 \| a_2, b = b_1 \| b_2$, тогда выполняются соотношения

$$\ell_H(a,b) - \ell_G(a,b) = \begin{cases} 0, & \langle a_1,\, x+y\rangle = 0 \text{ или } \langle b_1, \widehat{\pi}_1(x) + \widehat{\pi}_1(y)\rangle = 0, \\ (-1)^{\langle a_1,\, x\rangle + \langle b_1, \widehat{\pi}_1(x)\rangle + 1} \cdot 2 & \text{в противном случае.} \end{cases} \tag{12}$$

Следствие 2. В условиях утверждений (3) и (4) значения в следующих ячейках LAT исходной 8-битовой подстановки не изменяются при последовательном умножении в произвольном порядке вспомогательных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ на транспозиции (x, y), $x, y \in V_4^\times$, $x \neq y$:

$$\{(a_1\|a_2, b_1\|b_2) : a_1 = b_2 = 0, \text{ или } a_2 = b_1 = 0, \text{ или } a_1 = a_2 = 0, \text{ или } b_1 = b_2 = 0\}. \tag{13}$$

Доказательство. В силу утверждения 3 при умножении $\widehat{\pi}_2$ на транспозицию не изменяются значения LAT в ячейках с индексами $\{(a_1\|a_2, b_1\|b_2) : a_2 = 0 \text{ или } b_2 = 0\}$. В силу утверждения 4 при умножении $\widehat{\pi}_1$ на транспозицию не изменяются значения LAT в ячейках с индексами $\{(a_1\|a_2, b_1\|b_2) : a_1 = 0 \text{ или } b_1 = 0\}$. В пересечении этих множеств получим указанное в следствии множество индексов. ■

В свою очередь, из определений 10 и 11 и утверждений 3 и 4 непосредственно вытекает

Следствие 3. Для подстановок H и G, рассматриваемых в утверждениях 3 и 4, выполняются неравенства (5).

Утверждения 3 и 4 и следствие 2 можно использовать в алгоритме пересчёта линейного спектра, предложенном в [13, Algorithm 1, p. 114], применяя его к подстановкам, полученным путём умножения на транспозицию вспомогательных подстановок $\widehat{\pi}_1$ или $\widehat{\pi}_2$ в обобщённой конструкции (1). При этом значения ячеек LAT с индексами (13) можно не хранить в памяти, а также в силу формул (11) и (12) операции скалярного произведения выполнять для 4-битовых векторов, а не для 8-битовых (алгоритм 3 для $\widehat{\pi}_2$). Отметим, что утверждения 3 и 4 справедливы для обобщённой конструкции (1) при произвольном m. При этом асимптотическая оценка трудоёмкости нахождения линейности совпадет с приведённой в [13].

Алгоритм 3. Модификация [13, Algorithm 1, p. 114] при умножении на транспозицию подстановки $\widehat{\pi}_2$

Вход: 8-битовая подстановка $G = G(x_1, x_2) = y_1 \| y_2$, построенная по формулам (1) с использованием одного из 768 наборов параметров $(\alpha, \beta, \gamma, \delta)$ [12] и произвольных 4-битовых подстановок $\widehat{\pi}_1$, $\widehat{\pi}_2$ (2); $x, y \in V_4^{\times}$, $x \neq y$; LAT_G; линейный спектр L_G.

1: Вычислить элементы $a_2 = x + y$ и $b_2 = \widehat{\pi}_2(x) + \widehat{\pi}_2(y)$.
2: **Для всех** элементов $i = 1, \ldots, 2^4 - 1$ выполнить:
3: **Если** $\langle a_2, i\rangle > 0$, **то**
4: добавить i в список I_1.
5: **Если** $\langle b_2, i\rangle > 0$, **то**
6: добавить i в список I_2.
7: **Для всех** пар $(a_2, b_2) \in I_1 \times I_2$ выполнить:
8: вычислить $\Delta\ell(a_2, b_2) = (-1)^{\langle a_2, x\rangle + \langle b_2, \widehat{\pi}_2(x)\rangle + 1} \cdot 2$.
9: **Для всех** $a_1 \in V_4$, $b_1 \in V_4$
10: сформировать $a = a_1 || a_2$, $b = b_1 || b_2$;
11: вычислить $|L_G(|\ell_G(a, b)|)| := |L_G(|\ell_G(a, b)|)| - 1$;
12: вычислить значение $\ell_G(a, b) := \ell_G(a, b) + \Delta\ell(a_2, b_2)$;
13: вычислить $|L_G(|\ell_G(a, b)|)| := |L_G(|\ell_G(a, b)|)| + 1$.
14: Алгоритм останавливается после вычисления $H = G \circ (0\|x, 0\|y)$ и $L_H = L_G$.

Выход: Подстановка $H = G \circ (0\|x, 0\|y)$; LAT_H; линейный спектр L_H.

Заключение

Предложен эвристический алгоритм подбора таких вспомогательных 4-битовых подстановок в обобщённой конструкции, полученных посредством умножения на транспозиции, чтобы итоговая 8-битовая подстановка F обладала криптографическими характеристиками $N_F = 108$, $\delta_F = 6$. Экспериментально показана практическая применимость алгоритма, при этом удалось получить подстановки с минимальной степенью 7. Теоретически исследованы вопросы оптимизации вычисления линейного и разностного спектров на каждой итерации алгоритма. Алгоритмы вычисления линейного и разностного спектров из работы [13] применены к подстановкам, задаваемым конструкцией (1), с учётом того, что на транспозицию умножается одна из вспомогательных 4-битовых подстановок. При реализации алгоритмов можно получить выигрыш по памяти за счёт уменьшения числа хранимых ячеек в DDT и LAT. При этом операции побитового XOR и вычисления скалярного произведения осуществляются для 4-битовых векторов, а не для 8-битовых.

ЛИТЕРАТУРА

1. *Biham E. and Shamir A.* Differential cryptanalysis of the full 16-round DES // LNCS. 1993. V. 740. P. 487–496.
2. *Biham E. and Shamir A.* Differential cryptanalysis of DES-like cryptosystems // J. Cryptology. 1991. No. 4. P. 3–72.
3. *Matsui M.* Linear cryptanalysis method for DES cipher // LNCS. 1994. V. 765. P. 386–397.
4. *Shannon C. E.* Communication theory of secrecy systems // Bell Syst. Techn. J. 1949. V. 28. P. 656–715.
5. *Menyachikhin A. V.* Spectral-linear and spectral-differential methods for generating S-boxes having almost optimal cryptographic parameters // Матем. вопр. криптогр. 2017. Т. 8. № 2. С. 97–116.

6. *De la Cruz Jiménez R.A.* Generation of 8-bit S-Boxes having almost optimal cryptographic properties using smaller 4-bit S-Boxes and finite field multiplication // LNCS. 2019. V. 11368. P. 191–206.
7. *De la Cruz Jiménez R.A.* On Some Methods for Constructing Almost Optimal S-Boxes and their Resilience against Side-Channel Attacks. 2018. `https://eprint.iacr.org/2018/618`.
8. *De la Cruz Jiménez R.A.* A method for constructing permutations, involutions and orthomorphisms with strong cryptographic properties // Прикладная дискретная математика. Приложение. 2019. № 12. С. 145–151.
9. *Фомин Д. Б.* О подходах к построению низкоресурсных нелинейных преобразований // Обозрение прикладной и промышленной математики. 2018. Т. 25. Вып. 4. С. 379–381.
10. *Fomin D. B.* New classes of 8-bit permutations based on a butterfly structure // Матем. вопр. криптогр. 2019. Т. 10. № 2. С. 169–180.
11. *Фомин Д. Б.* Построение подстановок пространства V_{2m} с использованием $(2m, m)$-функций // Матем. вопр. криптогр. 2020. Т. 11. № 3. С. 121–138.
12. *Фомин Д. Б.* Об алгебраической степени и дифференциальной равномерности подстановок пространства V_{2m}, построенных с использованием $(2m, m)$-функций // Матем. вопр. криптогр. 2020. Т. 11. № 4. С. 133–149.
13. *Menyachikhin A.* The change in linear and differential characteristics of substitution after the multiplication by transposition // Матем. вопр. криптогр. 2020. Т. 11. № 2. С. 111–123.
14. *Fomin D. and Kovrizhnykh M.* On differential uniformity of permutations derived using a generalized construction // CTCrypt 2021. `https://ctcrypt.ru/files/files/2021/Fomin_Kovrizhnylh.pdf`.
15. *Ivanov G., Nikolov N., and Nikova S.* Reversed genetic algorithms for generation of bijective s-boxes with good cryptographic properties // Cryptogr. Commun. 2016. V. 8. No. 2. P. 247–276.
16. *Biryukov A., Perrin L., and Udovenko A.* Reverse-engineering the s-box of Streebog, Kuznyechik and STRIBOBr1 // LNCS. 2016. V. 9665. P. 372–402.
17. *Canteaut A. and Perrin L.* On CCZ-Equivalence, Extended-Affine Equivalence, and Function Twisting. Cryptology ePrint Archive, Report 2018/713. `https://eprint.iacr.org/2018/713`.
18. *Kazymyrov O. V., Kazymyrova V. N., and Oliynykov R. V.* A method for generation of high-nonlinear S-boxes based on gradient descent // Матем. вопр. криптогр. 2014. Т. 5. № 2. С. 71–78.
19. *Лидл Р., Нидеррайтер Г.* Конечные поля. В 2-х т. М.: Мир, 1988. 430 с.
20. *Кострикин А. И.* Введение в алгебру. Ч. I. Основы алгебры: учебник для вузов. 3-е изд. М.: Физматлит, 2004. 272 с.
21. *Browning K. A., Dillon J. F., McQuistan M. T., and Wolfe A. J.* An APN permutation in dimension six // 9th Int. Conf. Finite Fields Appl. 2009. Contemp. Math. 2010. V. 518. P. 33–42.
22. *Knuth D.* Art of Computer Programming. V. 2: Seminumerical Algorithms, 3rd ed. Addison-Wesley Professional, 1997.
23. *Казимиров А. В.* Методы и средства генерации нелинейных узлов замены для симметричных криптоалгоритмов: дис. ... канд. техн. наук. Харьков, 2013. 190 с.
24. *O'Connor L.* Properties of linear approximation tables // LNCS. 1995. V. 1008. P. 131–136.
25. *Carlet C.* Vectorial Boolean functions for cryptography // Boolean Models and Methods in Mathematics, Computer Science, and Engineering / eds. Y. Crama and P. Hammer. Cambridge: Cambridge University Press, 2010. P. 398–469.

26. *Yu Y., Wang M., and Li Y.* Constructing differential 4-uniform permutations from know ones. Cryptology ePrint Archive. Report 2011/047. 2011. https://eprint.iacr.org/2011/047.

REFERENCES

1. *Biham E. and Shamir A.* Differential cryptanalysis of the full 16-round DES. LNCS, 1993, vol. 740, pp. 487–496.
2. *Biham E. and Shamir A.* Differential cryptanalysis of DES-like cryptosystems. J. Cryptology, 1991, no. 4, pp. 3–72.
3. *Matsui M.* Linear cryptanalysis method for DES cipher. LNCS, 1994, vol. 765, pp. 386–397.
4. *Shannon C. E.* Communication theory of secrecy systems. Bell Syst. Techn. J., 1949, vol. 28, pp. 656–715.
5. *Menyachikhin A. V.* Spectral-linear and spectral-differential methods for generating S-boxes having almost optimal cryptographic parameters. Mat. Vopr. Kriptogr., 2017, vol. 8, iss. 2, pp. 97–116.
6. *De la Cruz Jiménez R.A.* Generation of 8-bit S-Boxes having almost optimal cryptographic properties using smaller 4-bit S-Boxes and finite field multiplication. LNCS, 2019, vol. 11368, pp. 191–206.
7. *De la Cruz Jiménez R.A.* On Some Methods for Constructing Almost Optimal S-Boxes and their Resilience against Side-Channel Attacks. 2018. https://eprint.iacr.org/2018/618.
8. *De la Cruz Jiménez R.A.* A method for constructing permutations, involutions and orthomorphisms with strong cryptographic properties. Prikladnaya Diskretnaya Matematika. Prilozhenie, 2019, no. 12, pp. 145–151.
9. *Fomin D. B.* O podkhodakh k postroeniyu nizkoresursnykh nelineynykh preobrazovaniy [On approaches to constructing low-resource nonlinear transformations]. Obozrenie Prikladnoy i Promyshlennoy Matematiki, 2018, vol. 25, iss. 4, pp. 379–381. (in Russian)
10. *Fomin D. B.* New classes of 8-bit permutations based on a butterfly structure. Mat. Vopr. Kriptogr., 2019, vol. 10, no. 2, pp. 169–180.
11. *Fomin D. B.* Postroenie podstanovok prostranstva V_{2m} s ispol'zovaniem $(2m, m)$-funktsiy [Construction of permutations on the space V_{2m} by means of $(2m, m)$-functions]. Mat. Vopr. Kriptogr., 2020, vol. 11, no. 3, pp. 121–138. (in Russian)
12. *Fomin D. B.* Ob algebraicheskoy stepeni i differentsial'noy ravnomernosti podstanovok prostranstva V_{2m}, postroyennykh s ispol'zovaniyem $(2m, m)$-funktsiy [On the algebraic degree and differential uniformity of permutations on the space V_{2m} constructed via $(2m, m)$-functions]. Mat. Vopr. Kriptogr., 2020, vol. 11, no. 4. pp. 133–149. (in Russian)
13. *Menyachikhin A.* The change in linear and differential characteristics of substitution after the multiplication by transposition. Mat. Vopr. Kriptogr., 2020, vol. 11, no. 2, pp. 111–123.
14. *Fomin D. and Kovrizhnykh M.* On differential uniformity of permutations derived using a generalized construction. CTCrypt 2021. https://ctcrypt.ru/files/files/2021/Fomin_Kovrizhnylh.pdf.
15. *Ivanov G., Nikolov N., and Nikova S.* Reversed genetic algorithms for generation of bijective s-boxes with good cryptographic properties. Cryptogr. Commun., 2016, vol. 8, no. 2, pp. 247–276.
16. *Biryukov A., Perrin L., and Udovenko A.* Reverse-engineering the s-box of Streebog, Kuznyechik and STRIBOBr1. LNCS, 2016, vol. 9665, pp. 372–402.
17. *Canteaut A. and Perrin L.* On CCZ-Equivalence, Extended-Affine Equivalence, and Function Twisting. Cryptology ePrint Archive, Report 2018/713. https://eprint.iacr.org/2018/713.

18. *Kazymyrov O. V., Kazymyrova V. N., and Oliynykov R. V.* A method for generation of high-nonlinear S-boxes based on gradient descent. Mat. Vopr. Kriptogr., 2014, vol. 5, no. 2, pp. 71–78.
19. *Lidl R. and Niederreiter H.* Finite Fields. 2nd ed. Cambridge, Cambridge University Press, 1997. 755 p.
20. *Kostrikin A. I.* Introduction to Algebra. N.Y., Springer Verlag, 1982. 577 p.
21. *Browning K. A., Dillon J. F., McQuistan M. T., and Wolfe A. J.* An APN permutation in dimension six. 9th Int. Conf. Finite Fields Appl. 2009; Contemp. Math., 2010, vol. 518, pp. 33–42.
22. *Knuth D.* Art of Computer Programming. Vol. 2: Seminumerical Algorithms, 3rd ed. Addison-Wesley Professional, 1997.
23. *Kazymyrov O. V.* Metody i sredstva generatsii nelineynykh uzlov zameny dlya simmetrichnykh kriptoalgoritmov [Methods and tools for generating nonlinear replacement nodes for symmetric cryptographic algorithms]. PhD Thesis, Kharkiv, 2013. 190 p. (in Russian)
24. *O'Connor L.* Properties of linear approximation tables. LNCS, 1995. vol. 1008, pp. 131–136.
25. *Carlet C.* Vectorial Boolean functions for cryptography. Boolean Models and Methods in Mathematics, Computer Science, and Engineering, Y. Crama and P. Hammer (eds.). Cambridge: Cambridge University Press, 2010, pp. 398–469.
26. *Yu Y., Wang M., and Li Y.* Constructing differential 4-uniform permutations from know ones. Cryptology ePrint Archive, Report 2011/047, 2011. `https://eprint.iacr.org/2011/047`.

УДК 621.391.7 DOI 10.17223/20710410/57/2

О СТРУКТУРНОЙ СТОЙКОСТИ КРИПТОСИСТЕМЫ ТИПА МАК-ЭЛИСА НА СУММЕ ТЕНЗОРНЫХ ПРОИЗВЕДЕНИЙ БИНАРНЫХ КОДОВ РИДА — МАЛЛЕРА

Ю. В. Косолапов, Е. А. Лелюк

Южный федеральный университет, г. Ростов-на-Дону, Россия

E-mail: yvkosolapov@sfedu.ru, lelukevgeniy@mail.ru

Актуальной задачей криптографии является разработка криптосистем, стойких к атакам с использованием квантовых вычислений. Одной из перспективных схем шифрования считается система Мак-Элиса на кодах Гоппы. Однако эта система обладает рядом недостатков, обусловленных структурой кодов Гоппы, что делает актуальным поиск других кодов для схемы Мак-Элиса. Важными требованиями для этих кодов являются наличие быстрого декодера и обеспечение стойкости соответствующей криптосистемы к известным атакам, в том числе с использованием произведения Шура — Адамара. Многие попытки заменить коды Гоппы не привели к успеху, поскольку соответствующие криптосистемы оказались нестойкими к структурным атакам. В настоящей работе в качестве кода предлагается использовать D-конструкцию (D-код) на бинарных кодах Рида — Маллера. Эта конструкция является суммой специального вида тензорных произведений бинарных кодов Рида — Маллера. Для неё имеется быстрый алгоритм декодирования. С целью анализа стойкости схемы Мак-Элиса на D-кодах построена структурная атака с использованием произведения Шура — Адамара D-кода. Для выбора параметров, обеспечивающих стойкость криптосистемы к построенной атаке, исследуется разложимость степени D-кода в прямую сумму кодов Рида — Маллера и делается вывод о множестве стойких ключей криптосистемы.

Ключевые слова: *криптосистема типа Мак-Элиса, структурная стойкость, бинарные коды Рида — Маллера, сумма тензорных произведений, произведение Шура — Адамара.*

ON THE STRUCTURAL SECURITY OF A McELIECE-TYPE CRYPTOSYSTEM BASED ON THE SUM OF TENSOR PRODUCTS OF BINARY REED — MULLER CODES

Yu. V. Kosolapov, E. A. Lelyuk

Southern Federal University, Rostov-on-Don, Russia

The current task of cryptography is the development of cryptosystems resistant to attacks using quantum computing. One of the promising encryption schemes is the McEliece system based on Goppa codes. However, this system has a number of disadvantages due to the structure of Goppa codes, which makes it relevant to search for other codes for the McEliece scheme. Important requirements for these codes are the presence of a fast decoder and ensuring the resistance of the corresponding cryptosystem to known attacks, including attacks with the Schur — Hadamard product. Many attempts to replace Goppa codes have failed because the corresponding cryptosystems

have proven to be unstable against structural attacks. In this paper, it is proposed to use the D-construction (D-code) on binary Reed — Muller codes in the McEliece cryptosystem. This construction is a sum of a special kind of tensor products of binary Reed — Muller codes. There is a fast decoding algorithm for it. To analyze the security of the McEliece scheme on D-codes, we have constructed a structural attack that uses the Schur — Hadamard product of a D-code. To select the parameters that ensure the resistance of the cryptosystem to the constructed attack, we investigate the decomposition of the degree of the D-code into the direct sum of Reed — Muller codes and conclude about the set of strong keys of the cryptosystem.

Keywords: *McEliece-type cryptosystem, structural security, binary Reed — Muller codes, sum of tensor products, Schur —Hadamard product.*

Введение

В 2016 г. Национальным институтом стандартов и технологий США был объявлен конкурс на разработку криптосистем, стойких в постквантовую эпоху [1]. Одним из основных претендентов является кодовая криптосистема Мак-Элиса на кодах Гоппы, стойкость которой основана на сложности декодирования кода в предположении неотличимости его от случайного [2]. Однако её недостатками являются большой размер ключа (от 260 кбайт до 1,3 Мбайт) и относительно высокая (хотя и полиномиальная от размера входных данных) вычислительная сложность алгоритма декодирования [3]. Для уменьшения размера ключа исследователями предлагалось заменить коды Гоппы другими помехоустойчивыми кодами [4 – 7]. Однако эти системы оказались неустойчивы к атакам на ключ [8 – 13], а именно: для предложенных кодов найдены эффективные криптоаналитические алгоритмы структурных атак, которые по публичному ключу находят подходящий секретный ключ. Позднее были предложены некоторые модификации взломанных криптосистем. В частности, в [6] в качестве помехоустойчивого кода используется случайно выбранный подкод известного кода с эффективным алгоритмом декодирования. Однако слабость такого подхода показана в [10, 14]. В [15] для усиления предложено добавлять к порождающей матрице используемого кода случайные столбцы. Эта конструкция также оказалась нестойкой [16, 17].

Заметим, что имеющийся прогресс в области криптоанализа кодовых криптосистем позволяет предположить, что не исключено появление в будущем эффективных структурных атак и на оригинальную криптосистему на кодах Гоппы. Такое предположение подкрепляется, в частности, результатами работы [18], в которой построена эффективная структурная атака для одного класса подпространственных подкодов кодов Рида — Соломона. При этом известно, что коды Гоппы также являются классом подпространственных подкодов кодов Рида — Соломона, и возможно, что идеи, используемые в [18], могут применяться и в случае кодов Гоппы. Поэтому, несмотря на имеющиеся стойкие схемы, актуальна задача поиска других эффективно декодируемых кодов, обеспечивающих высокую стойкость кодовых криптосистем.

Один из подходов построения эффективно декодируемых кодов — комбинирование известных кодов [19]. Однако стоит отметить, что для некоторых известных кодовых конструкций уже получены результаты успешного криптоанализа систем типа Мак-Элиса, основанных на таких конструкциях. В частности, использование в такой криптосистеме прямой суммы известных кодов, как предложено в [20], не усиливает стойкость системы по сравнению со стойкостью системы Мак-Элиса на кодах-слагаемых. Вычислительная эквивалентность структурных атак для таких систем показана

в [21]. Использование повторения [4] и псевдоповторения (соединения) [22] кодов также не позволяет усилить стойкость криптосистемы, что показано в [23].

Представляется, что потенциально стойкой системой типа Мак-Элиса может являться система на произведении кодов. Такие коды и их обобщение — D-коды [24] — относятся к эффективно декодируемым: для некоторых классов этих кодов имеются быстрые алгоритмы мажоритарного декодирования [25, 26]. При этом анализ криптосистемы типа Мак-Элиса на произведении кодов, проведённый в [25], показал её высокую стойкость к структурным атакам. Заметим, что при анализе в [25] не использовалось произведение Шура — Адамара, преобразующее публичный ключ и часто позволяющее получить дополнительную информацию о ключе по свойствам такого преобразования [27, 10, 9, 28]. Позднее для кодов произведения в [29] исследованы некоторые свойства квадрата Шура — Адамара, которые могут быть использованы для построения структурной атаки на соответствующую кодовую криптосистему.

В настоящей работе исследуется стойкость криптосистемы Мак-Элиса на одном обобщении произведения кодов — D-кодах на основе кодов Рида — Маллера. Проводится анализ стойкости этой системы к атаке на основе произведения Шура — Адамара. Для этого исследуются свойства степеней Шура — Адамара D-кодов на основе кодов Рида — Маллера. Соответствующие результаты приводятся в п. 1. В п. 2 строится структурная атака с использованием произведения Шура — Адамара и на основе результатов п. 1 делается вывод о параметрах D-кодов, позволяющих противодействовать построенной атаке.

1. Свойства произведения Шура — Адамара D-конструкции на основе бинарных кодов Рида — Маллера

1.1. Бинарные коды Рида — Маллера

Пусть $\mathbb{F}_q^n$ — векторное пространство над полем Галуа $\mathbb{F}_q$. Для вектора $\mathbf{x} \in \mathbb{F}_q^n$ множество его ненулевых координат называется носителем вектора $\mathbf{x}$ и обозначается $\mathrm{supp}(\mathbf{x})$. Вес $\mathrm{wt}(\mathbf{x})$ вектора $\mathbf{x}$ определяется как $|\mathrm{supp}(\mathbf{x})|$ (здесь и далее символом $|A|$ обозначается мощность множества A). Линейное подпространство C размерности k пространства $\mathbb{F}_q^n$ называется линейным $[n,k]_q$-кодом [30]; $[n,k]_q$-код C с минимальным кодовым расстоянием $d = \min\limits_{\mathbf{c}\in C, \mathbf{c}\neq \mathbf{0}} \{\mathrm{wt}(\mathbf{c})\}$ называется $[n,k,d]_q$-кодом. Порождающую матрицу кода C обозначим через G_C, а двойственный код к коду C — через $\overline{C}$. Коды C и D длины n и размерности k называются перестановочно эквивалентными, если в симметрической группе перестановок $\mathcal{P}_n$, действующей на множестве $\{1,\dots,n\}$, найдётся перестановка σ, такая,что

$$\sigma(C) = \{(c_{\sigma^{-1}(1)},\dots,c_{\sigma^{-1}(n)}) : (c_1,\dots,c_n) \in C\} = D.$$

Напомним, что код C называется *разложимым*, если он перестановочно эквивалентен прямой сумме двух или более кодов ненулевой длины. Для двух векторов $\mathbf{a} = (a_1,\dots,a_n)$ и $\mathbf{b} = (b_1,\dots,b_n)$ из $\mathbb{F}_q^n$ произведением Шура — Адамара называется вектор $\mathbf{a}\star\mathbf{b} = (a_1b_1,\dots,a_nb_n)$; произведением Шура — Адамара $k\times n$-матрицы $A = (\mathbf{a}_i)$ и $l\times n$-матрицы $B = (\mathbf{b}_j)$ называется матрица $A\star B = (\mathbf{a}_i\star\mathbf{b}_j)$, $i = 1,\dots,k$, $j = 1,\dots,l$. Для кодов C и D из $\mathbb{F}_q^n$ их произведение Шура — Адамара определяется следующим образом:

$$C\star D = \mathcal{L}(\{\mathbf{x}\star\mathbf{y} | \mathbf{x}\in C, \mathbf{y}\in D\}).$$

Здесь и далее через $\mathcal{L}(U)$ обозначается линейная оболочка множества U. Известно [27], что $C \star D = \mathcal{L}(G_C \star G_D)$. Произведение $C \star C$ далее обозначается C^2 и называется квадратом кода C.

Для $(k_1 \times n_1)$-матрицы $A = (a_{i,j})$ и $(k_2 \times n_2)$-матрицы B их тензорное произведение $A \otimes B$ определяется как $(k_1 k_2 \times n_1 n_2)$-матрица вида

$$\begin{pmatrix} a_{1,1}B & \cdots & a_{1,n_1}B \\ \vdots & \ddots & \vdots \\ a_{k_1,1}B & \cdots & a_{k_1,n_1}B \end{pmatrix}. \tag{1}$$

Тензорное произведение $C_1 \otimes C_2$ двух $[n_i, k_i, d_i]_q$-кодов $C_i \subset \mathbb{F}_q^{n_i}$, где $i \in \{1, 2\}$, можно определить как $\mathcal{L}(G_{C_1} \otimes G_{C_2})$. Известно, что $C_1 \otimes C_2$ является $[n_1 n_2, k_1 k_2, d_1 d_2]_q$-кодом [19, п. 6.2.3], а из [31, теорема 5] вытекает, что

$$\overline{C_1 \otimes C_2} = \mathbb{F}_q^{n_1} \otimes \overline{C_2} + \overline{C_1} \otimes \mathbb{F}_q^{n_2}. \tag{2}$$

Действительно, согласно [31, формула (7)], порождающая матрица кода $\overline{C_1 \otimes C_2}$ может быть представлена в виде

$$G_{\overline{C_1 \otimes C_2}} = \begin{pmatrix} G_{\overline{C_1}} \otimes G_{\overline{C_2}} \\ A_1 \otimes G_{\overline{C_2}} \\ G_{\overline{C_1}} \otimes A_2 \end{pmatrix},$$

где A_1, A_2 — такие $k_1 \times n_1$- и $k_2 \times n_2$-матрицы, что

$$\mathcal{L}\begin{pmatrix} G_{\overline{C_1}} \\ A_1 \end{pmatrix} = \mathbb{F}_q^{n_1}, \quad \mathcal{L}\begin{pmatrix} G_{\overline{C_2}} \\ A_2 \end{pmatrix} = \mathbb{F}_q^{n_2}.$$

Следовательно,

$$\begin{aligned} \overline{C_1 \otimes C_2} &= \overline{C_1} \otimes \overline{C_2} + \mathcal{L}(A_1) \otimes \overline{C_2} + \overline{C_1} \otimes \mathcal{L}(A_2) = \\ &= \overline{C_1} \otimes \overline{C_2} + \mathcal{L}(A_1) \otimes \overline{C_2} + \overline{C_1} \otimes \mathcal{L}(A_2) + \overline{C_1} \otimes \overline{C_2} = \\ &= (\overline{C_1} + \mathcal{L}(A_1)) \otimes \overline{C_2} + \overline{C_1} \otimes (\overline{C_2} + \mathcal{L}(A_2)) = \\ &= \mathbb{F}_q^{n_1} \otimes \overline{C_2} + \overline{C_1} \otimes \mathbb{F}_q^{n_2}. \end{aligned}$$

Отметим, что порождающая матрица кода $\mathbb{F}_q^n \otimes C$ имеет блочно-диагональный вид: $G_{\mathbb{F}_q^n \otimes C} = \operatorname{diag}(G_C, \ldots, G_C)$. Поэтому код $\mathbb{F}_q^n \otimes C$ можно представить как прямую (внешнюю) сумму n кодов C:

$$\mathbb{F}_q^n \otimes C = \underbrace{C \oplus \ldots \oplus C}_{n}. \tag{3}$$

Широко известным классом линейных кодов является класс бинарных кодов Рида — Маллера. Для их определения и исследования некоторых их свойств рассмотрим $\mathbb{F}_2[x_1, \ldots, x_m]$ — кольцо полиномов от m переменных над полем $\mathbb{F}_2$. Полиномы из $\mathbb{F}_2[x_1, \ldots, x_m]$ будем записывать в виде

$$f(x_1, \ldots, x_m) = \sum_{\boldsymbol{\alpha}=(\alpha_1,\ldots,\alpha_m) \in \mathbb{F}_2^m} f_{\boldsymbol{\alpha}} \bar{x}^{\boldsymbol{\alpha}},$$

где $\bar{x}^{\boldsymbol{\alpha}} = x_1^{\alpha_1} \ldots x_m^{\alpha_m}$ — моном степени $\operatorname{wt}(\boldsymbol{\alpha})$. Для вектора $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m) \in \mathbb{F}_2^m$ символом $f(\boldsymbol{\alpha})$ будем обозначать значение полинома $f(x_1, \ldots, x_m)$, вычисленное при $x_i = \alpha_i$, $i = 1, \ldots, m$. Степень полинома f определяется как максимальная степень

его ненулевых мономов. Пусть $\mathbb{F}_2^{(r)}[x_1,\ldots,x_m]$ — линейное пространство полиномов из $\mathbb{F}_2[x_1,\ldots,x_m]$ степени не выше r. Определим оператор $E_{(m,r)}:\mathbb{F}_2^{(r)}[x_1,\ldots,x_m]\to\mathbb{F}_2^n$ следующим образом:

$$E_{(m,r)}(f)=(f(\boldsymbol{\alpha}_1),\ldots,f(\boldsymbol{\alpha}_{2^m})),$$

где $\boldsymbol{\alpha}_i,\boldsymbol{\alpha}_j\in\mathbb{F}_2^m$, $\boldsymbol{\alpha}_i\neq\boldsymbol{\alpha}_j$ для $i\neq j$, $n=2^m$. Бинарный код Рида — Маллера $\mathrm{RM}(m,r)$ с параметрами m и r определяется следующим образом:

$$\mathrm{RM}(m,r)=\{E_{(m,r)}(f):f\in\mathbb{F}_2^{(r)}[x_1,\ldots,x_m]\},$$

при этом $\mathrm{RM}(m,r)=\mathrm{RM}(m,m)=\mathbb{F}_2^n$ для $r\geqslant m$. Известно, что $\overline{\mathrm{RM}(m,r)}=\mathrm{RM}(m,m-r-1)$, а в [9] доказано, что

$$\mathrm{RM}(m,r_1)\star\mathrm{RM}(m,r_2)=\mathrm{RM}(m,r_1+r_2). \tag{4}$$

Из (4) и определения кода Рида — Маллера вытекает, что $\mathrm{RM}(m,r_1)\star\mathrm{RM}(m,r_2)=\mathbb{F}_2^n$ при $r_1+r_2\geqslant m$.

Лемма 1. Пусть $\mathrm{RM}(m_i,r_i)$ — бинарный код Рида — Маллера, $i=1,2$. Тогда

$$\mathrm{RM}(m_1,r_1)\otimes\mathrm{RM}(m_2,r_2)\subset\mathrm{RM}(m_1+m_2,r_1+r_2).$$

Доказательство. Рассмотрим тензорное произведение пространств $\mathbb{F}_2^{(r_1)}[x_1,\ldots,x_{m_1}]$ и $\mathbb{F}_2^{(r_2)}[y_1,\ldots,y_{m_2}]$, которое, по определению, имеет вид

$$\mathbb{F}_2^{(r_1)}[x_1,\ldots,x_{m_1}]\otimes\mathbb{F}_2^{(r_2)}[y_1,\ldots,y_{m_2}]=$$
$$=\{f\cdot g:f\in\mathbb{F}_2^{(r_1)}[x_1,\ldots,x_{m_1}],\ g\in\mathbb{F}_2^{(r_2)}[y_1,\ldots,y_{m_2}]\},$$

где $f\cdot g$ — произведение полиномов в кольце $\mathbb{F}_2[x_1,\ldots,x_{m_1},y_1,\ldots,y_{m_2}]$. Очевидно, что

$$\mathbb{F}_2^{(r_1)}[x_1,\ldots,x_{m_1}]\otimes\mathbb{F}_2^{(r_2)}[y_1,\ldots,y_{m_2}]\subset\mathbb{F}_2^{(r_1+r_2)}[x_1,\ldots,x_{m_1},y_1,\ldots,y_{m_2}].$$

Пусть $f\in\mathbb{F}_2^{(r_1)}[x_1,\ldots,x_{m_1}]$, $g\in\mathbb{F}_2^{(r_2)}[y_1,\ldots,y_{m_2}]$. Покажем, что

$$E_{(m_1+m_2,r_1+r_2)}(f\cdot g)=E_{(m_1,r_1)}(f)\otimes E_{(m_2,r_2)}(g).$$

Так как $(f\cdot g)(\boldsymbol{\alpha},\boldsymbol{\beta})=f(\boldsymbol{\alpha})g(\boldsymbol{\beta})$ для $\boldsymbol{\alpha}\in\mathbb{F}_2^{m_1}$, $\boldsymbol{\beta}\in\mathbb{F}_2^{m_2}$, то, с одной стороны,

$$E_{(m_1+m_2,r_1+r_2)}(f\cdot g)=\big((f\cdot g)(\boldsymbol{\alpha}_i,\boldsymbol{\beta}_j)\big)_{i=1,\ldots,2^{m_1},j=1,\ldots,2^{m_2}}=$$
$$=\big(f(\boldsymbol{\alpha}_i)g(\boldsymbol{\beta}_j)\big)_{i=1,\ldots,2^{m_1},j=1,\ldots,2^{m_2}},$$

с другой стороны, по определению тензорного произведения (1), получаем

$$E_{(m_1,r_1)}(f)\otimes E_{(m_2,r_2)}(g)=\big(f(\boldsymbol{\alpha}_i)g(\boldsymbol{\beta}_j)\big)_{i=1,\ldots,2^{m_1},j=1,\ldots,2^{m_2}},$$

что доказывает утверждение. ■

1.2. D-конструкция на кодах Рида — Маллера

Рассмотрим два семейства двоичных кодов Рида — Маллера:

$$\mathcal{S}_1 = \{C_1(0){=}\mathrm{RM}(m_1, m_1), C_1(1){=}\mathrm{RM}(m_1, m_1-1), \ldots, C_1(m_1){=}\mathrm{RM}(m_1, 0), C_1(J_1){=}\{\bar{0}\}\},$$
$$\mathcal{S}_2 = \{C_2(0){=}\mathrm{RM}(m_2, m_2), C_2(1){=}\mathrm{RM}(m_2, m_2-1), \ldots, C_2(m_2){=}\mathrm{RM}(m_2, 0), C_2(J_2){=}\{\bar{0}\}\},$$

где $J_t = m_t + 1$, $t = 1, 2$. Будем считать, что $C_t(i) = C_t(m_t + 1)$ для $i \geqslant m_t + 1$, $C_t(i) = C_t(0)$ для $i \leqslant 0$, $t = 1, 2$. Эти семейства для $t \in \{1, 2\}$ удовлетворяют условиям

$$C_t(0) \supset C_t(1) \supset C_t(2) \supset \cdots \supset C_t(J_t); \tag{5}$$
$$\overline{C_t(0)} \subset \overline{C_t(1)} \subset \overline{C_t(2)} \subset \cdots \subset \overline{C_t(J_t)}. \tag{6}$$

Поэтому для $i_1 \leqslant i_2$, $j_1 \leqslant j_2$ получаем

$$C_1(i_1) \otimes C_2(j_1) \supseteq C_1(i_2) \otimes C_2(j_2), \tag{7}$$
$$\overline{C_1(i_1)} \otimes \overline{C_2(j_1)} \subseteq \overline{C_1(i_2)} \otimes \overline{C_2(j_2)}.$$

Пусть $D_0 = \{(i, j) : i = 0, \ldots, J_1, j = 0, \ldots, J_2\}$. Для любого $D \subseteq D_0$ определим код

$$C(D) = \mathcal{L}\left(\bigcup_{(i,j)\in D} C_1(i) \otimes C_2(j)\right),\ C_1(i) \in \mathcal{S}_1,\ C_2(j) \in \mathcal{S}_2. \tag{8}$$

Далее будем рассматривать только такие множества $D \subset D_0$, которые удовлетворяют следующему условию: если $(i, j) \in D$ и существует пара $(k, s) \in D_0$, такая, что $k \geqslant i$, $s \geqslant j$, то $(k, s) \in D$. Набор таких множеств D обозначим через $F(D_0)$. Если $D \in F(D_0)$, то *D-кодом* называется код $\overline{C(D)}$, двойственный коду $C(D)$ вида (8). Отметим, что семейства $\mathcal{S}_1, \mathcal{S}_2$ удовлетворяют условиям для построения мажоритарного декодера [26, пример 3], поэтому рассматриваемые D-коды могут быть эффективно декодированы.

Рассмотрим определённые в [24] подмножества D^*, D_b, D_b^* множества D_0, которые строятся по $D \in F(D_0)$:

$$D^* = D_1^* \cup D_2^* \cup D_3^*, \tag{9}$$

где

$$D_1^* = \{(i, j) : (i-1, j-1) \in D_0, (i-1, j-1) \notin D\};$$
$$D_2^* = \begin{cases} \{(i, 0) : i = 0, \ldots, \min\limits_{(v,0)\in D}(v)\}, & \text{если } \min\limits_{(i,j)\in D}(j) = 0, \\ \{(i, 0) : i = 0, \ldots, J_1\} & \text{иначе;} \end{cases}$$
$$D_3^* = \begin{cases} \{(0, j) : j = 0, \ldots, \min\limits_{(0,v)\in D}(v)\}, & \text{если } \min\limits_{(i,j)\in D}(i) = 0, \\ \{(0, j) : j = 0, \ldots, J_2\} & \text{иначе;} \end{cases}$$
$$D_b = \{(i, j) : (i, j) \in D \wedge (i-1, j) \notin D \wedge (i, j-1) \notin D\};$$
$$D_b^* = \{(i, j) : (i, j) \in D^* \wedge (i, j+1) \notin D^* \wedge (i+1, j) \notin D^*\}. \tag{10}$$

Эти подмножества и условия вложенности (5) и (6) для рассматриваемых семейств позволяют определить D-код как сумму тензорных произведений кодов, двойственных к кодам из $\mathcal{S}_1$, $\mathcal{S}_2$, а также использовать простую формулу для вычисления размерности D-кода. Согласно [24, леммы 2 и 3], справедлива следующая

Теорема 1. Пусть $D \in F(D_0)$, $C_1(i) \in \mathcal{S}_1$, $C_2(j) \in \mathcal{S}_2$. Тогда

1) $\dim(C(D)) = \sum\limits_{(i,j)\in D} k_1(i)k_2(j)$, где $k_t(i) = \dim(C_t(i)) - \dim(C_t(i+1))$, $t = 1, 2$;

2) $C(D) = \mathcal{L}\left(\bigcup\limits_{(i,j)\in D_b} C_1(i) \otimes C_2(j)\right)$, $\overline{C(D)} = \mathcal{L}\left(\bigcup\limits_{(i,j)\in D_b^*} \overline{C_1(i)} \otimes \overline{C_2(j)}\right)$.

Замечание 1. Приведённое определение множества D^* немного отличается от оригинального из [24], тем не менее теорема 1 остаётся справедливой. Определение D^* уточняет поведение этого множества на границах D_0, что, в свою очередь, влияет на определение D_b^*. Это множество может отличаться от определённого в [24] в точках (i, j), таких, что $i = 0$ или $j = 0$. В этих точках $\overline{C_1(i)} \otimes \overline{C_2(j)} = \{\overline{0}\} \subset \mathbb{F}_2^{n_1 n_2}$, поэтому в теореме 1 такие точки не влияют на вид кода $\overline{C(D)}$.

Так как $\mathcal{S}_1$, $\mathcal{S}_2$ — семейства кодов Рида — Маллера, то

$$\begin{gathered} k_t(l) = \dim(\mathrm{RM}(m_t, m_t - l)) - \dim(\mathrm{RM}(m_t, m_t - l - 1)) = \\ = \sum_{p=0}^{m_t-l} C_{m_t}^p - \sum_{p=0}^{m_t-l-1} C_{m_t}^p = C_{m_t}^{m_t-l} + \sum_{p=0}^{m_t-l-1} C_{m_t}^p - \sum_{p=0}^{m_t-l-1} C_{m_t}^p = C_{m_t}^{m_t-l} = C_{m_t}^l, \end{gathered} \tag{11}$$

где $l = i$ для $t = 1$ и $l = j$ для $t = 2$. Поэтому в соответствии с п. 1 теоремы 1

$$\dim(C(D)) = \sum_{(i,j)\in D} C_{m_1}^i C_{m_2}^j. \tag{12}$$

Лемма 2. Если $D = \{(i, j) : i + j > \mu\}$, то $C(D) = \mathrm{RM}(m_1 + m_2, m_1 + m_2 - \mu - 1)$, где $\mu \in \{0, \ldots, m_1 + m_2 - 1\}$.

Доказательство. Для заданного D, по формуле (9), $D^* = \{(i, j) : i + j \leqslant \mu + 2\}$. В соответствии с замечанием 1 можно предполагать $i, j > 0$, а по п. 2 теоремы 1 код $\overline{C(D)}$ представим в виде

$$\overline{C(D)} = \mathcal{L}\left(\bigcup_{(i,j)\in D_b^*} \overline{\mathrm{RM}(m_1, m_1 - i)} \otimes \overline{\mathrm{RM}(m_2, m_2 - j)}\right),$$

где $D_b^* = \{(i, j) : i + j = \mu + 2\}$ по определению (10). Из свойства кода, двойственного к коду Рида — Маллера, и леммы 1 для любого $(i, j) \in D_b^*$ выполняется

$$\begin{gathered} \overline{\mathrm{RM}(m_1, m_1 - i)} \otimes \overline{\mathrm{RM}(m_2, m_2 - j)} = \mathrm{RM}(m_1, i - 1) \otimes \mathrm{RM}(m_2, j - 1) \subset \\ \subset \mathrm{RM}(m_1 + m_2, i + j - 2) = \mathrm{RM}(m_1 + m_2, \mu). \end{gathered}$$

Из этого следует, что

$$\overline{C(D)} \subseteq \mathrm{RM}(m_1 + m_2, \mu). \tag{13}$$

Теперь покажем, что

$$\dim(\overline{C(D)}) = \dim(\mathrm{RM}(m_1 + m_2, \mu)). \tag{14}$$

Из [24, формула (30)] с учётом (11) получаем

$$\dim(\overline{C(D)}) = \sum_{(i,j)\notin D} k_1(i)k_2(j) = \sum_{i+j\leqslant\mu} C_{m_1}^i C_{m_2}^j = \sum_{\lambda=0}^{\mu} \sum_{i=0}^{\lambda} C_{m_1}^i C_{m_2}^{\lambda-i}.$$

Из тождества Вандермонда следует формула $\sum\limits_{i=0}^{\lambda} C_{m_1}^i C_{m_2}^{\lambda-i} = C_{m_1+m_2}^{\lambda}$. Следовательно, $\dim(\overline{C(D)}) = \sum\limits_{\lambda=0}^{\mu} C_{m_1+m_2}^{\lambda}$. С другой стороны, $\dim(\mathrm{RM}(m_1 + m_2, \mu)) = \sum\limits_{\lambda=0}^{\mu} C_{m_1+m_2}^{\lambda}$. Отсюда получаем (14). Из (13) и (14) следует, что $\overline{C(D)} = \mathrm{RM}(m_1 + m_2, \mu)$ и $C(D) = \overline{\overline{C(D)}}$. ■

1.3. О разложимости степеней D-конструкции на бинарных кодах Рида — Маллера

Далее нам понадобится следующая техническая лемма:

Лемма 3. Пусть $D_b^* = \{(k_1, l_1), (k_2, l_2), \ldots, (k_s, l_s)\}$ для некоторого $D \in F(D_0)$, где $s \leqslant \min\{m_1, m_2\} + 1$, $k_i \in \{0, \ldots, m_1 + 1\}$, $l_i \in \{0, \ldots, m_2 + 1\}$. Последовательность $(k_i)_{i=1}^s$ — возрастающая тогда и только тогда, когда $(l_i)_{i=1}^s$ — убывающая последовательность.

Доказательство. Пусть $(k_i)_{i=1}^s$ — возрастающая последовательность. Предположим, что последовательность $(l_i)_{i=1}^s$ при этом не является убывающей. Это означает, что для $j > i$ существуют такие точки $(k_i, l_i), (k_j, l_j) \in D_b^*$, что $k_j > k_i$ и $l_j \geqslant l_i$. По определению множества D_b^* имеем $(k_i, l_i), (k_j, l_j) \in D^*$, при этом $(k_i + 1, l_i) \notin D^*$, где $k_i + 1 \leqslant k_j$. По определению множества D^* если $(k_j, l_j) \in D^*$, то $(k_j - 1, l_j - 1) \notin D$, а так как $D \in F(D_0)$, то $(k_p, l_p) \notin D$ для любых $k_p \leqslant k_j - 1$, $l_p \leqslant l_j - 1$. Это означает, что $(k_p, l_p) \in D^*$ для любых $k_p \leqslant k_j$, $l_p \leqslant l_j$ по определению D^*. Поэтому если $(k_j, l_j) \in D^*$, $k_j \geqslant k_i + 1$, $l_j \geqslant l_i$, то и $(k_i + 1, l_i) \in D^*$. Приходим к противоречию, значит, $(l_i)_{i=1}^s$ — убывающая последовательность. Аналогично утверждение доказывается в обратную сторону. ■

Рассмотрим семейства кодов $\mathcal{S}_1$, $\mathcal{S}_2$, определённые в п. 1.2. Пусть

$$D_b^* = \{(k_1, l_1), (k_2, l_2), \ldots, (k_s, l_s)\}, \tag{15}$$

где $s \leqslant \min\{m_1, m_2\}+1$; $k_i \in \{0, \ldots, m_1+1\}$; $l_i \in \{0, \ldots, m_2+1\}$; $(k_i)_{i=1}^s$ — возрастающая последовательность. Тогда по лемме 3 $(l_i)_{i=1}^s$ — убывающая. Поэтому в соответствии с п. 2 теоремы 1

$$\overline{C(D)} = \sum_{i=1}^{s} \overline{C_1(k_i)} \otimes \overline{C_2(l_i)}. \tag{16}$$

Для $s \geqslant 2$ и $i < s$ во введённых обозначениях получаем

$$\overline{C_1(k_i)} \subset \overline{C_1(k_{i+1})}, \overline{C_2(l_{i+1})} \subset \overline{C_2(l_i)}. \tag{17}$$

Пусть $r_{k_i}^1 = m_1 - k_i$ — порядок кода Рида — Маллера $C_1(k_i) = \mathrm{RM}(m_1, r_{k_i}^1)$, $r_{l_i}^2 = m_2 - l_i$ — порядок кода Рида — Маллера $C_2(l_i) = \mathrm{RM}(m_2, r_{l_i}^2)$, а

$$\overline{r}_{k_i}^1 = m_1 - (m_1 - k_i) - 1 = k_i - 1, \quad \overline{r}_{l_i}^2 = m_2 - (m_2 - l_i) - 1 = l_i - 1 \tag{18}$$

— порядки двойственных кодов $\overline{C_1(k_i)} = \mathrm{RM}(\overline{r}_{k_i}^1, m_1)$ и $\overline{C_2(l_i)} = \mathrm{RM}(\overline{r}_{l_i}^2, m_2)$ соответственно. Тогда с учётом (4), (16) и (18) получаем

$$\overline{C(D)}^2 = \left(\sum_{i=1}^{s} \overline{C_1(k_i)} \otimes \overline{C_2(l_i)}\right)^2 =$$

$$= \sum_{i=1}^{s} \overline{C_1(k_i)}^2 \otimes \overline{C_2(l_i)}^2 + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} (\overline{C_1(k_p)} \star \overline{C_1(k_j)}) \otimes (\overline{C_2(l_p)} \star \overline{C_2(l_j)}) = \tag{19}$$

$$= \sum_{i=1}^{s} \mathrm{RM}(2\overline{r}_{k_i}^1, m_1) \otimes \mathrm{RM}(2\overline{r}_{l_i}^2, m_2) + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} \mathrm{RM}(\overline{r}_{k_p}^1 + \overline{r}_{k_j}^1, m_1) \otimes \mathrm{RM}(\overline{r}_{l_p}^2 + \overline{r}_{l_j}^2, m_2) =$$

$$= \sum_{i=1}^{s} \mathrm{RM}(2k_i - 2, m_1) \otimes \mathrm{RM}(2l_i - 2, m_2) + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} \mathrm{RM}(k_p + k_j - 2, m_1) \otimes \mathrm{RM}(l_p + l_j - 2, m_2) =$$

$$= \sum_{i=1}^{s} \overline{C_1(2k_i - 1)} \otimes \overline{C_2(2l_i - 1)} + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} \overline{C_1(k_p + k_j - 1)} \otimes \overline{C_2(l_p + l_j - 1)}. \tag{20}$$

Таким образом, код $\overline{C(D)}^2$ представляет собой сумму тензорных произведений двойственных кодов к кодам из семейств $\mathcal{S}_1$, $\mathcal{S}_2$. Отметим, что формула для нахождения размерности квадрата произвольного кода неизвестна. Но в случае кода $\overline{C(D)}$ эту формулу можно получить. Перепишем (20) в виде

$$\overline{C(D)}^2 = \sum_{i=1}^{s} C_1(m_1 - 2k_i + 2) \otimes C_2(m_2 - 2l_i + 2) + \\ + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} C_1(m_1 - (k_p + k_j) + 2) \otimes C_2(m_2 - (l_p + l_j) + 2).$$

Пусть $X = \{(m_1 - 2k_i + 2, m_2 - 2l_i + 2) : i \in \{1, \ldots, s\}\} \cup \{(m_1 - (k_p + k_j) + 2, m_2 - (l_p + l_j) + 2) : p, j \in \{1, \ldots, s\}, p \neq j\}$ — множество точек из D_0, которое соответствует коду $\overline{C(D)}^2$. Построим множество $D' \subseteq D_0$ следующим образом. Для каждой точки $(i, j) \in X$ добавим в D' все точки $(k, l) \in D_0$, такие, что $k \geqslant i$ и $l \geqslant j$. Отметим, что $D' \in F(D_0)$. Множеству D' соответствует код $C(D')$ вида (8). При этом коды, соответствующие точкам из D', либо совпадают с кодами, соответствующими точкам из X, либо, согласно (7), являются их подкодами. Следовательно,

$$C(D') = \overline{C(D)}^2.$$

Так как $D' \in F(D_0)$, то, согласно (12), размерность кода $\overline{C(D)}^2$ вычисляется по формуле

$$\dim(\overline{C(D)}^2) = \dim(C(D')) = \sum_{(i,j) \in D'} C^i_{m_1} C^j_{m_2}. \tag{21}$$

Рассмотрим случаи, при которых удаётся упростить представление (19).

Теорема 2. Пусть D^*_b вида (15), $\overline{C(D)}$ вида (16) и $\overline{r}^1_{k_i}, \overline{r}^2_{l_i}$ вида (18).

1) Если $\overline{r}^1_{k_1} \geqslant m_1/2$ и $\overline{r}^2_{l_1} < m_2/2$, то $\overline{C(D)}^2 = \mathbb{F}_2^{n_1} \otimes \overline{C_2(l_1)}^2$.

2) Если $\overline{r}^1_{k_s} < m_1/2$ и $\overline{r}^2_{l_s} \geqslant m_2/2$, то $\overline{C(D)}^2 = \overline{C_1(k_s)}^2 \otimes \mathbb{F}_2^{n_2}$.

Доказательство.

1) Согласно (4), если $\overline{r}^1_{k_1} \geqslant m_1/2$ и $\overline{r}^2_{l_1} < m_2/2$, то $\overline{C_1(k_1)}^2 = \mathbb{F}_2^{n_1}$ и $\overline{C_2(l_1)}^2 \neq \mathbb{F}_2^{n_2}$. Так как $(k_i)_{i=1}^s$ — возрастающая последовательность, из леммы 3 и (18) последовательность $(\overline{r}^1_{k_i})_{i=1}^s$ возрастающая, а $(\overline{r}^2_{l_i})_{i=1}^s$ — убывающая. Поэтому из (19) следует

$$\overline{C(D)}^2 = \sum_{i=1}^{s} \mathbb{F}_2^{n_1} \otimes \overline{C_2(l_i)}^2 + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} \mathbb{F}_2^{n_1} \otimes (\overline{C_2(l_p)} \star \overline{C_2(l_j)}) = \\ = \mathbb{F}_2^{n_1} \otimes \left(\sum_{i=1}^{s} \overline{C_2(l_i)}^2 + \sum_{\substack{p,j=1 \\ p \neq j}}^{s} \overline{C_2(l_p)} \star \overline{C_2(l_j)} \right) = \mathbb{F}_2^{n_1} \otimes \overline{C_2(l_1)}^2,$$

где последнее равенство вытекает из вложений $\overline{C_2(l_p)} \star \overline{C_2(l_j)} \subseteq \overline{C_2(l_1)}^2$ и $\overline{C_2(l_i)}^2 \subseteq \subseteq \overline{C_2(l_1)}^2$, которые следуют из (4) и (17).

2) Согласно (4), если $\overline{r}^1_{k_s} < m_1/2$ и $\overline{r}^2_{l_s} \geqslant m_2/2$, то $\overline{C_1(k_s)}^2 \neq \mathbb{F}_2^{n_1}$ и $\overline{C_2(l_s)}^2 = \mathbb{F}_2^{n_2}$. Поэтому из (19) следует

$$\overline{C(D)}^2 = \sum_{i=1}^{s} \overline{C_1(k_i)}^2 \otimes \mathbb{F}_2^{n_2} + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} (\overline{C_1(k_p)} \star \overline{C_1(k_j)}) \otimes \mathbb{F}_2^{n_2} =$$

$$= \left(\sum_{i=1}^{s} \overline{C_1(k_i)}^2 + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} \overline{C_1(k_p)} \star \overline{C_1(k_j)} \right) \otimes \mathbb{F}_2^{n_2} = \overline{C_1(k_s)}^2 \otimes \mathbb{F}_2^{n_2}.$$

Теорема 2 доказана. ■

Согласно [23], двоичные коды Рида — Маллера $\mathrm{RM}(m, r)$ для $r < m$ являются неразложимыми. Так как $\overline{C_2(l_1)}^2$ — код Рида — Маллера, не совпадающий со всем пространством при выполнении условий первого утверждения теоремы 2, то, как следует из (3), код $\overline{C(D)}^2$ раскладывается в прямую сумму неразложимых кодов. При выполнении условий второго утверждения теоремы 2 код $\overline{C_1(k_s)}^2$ также является кодом Рида — Маллера, не совпадающим со всем пространством. Согласно [32, формула (10)], для любых двух матриц A и B найдутся такие перестановочные матрицы P и Q подходящих размеров, что $A \otimes B = P(B \otimes A)Q$. Тогда найдётся такая перестановка $\sigma \in \mathcal{P}_{n_1 n_2}$, что $\sigma(\overline{C_1(k_s)}^2 \otimes \mathbb{F}_2^{n_2}) = \mathbb{F}_2^{n_2} \otimes \overline{C_1(k_s)}^2$. Следовательно, в этом случае код $\overline{C(D)}^2$ является перестановочно эквивалентным прямой сумме неразложимых кодов.

Теорема 3. Пусть $\overline{C(D)}$ — код вида (16) и выполняется хотя бы одно из условий:

1) существует $i \in \{1, \ldots, s\}$, что $\overline{r}^1_{k_i} \geqslant m_1/2$ и $\overline{r}^2_{l_i} \geqslant m_2/2$;
2) существуют $p \in \{1, \ldots, s\}$, $j \in \{1, \ldots, s\}$, $p \neq j$, что $\overline{r}^1_{k_p} + \overline{r}^1_{k_j} \geqslant m_1$ и $\overline{r}^2_{l_p} + \overline{r}^2_{l_j} \geqslant m_2$.

Тогда $\overline{C(D)}^2 = \mathbb{F}_2^{n_1 n_2}$.

Доказательство. Пусть выполняется первое условие, тогда из (4) следует, что $\overline{C_1(k_i)}^2 = \mathbb{F}_2^{n_1}$ и $\overline{C_2(l_i)}^2 = \mathbb{F}_2^{n_2}$. Значит, одно из слагаемых в (19) имеет вид $\mathbb{F}_2^{n_1} \otimes \mathbb{F}_2^{n_2}$. В этом случае $\overline{C(D)}^2 = \mathbb{F}_2^{n_1 n_2}$. Пусть теперь выполняется второе условие, тогда из (4) следует, что $\overline{C_1(k_p)} \star \overline{C_1(k_j)} = \mathbb{F}_2^{n_1}$ и $\overline{C_2(l_p)} \star \overline{C_2(l_j)} = \mathbb{F}_2^{n_2}$. Таким образом, одно из слагаемых в (19) имеет вид $\mathbb{F}_2^{n_1} \otimes \mathbb{F}_2^{n_2}$, поэтому $\overline{C(D)}^2 = \mathbb{F}_2^{n_1 n_2}$. ■

Отметим, что при выполнении условий теоремы 3 код $\overline{C(D)}^2$ совпадает с $\mathbb{F}_2^{n_1 n_2}$ и поэтому не эквивалентен прямой сумме нетривиальных кодов Рида — Маллера.

Теорема 4. Пусть $\overline{C(D)}$ — код вида (16) и выполняются условия $\overline{r}^1_{k_1} < m_1/2$, $\overline{r}^2_{l_1} \geqslant m_2/2$ и $\overline{r}^1_{k_j} \geqslant m_1/2$, $\overline{r}^2_{l_j} < m_2/2$ для любых $j \geqslant 2$. Если для любых $p \in \{1, \ldots, s\}$, $j \in \{1, \ldots, s\}$, $p \neq j$, выполняются неравенства $\overline{r}^1_{k_p} + \overline{r}^1_{k_j} \geqslant m_1$ и $\overline{r}^2_{l_p} + \overline{r}^2_{l_j} < m_2$, то

$$\overline{C(D)}^2 = \overline{\tilde{C}_1 \otimes \tilde{C}_2}, \quad \overline{C(D)}^3 = \mathbb{F}_2^{n_1 n_2},$$

где $\tilde{C}_1 = \overline{\overline{C_1(k_1)}^2}$; $\tilde{C}_2 = \overline{\overline{C_2(l_2)}^2 + \sum\limits_{\substack{p,j=1 \\ p\neq j}}^{s} \overline{C_2(l_p)} \star \overline{C_2(l_j)}}$.

Доказательство. При выполнении условий теоремы, согласно (4), формула (19) примет вид

$$\overline{C(D)}^2 = \overline{C_1(k_1)}^2 \otimes \mathbb{F}_2^{n_2} + \mathbb{F}_2^{n_1} \otimes \overline{C_2(l_2)}^2 + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} (\overline{C_1(k_p)} \star \overline{C_1(k_j)}) \otimes (\overline{C_2(l_p)} \star \overline{C_2(l_j)}) =$$

$$= \overline{\overline{C_1(k_1)}^2} \otimes \mathbb{F}_2^{n_2} + \mathbb{F}_2^{n_1} \otimes \overline{\overline{C_2(l_2)}^2} + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} \mathbb{F}_2^{n_1} \otimes (\overline{C_2(l_p)} \star \overline{C_2(l_j)}) =$$

$$= \overline{\overline{C_1(k_1)}^2} \otimes \mathbb{F}_2^{n_2} + \mathbb{F}_2^{n_1} \otimes (\overline{\overline{C_2(l_2)}^2} + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} \overline{C_2(l_p)} \star \overline{C_2(l_j)}).$$

Согласно (2), во введённых в условии теоремы обозначениях получаем доказываемое утверждение. При этом

$$\overline{C(D)}^3 = (\overline{C_1(k_1)}^2 \otimes \mathbb{F}_2^{n_2} + \mathbb{F}_2^{n_1} \otimes (\overline{C_2(l_2)}^2 + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} \overline{C_2(l_p)} \star \overline{C_2(l_j)})) \star \overline{C(D)} =$$

$$= (\overline{C_1(k_1)}^2 \otimes \mathbb{F}_2^{n_2}) \star \left(\sum_{i=1}^{s} \overline{C_1(k_i)} \otimes \overline{C_2(l_i)} \right) +$$

$$+ (\mathbb{F}_2^{n_1} \otimes (\overline{C_2(l_2)}^2 + \sum_{\substack{p,j=1 \\ p\neq j}}^{s} \overline{C_2(l_p)} \star \overline{C_2(l_j)})) \star \left(\sum_{i=1}^{s} \overline{C_1(k_i)} \otimes \overline{C_2(l_i)} \right).$$

Рассмотрим первое слагаемое:

$$(\overline{C_1(k_1)}^2 \otimes \mathbb{F}_2^{n_2}) \star \left(\sum_{i=1}^{s} \overline{C_1(k_i)} \otimes \overline{C_2(l_i)} \right) = (\overline{C_1(k_1)}^2 \star \overline{C_1(k_s)}) \otimes \mathbb{F}_2^{n_2} = \mathbb{F}_2^{n_1 n_2},$$

где последнее равенство вытекает из $\overline{r}^1_{k_p} + \overline{r}^1_{k_j} \geqslant m_1$. Тогда $\overline{C(D)}^3 = \mathbb{F}_2^{n_1 n_2}$. ■

Следующая теорема доказывается аналогично.

Теорема 5. Пусть $\overline{C(D)}$ — код вида (16) и выполняются условия $\overline{r}^1_{k_j} < m_1/2$, $\overline{r}^2_{l_j} \geqslant m_2/2$ для любых $j < s$ и $\overline{r}^1_{k_s} \geqslant m_1/2$, $\overline{r}^2_{l_s} < m_2/2$. Если для любых $p \in \{1, \ldots, s\}$, $j \in \{1, \ldots, s\}$, $p \neq j$, выполняются неравенства $\overline{r}^1_{k_p} + \overline{r}^1_{k_j} < m_1$ и $\overline{r}^2_{l_p} + \overline{r}^2_{l_j} \geqslant m_2$, то

$$\overline{C(D)}^2 = \overline{\hat{C}_1 \otimes \hat{C}_2}, \ \ \overline{C(D)}^3 = \mathbb{F}_2^{n_1 n_2},$$

где $\hat{C}_1 = \overline{\overline{C_1(k_{s-1})}^2 + \sum\limits_{\substack{p,j=1 \\ p\neq j}}^{s} \overline{C_1(k_p)} \star \overline{C_1(k_j)}}$, $\hat{C}_2 = \overline{C_2(l_s)}^2$.

Отметим, что в теоремах 4 и 5 коды $\tilde{C}_1$, $\tilde{C}_2$, $\hat{C}_1$, $\hat{C}_2$ — коды Рида — Маллера. Поэтому при выполнении условий теорем 4 и 5 код $\overline{C(D)}^2$ является неразложимым и не эквивалентен прямой сумме нетривиальных кодов Рида — Маллера.

В случаях, не рассмотренных в теоремах 2–5, представление (19) пока не удаётся упростить и сделать выводы о разложимости кода $\overline{C(D)}^2$ в прямую сумму кодов Рида — Маллера. Но можно вычислить размерность кода $\overline{C(D)}^2$ с помощью (21) и оценить возможность разложения этого кода в прямую сумму одинаковых кодов Рида — Маллера. Если код $\overline{C(D)}^2$ раскладывается в прямую сумму 2^{m_1} (2^{m_2}) одинаковых

кодов Рида — Маллера K длины 2^{m_2} (2^{m_1}), то, согласно (3), его размерность равна $2^{m_1}\dim(K)$ $(2^{m_2}\dim(K))$. Поэтому если размерность кода $\overline{C(D)}^2$ не делится на 2^{m_1} (2^{m_2}), то этот код не раскладывается в прямую сумму 2^{m_1} (2^{m_2}) одинаковых кодов Рида — Маллера и не является перестановочно эквивалентным такому коду.

2. Криптосистема типа Мак-Элиса на D-кодах и анализ её структурной стойкости

2.1. Криптосистема типа Мак-Элиса на D-кодах

Кодовая криптосистема, предложенная Р. Мак-Элисом в [2], строится на основе порождающей матрицы G_C линейного $[n,k,d]_q$-кода $C \subset \mathbb{F}_q^n$ и случайно выбранных невырожденной $(k\times k)$-матрицы S и перестановочной $(n\times n)$-матрицы P [2]. Пара $(\tilde{G},t)$, где $\tilde{G} = SG_CP$, $t = \lfloor (d-1)/2 \rfloor$, является публичным ключом, с помощью которого вектор $\mathbf{m} \in \mathbb{F}_q^k$ шифруется по правилу $\mathbf{c} = \mathbf{m}\tilde{G} + \mathbf{e}$, где вектор $\mathbf{e}$ обычно выбирается случайно и равновероятно, $\mathrm{wt}(\mathbf{e}) \leqslant t$. Для расшифрования применяется секретный ключ (S,P,C): $\mathbf{m} = S^{-1}\mathsf{Dec}_C(\mathbf{c}P^{-1})$, где $\mathsf{Dec}_C : \mathbb{F}_q^n \to \mathbb{F}_q^k$ — эффективный декодер для кода C. Криптосистему типа Мак-Элиса на коде C обозначим McE(C).

В настоящей работе обобщается криптосистема из [25] путём применения D-конструкции вместо тензорного произведения кодов, а именно: в качестве декодируемого кода используется код $\overline{C(D)}$ с быстрым мажоритарным декодером [26]. Отметим, что этот код может быть секретным для дополнительного усиления стойкости криптосистемы. Другими словами, множество D может быть частью секретного ключа. Перестановочная матрица P имеет размер $n_1n_2 \times n_1n_2$, а невырожденная матрица S — размер $\dim(\overline{C(D)}) \times \dim(\overline{C(D)})$. Матрица публичного ключа тогда имеет вид

$$\tilde{G} = S(G_{\overline{C(D)}})P. \tag{22}$$

Обозначим эту криптосистему McE$(C(D))$.

Отметим, что в соответствии с леммой 2 код $\overline{C(D)}$ может быть кодом Рида — Маллера. Учитывая результаты работ [9, 13], ключи криптосистемы McE$(C(D))$ при таких параметрах D-кода являются слабыми. Для выявления других возможных слабых ключей системы McE$(C(D))$ в настоящей работе используется произведение Шура — Адамара, а именно: строится атака для системы McE$(C(D))$, в которой используется разложимость степени Шура — Адамара D-кода, и на основе результатов п. 1.3 проводится анализ стойкости к разработанной атаке.

2.2. Атака на основе произведения Шура — Адамара

Рассмотрим код $\overline{C(D)}$ вида (16). Пусть $K = \dim(\overline{C(D)})$, $k = \dim(\overline{C_2(l_1)})$. Так как код $\overline{C(D)}$ имеет вид (16), то для $\tau_i = \{(i-1)n_2+1,\ldots,in_2\}$ проекция кода $\overline{C(D)}$ на множество τ_i, получаемая путём отбрасывания в кодовых словах всех координат, за исключением координат из множества τ_i, совпадает с $\overline{C_2(l_1)}$, $i = 1,\ldots,n_1$. Поэтому найдутся такие $(K\times k)$-матрицы M_1, …, M_{n_1} ранга k, что

$$G_{\overline{C(D)}} = (M_1|\ldots|M_{n_1})\mathrm{diag}(G_{\overline{C_2(l_1)}},\ldots,G_{\overline{C_2(l_1)}}). \tag{23}$$

Для кода $\overline{C_2(l_1)}$ и криптосистемы McE$(\overline{C_2(l_1)})$ обозначим через Attack алгоритм, принимающий на вход публичный ключ $\tilde{G}'$ криптосистемы Мак-Элиса на коде Рида — Маллера $\overline{C_2(l_1)}$ с порождающей матрицей $G_{\overline{C_2(l_1)}}$ и возвращающий невырожденную $(k\times k)$-матрицу S' и перестановочную $(n_2\times n_2)$-матрицу P', для которых $\tilde{G}' = S'G_{\overline{C_2(l_1)}}P'$.

Теорема 6. Пусть Attack — алгоритм полиномиальной сложности. Если $\overline{C(D)}^v = \mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}^v$ для некоторого $v(\in \mathbb{N})$ и код $\overline{C_2(l_1)}^v$ неразложимый, то существует алгоритм полиномиальной сложности, который по публичной матрице $\tilde{G}$ вида (22) находит такую перестановку π, что $\pi(\mathcal{L}(\tilde{G})) \subseteq \mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}$.

Доказательство. Публичная матрица $\tilde{G}$ вида (22), с учётом представления (23), имеет вид

$$\tilde{G} = S(G_{\overline{C(D)}})P = (\mathbf{M}_1 G_{\overline{C_2(l_1)}} | \dots | \mathbf{M}_{n_1} G_{\overline{C_2(l_1)}})P, \tag{24}$$

где $\mathbf{M}_i = SM_i$, $i = 1, \dots, n_1$. Обозначим через $\mathcal{P}_{n_1 n_2}$ симметрическую группу из $n_1 n_2$ элементов. Для удобства перестановку из этой группы, соответствующую матрице P в (24), обозначим ϕ. Вместо P иногда будем писать P_ϕ.

Из условия вытекает, что код с матрицей $\tilde{G}^v$ перестановочно эквивалентен тензорному произведению $\mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}^v$. Так как по условию код $\overline{C_2(l_1)}^v$ неразложимый, то, применяя полиномиальный алгоритм Decomposition из [21] для разложения кода $\mathcal{L}(\tilde{G}^v)$ в прямую сумму подкодов, по матрице $\tilde{G}^v$ можно получить такую перестановку $\sigma(\in \mathcal{P}_{n_1 n_2})$, что

$$\tilde{G}^v P_\sigma \sim \operatorname{diag}(\dot{\mathbf{G}}_1, \dots, \dot{\mathbf{G}}_{n_1}), \tag{25}$$

где матрицы $\dot{\mathbf{G}}_i$ полного ранга порождают коды $C_{2,i} = \mathcal{L}(\dot{\mathbf{G}}_i) \sim \overline{C_2(l_1)}^v$. Тогда

$$\mathcal{L}(\tilde{G}^v P_\sigma) = \sigma(\phi(\mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}^v)) = \sigma(\phi(\underbrace{\overline{C_2(l_1)}^v \oplus \dots \oplus \overline{C_2(l_1)}^v}_{n_1})) = C_{2,1} \oplus \dots \oplus C_{2,n_1}. \tag{26}$$

Из неразложимости кода $\overline{C_2(l_1)}^v$ вытекает, что группа перестановочных автоморфизмов кода $\mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}^v$ в этом случае состоит из перестановок вида

$$\omega = \begin{pmatrix} 1, & \dots, & n_2, & n_2+1, & \dots, & 2n_2, & \dots, & (n_1-1)n_2+1, & \dots, & n_1 n_2 \\ x_1, & \dots, & x_{n_2}, & x_{n_2+1}, & \dots, & x_{2n_2}, & \dots, & x_{(n_1-1)n_2+1}, & \dots, & x_{n_1 n_2} \end{pmatrix} \tag{27}$$

с двумя ограничениями: 1) для каждого $i \in \{0, \dots, n_1 - 1\}$ найдётся единственное $j \in \{0, \dots, n_1 - 1\}$, такое, что $\{x_{in_2+1}, \dots, x_{(i+1)n_2}\} = \{jn_2 + 1, \dots, (j+1)n_2\}$; 2) каждый набор $(x_{in_2+1}, \dots, x_{(i+1)n_2})$ упорядочен так, что проекция кода $\omega(\mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}^v)$, получаемая путём отбрасывания в кодовых словах всех координат, за исключением координат из множества $\{x_{in_2+1}, \dots, x_{(i+1)n_2}\}$, совпадает с $\overline{C_2(l_1)}^v$ [29, лемма 11]. Отсюда вытекает, что перестановка $\xi = \sigma \circ \phi$ в (26) имеет вид (27). Так как алгоритм Decomposition по разложимому коду находит перестановку, переводящую этот код в прямую сумму неразложимых кодов с точностью до перестановки координат в кодах — слагаемых, то для ξ выполняется только ограничение 1.

Учитывая (24), получаем

$$\begin{aligned} \tilde{G}P_\sigma &= S(G_{\overline{C_2(l_1)}})P_\phi P_\sigma = S(G_{\overline{C_2(l_1)}})P_\xi = (\mathbf{M}_1 G_{\overline{C_2(l_1)}} | \dots | \mathbf{M}_{n_1} G_{\overline{C_2(l_1)}})P_\xi = \\ &= (\mathbf{M}_{\theta^{-1}(1)} G_{\overline{C_2(l_1)}} P_{\gamma_1} | \dots | \mathbf{M}_{\theta^{-1}(n_1)} G_{\overline{C_2(l_1)}} P_{\gamma_{n_1}}) = \\ &= [\tilde{\mathbf{G}}_1 | \dots | \tilde{\mathbf{G}}_{n_1}], \quad \gamma_j \in \mathcal{P}_{n_2}, \quad j = 1, \dots, n_1, \end{aligned} \tag{28}$$

где θ — некоторая перестановка из $\mathcal{P}_{n_1}$, соответствующая ξ, а $\tilde{\mathbf{G}}_i = \mathbf{M}_{\theta^{-1}(i)} G_{\overline{C_2(l_1)}} P_{\gamma_i}$. При этом запись $\theta^{-1}(i)$ обозначает порядковый номер j матрицы M_j, которая после действия P_ξ находится в блочном представлении (28) на месте i-го блока.

Очевидно, что $\operatorname{rank}(\tilde{\mathbf{G}}_i) = k$ для всех $i = 1, \ldots, n_1$. По $\tilde{\mathbf{G}}_i$ легко построить соответствующую невырожденную $(k \times n_2)$-матрицу $\hat{\mathbf{G}}_i$, состоящую из k линейно независимых строк матрицы $\tilde{\mathbf{G}}_i$. Обозначим через F_i такую невырожденную $(K \times K)$-матрицу, что

$$F_i \tilde{\mathbf{G}}_i = \left[\frac{\hat{\mathbf{G}}_i}{O}\right],$$

где O — нулевая $((K-k) \times n_2)$-матрица. Матрица $\hat{\mathbf{G}}_i$ может рассматриваться как публичная матрица криптосистемы $\mathrm{McE}(\overline{C_2(l_1)})$. Применяя к каждой матрице $\hat{\mathbf{G}}_i$ алгоритм Attack, можно найти перестановочную матрицу P_{δ_i} ($\delta_i \in \mathcal{P}_{n_2}$) и невырожденную матрицу S'_i, что $\hat{\mathbf{G}}_i = S'_i G_{\overline{C_2(l_1)}} P_{\delta_i}$. Отсюда получаем

$$\tilde{G} P_\sigma \operatorname{diag}(P_{\delta_1}^{-1}, \ldots, P_{\delta_{n_1}}^{-1}) = [\tilde{\mathbf{G}}_1 P_{\delta_1}^{-1} | \ldots | \tilde{\mathbf{G}}_{n_1} P_{\delta_{n_1}}^{-1}] = \left[F_1^{-1}\left[\frac{\hat{\mathbf{G}}_1}{O}\right] P_{\delta_1}^{-1} \middle| \ldots \middle| F_{n_1}^{-1}\left[\frac{\hat{\mathbf{G}}_{n_1}}{O}\right] P_{\delta_{n_1}}^{-1}\right] =$$

$$= \left[F_1^{-1}\left[\frac{S'_1}{O'}\right] G_{\overline{C_2(l_1)}} \middle| \ldots \middle| F_{n_1}^{-1}\left[\frac{S'_{n_1}}{O'}\right] G_{\overline{C_2(l_1)}}\right] =$$

$$= [\hat{\mathbf{S}}_1 G_{\overline{C_2(l_1)}} | \ldots | \hat{\mathbf{S}}_{n_1} G_{\overline{C_2(l_1)}}] = [\hat{\mathbf{S}}_1 | \ldots | \hat{\mathbf{S}}_{n_1}](I_{n_1} \otimes G_{\overline{C_2(l_1)}}),$$

где O' — нулевая $((K-k) \times k)$-матрица; $\hat{\mathbf{S}}_i = F_i^{-1}\left[\frac{S'_i}{O'}\right]$. Следовательно, искомая перестановка имеет вид $\pi = \delta_1^{-1} \circ \ldots \circ \delta_{n_1}^{-1} \circ \sigma$. Так как Decomposition, Attack — алгоритмы полиномиальной сложности, то описанный алгоритм нахождения π также имеет полиномиальную сложность. ■

Отметим, что для произвольных линейных кодов C_1, C_2 над полем $\mathbb{F}_q$ существует такая перестановка σ, зависящая только от длин кодов C_1, C_2, что $C_1 \otimes C_2 = \sigma(C_2 \otimes \otimes C_1)$. Поэтому теорема 6 остаётся справедливой, если вместо $\overline{C(D)}^v = \mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}^v$ рассмотреть произведение $\overline{C(D)}^u = \overline{C_1(k_s)}^u \otimes \mathbb{F}_q^{n_2}$.

Рассмотрим криптосистему Мак-Элиса $\mathrm{McE}(\overline{C(D)})$. Теорема 6 показывает, что в ряде случаев по публичному ключу криптосистемы за полиномиальное время может быть найдена перестановка, позволяющая преобразовать этот ключ к более простому виду. Этот вид даёт возможность, например, использовать для дешифрования сообщений декодер кода $\mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}$, заключающийся в применении декодера кода Рида — Маллера $\overline{C_2(l_1)}$ к каждому из n_1 блоков длины n_2. Ясно, что с большой вероятностью такой декодер будет ошибаться, так как кодовое расстояние кода $\mathbb{F}_q^{n_1} \otimes \overline{C_2(l_1)}$ меньше кодового расстояния D-кода, используемого в криптосистеме. Представляется, что уменьшить вероятность неправильного декодирования можно, например, применяя метод декодирования по информационным совокупностям и учитывая структуру кода. Для D-кода $\overline{C(D)}$, удовлетворяющего условиям теоремы 2, выполняются условия теоремы 6. Это позволяет сделать вывод, что коды с такими параметрами не подходят для использования в криптосистеме Мак-Элиса. С другой стороны, D-коды с параметрами, удовлетворяющими условиям одной из теорем 3–5, обеспечивают устойчивость к нахождению перестановки π. Это вытекает из того факта, что их квадрат совпадает со всем пространством или является неразложимым кодом. При этом, в случае неразложимости квадрата D-кода, его третья степень совпадает со всем пространством.

ЛИТЕРАТУРА

1. `http://csrc.nist.gov/projects/post-quantum-cryptography` — National Institute of Standards and Technology (NIST). Post-Quantum Cryptography, 2021.

2. *McEliece R. J.* A public-key cryptosystem based on algebraic coding theory // DSN Progress Report. 1978. P. 42–44.
3. `https://classic.mceliece.org/nist/mceliece-20201010.pdf`. 2020.
4. *Sidel'nikov V. M.* Open coding based on Reed — Muller binary codes // Discr. Math. Appl. 1994. V. 4. No. 3. P. 191–207.
5. *Niederreiter H.* Knapsack-type cryptosystems and algebraic coding theory // Problems Control Inform. Theory. 1986. V. 15. No. 2. P. 159–166.
6. *Berger T. and Loidreau P.* How to mask the structure of codes for a cryptographic use // Des. Codes Cryptogr. 2005. V. 35. No. 1. P. 63–79.
7. *Janwa H. and Moreno O.* McEliece public cryptosystem using algebraic-geometric codes // Des. Codes Cryptogr. 1996. V. 8. P. 293–307.
8. *Couvreur A., Marquez-Corbella I., and Pellikaan R.* Cryptanalysis of McEliece cryptosystem based on algebraic geometry codes and their subcodes // IEEE Trans. Inform. Theory. 2017. V. 8. No. 63. P. 5404–5418.
9. *Chizhov I. V. and Borodin M. A.* Effective attack on the McEliece cryptosystem based on Reed — Muller codes // Discr. Math. Appl. 2014. V. 24. No. 5. P. 273–280.
10. *Wieschebrink C.* Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes // LNCS. 2010. V. 6061. P. 61–72.
11. *Sidel'nikov V. M. and Shestakov S. O.* On an encoding system constructed on the basis of generalized Reed — Solomon codes // Discr. Math. Appl. 1992. V. 4. No. 2. P. 439–444.
12. *Деундяк В. М., Дружинина М. А., Косолапов Ю. В.* Модификация криптоаналитического алгоритма Сидельникова — Шестакова для обобщенных кодов Рида — Соломона и ее программная реализация // Изв. вузов. Северо-Кавказский регион. Сер.: Технические науки. 2006. № 4. С. 15–19.
13. *Minder L. and Shokrollahi A.* Cryptanalysis of the Sidelnikov cryptosystem // LNCS. 2007. V. 4515. P. 347–360.
14. *Бородин М. А., Чижов И. В.* Классификация произведений Адамара подкодов коразмерности 1 кодов Рида — Маллера // Дискретная математика. 2020. Т. 32. № 1. С. 115–134.
15. *Wieschebrink C.* Two NP-complete problems in coding theory with an application in code based cryptography // IEEE Intern. Symp. Inform. Theory. 2006. P. 1733–1737.
16. *Couvreur A., Gaborit P., Gauthier-Umana V., et al.* Distinguisher-based attacks on public-key cryptosystems using Reed — Solomon codes // Des. Codes Cryptogr. 2014. V. 73. P. 641–666.
17. *Otmani A. and Kalachi H.* Square code attack on a modified Sidelnikov cryptosystem // LNCS. 2015. V. 9084. P. 173–183.
18. *Couvreur A. and Lequesne M.* On the security of subspace subcodes of Reed — Solomon codes for public key encryption // IEEE Trans. Inform. Theory. 2022. V. 68. No. 1. P. 632–648.
19. *Morelos-Zaragoza R. H.* The Art of Error Correcting Coding. John Wiley & Sons, Ltd. 2006. 263 p.
20. *Деундяк В. М., Косолапов Ю. В.* Криптосистема на индуцированных групповых кодах // Модел. и анализ информ. систем. 2016. Т. 23. № 2. С. 137–152.
21. *Деундяк В. М., Косолапов Ю. В.* Анализ стойкости некоторых кодовых криптосистем, основанный на разложении кодов в прямую сумму // Вестн. ЮУрГУ. Сер. Матем. моделирование и программирование. 2019. Т. 12. № 3. С. 89–101.
22. *Egorova E., Kabatiansky G., Krouk E., and Tavernier C.* A new code-based public-key cryptosystem resistant to quantum computer attacks // J. Phys. Conf. Ser. 2019. V. 1163. P. 1–5.
23. *Deundyak V. M., Kosolapov Yu. V., and Maystrenko I. A.* On the decipherment of Sidel'nikov-type cryptosystems // LNCS. 2020. V. 12087. P. 20–40.

24. *Kasami T. and Lin S.* On the construction of a class of majority-logic decodable codes // IEEE Trans. Inform. Theory. 1971. V. IT-17. No. 5. P. 600–610.
25. *Deundyak V. M., Kosolapov Yu. V., and Lelyuk E. A.* Decoding the tensor product of MLD codes and applications for code cryptosystems // Automatic Control Comput. Sci. 2018. V. 52. No. 7. P. 647–657.
26. *Deundyak V. M. and Lelyuk E. A.* A graph-theoretical method for decoding some group MLD-codes // J. Appl. Industr. Math. 2020. V. 14. No. 2. P. 265–280.
27. *Randriambololona H.* On products and powers of linear codes under componentwise multiplication // Algorithmic Arithmetic Geometry Coding Theory. 2015. V. 637. P. 3–78.
28. *Deundyak V. M. and Kosolapov Yu. V.* On the strength of asymmetric code cryptosystems based on the merging of generating matrices of linear codes // XVI Intern. Symp. Prob. of Redundancy in Information and Control Systems. Russia, 2019. P. 143–148.
29. *Деундяк В. М., Косолапов Ю. В.* О некоторых свойствах произведения Шура — Адамара для линейных кодов и их приложениях // Прикладная дискретная математика. 2020. № 50. С. 72–86.
30. *Сидельников В. М.* Теория кодирования. М.: Физматлит, 2008.
31. *Grassl M. and Rotteler M.* Quantum block and convolutional codes from self-orthogonal product codes // Proc. IEEE Int. Symp. Inf. Theory. 2005. P. 1018–1022.
32. *Henderson H. V. and Searle S. R.* The vec-permutation matrix, the vec operator and Kronecker products: A review // Linear and Multilinear Algebra. 1981. No. 9. P. 271–288.

REFERENCES

1. `http://csrc.nist.gov/projects/post-quantum-cryptography` — National Institute of Standards and Technology (NIST). Post-Quantum Cryptography, 2021.
2. *McEliece R. J.* A public-key cryptosystem based on algebraic coding theory. DSN Progress Report, 1978, pp. 42–44.
3. `https://classic.mceliece.org/nist/mceliece-20201010.pdf`, 2020.
4. *Sidel'nikov V. M.* Open coding based on Reed — Muller binary codes. Discr. Math. Appl., 1994, vol. 4, no. 3, pp. 191–207.
5. *Niederreiter H.* Knapsack-type cryptosystems and algebraic coding theory. Problems Control Inform. Theory, 1986, vol. 15, no. 2, pp. 159–166.
6. *Berger T. and Loidreau P.* How to mask the structure of codes for a cryptographic use. Des. Codes Cryptogr., 2005, vol. 35, no. 1, pp. 63–79.
7. *Janwa H. and Moreno O.* McEliece public cryptosystem using algebraic-geometric codes. Des. Codes Cryptogr., 1996, vol. 8, pp. 293–307.
8. *Couvreur A., Marquez-Corbella I., and Pellikaan R.* Cryptanalysis of McEliece cryptosystem based on algebraic geometry codes and their subcodes. IEEE Trans. Inform. Theory, 2017, vol. 8, no. 63, pp. 5404–5418.
9. *Chizhov I. V. and Borodin M. A.* Effective attack on the McEliece cryptosystem based on Reed — Muller codes. Discr. Math. Appl., 2014, vol. 24, no. 5, pp. 273–280.
10. *Wieschebrink C.* Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. LNCS, 2010, vol. 6061, pp. 61–72.
11. *Sidel'nikov V. M. and Shestakov S. O.* On an encoding system constructed on the basis of generalized Reed — Solomon codes. Discr. Math. Appl., 1992, vol. 4, no. 2, pp. 439–444.
12. *Deundyak V. M., Druzhinina M. A., and Kosolapov Yu. V.* Modifikatsiya kriptoanaliticheskogo algoritma Sidel'nikova — Shestakova dlya obobshchennykh kodov Rida — Solomona i ee programmnaya realizatsiya [Modification of the Sidelnikov — Shestakov cryptanalytic

algorithm for generalized Reed — Solomon codes and its software implementation]. Izv. Vuzov. Severo-Kavkazskiy Region. Ser. Tekhnicheskie Nauki, 2006, no. 4, pp. 15–19. (in Russian)

13. *Minder L. and Shokrollahi A.* Cryptanalysis of the Sidelnikov cryptosystem. LNCS, 2007, vol. 4515, pp. 347–360.
14. *Borodin M. A. and Chizhov I. V.* Klassifikatsiya proizvedeniy Adamara podkodov korazmernosti 1 kodov Rida — Mallera [Classification of Hadamard products of codimension 1 subcodes of Reed — Muller codes]. Diskretnaya Matematika, 2020, vol. 32, no. 1, pp. 115–134. (in Russian)
15. *Wieschebrink C.* Two NP-complete problems in coding theory with an application in code based cryptography. IEEE Intern. Symp. Inform. Theory, 2006, pp. 1733–1737.
16. *Couvreur A., Gaborit P., Gauthier-Umana V., et al.* Distinguisher-based attacks on public-key cryptosystems using Reed — Solomon codes. Des. Codes Cryptogr., 2014, vol. 73, pp. 641–666.
17. *Otmani A. and Kalachi H.* Square code attack on a modified Sidelnikov cryptosystem. LNCS, 2015, vol. 9084, pp. 173–183.
18. *Couvreur A. and Lequesne M.* On the security of subspace subcodes of Reed — Solomon codes for public key encryption. IEEE Trans. Inform. Theory, 2022, vol. 68, no. 1, pp. 632–648.
19. *Morelos-Zaragoza R. H.* The Art of Error Correcting Coding. John Wiley & Sons, Ltd, 2006. 263 p.
20. *Deundyak V. M. and Kosolapov Yu. V.* Kriptosistema na indutsirovannykh gruppovykh kodakh [Cryptosystem on induced group codes]. Model. i Analiz Inform. Sistem, 2016, vol. 23, no. 2, pp. 137–152. (in Russian)
21. *Deundyak V. M. and Kosolapov Yu. V.* Analiz stoykosti nekotorykh kodovykh kriptosistem, osnovannyy na razlozhenii kodov v pryamuyu summu [Analysis of the stability of some code cryptosystems based on the decomposition of codes into a direct sum]. Vestn. YuUrGU. Ser. Matem. Modelirovanie i Programmirovanie, 2019, vol. 12, no. 3, pp. 89–101. (in Russian)
22. *Egorova E., Kabatiansky G., Krouk E., and Tavernier C.* A new code-based public-key cryptosystem resistant to quantum computer attacks. J. Phys. Conf. Ser., 2019, vol. 1163, pp. 1–5.
23. *Deundyak V. M., Kosolapov Yu. V., and Maystrenko I. A.* On the decipherment of Sidel'nikov-type cryptosystems. LNCS, 2020, vol. 12087, pp. 20–40.
24. *Kasami T. and Lin S.* On the construction of a class of majority-logic decodable codes. IEEE Trans. Inform. Theory, 1971, vol. IT-17, no. 5, pp. 600–610.
25. *Deundyak V. M., Kosolapov Yu. V., and Lelyuk E. A.* Decoding the tensor product of MLD codes and applications for code cryptosystems. Automatic Control Comput. Sci., 2018, vol. 52, no. 7, pp. 647–657.
26. *Deundyak V. M. and Lelyuk E. A.* A graph-theoretical method for decoding some group MLD-codes. J. Appl. Industr. Math., 2020, vol. 14, no. 2, pp. 265–280.
27. *Randriambololona H.* On products and powers of linear codes under componentwise multiplication. Algorithmic Arithmetic Geometry Coding Theory, 2015, vol. 637, pp. 3–78.
28. *Deundyak V. M. and Kosolapov Yu. V.* On the strength of asymmetric code cryptosystems based on the merging of generating matrices of linear codes. XVI Intern. Symp. Prob. of Redundancy in Information and Control Systems. Russia, 2019, pp. 143–148.
29. *Deundyak V. M. and Kosolapov Yu. V.* O nekotorykh svoystvakh proizvedeniya Shura — Adamara dlya lineynykh kodov i ikh prilozheniyakh [On some properties of the Schur — Hadamard product for linear codes and their applications]. Prikladnaya Diskretnaya Matematika, 2020, no. 50, pp. 72–86.
30. *Sidel'nikov V. M.* Teoriya kodirovaniya [Coding Theory]. Moscow, Fizmatlit Publ., 2008.

31. *Grassl M. and Rotteler M.* Quantum block and convolutional codes from self-orthogonal product codes. Proc. IEEE Int. Symp. Inf. Theory, 2005, pp. 1018–1022.
32. *Henderson H. V. and Searle S. R.* The vec-permutation matrix, the vec operator and Kronecker products: A review. Linear and Multilinear Algebra, 1981, no. 9, pp. 271–288.

УДК 519.719.325 DOI 10.17223/20710410/57/3

РАНДОМИЗИРОВАННЫЙ АНАЛОГ НЕОСПОРИМОЙ ПОДПИСИ ЧАУМА — ВАН-АНТВЕРПЕНА

П. А. Полищук*, А. В. Черемушкин**

*Белорусский государственный университет, г. Минск, Беларусь
**Академия криптографии РФ, г. Москва, Россия

E-mail: poulpvp101@gmail.com, avc238@mail.ru

Предлагается модификация неоспоримой подписи Д. Чаума и Х. ван Антверпена, основанная на группе точек эллиптической кривой. Алгоритм формирования подписи дополнен предварительным этапом рандомизации. Для протоколов проверки подписи и отказа от неё предложено два варианта выполнения. Доказаны теоремы, показывающие, что эти протоколы отвечают своему назначению. Описан способ преобразования неоспоримой подписи в обычную цифровую подпись, проиллюстрированный на примере схемы цифровой подписи Шнорра.

Ключевые слова: *цифровая подпись, неоспоримая подпись, группа точек эллиптической кривой.*

A RANDOMIZED ANALOG OF CHAUM — VAN ANTWERPEN UNDENIABLE SIGNATURE

P. A. Polyschuk*, A. V. Cheremushkin**

*Belarusian State University, Minsk, Belarus,
**Academy of Cryptography of the Russian Federation, Moscow, Russia

We suggest an elliptic curve modification of the undeniable signature introduced by D. Chaum and H. van-Antwerpen. The signature generation algorithm is supplemented with a preliminary stage of randomization. For signature verification and disavowal protocols, two options are offered. Theorems showing that these protocols meet their purpose have been proven. A method for converting an undeniable signature into a regular digital signature is described, illustrated by the Schnorr electronic signature scheme as an example.

Keywords: *digital signature, undeniable signature, elliptic curve.*

Введение

Неоспоримая цифровая подпись, иначе называемая конфиденциальной, в отличие от обычной электронной подписи не может быть проверена без участия сформировавшего её лица. Поэтому она обеспечивает конфиденциальность и частичную анонимность подписавшей стороны. Название «неоспоримая подпись» объясняется тем, что если подписавшая сторона будет обязана участвовать в процедуре проверки подписи, то она не сможет отказаться от факта подписания документа, если подпись является настоящей, но сможет доказать, что это не её подпись, если она не подписывала документ.

Как отмечают авторы схемы, данный вид подписи удобен для компаний, занимающихся удалённым распространением своей продукции. Предположим, некоторая компания желает осуществить удалённую продажу произведённого ею программного продукта. В этом случае в роли текста выступает код программного продукта. Приобретя программные продукты, снабжённые лицензией с неоспоримыми подписями, покупатель после оплаты может получить доказательство её корректности, гарантирующее оригинальность продукта. Если же он не произвёл оплату, то никакими способами не сможет убедиться в достоверности подписи и лицензионности самого программного продукта, возможно, содержащего вредоносный код. С другой стороны, если вредоносный код будет обнаружен, то продавец может доказать, что выданную от его имени лицензию с неоспоримой подписью он никогда не создавал.

Для того чтобы у покупателя осталось подтверждение положительного результата проведённой проверки корректности неоспоримой подписи, она может быть конвертирована в обычную цифровую подпись, которая может быть привязана к имеющейся неоспоримой подписи, а также к идентификационным данным покупателя для того, чтобы независимый арбитр всегда мог убедиться в легитимности подписи и лицензионности продукта.

Исследованию неоспоримых подписей посвящено значительное количество публикаций. Их авторы сосредоточены в основном на распространении схем неоспоримой подписи на различные практические ситуации: (t, n)-пороговые схемы [1, 2], стойкие схемы неоспоримых подписей [3, 4], схема с возможностью конвертирования в обычную цифровую подпись [5, 6], схемы, позволяющие построить протокол честного обмена секретами [7], схемы с открытыми ключами на основе идентификационной информации [8], применение неоспоримых сертификатов для проверки подписей [9], схемы с делегированием права проверки подписи, позволяющие подписывающему лицу заранее назначить верификатора, способного проверить или отвергнуть подпись без участия подписавшего [10, 11], схемы подписи, не требующие сертификатов [12], содержащих указание на автора подписи, групповые неоспоримые подписи и др.

В исходных работах [13, 14] для построения схемы неоспоримой подписи Д. Чаум и Х. ван Антверпен использовали мультипликативную группу конечного поля. В дальнейшем было исследовано множество других вариантов построения схем неоспоримых цифровых подписей на основе различных математических конструкций: на основе RSA [5, 6], схемы с трудоёмким возведением в чётную степень по составному модулю [15], на основе логарифмирования в полной линейной группе над групповым кольцом [16], на основе группы кос [17], на основе операции билинейного спаривания [7, 8], на основе неабелевых групп [18], на основе изогений эллиптических кривых [19], на решётках [20] и др.

В связи с появлением субэкспоненциальных алгоритмов дискретного логарифмирования в целях повышения стойкости большинство вариантов схем цифровых подписей было адаптировано для циклических подгрупп группы точек эллиптической кривой путём замены операции умножения в конечном поле на операцию сложения в группе точек эллиптической кривой. Так, в США, России, Беларуси и др. государствах были приняты новые национальные стандарты цифровых подписей (ECDSA, ГОСТ Р 34.11-2012, СТБ 34.101.45-2013).

Для эллиптических кривых предложено много способов построения различных вариантов неоспоримых подписей. Вместе с тем эллиптические кривые с операцией спаривания составляют относительно небольшой класс эллиптических кривых, к тому же обладающий потенциальной уязвимостью к сведению двумерного представления

к одномерному. Поэтому представляет интерес нахождение способов построения таких подписей на основе групп точек эллиптических кривых без использования операции спаривания. Этой задаче в литературе уделено меньше внимания. Авторам удалось найти только одну публикацию [21], в которой схема Чаума — ван-Антверпена из [14] дословно перенесена с мультипликативной группы конечного поля на случай группы точек эллиптической кривой. Следует заметить, что предложенная в [21] схема обладает тем недостатком, что для неё необходим алгоритм вложения значения произвольной хеш-свёртки открытого текста в циклическую подгруппу группы точек эллиптической кривой, что представляет неудобство с практической точки зрения, ограничивая область применения.

В настоящей работе предложен способ модификации схемы неоспоримой цифровой подписи Чаума — ван Антверпена применительно к циклической подгруппе группы точек эллиптической кривой, не обладающий подобным недостатком. Для решения проблемы соответствия между числовыми значениями и точками этой подгруппы применён приём, решающий эту проблему за счёт введения в алгоритм формирования цифровой подписи предварительного этапа рандомизации алгоритма по аналогии со схемами цифровых подписей семейства Эль-Гамаля и подписи Шнорра, используемыми в национальных стандартах. Предложено два аналогичных варианта протоколов, а также алгоритм преобразования неоспоримой подписи в обычную цифровую подпись, сформированную аналогично схеме подписи Шнорра.

1. Протоколы неоспоримой подписи

Приведём необходимые для дальнейшего изложения сведения. Пусть p — простое, $p > 3$ и $\mathrm{GF}(p)$ — конечное поле из p элементов. Каждому элементу a поля $\mathrm{GF}(p)$ поставим в соответствие его представление в двоичном алфавите вектором длины $\lceil \log_2 p \rceil$, которое будем обозначать $\bar{a}$.

Для модификации схемы подписи Д. Шаума и Х. ван-Антверпена будем использовать такой же класс эллиптических кривых, как и в ГОСТ Р 34.10-2012 [22]. Уравнение эллиптической кривой $E_{a,b}(\mathrm{GF}(p))$ имеет вид

$$y^2 = x^3 + ax + b \pmod p,$$

где $a, b \in \mathrm{GF}(p)$. Точки кривой — это пары (x, y), удовлетворяющие этому уравнению, а также специальная бесконечно удалённая точка $\mathcal{O}$. Для обеспечения условия гладкости, которое позволяет ввести на множестве точек кривой групповую операцию, должно выполняться условие на дискриминант $4a^3 + 27b^2 \neq 0$.

Пусть $m = |E_{a,b}(\mathrm{GF}(p))|$ и $P = (x_P, y_P) \neq \mathcal{O}$ — точка эллиптической кривой, порождающая циклическую подгруппу $\langle P \rangle$ порядка q, где q — большое простое число, делящее порядок группы точек эллиптической кривой $m = qd$. Эллиптическая кривая и точка P должны быть выбраны таким образом, чтобы задача дискретного логарифмирования для элементов циклической группы $\langle P \rangle$ была труднорешаемой. Поэтому далее будем полагать, что эллиптическая кривая и точка P выбраны в соответствии с требованиями ГОСТ Р 34.10-2012. Кроме того, не ограничивая общности, будем полагать, что q^2 не делит m. Это условие не является исключительным, поскольку выполняется автоматически при выборе эллиптической кривой и циклической подгруппы с высокой эффективностью реализации: $q \approx m$ и соответственно $m \ll q^2$. Данные предположения делают невозможным успешное проведение MOV-атаки в задаче дискретного логарифмирования для циклической группы $\langle P \rangle$ большого простого

порядка q, относительного которого кривая обладает достаточно большой степенью вложения (в соответствии с ГОСТ Р 34.10-2012).

Обозначим $V_\infty = \bigcup\limits_{m \geqslant 1} V_m$, где V_m — множество двоичных последовательностей длины m. Пусть также $h : V_\infty \to V_{\lceil \log_2 p \rceil}$ — хеш-функция, удовлетворяющая условию однонаправленности, а также устойчивости к подбору коллизий и второго прообраза.

Ключом подписи является целое число k, $1 < k < q$, а ключом проверки подписи — точка $Q = [k]P = \underbrace{P + \ldots + P}_{k}$, точнее, строка $\bar{x}_Q || \bar{y}_Q$ — конкатенация двоичных векторов $\bar{x}_Q, \bar{y}_Q \in V_{\lceil \log_2 p \rceil}$, соответствующих координатам (x_Q, y_Q) точки Q.

Схема неоспоримой цифровой подписи включает три протокола:

— алгоритм формирования подписи;
— протокол её проверки при участии подписавшего;
— протокол отказа от подписи, т. е. доказательства того факта, что автором подписи данное лицо не является.

Предлагаемые ниже варианты схем неоспоримой подписи являются рандомизированными модификациями неоспориомой подписи Чаума — ван-Антверпена, адаптированными для случая группы точек эллиптической кривой.

1.1. Первый вариант

Пусть A и B обозначают соответственно доказывающую и проверяющую стороны.

1. *Алгоритм формирования подписи*

(1) вычислить хеш-код $h(T)$ сообщения $T \in V_\infty$;
(2) вычислить целое число t, двоичным представлением которого является вектор $h(T)$, и определить $l = t \bmod q$. Если $l = 0$, то определить $l = 1$;
(3) сгенерировать случайное (псевдослучайное) целое число r, $1 < r < q$, $r \neq l$;
(4) вычислить точку $R = [r]P = (x_R, y_R)$ эллиптической кривой $E_{a,b}(\mathrm{GF}(p))$;
(5) вычислить целое число $s = k(l - r) \bmod q$;
(6) вычислить двоичные векторы $\bar{x}_R, \bar{y}_R \in V_{\lceil \log_2 p \rceil}$ и $\bar{s} \in V_{\lceil \log_2 p \rceil}$, соответствующие координатам точки $R = (x_R, y_R)$ и числу s, определить цифровую подпись

$$sig(T) = \bar{x}_R || \bar{y}_R || \bar{s}.$$

Для краткости рассуждений будем полагать, что подписью является пара (R, s), не акцентируя внимание на том, что её в дальнейшем надо преобразовать в двоичную строку.

2. *Протокол проверки подписи*

(0) B: если $R \notin \langle P \rangle$ или $s = 0$, то подпись некорректна;
(1) B: генерирует $e_1, e_2 : 0 \leqslant e_1, e_2 \leqslant q - 1$;
вычисляет $C = [e_1]P + [e_2]R$;
(2) B: если $C = \mathcal{O}$, перейти к (1);
(3) $A \leftarrow B$: C;
(4) A: если $C \notin \langle P \rangle$, завершить выполнение;
(5) $A \to B$: $D = [k]C$;
(6) B: проверяет равенство: $D \stackrel{?}{=} [e_1]Q + [e_2]([l]Q - [s]P)$.

На шаге (0) выполняется проверка включения $R \in \langle P \rangle$, необходимого для корректности формирования подписи, а также для того, чтобы точки C и D принадлежали

циклической группе $\langle P\rangle$. Для этого надо сначала убедиться, что $R \in E_{a,b}(\mathrm{GF}(p))$, а затем — что её порядок равен q, путём проверки равенства $[m/q-1]R = -R$. Случай $s = 0$ недопустим при формировании подписи, так как в этом случае $r = l$, пара $(R, 0)$ не зависит от ключа подписи k и поэтому при любом ключе удовлетворяет протоколу проверки подписи.

Проверка условия $C = \mathcal{O}$ на шаге (2) проводится сравнением точек $[e_1]P \stackrel{?}{=} -[e_2]R$. Заметим, что нахождение такой пары e_1, e_2 приводит к раскрытию ключа k. Из сравнения $e_1 = -e_2 r \bmod q$ можно вычислить r, а затем $k = s(l-r)^{-1} \bmod q$. Вероятность такого события равна $1/q$.

Корректность протокола вытекает из равенств

$$\begin{aligned} D = kC &= [k]([e_1]P + [e_2]R) = [e_1]([k]P) + [e_2]([k]R) = \\ &= [e_1]Q + [e_2]([kl]P - [kl]P + [rk]P) = \\ &= [e_1]Q + [e_2]([lk]P - [(l-r)k]P) = \\ &= [e_1]Q + [e_2]([l]Q - [s]P). \end{aligned}$$

Данный протокол обеспечивает нулевое разглашение информации о ключе подписи, так как проверяющая сторона может для любого ключа проверки подписи Q самостоятельно получить любое число корректных стенограмм (e_1, e_2, C, D) выполнения протокола, задавая произвольные числа e_1, e_2, $0 \leqslant e_1, e_2 < q-1$, и вычисляя точки C и D по формулам

$$C = [e_1]P + [e_2]R, \quad D = [e_1]Q + [e_2]([l]Q - [s]P). \tag{1}$$

При этом ключ подписи k никак не задействован.

С другой стороны, в ходе реализации протокола будут возникать только точки R, C и D из циклической подгруппы $\langle P\rangle$ большого простого порядка q группы точек эллиптической кривой. Вычисленные по формулам (1) пары (C, D) ничего не дают для определения ключа подписи k подписавшей стороны, поскольку нахождение k равносильно решению задачи дискретного логарифмирования $D = [k]C$, или, что равносильно, решению уравнений $[l]Q - [s]P = [k]R$ или $Q = [k]P$. В силу выбора параметров кривой m и q MOV-атака для задачи дискретного логарифмирования в этой группе оказывается неэффективной, а шаг 4 защищает от возможности применения атаки на ключ k типа «малая подгруппа» со стороны B, когда B может отправлять стороне A специально подобранные элементы малых порядков из группы точек эллиптической кривой. Невыполнение проверяемого на шаге (4) равенства доказывает стороне A, что полученная от B точка C имеет порядок q.

Поэтому данный протокол обеспечивает конфиденциальность подписавшей стороны, а также защищает от возможности подделки подписи путём раскрытия ключа подписи.

Замечание 1. В протоколах, относящихся к семейству Эль-Гамаля, и в протоколе Шнорра повторение числа r, которое применяется для рандомизации протокола формирования подписи, приводит к возможности раскрытия ключа. В данном случае это также имеет место. Пусть имеются два сообщения T_1 и T_2, подписанные стороной A, подписи для которых (R, s_1) и (R, s_2) получены при одинаковых значениях r. В этом случае значения k и r можно найти из системы уравнений

$$\begin{cases} s_1 = k(l_1 - r), \\ s_2 = k(l_2 - r). \end{cases}$$

Это событие маловероятно, так как при равномерном распределении значений r его вероятность оценивается как $1/q$.

Можно предложить модификацию алгоритма формирования подписи, где значение $R = f(r, l)P$ зависит от r и l. Однако такое решение не устраняет проблему, так как если у подписей для сообщений T_1 и T_2 совпадут значения $f(r_1, l_1) = f(r_2, l_2)$, то получается аналогичная система:

$$\begin{cases} s_1 = k(l_1 - f(r_1, l_1)), \\ s_2 = k(l_2 - f(r_2, l_2)), \end{cases}$$

при этом вероятность данного события оценивается как $1/q$, точно так же, как и в случае повторения числа r в исходном варианте алгоритма.

3. *Протокол отказа от подписи* состоит в двукратном повторении протокола проверки подписи с последующей проверкой равенства на шаге (9), позволяющего обнаружить нечестное поведение подписывающей стороны:

(0) B: если $R \notin \langle P \rangle$ или $s = 0$, то подпись некорректна;
(1) B: генерирует $e_1, e_2 : 0 \leqslant e_1, e_2 \leqslant q - 1$;
вычисляет $C_1 = [e_1]P + [e_2]R$;
(2) B: если $C_1 = \mathcal{O}$, перейти к (1);
(3) $A \leftarrow B$: C_1;
(4) A: если $C_1 \notin \langle P \rangle$, завершить выполнение;
(5) $A \to B$: $D_1 = [k]C_1$;
(6) B: проверяет равенство: $D_1 \stackrel{?}{=} [e_1]Q + [e_2]([l]Q - [s]P)$;
(7) B: генерирует $f_1, f_2 : 0 \leqslant f_1, f_2 \leqslant q - 1$;
вычисляет $C_2 = [f_1]P + [f_2]R$;
(8) B: если $C_2 = \mathcal{O}$, перейти к (1);
(9) $A \leftarrow B$: C_2;
(10) A: если $C_2 \notin \langle P \rangle$, завершить выполнение;
(11) $A \to B$: $D_2 = [k]C_2$;
(12) B: проверяет равенство: $D_2 \stackrel{?}{=} [f_1]Q + [f_2]([l]Q - [s]P)$;
(13) B: проверяет равенство: $[f_2](D_1 - [e_1]Q) \stackrel{?}{=} [e_2](D_2 - [f_1]Q)$.

Протокол выполняется за 13 шагов: шаги 1–6 и 7–12 повторяют предыдущий протокол, а шаг 13 — это проверка, что A честно выполнял предыдущие действия. Если A не ставил свою подпись и правильно выполнял протокол, то шаги 6 и 12 протокола будут свидетельствовать о нарушении условия проверки подписи, а на шаге 13 будет подтверждено, что участник A корректно выполнял протокол. Тем самым будет с большой вероятностью доказано, что эту подпись поставил не он (см. теорему 2). Если же он обманывает, то для отказа от подписи участник A должен дважды отправить неправильные значения D_1 на шаге 5 и D_2 на шаге 11. Отправка неправильных значений на шагах 5 и 11 протокола со стороны A будет обнаружена проверяющей стороной B при нарушении последнего равенства на шаге 13 (см. теорему 3).

Теоретическое обоснование надежности схемы проводится аналогично рассуждениям в [14].

Теорема 1. Пусть сторона A, обладающая ключом подписи k, $Q = [k]P$, участвует в протоколе проверки подписи (R, s) для сообщения с хеш-свёрткой l, где $R = [r]P$, $r \neq l$, которую она не создавала. Если $s \neq k(l - r) \bmod q$, то вероятность того, что проверяющая сторона B признает корректность подписи, равна $1/q$.

Доказательство. Рассмотрим стенограмму (C, D) выполнения протокола проверки подписи

$$\begin{cases} C = [e_1]P + [e_2]R, \\ D = [k]C \end{cases}$$

относительно неизвестных упорядоченных пар (e_1, e_2).

Проверяющая сторона B признает корректность подписи только в случае, когда выполнено равенство

$$D = [e_1]Q + [e_2]([l]Q - [s]P). \tag{2}$$

Если бы подпись (R, s) была сформирована участником A, то вторая компонента подписи равнялась бы $s_0 = k(l - r)$, откуда

$$[k]C = [k]([e_1]P + [e_2]R) = [e_1]Q + [e_2]([l]Q - [s_0]P).$$

Поэтому равенство (2) можно переписать в виде

$$[e_1]Q + [e_2]([l]Q - [s_0]P) = [e_1]Q + [e_2]([l]Q - [s]P).$$

Оно будет выполнено только в случае

$$[e_2(s - s_0)]P = 0,$$

где $s - s_0 \neq 0$, а значит, $e_2 = 0$. Следовательно, для запроса $C = [e_1]P + [e_2]R$ ответ D на него может быть признан корректным, только если $(e_1, e_2) = (e_1, 0)$. Таким образом, среди q^2 пар (e_1, e_2) число пар, для которых ответ будет признан корректным, равно q. ■

Теорема 2. Пусть сторона A, обладающая ключом подписи k, $Q = [k]P$, участвует в протоколе отказа от подписи (R, s) для сообщения с хеш-сверткой l, где $R = [r]P$, $r \neq l$, которую она не создавала. Если $s \neq k(l - r) \bmod q$, но A и B корректно выполняют протокол отказа от подписи, то равенство

$$[f_2](D_1 - [e_1]Q) = [e_2](D_2 - [f_1]Q) \tag{3}$$

выполнено.

Доказательство. Доказательство основывается на том, что левая и правая части равенства (3) могут быть представлены следующим образом:

$$\begin{cases} [f_2](D_1 - [e_1]Q) & = [f_2]([k]([e_1]P + [e_2]R) - [e_1]Q) = \\ & = [f_2]([e_1k]P + [e_2k]R - [e_1]Q) = [f_2e_2k]R, \\ [e_2](D_2 - [f_1]Q) & = [e_2](([k][f_1]P + [f_2]R) - [f_1]Q) = \\ & = [e_2]([f_1k]P + [f_2k]R - [f_1]Q) = [e_2f_2k]R. \end{cases}$$

Теорема 2 доказана. ■

Теорема 3. Пусть сторона A, обладающая ключом подписи k, $Q = [k]P$, участвует в протоколе отказа от подписи (R, s) для сообщения с хеш-сверткой l, где $R = [r]P$, $r \neq l$, которую она создала, но не хочет в этом сознаваться. Пусть $s = k(l - r)$ и проверяющая сторона B корректно выполняет протокол. Если сторона A на шагах 4 и 10 протокола отправляет сообщения

$$D_1 \neq [e_1]Q + [e_2]([l]Q - [s]P), \quad D_2 \neq [f_1]Q + [f_2]([l]Q - [s]P),$$

то вероятность того, что равенство (3) будет нарушено, равна $1 - 1/q$.

Доказательство. Предположим, что равенство (3) выполнено. Преобразуем его:

$$[e_2]D_2 = [e_2 f_1]Q + [f_2](D_1 - [e_1]Q).$$

Если $e_2 = 0$, то оно имеет вид $[f_2](D_1 - [e_1]Q) = \mathcal{O}$, и поскольку по условию $D_1 \neq [e_1]Q$, то должно быть $f_2 = 0$. Поэтому вероятность того, что последнее равенство выполнено, равна $1/q$.

Если $e_2 \neq 0$, то равенство (3) можно привести к виду

$$D_2 = [f_1]Q + [f_2]D_0, \tag{4}$$

где $D_0 = [e_2^{-1}](D_1 - [e_1]Q)$ определяется действиями участников A и B на шагах 4 и 10 протокола отказа от подписи. Поскольку шаг 5 протокола отказа от подписи подтверждает, что $D_1 \in \langle P\rangle$, то D_0 также принадлежит циклической группе $\langle P\rangle$. Значит, найдётся элемент $l_0 \in \mathrm{GF}(p)$, для которого $D_0 = [l_0]Q - [s]P$. По условию теоремы $D_0 \neq [l]Q - [s]P$, и значит, $l_0 \neq l$. Согласно протоколу проверки подписи, выполнение равенства (4) доказывает, что подпись (R, s) соответствует сообщению со значением хеш-свертки $l_0 \neq l$, для которого выполняется неравенство $s \neq k(l_0 - r)$, причём в силу теоремы 1 вероятность того, что проверяющая сторона B признает корректность подписи (R, s) для сообщения с хеш-свёрткой l, равна $1/q$.

Поэтому вероятность того, что равенство (3) будет выполнено, равна $1/q$. ■

1.2. Второй вариант

1. ***Алгоритм формирования подписи*** полностью повторяет алгоритм первого варианта.

2. ***Протокол проверки подписи***:

(0)	B :	если $R \notin \langle P\rangle$ или $s = 0$, то подпись некорректна;
(1)	B :	генерирует $e_1, e_2 : 0 < e_1, e_2 < q - 1$;
		вычисляет $C = [e_1]([l]Q - [s]P)[+e_2]Q$;
(2)	B :	если $C = \mathcal{O}$, перейти к (1);
(3)	$A \leftarrow B$:	C;
(4)	A :	если $C \notin \langle P\rangle$, завершить выполнение;
(5)	$A \rightarrow B$:	$D = [k^{-1}]C$;
(6)	B :	проверяет равенство: $D \stackrel{?}{=} [e_1]R + [e_2]P$.

Корректность протокола вытекает из равенств

$$\begin{aligned} C &= [e_1]([l]Q - [s]P) + [e_2]Q = \\ &= [e_1]([l]Q - [(l-r)k]P) + [e_2 k]P = \\ &= [e_1]([l]Q - [l]Q + [rk]P) + [e_2 k]P = \\ &= [k]([e_1]R + [e_2]P) = [k]D. \end{aligned}$$

То, что данный протокол также обеспечивает нулевое разглашение информации о ключе подписи и конфиденциальность подписавшей стороны, показывается полностью аналогично.

3. *Протокол отказа от подписи*:

(0) B: если $R \notin \langle P \rangle$ или $s = 0$, то подпись некорректна;
(1) B: генерирует $e_1, e_2 : 0 < e_1, e_2 < q - 1$;
вычисляет $C_1 = [e_1]([l]Q - [s]P)[+e_2]Q$;
(2) B: если $C_1 = \mathcal{O}$, перейти к (1);
(3) $A \leftarrow B$: C_1;
(4) A: если $C_1 \notin \langle P \rangle$, завершить выполнение;
(5) $A \rightarrow B$: $D_1 = k^{-1}C_1$;
(6) B: проверяет равенство: $D_1 \stackrel{?}{=} [e_1]R + [e_2]P$;
(7) B: генерирует $f_1, f_2 : 0 < f_1, f_2 < q - 1$;
вычисляет $C_2 = [f_1]([l]Q - [s]P)[+f_2]Q$;
(8) B: если $C_2 = \mathcal{O}$, перейти к (1);
(9) $A \leftarrow B$: C_2;
(10) A: если $C_2 \notin \langle P \rangle$, завершить выполнение;
(11) $A \rightarrow B$: $D_1 = [k^{-1}]C_1$;
(12) B: проверяет равенство: $D_2 \stackrel{?}{=} [f_1]R + [f_2]P$;
(13) B: $M_1 = D_1 - ([e_1]R + [e_2]P)$,
(14) B: $M_2 = D_2 - ([f_1]R + [f_2]P)$,
(15) B: проверяет равенство: $[f_1]M_1 = [e_1]M_2$.

Теоретическое обоснование надёжности схемы повторяет рассуждения для первого варианта.

2. Преобразования в обычную цифровую подпись

Пусть имеется уже сформированная неоспоримая подпись (R, s) для сообщения T, полученная для ключевой пары (k, Q). Предлагаемый способ является по сути не конвертацией, а заменой неоспоримой подписи на обычную цифровую подпись, полученную при том же ключе подписи k с учётом зависимости от ранее сформированной неоспоримой подписи. Рассмотрим вариант преобразования, использующий схему цифровой подписи, аналогичную схеме подписи Шнорра [23].

Для преобразования неоспоримой подписи (R, s) в обычную владелец ключа подписи вычисляет пару (x, s') в соответствии со следующим алгоритмом:

***Алгоритм формирования цифровой подписи*:**

(1) выбрать случайный элемент r', $1 < r' < q$;
(2) вычислить точку эллиптической кривой $R' = [r']P = (x_{R'}, y_{R'})$;
(3) вычислить точку $Q' = [l]Q - [s]P = (x_{Q'}, y_{Q'})$;
(4) вычислить целое число t', двоичным представлением которого является вектор $h(T||\bar{x}_{Q'}||\bar{y}_{Q'}||\bar{x}_{R'}||\bar{y}_{R'})$, и определить $l' = t' \bmod q$. Если $l' = 0$, то определить $l' = 1$;
(5) вычислить $s' = kl' - r' \bmod q$.

Покажем, что $\bar{l}'||\bar{s}'$ является обычной цифровой подписью для сообщения T (для удобства будем, как и выше, использовать запись (l', s')).

Имея неоспоримую подпись (R, s), цифровую подпись (l', s') и открытый ключ Q, любой проверяющий B может осуществить проверку подписи в соответствии со следующим алгоритмом:

Алгоритм проверки цифровой подписи:

(1) вычислить $Y = lQ - sP = (x_Y, y_Y)$;
(2) вычислить $Z = l'Q - s'P = (x_Z, y_Z)$;
(3) вычислить целое число w, двоичным представлением которого является вектор $h(T||\bar{x}_Y||\bar{y}_Y||\bar{x}_Z||\bar{y}_Z)$, и определить $w' = w \bmod q$. Если $w' = 0$, то определить $w' = 1$;
(4) проверить соответствие $l' \stackrel{?}{\leftrightarrow} w$; если оно выполнено, то подпись (l', s') принимается и признается корректной; в противном случае отвергается.

Корректность алгоритма.
Из равенств $s = k(l - r)$ и $s' = kl' - r'$ следует

$$[s]P = [k(l-r)]P = [l]Q - [r]Q,$$
$$[s']P = [kl']P - [r']P = [l']Q - R',$$

а значит, должны выполняться равенства

$$Y = [l]Q - [s]P = Q',$$
$$Z = [l']Q - [s']P = R'.$$

Таким образом, $w' = l'$.

В случае, когда автор неоспоримой подписи, занимающийся удалённым распространением своей продукции, после проведённой проверки неоспоримой подписи пожелает осуществить привязку сформированной цифровой подписи к идентификационным данным ID покупателя, он может внести эти данные в хеш-функцию

$$h(T||x_{rQ}||y_{rQ}||x_{R'}||y_{R'}||ID),$$

тем самым подтверждая положительный результат проверки неоспоримой подписи и лицензионность приобретённого покупателем программного продукта.

Авторы выражают благодарность С. В. Агиевичу за многочисленные полезные замечания и исправления.

ЛИТЕРАТУРА

1. *Harn L. and Yang S.* Group-oriented undeniable signature schemes without the assistance of a mutually trusted party // LNCS. 1993. V. 718. P. 133–142.
2. *Wang G. and Qing S.* A threshold undeniable signature scheme without a trusted party // J. Software. 2002. V. 13. No. 9. P. 1757–1764.
3. *Chaum D., van Heijst E., and Pfitzmann B.* Cryptographically strong undeniable signatures, unconditionally secure for the signer // LNCS. 1992. V. 576. P. 470–484.
4. *Zhu H.* Universal Undeniable Signatures. ePrint Archive. `eprint.iacr.org/2004/005`.
5. *Boyar J., Chaum D., Damgard I., and Pedersen T.* Convertible undeniable signatures // LNCS. 1991. V. 537. P. 189–205.
6. *Gennaro R., Krawczyk H., and Rabin T.* RSA-based undeniable signature // LNCS. 1997. V. 1294. P. 132–148.
7. *Horng S.-J., Tzeng S.-F., Fan P., et al.* Secure convertible undeniable signature scheme using extended Euclidean algorithm without random oracles // KSII Trans. Internet Inform. Systems. 2013. V. 7. No. 6. `http://dx.doi.org/10.3837/tiis.2013.06.010`.
8. *Libert Q. and Quisquater J.-J.* Identity Based Undeniable Signatures. ePrint Archive. `eprint.iacr.org/2003/206`.

9. *Gennaro R., Krawczyk Y. M., and Rabin T. D.* Udeniable Certificates for Digital Signature Verification. United States Patent No. US 6292897 B1, Sep. 18, 2001.
10. *Chaum D.* Designated confirmer signatures // LNCS. 1994. V. 950. P. 86–91.
11. *Okamoto T.* Designated confirmer signatures and public-key encryption are equivalent // LNCS. 1994. V. 839. P. 61–74.
12. *Duan S.* Certificateless undeniable signature scheme // Inform. Sci. 2008. V. 178. Iss. 3. P. 742–755.
13. *Chaum D.* Zero-knowledge undeniable signatures // LNCS. 1991. V. 473. P. 458–464.
14. *Chaum D. and Van-Antwerpen H.* Undeniable signature // LNCS. 1990. V. 435. P. 212–216.
15. *Mao W.* New Zero-knowledge Undeniable Signatures — Forgery of Signature Equivalent to Factorisation. ePrint Archive. `eprint.iacr.org/2001-011`.
16. *Pandey A. and Gupta I.* A new undeniable signature scheme on general linear group over group ring // J. Discrete Math. Sci. Cryptography. 2020. P. 1–13. `https://doi.org/10.1080/09720529.2020.1744814`.
17. *Thomas T. and Lal A. K.* Undeniable Signature Schemes Using Braid Groups. `https://arxiv.org/abs/cs/0601049`.
18. *Thomas T. and Lal A. K.* A zero-knowledge undeniable signature scheme in non-Abelian group setting // Intern. J. Network Security. 2008. V. 6. No. 3. P. 265–269.
19. *Jao J. and Soukharev V.* Isogeny-based quantum-resistant undeniable signatures // LNSC. 2014. V. 8772. P. 160–179.
20. *Tian M. and Huang L.* Efficient Identity-Based Signature from Lattices. `https://hal.inria.fr/hal-01370379`. 2016.
21. *Александрова Е. Б., Шкоркина Е. Н.* Применение неоспоримой подписи на эллиптических кривых для верификации сервера при аутсорс-вычислениях // Проблемы информационной безопасности. Компьютерные системы. СПб.: Изд-во СПбТУ, 2018. С. 97–101.
22. ГОСТ Р 34.10-2012. Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи. `https://docs.cntd.ru/document/1200095034`.
23. *Schnorr C. P.* Efficient identification and signature for smart cards // LNCS. 1990. V. 435. P. 235–251.

REFERENCES

1. *Harn L. and Yang S.* Group-oriented undeniable signature schemes without the assistance of a mutually trusted party. LNCS, 1993, vol. 718, pp. 133–142.
2. *Wang G. and Qing S.* A threshold undeniable signature scheme without a trusted party. J. Software, 2002, vol. 13, no. 9, pp. 1757–1764.
3. *Chaum D., van Heijst E., and Pfitzmann B.* Cryptographically strong undeniable signatures, unconditionally secure for the signer. LNCS, 1992, vol. 576, pp. 470–484.
4. *Zhu H.* Universal Undeniable Signatures. ePrint Archive, `eprint.iacr.org/2004/005`.
5. *Boyar J., Chaum D., Damgard I., and Pedersen T.* Convertible undeniable signatures. LNCS, 1991, vol. 537, pp. 189–205.
6. *Gennaro R., Krawczyk H., and Rabin T.* RSA-based undeniable signature. LNCS, 1997, vol. 1294, pp. 132–148.
7. *Horng S.-J., Tzeng S.-F., Fan P., et al.* Secure convertible undeniable signature scheme using extended Euclidean algorithm without random oracles. KSII Trans. Internet Inform. Systems, 2013, vol. 7, no. 6, `http://dx.doi.org/10.3837/tiis.2013.06.010`.
8. *Libert Q. and Quisquater J.-J.* Identity Based Undeniable Signatures. ePrint Archive. `eprint.iacr.org/2003/206`.

9. *Gennaro R., Krawczyk Y. M., and Rabin T. D.* Udeniable Certificates for Digital Signature Verification. United States Patent No. US 6292897 B1, Sep. 18, 2001.
10. *Chaum D.* Designated confirmer signatures. LNCS, 1994, vol. 950, pp. 86–91.
11. *Okamoto T.* Designated confirmer signatures and public-key encryption are equivalent. LNCS, 1994, vol. 839, pp. 61–74.
12. *Duan S.* Certificateless undeniable signature scheme. Inform. Sci., 2008, vol. 178, iss. 3, pp. 742–755.
13. *Chaum D.* Zero-knowledge undeniable signatures. LNCS, 1991, vol. 473, pp. 458–464.
14. *Chaum D. and Van-Antwerpen H.* Undeniable signature. LNCS, 1990, vol. 435, pp. 212–216.
15. *Mao W.* New Zero-knowledge Undeniable Signatures — Forgery of Signature Equivalent to Factorisation. ePrint Archive. `eprint.iacr.org/2001-011`.
16. *Pandey A. and Gupta I.* A new undeniable signature scheme on general linear group over group ring. J. Discrete Math. Sci. Cryptography, 2020, pp. 1–13, `https://doi.org/10.1080/09720529.2020.1744814`.
17. *Thomas T. and Lal A. K.* Undeniable Signature Schemes Using Braid Groups. `https://arxiv.org/abs/cs/0601049`.
18. *Thomas T. and Lal A. K.* A zero-knowledge undeniable signature scheme in non-Abelian group setting. Intern. J. Network Security, 2008, vol. 6, no. 3, pp. 265–269.
19. *Jao J. and Soukharev V.* Isogeny-based quantum-resistant undeniable signatures. LNSC, 2014, vol. 8772, pp. 160–179.
20. *Tian M. and Huang L.* Efficient Identity-Based Signature from Lattices. `https://hal.inria.fr/hal-01370379`. 2016.
21. *Aleksandrova E. B. and Shkorkina E. N.* Primenenie neosporimoy podpisi na ellipticheskikh krivykh dlya verifikatsii servera pri autsors-vychisleniyakh [Elliptic curve undeniable signature for server verification in outsource computations]. Problemy Informatsionnoy Bezopasnosti. Komp'yuternye Sistemy. SPbTU Publ., 2018, pp. 97–101. (in Russian)
22. GOST R 34.10-2012. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Protsessy formirovaniya i proverki elektronnoy tsifrovoy podpisi [Informatiom technology. Cryptographic protection information. Electronic digital signature process]. `https://docs.cntd.ru/document/1200095034`. (in Russian)
23. *Schnorr C. P.* Efficient identification and signature for smart cards. LNCS, 1990, vol. 435, pp. 235–251.

УДК 519.719.2+512.542.74 DOI 10.17223/20710410/57/4

КРИПТОГРАФИЧЕСКИЕ СЛАБОСТИ АЛГОРИТМОВ ТИПА «ГИПЕРКУБ»

Д. И. Трифонов

Академия криптографии Российской Федерации, г. Москва, Россия

E-mail: d.arlekino@gmail.com

Рассмотрен класс блочных криптографических XSLP-алгоритмов, называемых «гиперкуб». Для алгоритмов данного класса получены оценки показателя рассеивания линейной среды для любого числа итераций. Показано, что при выборе преобразования P с использованием обобщённых графов де Брейна для рассматриваемых алгоритмов может не наступать лавинный эффект, вследствие чего ключ шифрования может быть определён с трудоёмкостью, существенной меньшей трудоёмкости тотального опробования ключей.

Ключевые слова: *XSLP-шифр, криптоанализ, линейный метод, показатель рассеивания, «гиперкуб».*

FLAWS OF HYPERCUBE-LIKE CIPHERS

D. I. Trifonov

Academy of Cryptograhy of Russian Federation, Moscow, Russia

A class of block XSLP cryptographic algorithms called “hypercube” is considered. These algorithms have a block size $n = n' \cdot m = n' \cdot m' \cdot k$ bits. A hypercube-like algorithm is an iterative block algorithm consisted of four main operations: (1) key addition (by XOR), (2) n'-bit S-box application, (3) block-diagonal diffusion matrix $\mathrm{diag}(A_1, \ldots, A_k)$, $A_i \in \mathrm{GF}(2)_{n'm',n'm'}$, multiplication with diffusion degree ρ, and (4) permutation. The main results are the following: 1) the idea of constructing linear correlations and probabilities of distribution of differences, determined by hypercube-like algorithms, has been described; 2) the linear environment propagation index for any number of rounds has been evaluated; 3) the relevance of branch number $\theta(r)$ for differential trails probability and correlation of linear trails for any $r \in \mathbb{N}$, $r \geqslant 2$, rounds has been formally represented; 4) for hypercube-like algorithms, it is shown that when constructing a P-transform using de Bruijn graphs, the avalanche effect may not occur, which means that the (time) complexity of determining the encryption key will be much less than the exhaustive key search (time) complexity. Let $n = n'(m')^d$ and $\mathrm{P} : V_n \to V_n$ affect $a = (a_0, \ldots, a_{m-1}) \in V_n$, $a_i \in V_{n'}$, as follows. Numbers $l \in \{0, \ldots, (m')^d - 1\}$ of $a_l \in V_{n'}$ in $a \in V_n$ are considered as $l = j_0 + j_1 m' + \ldots + j_{d-1}(m')^{d-1}$, $j_t = 0, \ldots, m' - 1$, $t = 0, \ldots, d - 1$. Let the mapping P is defined as $\mathrm{P}(a) = \mathrm{P}(a_0, \ldots, a_{(m')^d-1}) = (a_{\tau(0)}, \ldots, a_{\tau((m')^d-1)})$, $\tau \in S_{(m')^d}$, $\tau(l) = \tau(j_0, \ldots, j_{d-1})$, $l = 1, \ldots, (m')^d$. In the case $d = 3$ it is obtained that if P is rotation of hypercube, i.e., $\tau(j_0, j_1, j_2) = (j_1, j_2, j_0)$, then $\theta(r) \leqslant t(r)$, $t(1) = m'$, $t(r) = ((m')^2 + m')\,[r/2] + m'(r \bmod 2)$, $r \geqslant 2$. In the case $\tau(i_0, i_1, i_2) = (i_0, i_1 + i_0 \bmod m', i_2 + i_0 \bmod m')$ we obtain $\theta(r) = \theta(r - 4) + \rho^2$, $\theta(1) = 1$, $\theta(2) = \rho$, $\theta(3) = 2\rho - 1$, $r \in \mathbb{N}$, $r > 4$.

Keywords: *XSLP-ciphers, cryptoanalysis, linear method, branch numbers, hypercube structure.*

Введение

В [1] отмечено, что благодаря появлению линейного метода криптографического анализа в отечественной и зарубежной практике сформировался уже устоявшийся подход к построению блочных шифров: в качестве нелинейных необходимо использовать преобразования, обеспечивающие необходимые характеристики относительно линейного и разностного методов, а в качестве линейных преобразований — преобразования с достаточно высокими рассеивающими свойствами. По такому принципу построено много криптографических алгоритмов, среди которых можно отметить российский стандарт блочного шифрования «Кузнечик» [2].

Вместе с тем выбор криптографически качественных примитивов далеко не всегда означает построение блочного алгоритма с высокими криптографическими свойствами (см., например, [1, 3]). В данной работе показано, что для XSLP-алгоритмов с использованием в качестве преобразования L блочно-диагональной матрицы с максимально рассеивающими матрицами на диагонали, а в качестве преобразования P коммутации, построенной с использованием обобщённых графов де Брейна, существуют эффективные схемы согласования, позволяющие строить эффективные методы определения ключа.

1. Определения и обозначения

В работе используются следующие определения и обозначения [4, 5]:

— $\mathbb{Z}_n$ — кольцо вычетов по модулю n;
— $\mathbb{F}_{m,n}$ — множество всех матриц из m строк и n столбцов с элементами из поля $\mathbb{F}$; в случае поля $\mathbb{F}_2$ применяется обозначение $(\mathbb{F}_2)_{n,n} = M_n$;
— $S(A)$ — симметрическая группа подстановок на непустом множестве A; в случае $A = \{0, 1, \ldots, m-1\}$ применяется обозначение S_m.

Определение 1. Под наступлением лавинного эффекта на t-й итерации блочного криптографического алгоритма будем понимать существенную зависимость всех бит обрабатываемого информационного блока на t-й итерации от всех входных бит, $t \in \mathbb{N}$.

Определение 2 [6]. Пусть $m > 1$, $r > 1$ — натуральные числа, $n = m^r$. Множество вершин графа де Брейна порядка r на $n = m^r$ вершинах составляют все r-мерные векторы вида $(\nu_0, \nu_1, \ldots, \nu_{r-1})$, $\nu_i \in \mathbb{Z}_m$, $i = 0, \ldots, r-1$. Из вершины $(\nu_0, \nu_1, \ldots, \nu_{r-1})$ исходит m дуг в вершины $(\varepsilon, \nu_0, \nu_1, \ldots, \nu_{r-2})$; в неё также входит m дуг из вершин $(\nu_1, \ldots, \nu_{r-1}, \varepsilon)$, где ε — любой элемент $\mathbb{Z}_m$.

Замечание 1. В условиях определения 2 в графе де Брейна петли имеются в точности на m вершинах $(\nu, \ldots, \nu) \in \mathbb{Z}_m$.

Определение 3 [6]. Пусть n, r — натуральные числа, $n > 1$, $r \geqslant 1$. Ориентированный граф на n вершинах называется ∂-графом (или обобщённым графом де Брейна) порядка r, если для любой пары его вершин существует единственный направленный путь длины r из одной вершины в другую.

Замечание 2. Если в ∂-графе Γ порядка r изменить направление по каждой его дуге на противоположное, то получающийся граф $\overline{\Gamma}$ также будет ∂-графом порядка r.

2. Постановка задачи

В [7, п. 9.7.1] для построения блочных алгоритмов с высокими криптографическими характеристиками предлагается использовать структуру «гиперкуб» (d-мерный куб). Каждый алгоритм, имеющий структуру «гиперкуб», по сути представляет собой XSLP-алгоритм со следующими параметрами: n — размер блока в битах; d — размерность гиперкуба; m — число нелинейных преобразований (подстановок) на n'-подвекторах; m' — размер стороны гиперкуба; $n = n' \cdot m = n' \cdot m' \cdot k$; r — число итераций. Будем полагать $k = (m')^{d-1}$, то есть $n = n'(m')^d$, $m = m' \cdot k = (m')^d$.

Обрабатываемый данным алгоритмом информационный блок представляется следующим образом. Компоненты вектора $(a_1, \ldots, a_m) \in V_n$, $a_i \in V_{n'}$, $i = 1, \ldots, m$, записываются в ячейки d-мерного куба последовательно по всем граням. Так, в случае $d = 2$ запись происходит в таблицу размера $m' \times m'$.

При реализации алгоритмов из предлагаемого семейства на каждой итерации используются следующие преобразования:

Наложение ключа $\mathrm{X}[K] : V_n \to V_n$, где $\mathrm{X}[K](a) = K \oplus a$; $a, K \in V_n$.

Нелинейное преобразование $\mathrm{S} : V_n \to V_n$, $\mathrm{S}(a) = \mathrm{S}(a_0, \ldots, a_{m-1}) = (\pi_0(a_0), \ldots, \pi_{m-1}(a_{m-1}))$, $a_i \in V_{n'}$, $\pi_i \in S(V_{n'})$, $i = 0, \ldots, m-1$.

Рассеивающее преобразование $\mathrm{L} : V_n \to V_n$,

$$\mathrm{L}(a) = (a_0, \ldots, a_{m'-1}, a_{m'}, \ldots, a_{2m'-1}, \ldots a_{m'k-1}) = (b_0, \ldots, b_{m'-1}, b_{m'}, \ldots, b_{2m'-1}, \ldots b_{m'k-1}),$$

$a_i \in V_{n'}$, $i = 1, \ldots, m' \cdot k$ при этом $(b_{m'(i-1)}, \ldots, b_{im'-1}) = (a_{m'(i-1)}, \ldots, a_{im'-1})A_i$, где $A_i \in \mathrm{GF}(2)_{n'm',n'm'}$, $i = 1, \ldots, k$; $a_j, b_j \in V_{n'}$, $j = 0, \ldots, m'k-1$; $a_i \in V_{n' \cdot m'}$, $(a_1 \| \ldots \| a_k) \in V_n$;

Коммутация $\mathrm{P} : V_n \to V_n$, действует на вектор $a = (a_0, \ldots, a_{m-1}) \in V_n$, $a_i \in V_{n'}$, следующим образом. При $m = (m')^d$ представим номер $l \in \{0, \ldots, (m')^d - 1\}$ подвектора $a_l \in V_{n'}$ вектора $a \in V_n$ в виде m'-ичной записи $l = j_0 + j_1 m' + \ldots + j_{d-1}(m')^{d-1}$, $j_t = 0, \ldots, m'-1$, $t = 0, \ldots, d-1$, которой будем ставить во взаимно-однозначное соответствие набор $(j_0, j_1, \ldots, j_{d-1})$. Тогда $\mathrm{P}(a) = \mathrm{P}(a_0, \ldots, a_{(m')^d-1}) = (a_{\tau(0)}, \ldots, a_{\tau((m')^d-1)})$, где $\tau \in S_{(m')^d}$. $\tau(l) = \tau(j_0, \ldots, j_{d-1}) = (j_1, j_2, \ldots, j_0)$, $l = 1, \ldots, (m')^d$.

В рассматриваемой конструкции XSLP-алгоритма ключевую роль играет выбор перестановки на номерах подвекторов (коммутации) τ. В [7, п. 9.7.1] предлагается использовать коммутацию вида

$$(j_0, \ldots, j_{d-1}) \xrightarrow{\tau} (j_1, j_2, \ldots, j_0). \tag{1}$$

Такой выбор явился следствием глубокой проработанности вопроса использования данной коммутации в SP-сетях [8]. В [8] также указывается на то, что в SP-сетях степень рассеивания для преобразования P определяется по лавинному эффекту. Отметим, что для SP-сети преобразование P задаётся коммутацией на номерах координат вектора. Согласно [8, теорема 3], оптимальными в этом отношении являются подстановки $\mathrm{P} \in S_n$, для которых переходы $(i, \mathrm{P}(i))$, $i = 1, \ldots, n$, содержатся среди дуг обобщённых графов де Брейна [6].

Во введённых определениях, исходя из [8, теорема 3], с точки зрения качественного представления о рассеивании наиболее предпочтительными при $n = m^r$ являются коммутации $\mathrm{P} \in S_n$, для которых $\Gamma(P)$ или $\Gamma(P^{-1})$ является ∂-графом порядка r, в частности графом де Брейна, $r \in \mathbb{N}$. Данные рассуждения также переносятся на рассматриваемый в настоящей работе случай $n = n'(m')^d$, когда преобразование P используется для рассеивания подвекторов длины n'. В этом случае дугами графа де

Брейна Γ_0 на множестве $\{0, 1, \ldots, (m')^d - 1\}$ являются пары вершин

$$\big((i_0, i_1, \ldots, i_{d-1}), (i_1, \ldots, i_{d-1}, j)\big) : i_0, i_1, \ldots, i_{d-1}, j \in \{0, 1, \ldots, m' - 1\}.$$

В «канонической» SP-сети [8] в качестве коммутации $P_0 \in S_{m^r}$, для которой $\Gamma(P_0^{-1}) = \overline{\Gamma}_0$, рекомендуется брать подстановку P_0, задаваемую коммутацией вида (1):

$$P_0(i) = P_0(i_0, \ldots, i_{r-1}) = (i_1, \ldots, i_{r-1}, i_0).$$

При $d = 2$ коммутация τ вида (1) хорошо зарекомендовала себя при использовании в блочных криптографических алгоритмах и хэш-функциях (выделим в первую очередь алгоритмы AES [9] и Стрибог [10]). Вместе с тем далее показано, что при $d = 3$ коммутация τ вида (1) обладает недостаточными рассеивающими свойствами. Данное обстоятельство приводит к отсутствию лавинного эффекта и как следствие — к возможности построения эффективных схем согласования для определения ключа алгоритма.

Замечание 3. Коммутация P вида (1) для XSLP-алгоритма со структурой «гиперкуб» соответствует с точностью до переобозначений повороту d-мерного куба относительно одной из осей. Согласно [7, с. 145], при $d > 1$ данное преобразование является «оптимальным» в смысле обеспечения существенной зависимости бит промежуточного блока от входных данных.

Пример 1. Пример преобразования P в случаях $d = 2$ и 3 представлен на рис. 1.

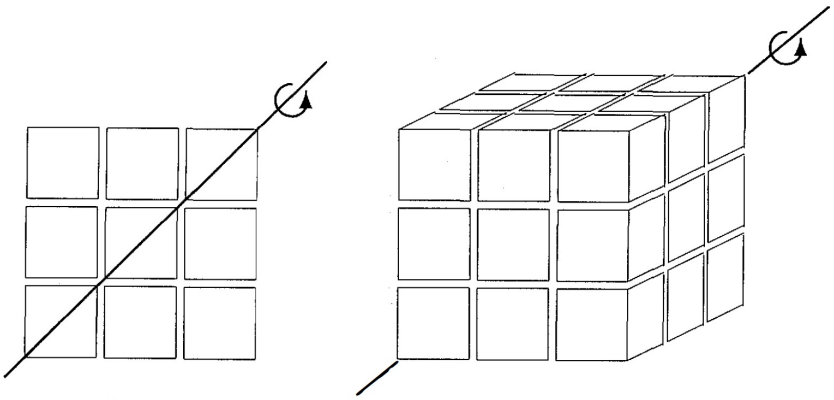

Рис. 1. Действие коммутации P в случаях $d = 2$ и 3

В случае $d = 3$ нижняя строка подстановки τ имеет следующий вид:

$$1, 10, 19, 2, 11, 20, 3, 12, 21, 4, 13, 22, 5, 14, 23, 6, 15, 24, 7, 16, 25, 8, 17, 26, 9, 18, 27$$

Графически перестановка вида (1) номеров подвекторов, записанных в 3-мерный куб, приведена на рис. 2. Для большей наглядности вместо подстановки τ приведена подстановка τ^{-1}, нижняя строка которой имеет следующий вид:

$$1, 4, 7, 10, 13, 16, 19, 22, 25, 2, 5, 8, 11, 14, 17, 20, 23, 26, 3, 6, 9, 12, 15, 18, 21, 24, 27.$$

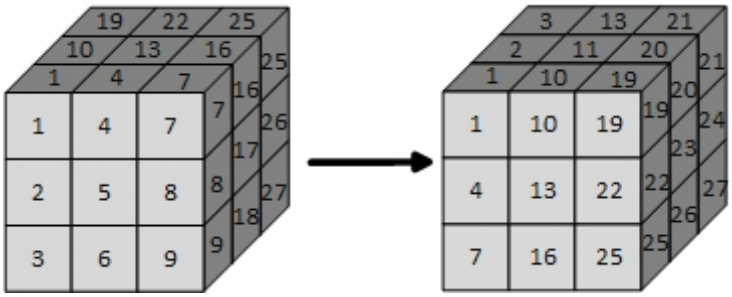

Рис. 2. Коммутация τ^{-1} при $d = 3$

3. Примеры коммутаций

Приведём примеры коммутаций τ вида (1), используемых в различных криптографических алгоритмах. На некоторые из рассмотренных примеров автору указали А. В. Анашкин и Ф. М. Малышев.

Алгоритм LOKI91 [11, 12]. Данный алгоритм схож по своему строению с алгоритмом DES [13]. В алгоритме LOKI91 используется перестановка $t \in S_{32}$, для которой нижняя строка подстановки t^{-1} имеет следующий вид:

$$31, 23, 15, 7;\ 30, 22, 14, 6;\ 29, 21, 13, 5;\ 28, 20, 12, 4;$$
$$27, 19, 11, 3;\ 26, 18, 10, 2;\ 25, 17, 9, 1;\ 24, 16, 8, 0.$$

Естественней здесь выглядит подстановка $\sigma \in S_{32}$, $\sigma(i) = t(32 - i)$, $i = 0, 1, \ldots, 31$. Для задания подстановки σ верхняя строка разбивается на четыре восьмёрки подряд идущих чисел, а нижняя — на восемь четвёрок подряд идущих чисел. Тогда верхняя строка параметризуется парами (i, j), а нижняя — парами (j, i), $i = 0, \ldots, 3$, $j = 0, \ldots, 7$. В этом случае подстановка σ определяется правилом $\sigma(i, j) = (j, i)$.

Алгоритм SAFER K-64 [14, 15]. Данный алгоритм представляет собой SP-сеть. В алгоритме используется подстановка $t \in S_8$, нижняя строка которой имеет вид $(0, 4, 1, 5, 2, 6, 3, 7)$. Подстановка t может быть задана как подстановка σ в алгоритме LOKI91, а именно: $t(i, j) = [j, i]$, где $(i, j) = 2i + j$, $[j, i] = 4j + i$, $i = 0, \ldots, 3$, $j = 0, 1$. Если $i \in \{0, \ldots, 7\}$ представлять в двоичном виде $i = (i_0, i_1, i_2) = i_0 \cdot 2^2 + i_1 \cdot 2 + i_2$, то $i = (i_0, i_1)$, $j = i_2$ и $t(i_0, i_1, i_2) = (i_2, i_0, i_1)$.

Нетрудно убедиться, что ориентированный граф Γ на множестве вершин $\{0, \ldots, 7\}$ с дугами $i \to t^{-1}(i)$, $i \to t^{-1}(i) \pm 1$, где знак «+» берётся при чётном $t^{-1}(i)$ и знак «−» иначе, является графом де Брейна.

Далее приведём два примера криптографических алгоритмов с длиной блока $n = n'(m')^2$, в которых коммутация τ рассматривается в несколько обобщённом виде, а именно:

$$(j_0, \ldots, j_{d-1}) \xrightarrow{\tau} (j_1, j_2, \ldots, T(j_0, j_1, \ldots, j_{d-1})), \tag{2}$$

где $T : \mathbb{Z}_{m'}^d \to \mathbb{Z}_{m'}$ — некоторая функция.

Алгоритм AES [9]. В криптографическом алгоритме AES с длиной блока $128 = 8 \cdot 4^2$ бит и длиной ключа 256 бит в качестве рассеивающего преобразования P (ShiftRow Transformation) используется следующая подстановка:

$$\mathrm{P} = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 0 & 13 & 10 & 7 & 4 & 1 & 14 & 11 & 8 & 5 & 2 & 15 & 12 & 9 & 6 & 3 \end{pmatrix}.$$

Тогда, представив числа из множества $\{0,\dots,15\}$ в виде $i = i_0 + i_1 \cdot 4$, $i_0, i_1 = 0,1,2,3$, получаем, что преобразование P имеет вид

$$(i_0, i_1) \to (i_0,\ (i_1 - i_0) \bmod 4).$$

Строго говоря, данное преобразование не является преобразованием вида (2). Для того чтобы привести его к виду (2), произведём перенумерацию подвекторов длины n', которые являются входами для соответствующих нелинейных преобразований. Перенумерация заключается в переходе к новым младшим цифрам в номере с сохранением старших. Другими словами, вместо (i_0, i_1) рассматриваются (j_0, j_1), $j_1 = i_1$, $j_0 = i_1 + i_0$. Тогда номера подвекторов длины n' изменятся по следующей подстановке:

$$\tau = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 0 & 1 & 2 & 3 & 5 & 6 & 7 & 4 & 10 & 11 & 8 & 9 & 15 & 12 & 13 & 14 \end{pmatrix},$$

а коммутация AES примет вид

$$\mathrm{P}(i_0, i_1) = \tau^{-1}(i_1, T_{i_1}(i_0)),$$

где

$$T_0 = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 0 & 3 & 2 & 1 \end{pmatrix},\ T_1 = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 2 \end{pmatrix},\ T_2 = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 2 & 1 & 0 & 3 \end{pmatrix},\ T_3 = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 3 & 2 & 1 & 0 \end{pmatrix}.$$

Алгоритм LUCIFER [16]. Здесь рассматривается изменённая версия алгоритма Люцифер (иногда обозначается Lucifer (v1)), которая была выдвинута компанией IBM в качестве национального стандарта США. Алгоритм представляет собой 8-раундовую SP-сеть и имеет длину блока 128 бит и длину ключа 256 бит. При длине блока $n = (m')^2$ в качестве подстановки $t \in S_{(m')^2}$ используется подстановка $t(i_0, i_1) = (i_1, w(i_1) + i_0)$, $i_0, i_1 \in \{0,\dots,m'-1\}$, сложение производится по модулю m' и $w \in S_{m'}$.

Перенумерацией, рассмотренной для алгоритма AES, можно прийти к аналогичному виду коммутации:

$$(j_0, j_1) \to (j_1, 2j_1 - j_0).$$

4. Показатель рассеивания

Согласно [17, 8, 18], линейный метод нацелен на получение вероятностного линейного соотношения шифрпреобразования

$$F : V_N \times V_K \to V_M,\quad V_N \times V_K \ni (a, z) \mapsto b = F(a, z) \in V_M \tag{3}$$

общего вида, задаваемого функциональной схемой, которая представляется последовательностью выполняемых в определённом порядке линейных и нелинейных преобразований. Для блочного шифрования в (3) $M = N$, $a \in V_N$ и $b \in V_N$ — соответствующие блоки открытого и шифрованного текстов, $z \in V_K$ — ключ. Предполагается, что биты ключа складываются по модулю 2 с соответствующими промежуточными знаками шифрпреобразования.

Последовательность преобразований функциональной схемы можно представлять в виде условных или безусловных шагов алгоритма, не содержащего операторов перехода на предыдущие операторы. Пусть нелинейные операции в этой программе

$$f_i : V_{n_i} \to V_{m_i},\ x_i \mapsto y_i = f_i(x_i),\ i = 1,\dots,k, \tag{4}$$

следуют в указанном порядке. Значения переменных $x_i \in V_{n_i}$, $y_i \in V_{m_i}$ и $b \in V_M$ однозначно задаются значениями $a \in V_N$ и $z \in V_K$. В промежутках между операциями (4) (как и в начале и в конце программы) могут быть линейные преобразования.

Основным этапом в линейном методе применительно к шифрпреобразованию (3) является нахождение трёх конкретных вектор-столбцов L', L и $L'' \neq 0$ размеров N, K и M соответственно, для которых желательно, чтобы булева случайная величина $\eta = aL' + zL + bL''$ имела как можно большее значение $\delta = |2\,\mathsf{P}[\eta = 0] - 1|$ для как можно большего числа ключей $z \in V_K$, где вероятность $\mathsf{P}[\cdot]$ вычисляется при случайном равновероятном выборе вектора $a \in V_N$, ключ $z \in V_K$ фиксирован.

Случайная величина η определяется по набору вектор-столбцов

$$\pounds = ((l'_i, l''_i), i = 1, \ldots, k)$$

размеров соответственно n_i и m_i как

$$\eta = \sum_{i=1}^{s} (x_i l'_i + y_i l''_i). \tag{5}$$

Чтобы такая сумма приняла вид $aL' + zL + bL''$ для каких-либо вектор-столбцов L', L и $L'' \neq 0$, набор $\pounds$ должен удовлетворять специальным условиям согласованности. Для их формулировки компоненты l'_i припишем на функциональной схеме в виде входных пометок к соответствующим компонентам x_i, а компоненты l''_i аналогично припишем к компонентам y_i в виде выходных пометок. Расстановка пометок в виде элементов из GF(2) распространяется и на каждую линейную операцию в шифрпреобразовании (3). При этом если $g : V_n \to V_m$, $V_n \ni x \mapsto y = xg \in V_m$, — одна из линейных операций в шифрпреобразовании (3) и входные пометки g образуют вектор-столбец α' размера n, а выходные пометки g — вектор-столбец α'' размера m, то в целях обеспечения равенства $x\alpha' = y\alpha''$ условие согласованности (в соответствии с $x\alpha' = xg\alpha''$) предполагает выполнение равенства $\alpha' = g\alpha''$. В частности, пометки вокруг сумматора

$$(x_1, \ldots, x_n) \mapsto x_1 + \ldots + x_n = y_1$$

одинаковы, а пометки вокруг узла размножения

$$x \mapsto (x_1, \ldots, x_n) = (y_1, \ldots, y_m)$$

в сумме дают нуль. При $n = m$ и $g = \mathrm{id}_{V_n}$ обязательно $\alpha' = \alpha''$. Последнее означает, что если выход некоторого отображения в шифрпреобразовании (3) является входом какого-то одного отображения, то к этим выходу и входу приписываются одинаковые пометки.

Пара (l'_i, l''_i) в наборе $\pounds$, такая, что $l'_i \neq 0$, $l''_i = 0$ или $l'_i = 0$, $l''_i \neq 0$, а f_i биективно, считается дефектной. Тогда слагаемые $x_i l'_i + y_i l''_i$ в (5) при случайном равновероятном $x_i \in V_{n_i}$ обладают равномерным распределением на GF(2), то есть имеет максимальную неопределённость.

Пусть W — совокупность всех согласованных наборов пар вектор-столбцов $\pounds = ((l'_i, l''_i), i = 1, \ldots, k)$ без дефектов, в которых хотя бы один вектор-столбец ненулевой (данный набор будем также называть $\pounds$-путём). Обозначим

$$\theta_{\pounds} = |\{i = 1, \ldots, k \ : l''_i \neq 0\}|\,.$$

Для уменьшения неопределённости суммы (5) приходится следить за уменьшением неопределённости отдельных слагаемых. Она минимальна при $l'_i = 0$, $l''_i = 0$. В этой связи становится важной характеристика

$$\theta = \min_{\pounds \in W} \theta_{\pounds} \tag{6}$$

для шифрпреобразования (3), точнее, для линейной среды этого шифрпреобразования, образованной всеми линейными преобразованиями соответствующей функциональной схемы. Замена совокупности нелинейных отображений (4) на любые другие отображения при тех же n_i, m_i никак не скажется на величине θ. Чем больше θ, тем соответствующий шифр является более стойким к линейному методу и тем легче можно этого добиваться путём специального выбора нелинейных отображений (4). Более тщательный выбор последних отображений позволяет также (в большинстве случаев) повышать стойкость к другим методам криптографического анализа.

5. Основной результат

Не ограничивая общность, для простоты изложения будем рассматривать случай $d = 3$, $n = n'(m')^3$. Приведённые далее рассуждения могут быть без труда обобщены на случай $d > 3$.

Пусть также коммутация τ на номерах координат имеет вид

$$(j_0, j_1, j_2) \to (j_1, j_2, j_0), \quad j_0, j_1, j_2 \in \{0, 1, \ldots, m' - 1\}, \tag{7}$$

где при фиксированных j_1, j_2 число j_0 принимает всевозможные значения из множества $\{0, 1, \ldots, m' - 1\}$. Как и в примере 1, в рассматриваемом алгоритме преобразование P будем отождествлять с поворотом влево 3-мерного куба относительно оси z в системе координат (x, y, z).

Покажем, что в данном случае для XSPL-алгоритма при любом числе итераций не наступает существенная зависимость бит промежуточного блока от каждого бита входного блока. Эта слабость приводит к появлению эффективной схемы согласования, что существенно влияет на стойкость рассматриваемого криптографического алгоритма.

Следуя работе [19], введём специальное представление XSLP-алгоритма в виде графа. Данное представление тесно связано с оценкой эффективности линейного метода криптографического анализа. В частности, теоретико-графовый подход хорошо зарекомендовал себя при вычислении показателя рассеивания линейной среды шифрпреобразования (более подробно см. [19, 20]).

Пусть $\pounds = ((l'_{ij}, l''_{ij}), i = 1, \ldots, r,\ j = 0, \ldots, m - 1)$ — произвольный ненулевой согласованный набор столбцов без дефектов, входящий в определение (6) показателя θ. Столбец l'_{ij} состоит из n' бит меток входов подстановки π_{ij} (без ограничения общности будем считать, что $\pi_{ij} = \pi$), а столбец l''_{ij} — из n' меток выходов этой же подстановки. Отсутствие дефектов означает, что $l'_{ij} = 0$ тогда и только тогда, когда $l''_{ij} = 0$, $i = 1, \ldots, r$, $j = 0, \ldots, m - 1$. Согласованность $\pounds$ означает, что

$$l''_i = \widehat{\mathrm{P}}(\mathrm{L}(l'_{i+1})),\ i = 0, 1, \ldots, r - 1,$$

где

— столбец l''_i составлен из столбцов $l''_{i,j}$, $j = 0, \ldots, m - 1$;
— столбец l'_{i+1} составлен из столбцов $l'_{i+1,j}$, $j = 0, \ldots, m - 1$;

— $L \in \mathrm{GF}(2)_{n'm,n'm}$ — блочно-диагональная матрица, составленная из матриц B_{i,j_1}, $j_1 = 1,\ldots,k$, $i = 1,\ldots,r-1$, которые имеют показатель рассеивания ρ [19] относительно разбиения вектора длины $n' \cdot m'$ на подвектора длины n';

— $\widehat{\mathrm{P}} \in \mathrm{GF}(2)_{n'\cdot m,n'\cdot m}$ — подстановочная матрица, осуществляющая перестановку n'-векторов в соответствии с коммутацией τ. При этом меняются местами только n'-векторы, соответствующие подстановкам

$$\pi_{i,(j_0,j_1,j_2)},\ \pi_{i,(j_1,j_2,j_0)},$$

для всех $i = 1,\ldots,r$ и всех $j = j_0 + j_1 m' + j_2 (m')^2$, $j_0, j_1, j_2 \in \{0,1,\ldots,m'-1\}$.

Набору $\pounds$ поставим в соответствие неориентированный граф Γ. *Вершинами* графа Γ, располагающимися в следующих сверху вниз $r-1$ ярусах, являются те блоки $B_{i,(j_1,j_2)}$, $i = 1,\ldots,r-1$, $j_1, j_2 = 0,\ldots,m'-1$, для которых $l''_{i,*(j_1,j_2)} \neq 0$, а значит, и $l'_{i,*(j_1,j_2)} \neq 0$. Здесь столбец $l''_{i,*(j_1,j_2)}$ ($l'_{i,*(j_1,j_2)} \neq 0$) составлен из столбцов $l''_{i,(j_0,j_1,j_2)}$ (соответственно $l'_{i,(j_0,j_1,j_2)}$), $j_0 = 0,\ldots,m'-1$.

Ребром графа Γ будем называть пару вершин $(B_{i,(j_1,j_2)}, B_{i+1,(j_2,j_0)})$ при некотором $j_0 = 0,\ldots,m'-1$, таком, что $l'_{i+1,*(j_2,j_0)} \neq 0$.

Дополнительно к каждой вершине $B_{1,(j_1,j_2)}$ верхнего яруса добавим

$$[[l''_{1,*(j_1,j_2)}]] = \left|\{j_0 = 0,\ldots,m'-1 : l''_{1,(j_0,j_1,j_2)} \neq 0\}\right|$$

рёбер, а к каждой вершине $B_{r-1,(j_1,j_2)}$ нижнего яруса — $[[B^{-1}_{r-1,(j_1,j_2)} l''_{1,*(j_1,j_2)}]]$ рёбер. После этого величина $\theta_{\pounds}$ будет совпадать с числом рёбер в графе Γ.

Для удобства графического представления графа Γ будем изображать его следующим образом. Все $(m')^2$ вершин графа Γ запишем в таблицу из m' строк и столбцов. Запись в таблицу будем производить по строкам слева направо. Таким образом, граф Γ имеет вид, приведённый на рис. 3.

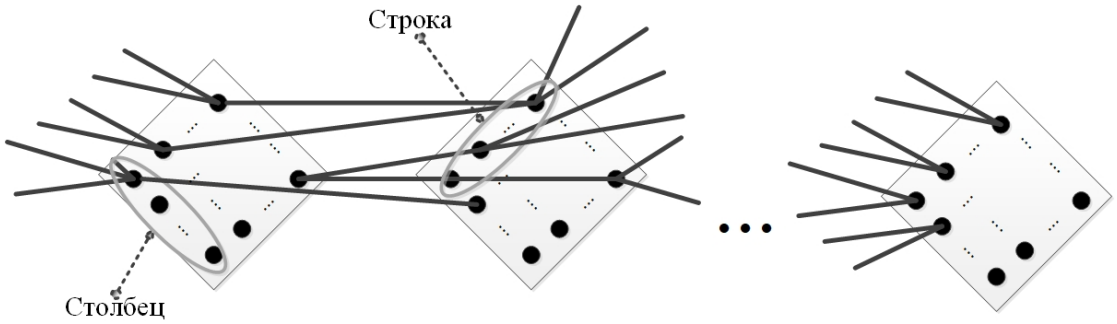

Рис. 3. Общий вид графа Γ для XSLP-алгоритма с длиной блока $n = n'(m')^3$

Граф Γ для XSLP-алгоритма, состоящего из $r \in \mathbb{N}$ итераций, будем обозначать $\Gamma^{(r)}$. Часть графа $\Gamma^{(r)}$, соответствующую i-й итерации, будем называть $(i-1)$-м слоем, $i \leqslant r$. Слои в графе $\Gamma^{(r)}$ занумеруем справа налево начиная с нуля. Количество неизолированных вершин в i-м слое обозначим через M_i, $0 \leqslant i \leqslant r-2$.

5.1. Показатель рассеивания алгоритма типа гиперкуб

Лемма 1. Для любого $r \in \mathbb{N}$, $r \geqslant 2$, и графа $\Gamma^{(r)}$ каждая неизолированная вершина графа i-го слоя смежна лишь с такими вершинами $(i+1)$-го слоя, которые стоят в одном столбце и в разных строках $(i+1)$-го слоя, $0 \leqslant i \leqslant r-2$.

Доказательство. Рассмотрим произвольную неизолированную вершину графа $\Gamma^{(i)}$. Из неё выходят рёбра с номерами $(j_0, j_1, j_2) = j_0 + j_1 m' + j_2 (m')^2$; $j_1, j_2 \in \in \{0, \ldots, m'-1\}$ фиксированы, j_0 принимает все значения из множества $\{0, \ldots, m'-1\}$. Согласно определению коммутации τ (7), данные рёбра (подвекторы длины n') будут переставляться в соответствии с правилом

$$\tau(j_0, j_1, j_2) = (j_1, j_2, j_0).$$

Для завершения доказательства необходимо заметить, что ребро с номером (j_0, j_1, j_2) инцидентно вершине, стоящей с строке с номером j_2 и столбце с номером j_1. ■

Следствие 1. В условиях леммы 1, вершины i-го слоя графа $\Gamma^{(r)}$, лежащие в j-й строке, смежны лишь с такими вершинами $(i+1)$-го слоя, которые лежат в j-м столбце, $j = 1, 2, \ldots, m'$.

Лемма 2. Для любого $r \in \mathbb{N}$, $r \geqslant 2$, и графа $\Gamma^{(r)}$ выполнены следующие неравенства:

$$M_i + M_{i+2} \geqslant \rho, \quad 0 \leqslant i \leqslant r-3. \tag{8}$$

Доказательство. Рассмотрим три подряд идущих слоя графа $\Gamma^{(r)}$. Нетрудно убедиться в том, что неравенство (8) обращается в равенство при $M_{i+1} = 1$. Так как в графе $\Gamma^{(r)}$ отсутствуют петли и кратные дуги, а также в силу $\rho(A_i) = \rho$, $i = 1, \ldots, k$, получаем, что единственная вершина $(i+1)$-го слоя инцидентна по крайней мере ρ рёбрам, а значит, смежна по крайней мере с ρ вершинами, находящимися в i-м и $(i+2)$-м слоях. Следовательно, $M_i + M_{i+2} \geqslant \rho$. Графическая интерпретация ситуации, когда неравенство (8) обращается в равенство, при $m' = 3$ представлена на рис. 4.

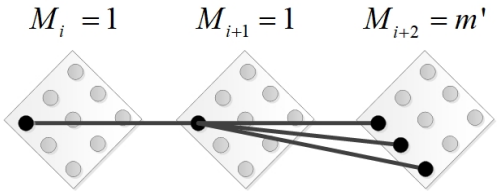

Рис. 4. Пример графа $\Gamma^{(4)}$, для которого $M_i + M_{i+2} = \rho$ при $m' = 3$ и $\rho = m' + 1$

Если предположить, что $M_i + M_{i+2} < \rho$, то неизолированная вершина $(i+1)$-го слоя будет инцидентна менее чем ρ рёбрам, что противоречит условию $\rho(A_i) = \rho$, $i = 1, \ldots, k$. ■

Теорема 1. Пусть $\rho = m' + 1$. Тогда для показателя рассеивания $\theta = \theta(r)$ линейной среды XSLP-алгоритма на $r \in \mathbb{N}$, $r \geqslant 2$, итераций справедлива следующая оценка:

$$\theta(r) \leqslant t(r), \quad t(1) = m',$$
$$t(r) = ((m')^2 + m') \left[\frac{r}{2}\right] + m'(r \bmod 2), \quad r \geqslant 2.$$

Доказательство. Используем лемму 1, а также тот факт, что для любого графа $\Gamma^{(r)}$ выполнено $\theta(r) \leqslant \upsilon(\Gamma^{(r)})$, где $\upsilon(\Gamma^{(r)})$ — количество рёбер в графе $\Gamma^{(r)}$. Построим граф $\Gamma^{(r)}$, такой, что $t(r) = \upsilon(\Gamma^{(r)})$.

Выберем $l \in \{1, \ldots, m'\}$. Начнём построение графа с первого слоя, в котором m' неизолированных вершин находятся в l-м столбце. Из каждой неизолированной

вершины выходит по одному ребру, причём все рёбра смежны лишь с вершинами в l-й строке. Второй слой состоит из m' неизолированных вершин, находящихся в l-й строке. В каждую вершину входит одна и выходит m' рёбер таким образом, что третий слой содержит m' неизолированных вершин, лежащих в l-м столбце. Таким образом, в каждую неизолированную вершину третьего слоя входит m' рёбер и выходит одно ребро. Далее граф строится по индукции аналогичным образом.

Пример описываемого графа $\Gamma^{(r)}$ при $m' = 3$ приведён на рис. 5. ■

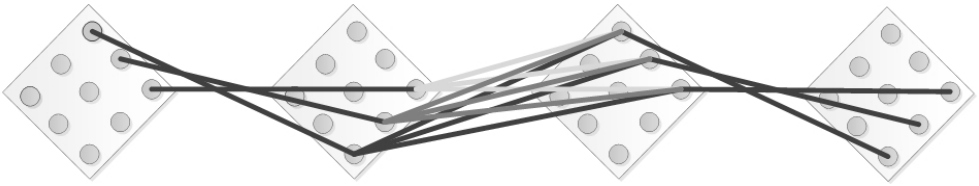

Рис. 5. Граф XSLP-алгоритма с длиной блока $n = n'(m')^3$ при $m' = 3$, для которого $\upsilon(\Gamma^{(r)}) = t(r)$, $r = 1, 2, \ldots$

5.2. Один подход к определению ключа алгоритмов типа «гиперкуб»

Более подробно остановимся на существенной слабости рассматриваемого XSLP-алгоритма, имеющего структуру «гиперкуб». Для этого приведём подход к методу определения ключа для данного алгоритма и оценим его трудоёмкость. Пусть для XSLP-алгоритма на каждой итерации используются случайно равновероятно выбранные ключи $K \in V_n$. Разобъём XSLP-алгоритм на m' параллельно действующих алгоритмов меньшей размерности. При этом каждый из таких алгоритмов получается, согласно доказательству теоремы 1, путём выбора номера $l \in \{1, \ldots, m'\}$ столбца, содержащего неизолированные вершины в первом слое соответствующего графа. Тогда, используя результаты и терминологию теоремы 1, получаем, что в каждом алгоритме меньшей размерности при любом числе итераций $r \in \mathbb{N}$ вместо $n \cdot r$ бит ключа будет использоваться только $n' \cdot t(r)$ бит.

Учитывая, что параллельно действующие алгоритмы меньшей размерности между собой никак не связаны, получаем следующую оценку трудоёмкости метода определения ключа:

$$m'(n' \cdot t(r)),$$

что существенно меньше трудоёмкости полного опробования $n \cdot r$.

5.3. Ещё один вариант коммутации для алгоритмов типа «гиперкуб»

Покажем, что, вообще говоря, выбор преобразования P в XSLP-алгоритмах представляет собой нетривиальную задачу, а обоснование криптографических качеств таких алгоритмов в целом — трудоёмкий процесс, в котором можно легко пойти по ложному пути.

Рассмотрим вопрос выбора коммутации в описанном XSLP-алгоритме типа «гиперкуб» с длиной блока $n = n'(m')^3$. За основу выбран принцип, используемый в алгоритме типа «гиперкуб» при $d = 2$ и $n = n'(m')^2$. Рассмотрев граф Γ такого алгоритма, можно убедиться, что он состоит из m' вершин, в каждую из которой в общем случае приходит не более m' рёбер (каждое ребро выходит из своей вершины) и выходит не более m' вершин (опять-таки в разные вершины).

Сохранение этого принципа при $d = 3$ в явном виде невозможно ввиду большого количества вершин в графе Γ. В качестве альтернативы коммутации вида (1) рассмотрим коммутацию

$$(i_0, i_1, i_2) \xrightarrow{\tau} (i_0, i_1 + i_0 \bmod m', i_2 + i_0 \bmod m'). \tag{9}$$

При $m' = 3$ коммутация имеет следующий вид:

$$1, 14, 27, 4, 17, 21, 7, 11, 24, 10, 23, 9, 13, 26, 3, 1620, 6, 19, 5, 18, 22, 8, 12, 25, 2, 15.$$

С одной стороны, данная коммутация обеспечивает указанный принцип: в графе Γ в каждую вершину заходит и выходит по m' рёбер в разные вершины соответствующих слоёв. Однако эта коммутация сохраняет слабости коммутации вида (1). На это обстоятельство автору указал Ф. М. Малышев. Таким образом, имеет место следующее

Утверждение 1. Пусть рассматриваемый XSLP-алгоритм имеет длину блока $n = n'(m')^3$ и преобразование P задаётся коммутацией вида (9). Тогда для показателя рассеивания $\theta = \theta(r)$ линейной среды XSLP-алгоритма на $r \in \mathbb{N}$ итераций выполнено:

$$\theta(r) = \begin{cases} 1, & \text{если } r = 1, \\ \rho, & \text{если } r = 2, \\ 2\rho - 1, & \text{если } r = 3, \\ \rho^2, & \text{если } r = 4, \\ \theta(r-4) + \rho^2, & \text{если } r > 4. \end{cases}$$

Доказательство. Для доказательства утверждения нужно лишь заметить, что в графе Γ существует подграф Γ_1, который соответствует XSLP-алгоритму с длиной блока $n_1 = n' \cdot m^2$ и равномерно рассеивающими перестановками (в терминологии [20]), то есть коммутацией вида (1).

Действительно, выделим одну вершину в графе Γ с номером $i \in \{1, \ldots, (m')^2\}$. Данная вершина инцидентна рёбрам, которые соответствуют подстановкам с номерами $(i-1)m'+1$, $(i-1)m'+2$, …, $(i-1)m'+m'$. Представив эти номера в виде $i_0 + m'j_1 + (m')^2 j_2$, получаем, что коммутация τ действует следующим образом:

$$(i_0, i_1, i_2) \xrightarrow{\tau} (i_0, i_1 + i_0 \bmod m', i_2 + i_0 \bmod m') \xrightarrow{\tau} (i_0, i_1, i_2).$$

Это с точностью до перенумерации соответствует коммутации вида (1). Таким образом, в каждом слое графа Γ_1 всего m' вершин, в каждую из которой входит не более m' рёбер и выходит не более m' рёбер, причём это справедливо при любой выборе $i \in \{1, \ldots, (m')^2\}$.

Осталось заметить, что для графа Γ_1, согласно [20], выполнено

$$\upsilon(\Gamma^{(1)}) = 1,\ \upsilon(\Gamma^{(2)}) = \rho,\ \upsilon(\Gamma^{(3)}) = 2\rho - 1,\ \upsilon(\Gamma^{(4)}) = \rho^2,\ \upsilon(\Gamma^{(5)}) = \upsilon(\Gamma^{(r-4)}) + \rho^2,$$

что завершает доказательство утверждения. ■

В качестве примера, демонстрирующего результат утверждения 1, рассмотрим случай $m' = 3$. Для получаемого в этом случае XSLP-алгоритма подграф Γ_1 графа Γ приведён на рис. 6.

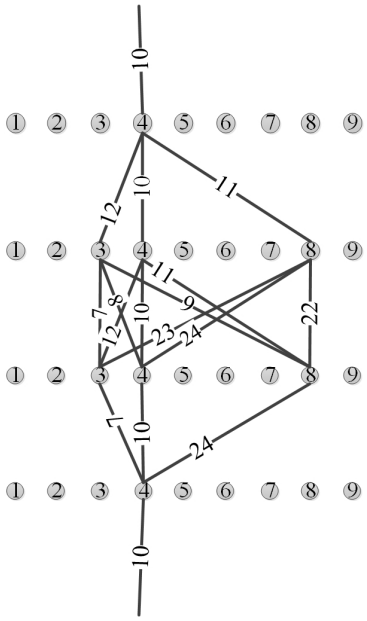

Рис. 6. Подграф Γ_1 графа Γ XSLP-алгоритма с длиной блока $n = n'(m')^3$ при $m' = 3$ и коммутации вида (9). Рёбра и вершины занумерованы слева направо, начиная с 1

ЛИТЕРАТУРА

1. *Трифонов Д. И., Фомин Д. Б.* Об инвариантных подпространствах в XSL-шифрах // Прикладная дискретная математика. 2021. Т. 54. С. 59–77.
2. ГОСТ Р 34.12-2015. Информационная технология. Криптографическая защита информации. Блочные шифры. М.: Стандартинформ, 2015.
3. *Burov D. A. and Pogorelov B. A.* An attack on 6 rounds of KHAZAD // Матем. вопр. криптогр. 2016. Т. 7. № 2. С. 35–46.
4. *Глухов М. М., Елизаров В. П., Нечаев А. А.* Алгебра. В 2-х т. Т. 1. М.: Гелиос АРВ, 2003. 336 с.
5. *Глухов М. М., Елизаров В. П., Нечаев А. А.* Алгебра. В 2-х т. Т. 2. М.: Гелиос АРВ, 2003. 416 с.
6. *Малышев Ф. М., Тараканов В. Е.* Обобщенные графы де Брейна // Матем. заметки. 1997. Т. 62. Вып. 4. С. 540–548.
7. *Daemon J. and Rijmen V.* The Design of Rijndael: AES — The Advanced Encryption Standard. Berlin; Heidelberg: Springer, 2002. 238 p.
8. *Ерохин А. В., Малышев Ф. М., Тришин А. Е.* Многомерный линейный метод и показатели рассеивания линейной среды шифрпреобразований // Матем. вопр. криптогр. 2017. Т. 8. № 4. С. 29–62.
9. Advanced Encryption Standard (AES). `https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf`. 2001.
10. ГОСТ Р 34.11-2012. Информационная технология. Криптографическая защита информации. Функция хэширования. М.: Стандартинформ, 2012.
11. *Brown L., Kwan M., Pieprzyk J., and Seberry J.* Improving resistance to differential cryptanalysis and the redesign of LOKI // LNCS. 1993. V. 739. P. 36–50.
12. *Knudsen L. R.* Cryptanalysis of LOKI // LNCS. 1993. V. 739. P. 22–35.
13. Data Encryption Standard (DES). NIST FIPS PUB 46. 1977.

14. *Massey J. L.* SAFER K-64: A byte-oriented block-ciphering algorithm // LNCS. 1994. V. 809. P. 1–17.
15. *Massey J. L.* SAFER K-64: One year later // LNCS. 1995. V. 1008. P. 212–241.
16. *Feistel H.* Cryptography and computer privacy // Scientific Amer. 1973. V. 228. No. 5. P. 15–23.
17. *Малышев Ф. М.* Двойственность разностного и линейного методов в криптографии // Матем. вопр. криптогр. 2014. Т. 5. № 3. С. 35–48.
18. *Malyshev F. M. and Trishin A. E.* Linear and differential cryptanalysis: Another viewpoint // Матем. вопр. криптогр. 2020. Т. 11. № 2. С. 83–98.
19. *Малышев Ф. М., Трифонов Д. И.* Рассеивающие свойства XSLP-шифров // Матем. вопр. криптогр. 2016. Т. 7. № 3. С. 47–60.
20. *Федченко В. А.* Показатели рассеивания линейной среды AES-подобных алгоритмов шифрования // Матем. вопр. криптогр. 2017. Т. 8. № 3. С. 109–126.

REFERENCES

1. *Trifonov D. I. and Fomin D. B.* Ob invariantnykh podprostranstvakh v XSL-shifrakh [Invariant subspaces in SPN block c ipher]. Prikladnaya Diskretnaya Matematika, 2021, no. 54, pp. 59–77. (in Russian)
2. GOST R 34.12-2015. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Blochnyye shifry. [Information Technology. Cryptographic Information Protection. Block ciphers]. Moscow, Standartinform, 2015. (in Russian)
3. *Burov D. A. and Pogorelov B. A.* An attack on 6 rounds of KHAZAD. Matem. Vopr. Kriptogr., 2016, vol. 7, no. 2, pp. 35–46.
4. *Gluhov M. M., Elizarov V. P., and Nechaev A. A.* Algebra [Algebra]. Vol. 1. Moscow, Gelios ARV Publ., 2003. 336 p. (in Russian)
5. *Gluhov M. M., Elizarov V. P., and Nechaev A. A.* Algebra [Algebra]. Vol. 2. Moscow, Gelios ARV Publ., 2003. 416 p. (in Russian)
6. *Malyshev F. M. and Tarakanov V. E.* Obobshchennyye grafy de Breyna [Generalized de Bruijn graphs]. Matem. Zametki, 1997, vol. 62, no. 4, pp. 540–548. (in Russian)
7. *Daemon J. and Rijmen V.* The Design of Rijndael: AES — The Advanced Encryption Standard. Berlin; Heidelberg, Springer, 2002, 238 p.
8. *Erokhin A. V., Malyshev F. M., and Trishin A. E.* Mnogomernyy lineynyy metod i pokazateli rasseivaniya lineynoy sredy shifrpreobrazovaniy [Multidimensional linear method and diffusion characteristics of linear medium of ciphering transform]. Matem. Vopr. Kriptogr., 2017, vol. 8, no. 4, pp. 29–62. (in Russian)
9. Advanced Encryption Standard (AES). `https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf`, 2001.
10. GOST R 34.11-2012. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Funktsiya kheshirovaniya. [Information Technology. Cryptographic Information Protection. Hash Function]. Moscow, Standartinform, 2012. (in Russian)
11. *Brown L., Kwan M., Pieprzyk J., and Seberry J.* Improving resistance to differential cryptanalysis and the redesign of LOKI. LNCS, 1993, vol. 739, pp. 36–50.
12. *Knudsen L. R.* Cryptanalysis of LOKI. LNCS, 1993, vol. 739, pp. 22–35.
13. Data Encryption Standard (DES). NIST FIPS PUB 46. 1977.
14. *Massey J. L.* SAFER K-64: A byte-oriented block-ciphering algorithm. LNCS, 1994, vol. 809, pp. 1–17.
15. *Massey J. L.* SAFER K-64: One year later. LNCS, 1995, vol. 1008, pp. 212–241.

16. *Feistel H.* Cryptography and computer privacy. Scientific Amer., 1973, vol. 228, no. 5, pp. 15–23.
17. *Malyshev F. M.* Dvojstvennost' raznostnogo i linejnogo metodov v kriptografii [The duality of differential and linear methods in cryptography]. Matem. Vopr. Kriptogr., 2014, vol. 5, no. 3, pp. 35–48. (in Russian)
18. *Malyshev F. M. and Trishin A. E.* Linear and differential cryptanalysis: Another viewpoint. Matem. Vopr. Kriptogr., 2020, vol. 11, no. 2, pp. 83–98.
19. *Malyshev F. M. and Trifonov D. I.* Rasseivayushchiye svoystva XSLP-shifrov [Diffusion properties of XSLP-ciphers]. Matem. Vopr. Kriptogr., 2016, vol. 7, no. 3, pp. 47–60. (in Russian)
20. *Fedchenko V. A.* Pokazateli rasseivaniya lineynoy sredy AES-podobnykh algoritmov shifrovaniya [Diffusion rates of linear medium in AES-like ciphers]. Matem. Vopr. Kriptogr., 2017, vol. 8, no. 3, pp. 109–126. (in Russian)

УДК 519.719.2 DOI 10.17223/20710410/57/5

THE SECURITY OF THE CODE-BASED SIGNATURE SCHEME BASED ON THE STERN IDENTIFICATION PROTOCOL

V. V. Vysotskaya*,**, I. V. Chizhov*,**,***

* *JSC "NPK Kryptonite", Moscow, Russia*
** *Lomonosov Moscow State University, Moscow, Russia*
*** *Federal Research Center "Informatics and Control" of Russian Academy of Science, Moscow, Russia*

E-mail: {v.vysotskaya, i.chizhov}@kryptonite.ru

The paper provides a complete description of the digital signature scheme based on the Stern identification protocol. We also present the proof of the existential unforgeability of the scheme under the chosen message attack (EUF-CMA) in the random oracle model (ROM). Finally, we discuss the choice of the signature parameters, in particular providing 70-bit security.

Keywords: *post-quantum cryptography, code-based cryptography, digital signature, Stern's scheme, Fiat-Shamir transform, provable security, EUF-CMA security.*

О СТОЙКОСТИ КОДОВОЙ ЭЛЕКТРОННОЙ ПОДПИСИ НА ОСНОВЕ ПРОТОКОЛА ИДЕНТИФИКАЦИИ ШТЕРНА

В. В. Высоцкая*,**, И. В. Чижов*,**,***

* *АО НПК «Криптонит», г. Москва, Россия*
** *МГУ им М. В. Ломоносова, г. Москва, Россия*
*** *ФИЦ ИУ РАН, г. Москва, Россия*

Представлено полное описание схемы электронной подписи на основе схемы идентификации Штерна. Доказана стойкость схемы относительно построения экзистенциальной подделки при атаке с выбором сообщений (EUF-CMA) в модели со случайным оракулом. Обсуждается выбор параметров подписи, в частности обеспечивающий стойкость, равную 70 битам.

Ключевые слова: *постквантовая криптография, кодовая криптография, электронная подпись, схема Штерна, преобразование Фиата — Шамира, доказуемая стойкость, EUF-CMA-стойкость.*

1. Introduction

The security of all standardized cryptographic algorithms used all around the world is based on the complexity of several number-theoretical problems. The latter includes the discrete logarithm and factorization problems. However, in 1994 P. Shor showed [1] that quantum computers could break all schemes constructed this way. And in 2001 the Shor's algorithm was implemented on a 7-qubit quantum computer. Since then, various companies have been actively developing more powerful quantum computers. Potential progress in this area poses a real threat to modern public-key cryptography.

This led to the emergence of so-called post-quantum cryptographic schemes. Most of them can be categorized into the following classes: code-based, lattice-based, multivariate,

hash-based, and isogeny-based. No successful quantum-computer attacks on "hard" problems from these areas are known.

The interest in code-based schemes as post-quantum ones can be noticed in the works submitted to the contest for prospective public-key post-quantum algorithms which was announced in 2016 by the US National Institute of Standards and Technology (NIST) [2]. The algorithms that win this contest will be accepted as US national standards. 21 of 69 applications filed (that is, almost a third of all works) were based on coding theory. However, it is worth noting that only three of them presented digital signature schemes. These were pqsigRM [3], RaCoSS [4] and RankSign [5] schemes. However, attacks on each of them were built during the peer review. The attack on the RankSign scheme was presented on Asiacrypt conference [6]. Out of the competition pqsigRM and RaCoSS schemes were fixed and presented as Modified pqsigRM [7] and RaCoSS-R [8], respectively. However, the RaCoSS-R scheme was also proven to be insecure [9]. As a result, none of the signatures based on the error-correcting codes made it to the final of the NIST competition.

In general, the development of code-based signature schemes was advancing less successfully than of the encryption ones. The first signature scheme of this type was KKS, presented by G. Kabatianskii, E. Krouk, and B. Smeets in [10] in 1997. However, in 2007 it was shown [11] that re-signing on one key pair leads to the disclosure of some information about the secret key. Thus, it is necessary either to use the signature as a one-time one or use additional resources for building and maintaining auxiliary structure.

After that for a rather long time, attacks on all proposed signature schemes were built so quickly that there was a fear that such schemes could not be created at all [12].

In 2001, N. Courtois, M. Finiasz and N. Sendrier presented a digital signature CFS [13] based on encryption schemes by R. McEliece [14] and H. Niederreiter [15] (provably secure version of this signature, called mCFS, was later proposed by L. Dallot in [16]). The authors used a decryption algorithm as the signature generation one. Unfortunately, due to the inner decoding procedure with extremely small probability of success on a random input, the signature generation algorithm has to be repeated many times. Also, a significant disadvantage of CFS-type schemes is that their security depends on the assumption that the base code is indistinguishable from a random one. This leads to the emergence of attacks on signatures, previously considered provably secure. One of the latest schemes of this type is Wave [17], based on generalized $(U, U + V)$ codes.

Another approach to constructing a signature scheme is to apply the Fiat — Shamir transformation [18] to an identification protocol. For example, one may use identification schemes by J. Stern [19], A. Jain et al. [20], or CVE [21]. This method does not take into account features of codes. But it allows to prove the security without assumptions that depend on their structure. However, due to the fact that the basic scheme has a certain cheating probability, the signature algorithm has to be repeated several times, that leads to an increase in its operation time and in the resulting signature length.

This drawback is overcome in Lyubashevsky-type signatures, the original version of which is lattice-based [22]. Despite of the fact that the original version remains secure, all known attempts to replace lattices with codes in Hamming metric resulted in the loss of security. However, a code-based signature in the rank metric called Durandal was proposed in [23] and is still considered secure. Yet it is not proven that the signature distribution is independent from the secret key and reveals no information on it. Moreover, the security proof is based on the hardness of a new problem PSSI+. In turn, the security of Stern-type schemes is based only on NP-hard problems and the hardness of finding a collision of the underlying hash function.

Despite the fact that the signature based on the Stern identification scheme has been repeatedly mentioned in the literature, it has never been fully presented. For example, the review [24] by R. Overbeck and N. Sendrier only mentions the possibility of constructing such a signature without giving the algorithm itself. In the paper [25] the scheme is formulated with an error, which leads to the significant decrease of the security level compared to the expected value. A correct but short description of the scheme can be found in [26].

Moreover, the security proof of the scheme is considered to be provided by D. Pointcheval and J. Stern in [27]. This paper presents so-called Forking lemma, by which the security of the signature scheme to existential forgery under an adaptively chosen-message attack in the random oracle model may be proved. The authors mention there the applicability of the Forking lemma to the Stern signature scheme proof. However, this fact was not proven neither in this paper nor elsewhere later.

In the paper we provide a complete description of the signature scheme based on the Stern identification scheme along with the proof of the existential unforgeability under the chosen message attack (EUF-CMA) under assumptions of hardness of syndrome decoding and hash function collision finding problems.

The rest of this paper is structured as follows. In Section 2 we give basic definitions, describe some hard problems and show the original Stern identification protocol. We present the signature scheme together with the security model in Section 3. Section 4 is devoted to the security proof of our signature in the EUF-CMA model. We give some restrictions on the scheme parameters and introduce an example parameter set in Section 5. Finally, conclusions are presented in Section 6.

2. Definitions and Preliminary Results

Our signature is based on linear block error-correcting codes. We will call them *codes* for brevity. The set of all binary strings of length n we denote by $\{0,1\}^n$ and the set of strings of arbitrary length by $\{0,1\}^*$. We denote the symmetric group of order n by S_n, i.e., the group of all permutations of elements of the set $\{1,\ldots,n\}$. If $\sigma \in S_n$, $u \in \{0,1\}^n$, then $\sigma(u) \in \{0,1\}^n$, $\sigma(u)_i = u_{\sigma(i)}$. The weight of the vector u is the number of its nonzero elements. It is denoted by $\mathrm{wt}(u)$.

The security of the signature scheme is based on the hardness of the following problems.

Problem SD(H, y, ω). **Syndrome Decoding**
Input: $(n-k)\times n$ parity-check matrix H of some binary code, nonzero vector $y \in \{0,1\}^{n-k}$, called *syndrome*, and number $\omega > 0$.
Output: vector $e \in \{0,1\}^n$ such that $\mathrm{wt}(e) = \omega$ and $He^{\mathrm{T}} = y^{\mathrm{T}}$.

Problem Coll(h). **Collision Finding**
Input: hash function $h : \{0,1\}^* \to \{0,1\}^\ell$.
Output: vectors $x', x'' \in \{0,1\}^*, x' \neq x''$, such that $h(x') = h(x'')$.

The former problem is known to be NP-hard [28]. The best known algorithm solves it in $\mathcal{O}\big(2^{0.0885n}\big)$ bit operations [29]. The complexity of the latter problem depends on the structure of the function h. In the general case, the complexity of solving such a problem using the birthday paradox can be estimated as $\mathcal{O}\big(2^{\ell/2}\big)$.

Let us recall the Stern identification protocol presented in [19]. The protocol parameters depend on the parameters of the underlying code: its length n, dimension k and minimum

distance ω. The parity-check matrix of this code is a random matrix $H \in \{0,1\}^{(n-k)\times n}$. Also, the protocol is based on a hash function $h(\cdot) : \{0,1\}^* \to \{0,1\}^\ell$.

To generate a secret key, one randomly uniformly chooses $s \in \{0,1\}^n$ such that $\mathrm{wt}(s) = \omega$. Now public key can be derived as $y = Hs^{\mathrm{T}}$. The description of the identification protocol is shown on Fig. 1. Here the notation $s \xleftarrow{\mathcal{U}} S$ means that s is chosen from the set S uniformly at random. We also denote the assignment of value v to x by $x \leftarrow v$.

Prover(s)		**Verifier**(y)
$u \xleftarrow{\mathcal{U}} \{0,1\}^n, \sigma \xleftarrow{\mathcal{U}} S_n$		
$c_0 \leftarrow h(\sigma \Vert Hu^T)$		
$c_1 \leftarrow h(\sigma(u))$		
$c_2 \leftarrow h(\sigma(u \oplus s))$		
	$\xrightarrow{c_0, c_1, c_2}$	
		$b \xleftarrow{\mathcal{U}} \{0,1,2\}$
	$\xleftarrow{b}$	
if $b = 0 : r_0 \leftarrow \sigma, r_1 \leftarrow u$		
if $b = 1 : r_0 \leftarrow \sigma, r_1 \leftarrow u \oplus s$		
if $b = 2 : r_0 \leftarrow \sigma(u), r_1 \leftarrow \sigma(s)$		
	$\xrightarrow{r_0, r_1}$	
		if $b = 0$:
		Check $c_0 \stackrel{?}{=} h(r_0 \Vert Hr_1^T)$,
		$c_1 \stackrel{?}{=} h(r_0(r_1))$
		if $b = 1$:
		Check $c_0 \stackrel{?}{=} h(r_0 \Vert (Hr_1^T \oplus y))$,
		$c_2 \stackrel{?}{=} h(r_0(r_1))$
		if $b = 2$:
		Check $c_1 \stackrel{?}{=} h(r_0)$,
		$c_2 \stackrel{?}{=} h(r_0 \oplus r_1)$,
		$\mathrm{wt}(r_1) \stackrel{?}{=} \omega$

Fig. 1. Stern identification scheme

In his paper, Stern proposes a strategy for an adversary to pass identification without knowing the secret key with probability of success equal to $2/3$. So to reduce this value and to reach the required level of security, one should repeat the algorithm several times.

Recall the general definition of a digital signature scheme.

Definition 1. A digital signature scheme is a triple $\Sigma = (\mathrm{KeyGen}, \mathrm{Sig}, \mathrm{Ver})$ of (possibly probabilistic) polynomial time algorithms, where

1) KeyGen() outputs a key pair (pk, sk);
2) $\mathrm{Sig}(sk, m)$ receives as input the secret key sk and a message $m \in \{0,1\}^*$ and outputs a signature ζ;
3) $\mathrm{Ver}(pk, m, \zeta)$ receives as input the public key pk, a message m and a signature ζ. It outputs 0 or 1, where 1 means that ζ is accepted as a signature for message m and public key pk, 0 means that the signature is not accepted. Moreover, $\mathrm{Ver}(pk, m, \mathrm{Sig}(sk, m)) = 1$ for a correct key pair (pk, sk).

3. Signature scheme

In this Section, we show the digital signature scheme that is the result of the Fiat — Shamir transformation applied to the Stern identification scheme. The transformation consists of replacing the random value b generated by the verifier, by some function f of the message and values received from the prover. It is important for f to depend on all of these values at once.

Parameters of the signature are the same as in the original identification protocol described in Section 2. Additionally, the scheme uses a hash function $f(\cdot) : \{0,1\}^* \to \{0,1,2\}^\delta$. The length of the signature depends on the parameter δ that is determined by the security parameter λ.

Stern.KeyGen()

$s \xleftarrow{\mathcal{U}} \{x \in \{0,1\}^n : \mathrm{wt}(x) = \omega\}$
$y \leftarrow Hs^T$
return (y, s)

Stern.Sig(s, m)

foreach $0 \leqslant i < \delta$:
  $u_i \xleftarrow{\mathcal{U}} \{0,1\}^n, \sigma_i \xleftarrow{\mathcal{U}} S_n$
  $c_{i,0} \leftarrow h(\sigma_i \| Hu_i^T)$
  $c_{i,1} \leftarrow h(\sigma_i(u_i))$
  $c_{i,2} \leftarrow h(\sigma_i(u_i \oplus s))$
  $c_i \leftarrow c_{i,0} \| c_{i,1} \| c_{i,2}$
$c \leftarrow c_0 \| \ldots \| c_{\delta-1}$
$b \leftarrow f(m \| c)$
foreach $0 \leqslant i < \delta$:
  if $b_i = 0 : r_i \leftarrow \sigma_i \| u_i$
  if $b_i = 1 : r_i \leftarrow \sigma_i \| (u_i \oplus s)$
  if $b_i = 2 : r_i \leftarrow \sigma_i(u_i) \| \sigma_i(s)$
$r \leftarrow r_0 \| \ldots \| r_{\delta-1}$
return $c \| r$

Stern.Ver$(y, m, (c\|r))$

$b \leftarrow f(m\|c)$
foreach $0 \leqslant i < \delta$:
  if $\big[b_i = 0\big] \wedge \Big[\big[c_{i,0} \neq h(r_{i,0}\|Hr_{i,1}^T)\big] \vee \big[c_{i,1} \neq h(r_{i,0}(r_{i,1}))\big]\Big]$:
    return 0
  if $\big[b_i = 1\big] \wedge \Big[\big[c_{i,0} \neq h(r_{i,0}\|(Hr_{i,1}^T \oplus y))\big] \vee \big[c_{i,2} \neq h(r_{i,0}(r_{i,1}))\big]\Big]$:
    return 0
  if $\big[b_i = 2\big] \wedge \Big[\big[c_{i,1} \neq h(r_{i,0})\big] \vee \big[c_{i,2} \neq h(r_{i,0} \oplus r_{i,1})\big] \vee \big[\mathrm{wt}(r_{i,1}) \neq \omega\big]\Big]$:
    return 0
return 1

To estimate the scheme security, we construct experiments where the adversary is represented by a probabilistic polynomial-time Turing machine. The notation $\mathbf{Exp} \Rightarrow b$ means that b is the output of the experiment **Exp**. We write **abort** in the oracle pseudocode to denote that experiment should stop and return 0. We denote the set of all mappings from set A to set B by $Func(A, B)$. To emphasize the fact that x is the result of a probabilistic algorithm A, we write $x \leftarrow\!\$\ A(\ldots)$.

To model a random oracle $F : \{0,1\}^* \to \{0,1,2\}^\delta$, we use lazy sampling. We introduce the set Π^F containing pairs of the form $(\alpha, F(\alpha))$. Further we write $(\alpha, \cdot) \in \Pi^F$ for $\alpha \in \{0,1\}^*$ to show that there exists $\beta \in \{0,1,2\}^\delta$ such that $(\alpha, \beta) \in \Pi^F$. As far as Π^F contains not more than one pair (α, β) for each α, then $\Pi^F(\alpha)$ denotes either β if $(\alpha, \beta) \in \Pi^F$ or special value $\perp$ if there is no such a pair.

Definition 2. For the signature scheme Stern.Σ, we denote the advantage of the adversary $\mathcal{A}$ in the EUF-NMA model with a random oracle access by

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) = \mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) \Rightarrow 1],$$

where the experiment $\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A})$ is defined as follows:

$\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A})$
$(pk, sk) \leftarrow\!\$\ \text{Stern.KeyGen}()$
$\Pi^F \leftarrow \varnothing$
$(m, \zeta) \leftarrow\!\$\ \mathcal{A}^F(pk)$
return Stern.Ver(pk, m, ζ)

Oracle $F(\alpha)$
if $\alpha \in \Pi^F : \beta \leftarrow \Pi^F(\alpha)$
else
  $\beta \overset{\mathcal{U}}{\leftarrow} \{0,1,2\}^\delta$
  $\Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
return β

Definition 3. For the signature scheme Stern.Σ, we denote the advantage of the adversary $\mathcal{A}$ in the EUF-CMA model with random oracle access by

$$\mathsf{Adv}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A}) = \mathsf{P}[\mathbf{Exp}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A}) \Rightarrow 1],$$

where the experiment $\mathbf{Exp}^{\text{EUF-CMA}}_{\text{Stern}}$ is defined as follows:

$\mathbf{Exp}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A})$

1 : $(pk, sk) \leftarrow\$ \text{Stern.KeyGen}()$
2 : $\mathcal{L} \leftarrow \varnothing$
3 : $\Pi^F \leftarrow \varnothing$
4 : $(m, \zeta) \leftarrow\$ \mathcal{A}^{\text{Sign},F}(pk)$
5 : **if** $m \in \mathcal{L}$: **return** 0
6 : **return** $\text{Stern.Ver}(pk, m, \zeta)$

Oracle $F(\alpha)$

1 : **if** $\alpha \in \Pi^F : \beta \leftarrow \Pi^F(\alpha)$
2 : **else**
3 : $\quad \beta \xleftarrow{\mathcal{U}} \{0,1,2\}^\delta$
4 : $\quad \Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
5 : **return** β

Oracle $\text{Sign}(m)$

1 : $\zeta \leftarrow\$ \text{Stern.Sig}(sk, m)$
2 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{m\}$
3 : **return** ζ

4. Security bounds

Let us give several definitions that will be needed below.

Definition 4. If T is a ternary tree of depth δ with N leaves, then the density of T is defined as $N/3^\delta$.

Definition 5. Let us call a tree ρ-dense if its density is not less than ρ.

Definition 6. We call a tree uniformly ρ-dense if each of its subtrees, excluding leaves, is a ρ-dense tree.

Proposition 1. A ρ-dense tree T with all leaves having depth δ, considered as a graph, contains a subgraph that is a uniformly $\dfrac{\rho}{\delta}$-dense tree with the same root. Moreover, each of its leaves has depth δ.

Proof. Let us describe the algorithm to choose such a subgraph. We start from the $(\delta-1)$-th level of the tree and move to the root (level 0) disposing of vertices that are roots of subtrees of density less than $\theta = \rho/\delta$. Note that until the algorithm stops (i.e. reaches the root), some leaves may have depth less than δ. However, after the algorithm completion each of the survived leaves will have depth δ.

Let us show that at each step of this algorithm the root density decreases by no more than θ. Suppose that there are $n_{i,3}$ vertices at the i-th level of the original tree T. The densities of subtrees formed by them are $\rho_{i,1}, \ldots, \rho_{i,n_{i,3}}$. If we denote the number of leaves of T by t, then

$$\rho_{i,1} + \ldots + \rho_{i,n_{i,3}} = \frac{t}{3^{\delta-i}} = 3^i\rho.$$

After the step of the algorithm at the i-th level, some of these vertices may be disposed of, namely those that have density less than θ. Thus, the new density $\rho'_{i,j}$ may either be equal to $\rho_{i,j}$ or become 0 if $\rho_{i,j} < \theta$. So

$$\rho'_{i,1} + \ldots + \rho'_{i,n_{i,3}} \geqslant \rho_{i,1} + \ldots + \rho_{i,n_{i,3}} - n_{i,3}\theta \geqslant 3^i\rho - n_{i,3}\theta.$$

Thus, for the new density of the root ρ' holds $3^i\rho' \geqslant 3^i\rho - n_{i,3}\theta$ and

$$\rho' \geqslant \rho - \frac{n_{i,3}}{3^i}\theta \geqslant \rho - \theta.$$

As a result of all deletions, ρ has decreased by at most $(\delta-1)\theta$. Since $\theta=\rho/\delta$, then

$$\rho-(\delta-1)\frac{\rho}{\delta}=\frac{\rho\delta-(\delta-1)\rho}{\delta}=\frac{\rho}{\delta}=\theta.$$

So the resulting tree is uniformly θ-dense. ■

Theorem 1. Let $\mathcal{A}$ be an adversary with time complexity at most T in the EUF-NMA model for the Stern signature scheme, making at most one query to the hashing oracle F, then it holds that

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A})\leqslant \max\left\{15\sqrt[3]{\frac{\delta^2 T}{\min\{T_{SD},T_{Coll}\}}}+\left(\frac{2}{3}\right)^{\delta},\ \left(\frac{2}{3}\right)^{\delta}\left(1+2\delta\cdot 1.1^{\delta}\right)\right\},$$

where T_{SD} and T_{Coll} are complexities of optimal algorithms solving $\mathrm{SD}(H,y,\omega)$ and $\mathrm{Coll}(h)$ problems with probabilities of success at least $1-1/e$.

Proof.

Denote

$$\varepsilon=\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A})-\left(\frac{2}{3}\right)^{\delta}. \tag{1}$$

In case $\varepsilon\leqslant 0$ the proof is complete. Therefore, further we will consider the case

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A})=\left(\frac{2}{3}\right)^{\delta}+\varepsilon,\ \varepsilon>0.$$

We can represent the execution of the adversary $\mathcal{A}$ at all outputs of the random oracle F as an incomplete ternary tree $T(x)$, each leaf of which has depth δ. It is determined by $\mathcal{A}$'s random tape x. Each output b of the random oracle corresponds to a certain path in the tree. If the corresponding b_i equals 0, then the vertex has the left child, if $b_i=1$, then the vertex has the middle one, and if $b_i=2$, then it has the right one. If the adversary was not able to build a signature for some output of the random oracle correctly, then the corresponding branch is removed from the tree. Note that fixing the adversary's random tape, we guarantee that at each level of the tree the same part of the signature corresponding to $c_{i,0}\|c_{i,1}\|c_{i,2}$ is checked.

Let us show that if there exists a level i with a vertex with a left child, a vertex with a middle child and a vertex with a right child (denoted respectively $v_{i,0}$, $v_{i,1}$ and $v_{i,2}$), then one of the $\mathrm{SD}(H,y,\omega)$ and $\mathrm{Coll}(h)$ problems can be solved. Note that some of vertices $v_{i,0}, v_{i,1}$, and $v_{i,2}$ may coincide. Later we will present the algorithm that let some adversary $\mathcal{B}$ find such vertices in the tree $T(x)$ with probability $1-1/e$.

Let the tree has such vertices. In this case, the adversary has successfully generated three signatures on outputs of random oracle that all differ in the i-th trit. Let $r_{i,0}=\tilde{\sigma}_0$ and $r_{i,1}=\tilde{u}_0$ for $b_i=0$. For $b_i=1$, let $r_{i,0}=\tilde{\sigma}_1$ and $r_{i,1}=\tilde{w}_1$, where $\tilde{w}_1$ corresponds to $u_i\oplus s$. Finally, for $b_i=2$, let $r_{i,0}=\tilde{z}_2$ and $r_{i,1}=\tilde{t}_2$, where $\tilde{z}_2$ corresponds to $\sigma_i(u_i)$ and $\tilde{t}_2$ corresponds to $\sigma_i(s)$. Since $c_{i,0}$ can be obtained in two cases ($b_i=0$ and $b_i=1$), then

$$c_{i,0}=h(\tilde{\sigma}_0\|H\tilde{u}_0^{\mathrm{T}})=h(\tilde{\sigma}_1\|H\tilde{w}_1^{\mathrm{T}}\oplus y).$$

Hence, either collision of hash function h can be found, or $\tilde{\sigma}_0=\tilde{\sigma}_1$ and $H\tilde{u}_0^{\mathrm{T}}=H\tilde{w}_1^{\mathrm{T}}\oplus y$. Similarly, it can be shown that if no collisions were found, then $\tilde{z}_2=\tilde{\sigma}_0(\tilde{u}_0)$ and

$\tilde{z}_2 \oplus \tilde{t}_2 = \tilde{\sigma}_1(\tilde{w}_1)$. Note that as the third answer was accepted, $\tilde{t}_2$ satisfies the weight constraint. Denoting $\sigma = \tilde{\sigma}_0 = \tilde{\sigma}_1$, we have

$$\tilde{t}_2 = \tilde{z}_2 \oplus (\tilde{t}_2 \oplus \tilde{z}_2) = \sigma(\tilde{u}_0 \oplus \tilde{w}_1).$$

Therefore, $\tilde{u}_0 \oplus \tilde{w}_1$ also has the acceptable weight. Then

$$H(\tilde{u}_0 \oplus \tilde{w}_1)^{\mathrm{T}} = H\tilde{u}_0^{\mathrm{T}} \oplus H\tilde{w}_1^{\mathrm{T}} = y$$

and $\tilde{u}_0 \oplus \tilde{w}_1$ is an acceptable secret key.

Let us denote $\theta = \dfrac{\varepsilon}{2\delta}$ and describe an algorithm that finds a tree with vertices $v_{i,0}, v_{i,1}$, and $v_{i,2}$ with some probability.

Algorithm 1

1) Randomly choose a value x of the adversary's random tape (i.e., fix the tree $T(x)$).
2) Randomly choose $60/\theta^2$ inputs of the random oracle and evaluate its outputs (defining the of branches of the tree).
3) Traverse the tree level by level to find vertices $v_{i,0}, v_{i,1}$ and $v_{i,2}$. If they are found, then solve either $\mathrm{SD}(H, y, \omega)$ or $\mathrm{Coll}(h)$ problem. Otherwise, return to Step 1.

Lemma 1. Under the assumptions of Theorem 1, the success probability of each run of Algorithm 1 is not less than $\varepsilon/4$, where ε is defined as in (1).

Proof. Define the set X as

$$X = \left\{ x : \text{ there are at least } 2^\delta + \frac{\varepsilon}{2} \cdot 3^\delta \text{ branches in } T(x) \right\}.$$

Then $\mathsf{P}[x \in X] \geqslant \varepsilon/2$.

Let, on the contrary, $\mathsf{P}[x \in X] < \varepsilon/2$. Let us denote the number of leaves of $T(x)$ by t. Then $\mathsf{P}[\mathcal{A} \Rightarrow 1 \wedge x \notin X] = t/3^\delta < (2/3)^\delta + \varepsilon/2$. Therefore, the success probability of $\mathcal{A}$ is

$$\begin{aligned}\mathsf{P}[\mathcal{A} \Rightarrow 1] = \mathsf{P}[\mathcal{A} \Rightarrow 1 \wedge x \in X] + \mathsf{P}[\mathcal{A} \Rightarrow 1 \wedge x \notin X] \leqslant \mathsf{P}[x \in X] + \\ + \mathsf{P}[\mathcal{A} \Rightarrow 1 \wedge x \notin X] < \varepsilon/2 + \big((2/3)^\delta + \varepsilon/2\big) = (2/3)^\delta + \varepsilon.\end{aligned}$$

And we came to a contradiction.

Let us consider separately the case $x \in X$. Note that X defines a set of $\varepsilon/2$-dense trees. Therefore, by the Proposition 1 one can select a uniformly θ-dense subtree with leaves of depth δ from each such tree. Let us call this tree $T_1(x)$.

For any index i, $0 \leqslant i \leqslant \delta$, we denote the number of vertices of $T_1(x)$ at i-th level having one, two, and three children by $n_{i,1}, n_{i,2}$, and $n_{i,3}$. We denote the total number of vertices at the i-th level by n_i. Then for $0 \leqslant i < \delta$ it holds that

$$n_{i+1} = n_{i,1} + 2n_{i,2} + 3n_{i,3} = n_i + n_{i,2} + 2n_{i,3}. \tag{2}$$

Let $q = \max\limits_i n_{i,3}/n_i$. Then $n_{i,3} \leqslant qn_i$. From (2) it holds that

$$n_{i+1} \leqslant n_i + n_{i,2} + 2qn_i \leqslant 2n_i + 2qn_i = 2n_i(1+q).$$

From the definition of uniformly θ-dense tree for each i, $0 \leqslant i \leqslant \delta$, holds the inequality

$$n_i \geqslant 3^i\theta.$$

Then

$$3^{\delta}\theta \leqslant n_{\delta} \leqslant 2^{\delta} n_0 (1+q)^{\delta}.$$

Since $n_0 = 1$ ($i = 0$ corresponds to the root of the tree), we have

$$\delta \ln(1+q) + \delta \ln 2 \geqslant \ln\theta + \delta \ln 3.$$

Dividing by δ we finally obtain

$$q \geqslant \frac{3}{2} \cdot \theta^{1/\delta} - 1.$$

Now let us fix $\alpha = \log_{3/2}\left(1.1\,(2\delta)^{1/\delta}\right)$ and consider separately two cases.

If $\varepsilon \leqslant (2/3)^{\delta(1-\alpha)}$, then

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) \leqslant \left(\frac{2}{3}\right)^{\delta}\left(1 + \left(\frac{2}{3}\right)^{-\alpha\delta}\right) = \left(\frac{2}{3}\right)^{\delta}\left(1 + 2\delta \cdot 1.1^{\delta}\right).$$

Otherwise, if $\varepsilon > (2/3)^{\delta(1-\alpha)}$ it holds that

$$\theta^{1/\delta} = \left(\frac{\varepsilon}{2\delta}\right)^{1/\delta} = \left(\frac{(2/3)^{\delta(1-\alpha)}}{2\delta}\right)^{1/\delta} = \frac{(2/3)^{(1-\alpha)}}{\sqrt[\delta]{2\delta}}.$$

It can be checked that for α defined as above it holds that $q \geqslant 0.1$. Note that q does not actually depend on θ.

Let j be the number of the level at which the maximum value of q is achieved. Then $n_{j,3} = qn_j$. We denote by $L_j(\pi)$ the predicate that the path π lies in $T_1(x)$ and goes through the left child of some vertex of level j. Similarly, we define predicates $C_j(\pi)$ and $R_j(\pi)$. By L_j we denote the predicate $\exists\pi\,(L_j(\pi))$.

We can write for the probability of the event L_j:

$$\mathsf{P}[L_j] \geqslant \mathsf{P}[\exists\pi((v_1 \in \pi \vee v_2 \in \pi \vee \ldots \vee v_{n_{j,3}} \in \pi) \wedge L_j(\pi))] = \sum_{i=1}^{n_{j,3}} \mathsf{P}[\exists\pi(v_i \in \pi \wedge L_j(\pi))].$$

Here v_i is a vertex from j-th level which has three children. For such a vertex the existence of the left child is guaranteed.

The probability $\mathsf{P}[\exists\pi(v_i \in \pi \wedge L_j(\pi)]$ for $1 \leqslant i \leqslant n_{j,3}$ is equal to the number S of paths in $T_1(x)$ passing through the left child of v_i divided by 3^{δ}. There are at most $3^{\delta-j-1}$ leaves in the subtree with root v_i. But since $T_1(x)$ is a uniformly θ-dense tree, it holds that

$$S \geqslant 3^{\delta-j-1}\theta \Rightarrow \mathsf{P}[\exists\pi(v_i \in \pi \wedge L_j(\pi))] \geqslant \frac{3^{\delta-j-1}\theta}{3^{\delta}}.$$

From this we can conclude that

$$\mathsf{P}[L_j] \geqslant n_{j,3} \cdot \frac{3^{\delta-j-1}\theta}{3^{\delta}} = \frac{n_{j,3}}{3^j} \cdot \frac{\theta}{3} = \frac{n_{j,3}}{n_j} \cdot \frac{n_j}{3^j} \cdot \frac{\theta}{3} \geqslant q \cdot \theta \cdot \frac{\theta}{3} \geqslant \frac{\theta^2}{30}.$$

Now let us find the probability P that by choosing $60/\theta^2$ branches π_j we find vertices $v_{j,0}, v_{j,1}$, and $v_{j,2}$ at the j-th level of $T_1(x)$.

$$\begin{gathered}
P = \mathsf{P}[\exists j_0, j_1, j_2\,(L_j(\pi_{j_0}) \wedge C_j(\pi_{j_1}) \wedge R_j(\pi_{j_2}))] = \\
= 1 - \mathsf{P}[\nexists j_0\, L_j(\pi_{j_0}) \vee \nexists j_1\, C_j(\pi_{j_1}) \vee \nexists j_2\, R_i(\pi_{j_2})] \geqslant \\
\geqslant 1 - \mathsf{P}[\nexists j_0\, L_j(\pi_{j_0})] - \mathsf{P}[\nexists j_1\, C_j(\pi_{j_1})] - \mathsf{P}[\nexists j_2\, R_j(\pi_{j_2})] = \\
= 1 - 3\,\mathsf{P}[\nexists j_0\, L_j(\pi_{j_0})] = 1 - 3\,\mathsf{P}[\overline{L}_j]^{60/\theta^2} = 1 - 3(1 - \mathsf{P}[L_j])^{60/\theta^2} \geqslant \\
\geqslant 1 - 3\left(1 - \frac{\theta^2}{30}\right)^{60/\theta^2} \geqslant 1 - \frac{3}{e^2}.
\end{gathered}$$

Thus, the success probability of Algorithm 1 searching vertices $v_{i,0}, v_{i,1}$, and $v_{i,2}$ is obtained from the probability of choosing a dense tree $T(x)$ and the probability P. It equals $p = \varepsilon/2\,(1 - 3/e^2) > \varepsilon/4$. ■

$\mathcal{B}$ runs the algorithm $1/p$ times. The complexity of one run is $T' = 60\,T/\theta^2$. The probability of failure is $(1-p)^{1/p}$ and, accordingly, the probability of success is $1-(1-p)^{1/p}$. Let us show that

$$1 - (1-p)^{1/p} > 1 - \frac{1}{e}.$$

Indeed, the Maclaurin series for $1/(1-p)$ and e^p are:

$$\frac{1}{1-p} = 1 + p + p^2 + \ldots, \quad e^p = 1 + \frac{p}{1!} + \frac{p^2}{2!} + \ldots,$$

thus, for all $p \in (0,1)$ holds

$$1/(1-p) > e^p \Rightarrow (1-p) < \frac{1}{e^p} \Rightarrow (1-p)^{1/p} < \frac{1}{e}.$$

The resulting complexity of the adversary $\mathcal{B}$ is $T'' = T'/p < 240\,T/(\theta^2\varepsilon)$. Let $\mathcal{B}$ solve $\mathrm{SD}(H, y, \omega)$ and $\mathrm{Coll}(h)$ with probabilities p_1 and p_2 respectively. Then

$$p_1 + p_2 \geqslant 1 - \frac{1}{e}.$$

We denote complexities of optimal algorithms solving $\mathrm{SD}(H, y, \omega)$ and $\mathrm{Coll}(h)$ with the success probability $1 - 1/e$ by $T_{\mathrm{SD},(1-1/e)}$ and $T_{\mathrm{Coll},(1-1/e)}$. Then

$$T_{\mathrm{SD},(1-1/e)} \leqslant \frac{1}{p_1} T_{\mathrm{SD},p_1} \leqslant \frac{1}{p_1} T'',$$
$$T_{\mathrm{Coll},(1-1/e)} \leqslant \frac{1}{p_2} T_{\mathrm{Coll},p_2} \leqslant \frac{1}{p_2} T''.$$

The first inequalities follow from the fact that repeating an algorithm with the success probability p_1 for $1/p_1$ times gives an algorithm with the success probability $1 - 1/e$, but possibly suboptimal. The second inequality follows from the fact that $\mathcal{B}$ solves one of two problems. Accordingly, its complexity cannot be less than the complexity of the algorithm that solves one of them. Hence,

$$T'' \geqslant p_1 T_{\mathrm{SD},(1-1/e)} \quad \text{and} \quad T'' \geqslant p_2 T_{\mathrm{Coll},(1-1/e)}.$$

Therefore, denoting $\widetilde{T} = \min\{T_{\mathrm{SD},(1-1/e)}, T_{\mathrm{Coll},(1-1/e)}\}$, we can write

$$T'' \geqslant \frac{1}{2}\big(p_1 T_{\mathrm{SD},(1-1/e)} + p_2 T_{\mathrm{Coll},(1-1/e)}\big) \geqslant \frac{1}{2}(p_1 + p_2)\,\widetilde{T} \geqslant \frac{1 - 1/e}{2}\,\widetilde{T}.$$

Equivalently,

$$\frac{960\delta^2 T}{\varepsilon^3} \geqslant \frac{1 - 1/e}{2}\,\widetilde{T}.$$

Finding ε from the last inequality and noting that $\sqrt[3]{\dfrac{1920}{1-1/e}} \leqslant 15$, we obtain

$$\varepsilon \leqslant 15\,\sqrt[3]{\delta^2 T/\widetilde{T}}.$$

Finally, for $\varepsilon > (2/3)^{\delta(1-\alpha)}$ we obtain

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) \leqslant 15 \sqrt[3]{\frac{\delta^2 T}{\min\{T_{SD}, T_{Coll}\}}} + \left(\frac{2}{3}\right)^{\delta}.$$

For an arbitrary ε it holds that

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) \leqslant \max\left\{15 \sqrt[3]{\frac{\delta^2 T}{\min\{T_{\text{SD}}, T_{\text{Coll}}\}}} + \left(\frac{2}{3}\right)^{\delta}, \left(\frac{2}{3}\right)^{\delta}\left(1 + 2\delta \cdot 1.1^{\delta}\right)\right\}.$$

Theorem 1 is proven. ■

Theorem 2. Let $\mathcal{A}$ be an adversary in the EUF-NMA model for the Stern signature scheme making at most q_f queries to the hashing oracle F. Then there exists an adversary $\mathcal{B}$ in the EUF-NMA model for the Stern signature scheme making at most one query to the hashing oracle and satisfying

$$q_f \cdot \mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \geqslant \mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) - 3^{-\delta}.$$

Furthermore, if the complexity of $\mathcal{A}$ is T, then the complexity of $\mathcal{B}$ is $T + c'q_f$, where c' is a constant depending on the model of computation.

Proof. Let $\mathbf{Exp}^0$ denote the original experiment in the EUF-NMA security model with q_f queries to the hashing oracle F. In this experiment, $\mathcal{A}$ is the adversary that makes an existential forgery for the Stern signature scheme using the random oracle F. Therefore,

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) = \mathsf{P}[\mathbf{Exp}^0(\mathcal{A}) \Rightarrow 1].$$

$\mathbf{Exp}^0(\mathcal{A})$

1 : $s \xleftarrow{\mathcal{U}} \{x \in \{0,1\}^n : \text{wt}(x) = \omega\}$
2 : $y \leftarrow Hs^T$
3 : $\Pi^F \leftarrow \varnothing$
4 : $(m, c\|r) \leftarrow\$\ \mathcal{A}^F(y)$
5 : **return** Stern.Ver$(y, m, c\|r)$

Oracle $F(\alpha)$

1 : **if** $\alpha \in \Pi^F : \beta \leftarrow \Pi^F(\alpha)$
2 : **else**
3 : $\quad\beta \xleftarrow{\mathcal{U}} \{0,1,2\}^{\delta}$
4 : $\quad\Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
5 : **return** β

Now, basing on the adversary $\mathcal{A}$, we construct an adversary $\mathcal{B}$ that makes an existential forgery in the model with one query to the random oracle. $\mathcal{B}$ simulates the oracle F that can give q_f answers to $\mathcal{A}$'s queries using algorithm SimF_t. Here the notation $\mathcal{A}^{\text{Sim}F_t}$ means that the only $\mathcal{B}$'s query to its own random oracle F^* matches the $\mathcal{A}$'s t-th query to the oracle F. Note that the output of the oracle F^* has a uniform distribution, i.e., values β obtained on lines 3 and 4 of SimF_t cannot be distinguished.

$\mathcal{B}^{F^*}(y)$

1 : $\Pi^F \leftarrow \varnothing$
2 : $j \leftarrow 0$
3 : $t \xleftarrow{\mathcal{U}} \{1, \ldots, q_f\}$
4 : $(m, c\|r) \leftarrow\$\ \mathcal{A}^{SimF_t}(y)$
5 : **return** $(m, c\|r)$

Sim$F_t(\alpha)$

1 : $j \leftarrow j + 1$
2 : **if** $(\alpha, \cdot) \in \Pi^F : \beta \leftarrow \Pi^F(\alpha)$
3 : **elseif** $j = t : \beta \leftarrow F^*(\alpha)$
4 : **else** $: \beta \xleftarrow{\mathcal{U}} \{0,1,2\}^{\delta}$
5 : $\Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
6 : **return** β

The adversary $\mathcal{A}$ can make a signature including answer for one of the queries made to the oracle F or none of them. Let I be a random variable that corresponds to the number of $\mathcal{A}$'s query to the oracle F that it uses to create a forgery. In case $\mathcal{A}$ does not use any, let $I = 0$. Hence,

$$\mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1] \geqslant \mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge t = I] \geqslant \mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge t = I \wedge I \geqslant 1] =$$
$$= \mathsf{P}[t = I]\,\mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge I \geqslant 1] \geqslant \frac{1}{q_f}\,\mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge I \geqslant 1].$$

The equality here follows from the independent random choice of t.

Note that $\mathcal{A}$'s probability of success in case it does not use any query to random oracle F is no more than as it has to guess full output $b = F(\alpha)$. From this and the definition of conditional probability holds

$$\mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1] \leqslant \mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge I \geqslant 1] + \mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge I = 0] \leqslant$$
$$\leqslant \mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1 \wedge I \geqslant 1] + 3^{-\delta}.$$

Consequently,

$$\mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1] - 3^{-\delta} \leqslant q_f \cdot \mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1].$$

From the above holds

$$q_f \cdot \mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) = q_f \cdot \mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1] \geqslant$$
$$\geqslant \mathsf{P}[\mathbf{Exp}^{0}(\mathcal{A}) \Rightarrow 1] - 3^{-\delta} = \mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{A}) - 3^{-\delta}.$$

$\mathcal{B}$ runs $\mathcal{A}$ and simulate q_f queries to oracle F, i.e., if the complexity of $\mathcal{A}$ is T, then the complexity of $\mathcal{B}$ does not exceed $T + c'q_f$ for some constant c'. ■

Theorem 3. Let $\mathcal{A}$ be an adversary in the EUF-CMA model for the Stern signature scheme making at most q_f queries to the hashing oracle F and at most q_s queries to the signing oracle Sign. Then there exists an adversary $\mathcal{B}$ in the EUF-NMA model for the Stern signature scheme making at most q_f queries to the hashing oracle and

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \geqslant \mathsf{Adv}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A}) - q_s \left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^{\delta},$$

where T_{Coll} is the complexity of optimal algorithm solving $\text{Coll}(h)$ problem with probability of success at least $1 - 1/e$ and $\tilde{c}$ is a constant depending on the model of computation.

Furthermore, if the complexity of $\mathcal{A}$ is T, then the complexity of $\mathcal{B}$ is upper bounded by $T + c''(q_f + q_s T^{\text{Sig}}_{\text{Stern}})$, where $T^{\text{Sig}}_{\text{Stern}}$ is the complexity of the signature generation algorithm and c'' is a constant depending on the model of computation.

Proof. Let $\mathbf{Exp}^0$ denote the original experiment in the EUF-CMA security model. In this experiment, $\mathcal{A}$ is the adversary that makes an existential forgery for the Stern signature scheme using the random oracle F and signing oracle Sign. $\mathcal{A}$ can make at most q_f queries to F and at most q_s queries to Sign.

$\mathbf{Exp}^0(\mathcal{A}) = \mathbf{Exp}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A})$

1 : $s \xleftarrow{\mathcal{U}} \{x \in \{0,1\}^n : \text{wt}(x) = \omega\}$
2 : $y \leftarrow Hs^T$
3 : $\mathcal{L} \leftarrow \varnothing$
4 : $\Pi^F \leftarrow \varnothing$
5 : $(m, c\|r) \leftarrow\$\ \mathcal{A}^{\text{Sign},F}(y)$
6 : **if** $m \in \mathcal{L}$: **return** 0
7 : **return** Stern.Ver$(y, m, c\|r)$

Oracle $F(\alpha)$

1 : **if** $\alpha \in \Pi^F : \beta \leftarrow \Pi^F(\alpha)$
2 : **else**
3 : $\quad\beta \xleftarrow{\mathcal{U}} \{0,1,2\}^\delta$
4 : $\quad\Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
5 : **return** β

Oracle Sign(s, m)

1 : **foreach** $0 \leqslant i < \delta$:
2 : $\quad u_i \xleftarrow{\mathcal{U}} \{0,1\}^n, \sigma_i \xleftarrow{\mathcal{U}} S_n$
3 : $\quad c_{i,0} \leftarrow h(\sigma_i \| Hu_i^T)$
4 : $\quad c_{i,1} \leftarrow h(\sigma_i(u_i))$
5 : $\quad c_{i,2} \leftarrow h(\sigma_i(u_i \oplus s))$
6 : $\quad c_i \leftarrow c_{i,0}\|c_{i,1}\|c_{i,2}$
7 : $c \leftarrow c_0\| \ldots \|c_{\delta-1}$
8 : $b \leftarrow F(m\|c)$
9 : **foreach** $0 \leqslant i < \delta$:
10 : $\quad$ **if** $b_i = 0 : r_i \leftarrow \sigma_i\|u_i$
11 : $\quad$ **if** $b_i = 1 : r_i \leftarrow \sigma_i\|(u_i \oplus s)$
12 : $\quad$ **if** $b_i = 2 : r_i \leftarrow \sigma_i(u_i)\|\sigma_i(s)$
13 : $r \leftarrow r_0\| \ldots \|r_{\delta-1}$
14 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{m\}$
15 : **return** $c\|r$

$$\mathsf{Adv}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A}) = \mathsf{P}[\mathbf{Exp}^0(\mathcal{A}) \Rightarrow 1].$$

The experiment $\mathbf{Exp}^1$ is a modification of $\mathbf{Exp}^0$ obtained by introducing sets $\Pi^S, \Pi \subset \{0,1\}^* \times \{0,1,2\}^\delta$, Π^S is filled while communicating with the oracle Sign and $\Pi = \Pi^F \cup \Pi^S$.

Modifications of algorithms F and Sign do not affect the distributions of their outputs, therefore,

$$\mathsf{P}[\mathbf{Exp}^0(\mathcal{A}) \Rightarrow 1] = \mathsf{P}[\mathbf{Exp}^1(\mathcal{A}) \Rightarrow 1].$$

The experiment $\mathbf{Exp}^2$ differs from $\mathbf{Exp}^1$ only in algorithm Sign. Now it does not use the secret key, and the result is formed by a random vector b.

We show that the distributions of outputs $c\|r$ of the algorithm Sign in experiments $\mathbf{Exp}^1$ and $\mathbf{Exp}^2$ are indistinguishable if the condition on the line 23 was not satisfied. If we show that the distribution of each such part $c_i\|r_i = c_{i,0}\|c_{i,1}\|c_{i,2}\|r_{i,0}\|r_{i,1}$ for $0 \leqslant i \leqslant \delta - 1$ in $\mathbf{Exp}^2$ coincides with the distribution of the corresponding part in $\mathbf{Exp}^1$, then the distributions of the signatures also coincide.

Further, we will consider arguments of the hash function h corresponding to $c_{i,j}$ instead of the values themselves. The reason for it is the fact that if distributions of variables ξ and η coincide, then distributions of variables $h(\xi)$ and $h(\eta)$ also coincide. Indeed,

$$\mathsf{P}[h(\xi) = a] = \mathsf{P}[\xi \in h^{-1}(a)] = \mathsf{P}[\eta \in h^{-1}(a)] = \mathsf{P}[h(\eta) = a].$$

$\mathbf{Exp}^1(\mathcal{A})$

$s \xleftarrow{\mathcal{U}} \{x \in \{0,1\}^n : \mathrm{wt}(x) = \omega\}$
$y \leftarrow Hs^T$
$\mathcal{L} \leftarrow \varnothing$
$(\Pi^F, \Pi^S) \leftarrow (\varnothing, \varnothing)$
$\Pi \leftarrow \Pi^F \cup \Pi^S$
$(m, c\|r) \leftarrow\$\ \mathcal{A}^{\mathrm{Sign},F}(y)$
if $m \in \mathcal{L}$: **return** 0
return Stern.Ver$(y, m, c\|r)$

Oracle $F(\alpha)$

if $(\alpha, \cdot) \in \Pi$: **return** $\Pi(\alpha)$
$\beta \xleftarrow{\mathcal{U}} \{0,1,2\}^\delta$
$\Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
$\Pi \leftarrow \Pi^F \cup \Pi^S$
return β

Oracle Sign(s, m) $(\mathbf{Exp}^1)$

foreach $0 \leqslant i < \delta$:
  $u_i \xleftarrow{\mathcal{U}} \{0,1\}^n, \sigma_i \xleftarrow{\mathcal{U}} S_n$
  $c_{i,0} \leftarrow h(\sigma_i\|Hu_i^T)$
  $c_{i,1} \leftarrow h(\sigma_i(u_i))$
  $c_{i,2} \leftarrow h(\sigma_i(u_i \oplus s))$
  $c_i \leftarrow c_{i,0}\|c_{i,1}\|c_{i,2}$
$c \leftarrow c_0\| \ldots \|c_{\delta-1}$
if $(m\|c, \cdot) \in \Pi : b \leftarrow \Pi(m\|c)$
else
  $b \xleftarrow{\mathcal{U}} \{0,1,2\}^\delta$
  $\Pi^S \leftarrow \Pi^S \cup \{(m\|c, b)\}$
  $\Pi \leftarrow \Pi^F \cup \Pi^S$
foreach $0 \leqslant i < \delta$:
  if $b_i = 0 : r_i \leftarrow \sigma_i\|u_i$
  if $b_i = 1 : r_i \leftarrow \sigma_i\|(u_i \oplus s)$
  if $b_i = 2 : r_i \leftarrow \sigma_i(u_i)\|\sigma_i(s)$
$r \leftarrow r_0\| \ldots \|r_{\delta-1}$
$\mathcal{L} \leftarrow \mathcal{L} \cup \{m\}$
return $c\|r$

Oracle Sign(m) $(\mathbf{Exp}^2)$

$s' \xleftarrow{\mathcal{U}} \{x \in \{0,1\}^n : \mathrm{wt}(x) = \omega\}$
foreach $0 \leqslant i < \delta$:
  $b_i \xleftarrow{\mathcal{U}} \{0,1,2\}$
  $u'_i \xleftarrow{\mathcal{U}} \{0,1\}^n, \sigma'_i \xleftarrow{\mathcal{U}} S_n,$
  if $b_i = 0$:
    $c_{i,0} \leftarrow h(\sigma'_i\|Hu_i'^T)$
    $c_{i,1} \leftarrow h(\sigma'_i(u'_i))$
    $c_{i,2} \leftarrow h(\sigma'_i(u'_i \oplus s'))$
    $r_i \leftarrow \sigma'_i\|u'_i$
  if $b_i = 1$:
    $c_{i,0} \leftarrow h(\sigma'_i\|(Hu_i'^T \oplus y))$
    $c_{i,1} \leftarrow h(\sigma'_i(s'))$
    $c_{i,2} \leftarrow h(\sigma'_i(u'_i))$
    $r_i \leftarrow \sigma'_i\|u'_i$
  if $b_i = 2$:
    $c_{i,0} \leftarrow h(\sigma'_i\|H(u'_i \oplus s')^T)$
    $c_{i,1} \leftarrow h(\sigma'_i(u'_i \oplus s'))$
    $c_{i,2} \leftarrow h(\sigma'_i(u'_i))$
    $r_i \leftarrow \sigma'_i(u'_i \oplus s')\|\sigma'_i(s')$
  $c_i \leftarrow c_{i,0}\|c_{i,1}\|c_{i,2}$
$c \leftarrow c_0\| \ldots \|c_{\delta-1}$
$r \leftarrow r_0\| \ldots \|r_{\delta-1}$
if $(m\|c, \cdot) \in \Pi^F$: **abort**
$\Pi^S \leftarrow \Pi^S \cup \{(m\|c, b)\}$
$\Pi \leftarrow \Pi^F \cup \Pi^S$
$\mathcal{L} \leftarrow \mathcal{L} \cup \{m\}$
return $c\|r$

In the case $b_i = 0$, for an external observer, the secret key s is a random variable. And other values are randomly selected in the same way as in the original protocol. So distributions, obviously, coincide.

If $b_i = 1$, then the probability that in $\mathbf{Exp}^1$ the string $c_i \| r_i$ equals $a_1 \| a_2 \| a_3 \| a_4 \| a_5 \| a_6$ is

$$P_{a_1,a_2,a_3,a_4,a_5,a_6} = \mathsf{P}\left[\sigma_i = a_1, Hu_i^{\mathrm{T}} = a_2, \sigma_i(u_i) = a_3, \sigma_i(u_i \oplus s) = a_4, \sigma_i = a_5, u_i \oplus s = a_6\right] =$$
$$= \mathbb{I}\big[a_1 = a_5, H(a_6 \oplus s)^{\mathrm{T}} = a_2, a_1(a_6) = a_4\big]\ \mathsf{P}\left[\sigma_i = a_1, s = a_1^{-1}(a_3) \oplus a_6, u_i = a_1^{-1}(a_3)\right],$$

where $\mathbb{I}\big[\theta\big]$ is the indicator of expression θ. In $\mathbf{Exp}^2$ this probability is

$$\widehat{P}_{a_1,a_2,a_3,a_4,a_5,a_6} = \mathsf{P}\left[\sigma'_i = a_1, Hu_i'^{\mathrm{T}} \oplus y = a_2, \sigma'_i(s') = a_3, \sigma'_i(u'_i) = a_4, \sigma'_i = a_5, u'_i = a_6\right] =$$
$$= \mathbb{I}\big[a_1 = a_5, Ha_6^{\mathrm{T}} \oplus y = a_2, a_1(a_6) = a_4\big]\ \mathsf{P}\left[\sigma'_i = a_1, s' = a_1^{-1}(a_3), u'_i = a_6\right].$$

As far as $H(a_6 \oplus s)^{\mathrm{T}} = Ha_6^{\mathrm{T}} \oplus y$, indicators of these two expressions coincide.

Let us evaluate the probabilities. Note that since all random variables are selected independently, the probability of the conjunction of events equals to the product of their probabilities. So we can find them separately:

$$\mathsf{P}[\sigma_i = a_1] = \Pr[\sigma'_i = a_1] = \frac{1}{n!},$$
$$\mathsf{P}[s = a_1^{-1}(a_3) \oplus a_6 = a'] = \frac{1}{2^n},$$
$$\mathsf{P}[u_i = a_1^{-1}(a_3) = a''] = \frac{1}{2^n},$$
$$\mathsf{P}[s' = a_1^{-1}(a_3) = a'''] = \frac{1}{2^n},$$
$$\mathsf{P}[u'_i = a_6] = \frac{1}{2^n}$$

for any constants a', a'' and a'''. Then

$$\mathsf{P}\left[\sigma_i = a_1, s = a_1^{-1}(a_3) \oplus a_6, u_i = a_1^{-1}(a_3)\right] = \mathsf{P}\left[\sigma'_i = a_1, s' = a_3, u'_i = a_6\right] = \frac{1}{n!2^{2n}}$$

and distributions are indistinguishable.

Finally, if $b_i = 2$, then the similar probability in $\mathbf{Exp}^1$ is

$$P_{a_1,a_2,a_3,a_4,a_5,a_6} = \mathsf{P}\left[\sigma_i = a_1, Hu_i^{\mathrm{T}} = a_2, \sigma_i(u_i) = a_3, \sigma_i(u_i \oplus s) = a_4, \sigma_i(u_i) = a_5, \sigma_i(s) = a_6\right] =$$
$$= \mathbb{I}\big[a_3 = a_5, a_3 \oplus a_6 = a_4, H(a_1^{-1}(a_3))^{\mathrm{T}} = a_2\big]\ \mathsf{P}\left[\sigma_i = a_1, u_i = a_1^{-1}(a_3), s = a_1^{-1}(a_6)\right]$$

and in $\mathbf{Exp}^2$ it is

$$\widehat{P}_{a_1,a_2,a_3,a_4,a_5,a_6} =$$
$$= \mathsf{P}\left[\sigma'_i = a_1, H(u'_i \oplus s')^{\mathrm{T}} = a_2, \sigma'_i(u'_i \oplus s') = a_3, \sigma'_i(u'_i) = a_4, \sigma'_i(u'_i \oplus s') = a_5, \sigma'_i(s') = a_6\right] =$$
$$= \mathbb{I}\big[a_3 = a_5, a_4 \oplus a_6 = a_3, H(a_1^{-1}(a_3))^{\mathrm{T}} = a_2\big]\ \mathsf{P}\left[\sigma'_i = a_1, u'_i = a_1^{-1}(a_4), s' = a_1^{-1}(a_6)\right].$$

Reasoning similar to above, we get

$$\mathsf{P}\left[\sigma_i = a_1, u_i = a_1^{-1}(a_3), s = a_1^{-1}(a_6)\right] = \mathsf{P}\left[\sigma'_i = a_1, u'_i = a_1^{-1}(a_4), s' = a_1^{-1}(a_6)\right] = \frac{1}{n!2^{2n}},$$

and distributions coincide.

The check on the line 23 corresponds to the case when the value c, created while generating a signature for the message m, already exists in the set Π^F. Let us consider the

worst case and suppose that $\mathcal{A}$ makes q_f queries to the hashing oracle F first. We denote by p' the probability that for some message at one of q_s queries the following condition was satisfied:

$$p' = \mathsf{P}\left[c \in \Pi_c^F\right],$$

where

$$\Pi_c^F = \{c \in \{0,1\}^{3\delta\ell} : \exists m \in \{0,1\}^* \, \exists \beta \in \{0,1,2\}^\delta \ \left((m\|c,\beta) \in \Pi^F\right)\}.$$

For a string c and a set $\Pi \subset \{0,1\}^* \times \{0,1\}^{3\delta\ell} \times \{0,1,2\}^\delta$, we can introduce the projection of this set to the part of the string:

$$\Pi_{c,(0,1)} = \{c_{0,1} : \exists m \in \{0,1\}^* \, \exists c \in \{0,1\}^{3\delta\ell} \, \exists \beta \in \{0,1,2\}^\delta \ ((m\|c,\beta) \in \Pi)\}.$$

Similarly, for a tuple $c_{*,1} = (c_{0,1}, \ldots, c_{\delta-1,1})$ we define

$$\Pi_{c,(*,1)} = \{c_{*,1} : \exists m \in \{0,1\}^* \, \exists c \in \{0,1\}^{3\delta\ell} \, \exists \beta \in \{0,1,2\}^\delta \ ((m\|c,\beta) \in \Pi)\}.$$

Then for $\Pi \subset \{0,1\}^* \times \{0,1\}^{3\delta\ell} \times \{0,1,2\}^\delta$ we can claim

$$p' \leqslant \sum_{\Pi} \mathsf{P}[\Pi^F = \Pi \wedge c_{*,1} \in \Pi_{c,(*,1)}].$$

Note that the first event is determined by $\mathcal{A}$'s random oracle whereas the second is determined by the signing one. That is, the events are independent and

$$p' \leqslant \sum_{\Pi} \mathsf{P}[\Pi^F = \Pi] \ \mathsf{P}[c_{*,1} \in \Pi_{c,(*,1)}].$$

Since each c_i is a function of independent random variables for $i \in \{0, \delta-1\}$, then events $c_{i,1} \in \Pi^F_{c,(i,1)}$ are also independent. Hence

$$p' \leqslant \sum_{\Pi} \mathsf{P}[\Pi^F = \Pi] \prod_{i=0}^{\delta-1} \mathsf{P}[c_{i,1} \in \Pi_{c,(i,1)}] \leqslant \sum_{\Pi} \mathsf{P}[\Pi^F = \Pi] \left(\max_{i\in\{0,\delta-1\}} \mathsf{P}[c_{i,1} \in \Pi_{c,(i,1)}]\right)^\delta \leqslant$$

$$\leqslant \left(\max_{\Pi} \max_{i\in\{0,\delta-1\}} \mathsf{P}[c_{i,1} \in \Pi_{c,(i,1)}]\right)^\delta \sum_{\Pi} \mathsf{P}[\Pi^F = \Pi] =$$

$$= \left(\max_{\Pi} \max_{i\in\{0,\delta-1\}} \mathsf{P}[c_{i,1} \in \Pi_{c,(i,1)}]\right)^\delta \leqslant \left(\max_{\Pi} \max_{i\in\{0,\delta-1\}} \sum_{y\in\Pi_{c,(i,1)}} \mathsf{P}[c_{i,1} = y]\right)^\delta \leqslant$$

$$\leqslant \left(\max_{\Pi} \max_{i\in\{0,\delta-1\}} \left(\left|\Pi_{c,(i,1)}\right| \max_{y\in\Pi_{c,(i,1)}} \mathsf{P}[c_{i,1} = y]\right)\right)^\delta \leqslant \left(q_f \max_{i\in\{0,\delta-1\}} \max_{y\in\{0,1\}^\ell} \mathsf{P}[h(x_i) = y]\right)^\delta \leqslant$$

$$\leqslant \left(q_f \sum_{i\in\{0,\delta-1\}} \max_{y\in\{0,1\}^\ell} \mathsf{P}[h(x_i) = y]\right)^\delta \leqslant \left(\delta q_f \max_{y\in\{0,1\}^\ell} \mathsf{P}[h(x) = y]\right)^\delta.$$

We can denote $p_h = \max\limits_{y\in\{0,1\}^\ell} \mathsf{P}[h(x) = y]$. Then there exists an algorithm $\mathcal{C}$ that evaluates hash function collision by the following steps: $\mathcal{C}$ chooses t random inputs x_i and evaluates their hash values. It stops after finding a collision. Then the probability that $\mathcal{C}$ fails decomposes into two incompatible events: there is no such $x \in \{x_i : i = 1, \ldots, t\}$ that $h(x) = y$ and there is only one such x. We claim the algorithm makes not more than $14/p_h$ steps and has the success probability at least $1 - 1/e$.

To prove this fact, we separate to cases: $p_h \leqslant 0.25$ and $p_h > 0.25$. In the first one, the number of steps can be taken equal to $t_1 = 3/p_h$. Then

$$\mathsf{P}[\mathcal{C} \text{ failes}] = (1-p_h)^{t_1} + \sum_{i=1}^{t_1} p_h(1-p_h)^{t_1-1} = (1-p_h)^{t_1} + t_1 p_h (1-p_h)^{t_1-1} \leqslant$$
$$\leqslant e^{-3} + \frac{(3/p_h)\, p_h e^{-3}}{1-p_h} = e^{-3}\left(1 + \frac{3}{1-p_h}\right) \leqslant 5e^{-3} < e^{-1}.$$

Here we used the fact that for $z > 0$ the inequality $(1+w)^z \leqslant e^{wz}$ holds.

If $p_h > 0.25$, we can fix $t_2 = 14$ and

$$\mathsf{P}[\mathcal{C} \text{ failes}] = (1-p_h)^{t_2} + \sum_{i=1}^{t_2} p_h(1-p_h)^{t_2-1} = (1-p_h)^{t_2} + t_2 p_h (1-p_h)^{t_2-1} \leqslant$$
$$\leqslant 0.75^{t_2} + t_2 p_h \cdot 0.75^{t_2-1} = 0.75^{t_2-1}(0.75 + t_2 p_h) < 0.75^{t_2-1}(1+t_2) < e^{-1}.$$

The complexity in the first case is $3\tilde{c}/p_h$ and in the second one it is $14\,\tilde{c}$, where $\tilde{c}$ is a constant depending on the model of computation and corresponding to the complexity of one hash function evaluation. So we can estimate the resulting complexity of the whole algorithm as $\check{T} = 14\tilde{c}/p_h$. Note that if $p_h > 0$, then the algorithm stops with probability equal to 1. Indeed, $\mathsf{P}[\mathcal{C} \text{ failes}]$ tends to zero as t tends to infinity.

Then for the complexity of the optimal algorithm T_{Coll} solving the problem $\text{Coll}(h)$ with probability at least $1 - 1/e$ it holds that

$$T_{\text{Coll}} \leqslant \check{T} \quad \text{and} \quad p_h \leqslant \frac{14\,\tilde{c}}{T_{\text{Coll}}}.$$

Finally, we obtain

$$p' \leqslant \left(\frac{14\,\tilde{c}\,\delta\, q_f}{T_{\text{Coll}}}\right)^{\delta}.$$

Let us denote by G the event that the condition on the line 23 was never satisfied. On the other hand, if the event $\overline{G}$ happened, then the condition was satisfied at least once during q_f $\mathcal{A}$'s queries. If $m^i \| c^i$ are strings formed in signature generating algorithm, then

$$\mathsf{P}[\overline{G}] = \mathsf{P}[m^1\|c^1 \in \Pi^F \vee \cdots \vee m^{q_s}\|c^{q_s} \in \Pi^F] \leqslant \mathsf{P}[c^1 \in \Pi_c^F \vee \cdots \vee c^{q_s} \in \Pi_c^F] =$$
$$= 1 - \mathsf{P}[c^1 \notin \Pi_c^F \wedge \cdots \wedge c^{q_s} \notin \Pi_c^F] = 1 - \prod_{i=1}^{q_s} \mathsf{P}[c^i \notin \Pi_c^F] =$$
$$= 1 - \prod_{i=1}^{q_s} (1 - \mathsf{P}[c^i \in \Pi_c^F]) = 1 - (1-p')^{q_s}.$$

Then, using the Bernoulli's inequality together with the fact $p' \leqslant 1$ and $q_s > 0$, we obtain

$$\mathsf{P}[\overline{G}] \leqslant 1 - 1 + q_s p' \leqslant q_s \left(\frac{14\,\tilde{c}\,\delta\, q_f}{T_{\text{Coll}}}\right)^{\delta}.$$

Since

$$\mathsf{P}[\mathbf{Exp}^1(\mathcal{A}) \Rightarrow 1] = \mathsf{P}[\mathbf{Exp}^1(\mathcal{A}) \Rightarrow 1 \wedge G] + \mathsf{P}[\mathbf{Exp}^1(\mathcal{A}) \Rightarrow 1 \wedge \overline{G}] \leqslant$$
$$\leqslant \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1] + \mathsf{P}[\overline{G}],$$

then

$$\mathsf{P}[\mathbf{Exp}^1(\mathcal{A}) \Rightarrow 1] - \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1] \leqslant \mathsf{P}[\overline{G}] = q_s \left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^{\delta}.$$

Now, based on the adversary $\mathcal{A}$, we construct an adversary $\mathcal{B}$ that makes an existential forgery in the EUF-NMA model. It simulates oracles F and Sign using algorithms Sim$F$ and SimSign. The algorithm SimSign repeats the algorithm Sign in the experiment $\mathbf{Exp}^2$. The oracle F^* is the random oracle of $\mathcal{B}$.

$\mathcal{B}^{F^*}(y)$

1 : $\mathcal{L} \leftarrow \varnothing$
2 : $\Pi^F \leftarrow \varnothing$
3 : $(m, c\|r) \leftarrow\$\ \mathcal{A}^{\text{SimSign},\text{Sim}F}(y)$
4 : **if** $m \in \mathcal{L}$: **return** 0
5 : **return** $(m, c\|r)$

Oracle Sim$F(\alpha)$

1 : $\beta \leftarrow F^*(\alpha)$
2 : $\Pi^F \leftarrow \Pi^F \cup \{(\alpha, \beta)\}$
3 : **return** β

Now let us denote by $\text{Out}(\mathcal{A})$ and $\text{Out}(\mathcal{B})$ pairs $(m, c\|r)$ that are outputs of the adversary $\mathcal{A}$ in the experiment $\mathbf{Exp}^2$ and the adversary $\mathcal{B}$ in the experiment EUF-NMA, respectively. $\text{Out}(\mathcal{A})_{m,c}$ and $\text{Out}(\mathcal{B})_{m,c}$ are projections of the adversaries outputs to $m\|c$. Note that the projection is not defined if an adversary returns zero (in case $m \in \mathcal{L}$). But further we consider only $\mathbf{Exp}^2$ and EUF-NMA experiments with output equal to one, which excludes this case. We also define

$$\text{V}(G, m, c, r) = \text{Ver}(y, m, c\|r),$$

where Ver is the signature verification algorithm that uses function G. Then for $m \in \{0,1\}^*$, $c \in \{0,1\}^{3\delta\ell}$ we define

$$\begin{aligned} p_{m,c} &= \mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1 \wedge \text{Out}(\mathcal{B})_{m,c} = m\|c] = \\ &= \mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1 \wedge \text{Out}(\mathcal{B})_{m,c} = m\|c \wedge m\|c \in \Pi^F] + \\ &+ \mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1 \wedge \text{Out}(\mathcal{B})_{m,c} = m\|c \wedge m\|c \notin \Pi^F]. \end{aligned}$$

If $m\|c \notin \Pi^F$, then the adversary does not know the correct hash-value for this string and has to guess it. This is possible with the probability equal to $3^{-\delta}$. Also, note that the result of the EUF-NMA experiment equals the result of the signature verification algorithm, so it holds

$$\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) = \text{V}(F, m, c, r). \tag{3}$$

Hence,

$$p_{m,c} = \mathsf{P}[\text{V}(F, m, c, r) \wedge \text{Out}(\mathcal{B})_{m,c} = m\|c \wedge m\|c \in \Pi^F] + 3^{-\delta}.$$

For $m\|c \in \Pi^F$ it holds that

$$\text{V}(F^*, m, c, r) = \text{V}(F, m, c, r)$$

as $\mathcal{A}$'s hashing oracle F is strictly determined by $\mathcal{B}$'s oracle F^*. Then

$$p_{m,c} = \mathsf{P}[\text{V}(F^*, m, c, r) \wedge \text{Out}(\mathcal{B})_{m,c} = m\|c \wedge m\|c \in \Pi^F] + 3^{-\delta}.$$

Similar to (3) we claim

$$\mathbf{Exp}^2(\mathcal{A}) = \mathrm{V}(F^*, m, c, r).$$

The adversary $\mathcal{B}$ always returns $\mathcal{A}$'s output so

$$\mathrm{Out}(\mathcal{B})_{m,c} = \mathrm{Out}(\mathcal{A})_{m,c}.$$

Finally, the $\mathcal{A}$'s strategy in the case $m\|c \notin \Pi^F$ is the same as $\mathcal{B}$'s and therefore has the same success probability $(1/3)^\delta$.

From the proceeding argument it becomes clear that

$$\begin{gathered} p_{m,c} = \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1 \wedge \mathrm{Out}(\mathcal{A})_{m,c} = m\|c \wedge m\|c \in \Pi^F] + \\ + \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1 \wedge \mathrm{Out}(\mathcal{A})_{m,c} = m\|c \wedge m\|c \notin \Pi^F] = \\ = \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1 \wedge \mathrm{Out}(\mathcal{A})_{m,c} = m\|c]. \end{gathered}$$

Thus,

$$\mathsf{P}[\mathbf{Exp}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \Rightarrow 1] = \mathsf{P}[\mathbf{Exp}(\mathcal{B}) \Rightarrow 1 \wedge \bigvee_{(m,c)} \mathrm{Out}(\mathcal{B})_{m,c} = m\|c] =$$

$$= \sum_{(m,c)} p_{m,c} = \sum_{(m,c)} \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1 \wedge \mathrm{Out}(\mathcal{A})_{m,c} = m\|c] = \mathsf{P}[\mathbf{Exp}^2(\mathcal{A}) \Rightarrow 1].$$

Consequently,

$$\mathsf{Adv}^{\text{EUF-NMA}}_{\text{Stern}}(\mathcal{B}) \geqslant \mathsf{Adv}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A}) - q_s \left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^\delta.$$

The adversary $\mathcal{B}$ runs $\mathcal{A}$ and simulates q_f queries to the oracle F and q_s queries to the oracle Sign. Note that the complexity of the oracle Sign does not exceed the complexity of the original signing algorithm. Hence, $\mathcal{B}$'s complexity is no more than $T + c''(q_f + q_s T^{\text{Sig}}_{\text{Stern}})$. ■

Corollary 1. Let $\mathcal{A}$ be an adversary in the EUF-CMA model for the Stern signature scheme making at most q_f queries to the hashing oracle F and at most q_s queries to the signing oracle Sign. Then it holds that

$$\begin{gathered} \mathsf{Adv}^{\text{EUF-CMA}}_{\text{Stern}}(\mathcal{A}) \leqslant \\ \leqslant \max\left\{ 15 q_f \sqrt[3]{\frac{\delta^2 (T + \tilde{\tilde{c}}(2q_f + q_s T^{\text{Sig}}_{\text{Stern}}))}{\min\{T_{\text{SD}}, T_{\text{Coll}}\}}} + \left(\frac{2}{3}\right)^\delta (1 + q_f) + q_s \left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^\delta, \right. \\ \left. \left(\frac{2}{3}\right)^\delta \left(1 + q_f \left(1 + 2\delta \cdot 1.1^\delta\right)\right) + q_s \left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^\delta \right\}, \end{gathered}$$

where $T^{\text{Sig}}_{\text{Stern}}$ is the complexity of the signature generation algorithm, T is the maximum possible time complexity of $\mathcal{A}$, T_{SD} and T_{Coll} are complexities of optimal algorithms solving $\mathrm{SD}(H, y, \omega)$ and $\mathrm{Coll}(h)$ problems with probabilities of success at least $1 - 1/e$, $\tilde{c}$ and $\tilde{\tilde{c}}$ are constants depending on the model of computation.

Proof. The complexity $\hat{T}$ of an adversary in the EUF-NMA model with one query to the hashing oracle from Theorems 1–3 does not exceed $T + \tilde{\tilde{c}}(2q_f + q_s T^{\text{Sig}}_{\text{Stern}})$, where T is the complexity of an adversary in the EUF-CMA model and $\tilde{\tilde{c}} = \max\{c', c''\}$.

Also, for an adversary $\mathcal{B}$ in the EUF-NMA model making at most q_f queries to the hashing oracle follows that

$$\mathsf{Adv}_{\text{Stern}}^{\text{EUF-NMA}}(\mathcal{B}) \leqslant$$

$$\leqslant \max\left\{15q_f\sqrt[3]{\frac{\delta^2\hat{T}}{\min\{T_{\text{SD}},T_{\text{Coll}}\}}}+\left(\frac{2}{3}\right)^{\delta}(1+q_f),\left(\frac{2}{3}\right)^{\delta}\left(1+q_f\left(1+2\delta\cdot 1.1^{\delta}\right)\right)\right\},$$

$$\mathsf{Adv}_{\text{Stern}}^{\text{EUF-CMA}}(\mathcal{A}) \leqslant \mathsf{Adv}_{\text{Stern}}^{\text{EUF-NMA}}(\mathcal{B}) + q_s\left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^{\delta} \leqslant$$

$$\leqslant \max\left\{15q_f\sqrt[3]{\frac{\delta^2\hat{T}}{\min\{T_{\text{SD}},T_{\text{Coll}}\}}}+\left(\frac{2}{3}\right)^{\delta}(1+q_f)+q_s\left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^{\delta},\right.$$

$$\left.\left(\frac{2}{3}\right)^{\delta}\left(1+q_f\left(1+2\delta\cdot 1.1^{\delta}\right)\right)+q_s\left(\frac{14\,\tilde{c}\,\delta\,q_f}{T_{\text{Coll}}}\right)^{\delta}\right\}.$$

Corollary 1 is proven. ■

5. Parameters

In this Section, we mention different constraints on the signature parameters and introduce a parameter set.

5.1. Choosing general parameters

J. Stern showed [19] that his identification scheme can be forged with probability equal to 2/3. Similar reasoning allows to assert that an adversary can build a forgery of the signature without knowing the secret key with the probability $(2/3)^{\delta}$. Thus, parameter δ should be chosen to satisfy the condition

$$\left(\frac{2}{3}\right)^{\delta} < 2^{-\lambda},$$

where λ is the security parameter.

As it was mentioned above, the complexity of the best collision-finding algorithm is $\mathcal{O}\left(2^{\ell/2}\right)$, where ℓ is the length of the hash value. Since we want to maximize this complexity, it is worth using hash functions with the maximal ℓ. This can be, for example, such well-known functions as the American standard SHA3-512 or the Russian standard Streebog-512, in which $\ell = 512$ bits. For them $T_{\text{Coll}} \approx 2^{256}$ bit operations. Further by default we suppose that hash function Streebog-512 is used.

In order to maximize the value of $\min\{T_{\text{SD}}, T_{\text{Coll}}\}$, it is necessary to choose the length of the code so that $T_{\text{SD}} \geqslant T_{\text{Coll}}$. From the fact that $T_{\text{SD}} \approx 2^{0.0885n}$ [29], a lower estimate for the length of the code n can be found. We choose the code dimension as $k = n/2$ and require the code to lie on the Varshamov — Gilbert boundary:

$$\frac{k}{n} = 1 - H\left(\frac{\omega}{n}\right),$$

whence it follows that $\omega \approx 0.11n$.

5.2. Public data and signature sizes

The public key is a vector y of $n-k$ bits in size. The public parameter H is an $(n-k)\times n$-matrix that can be stored as $k(n-k)$ bits in systematic form.

The size of c is $3\delta\ell$ bits. The maximal size of r_i is $n + n\log_2 n$ bits and, accordingly, size of r can be upper estimated as $\delta(n + n\log_2 n)$ bits. So the total size of the signature can be estimated from above as $\delta(3\ell + n + n\log_2 n)$ bits.

5.3. Example parameter set

Parameters q_f, q_s, T, and $T^{\text{Sig}}_{\text{Stern}}$ on practice depend on the desired level of security and the realization of hash functions. We assume that the maximal complexity T of the adversary does not exceed 2^{70} bit operations. The structure of function f we specify as follows:

$$f(x) = \left\lfloor \frac{h'(x)\, 3^{\delta}}{2^{256}} \right\rfloor,$$

where h' is Streebog-256 hash function and the output is considered as a ternary vector. So the complexity of h' dominates in the complexity of f. Each of q_f queries to the hashing oracle consists in evaluation of the hash function of messages, which in practice can be several megabytes in size. In this case, according to [30], the complexity of Streebog is about 2^{25} CPU cycles or more than 2^{30} bit operations. Signing oracle has to evaluate at least hash function f, thus $q_s \leqslant q_f$. As a result, an adversary with complexity T is able to do no more that 2^{40} queries to each oracle.

The value $T^{\text{Sig}}_{\text{Stern}}$ consists of the complexity of function f and the complexity of δ computations of the signature components. The complexity of one component (c_i, r_i) computation includes the triple calculation of the hash function h, which can be estimated as $2^{23.5}$ bit operations using Streebog realization from [30], and other calculations, the complexity of which in total equals 2^{22} bit operations. Finally, the computation of δ components requires 2^{31} bit operations, and $T^{\text{Sig}}_{\text{Stern}}$ is about $2^{31.5}$ bit operations.

The table presents an example parameter set for signature with 70-bit security.

λ	n	k	ω	δ	ℓ	H, MB	y, KB	ζ, MB
70	2896	1448	318	137	512	0.25	0.18	0.62

The adversary's advantage in this case equals approximately $1.5 \cdot 10^{-4}$.

Although the introduced parameter set guarantees 70 bits of security, we presume that our estimate is rather rough and in fact one can count on larger security level.

6. Conclusion

The paper presents the security bounds for a digital signature based on the Stern identification protocol. We connect the security of the scheme with the hardness of syndrome decoding and hash function collision finding problems. Basing on the security notions, we introduce a parameter set providing 70-bit security of the signature. As a direction for further research, we consider the refinement of the obtained estimate and the extension of the security proof to a model with quantum access to the random oracle.

7. Acknowledgments

The authors thank Lev Vysotsky, Liliya Akhmetzyanova, Alexandra Babueva and Kirill Tsaregorodtsev for helpful discussions.

REFERENCES

1. *Shor P. V.* Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Computing, 1997, vol. 26, no. 5, pp. 1484–1509.

2. https://csrc.nist.gov/Projects/post-quantum-cryptography/Post-Quantum-Cryptography-Standardization/Call-for-Proposals — NIST PQC Call for Proposals, 2016.
3. *Lee W., Kim Y.-S., Lee Y.-W., and No J.-S.* Post quantum signature scheme based on modified Reed — Muller code pqsigRM. First round submission to the NIST post-quantum cryptography call, 2017, https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/pqsigRM.zip.
4. *Fukushima K., Roy P. S., Xu R., et al.* Supporting documentation of RaCoSS (Random Code-based Signature Scheme). First round submission to the NIST post-quantum cryptography call, 2017, https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/RaCoSS.zip.
5. *Aragon N., Gaborit P., Hauteville A., et al.* RankSign — a signature proposal for the NIST's call. First round submission to the NIST post-quantum cryptography call, 2017, https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/RankSign.zip.
6. *Debris-Alazard T. and Tillich J.-P.* Two attacks on rank metric code-based schemes: RankSign and an IBE scheme. LNCS, 2018, vol. 11272, pp. 62–92.
7. *Lee Y., Lee W., Kim Y. S., and No J.-S.* Modified pqsigRM: RM code-based signature scheme. IEEE Access, 2020, vol. 8, pp. 177506–177518.
8. *Roy P. S., Morozov K., Fukushima K., et al.* Code-based signature scheme without trapdoors. IEICE Tech. Rep., 2018, vol. 118, no. 151, pp. 17–22.
9. *Xagawa K.* Practical Attack on RaCoSS-R. Cryptology ePrint Archive, 2018, Report 2018/831, http://eprint.iacr.org/
10. *Kabatianskii G., Krouk E., and Smeets B.* A digital signature scheme based on random error-correcting codes. LNCS, 1997, vol. 1355, pp. 161–167.
11. *Cayrel P.-L., Otmani A., and Vergnaud D.* On Kabatianskii — Krouk — Smeets signatures. LNCS, 2007, vol. 4547, pp. 237–252.
12. *Stern J.* Can one design a signature scheme based on error-correcting codes? LNCS, 1995, vol. 917, pp. 424–426.
13. *Courtois N., Finiasz M., and Sendrier N.* How to achieve a McEliece-based digital signature scheme. LNCS, 2001, vol. 2248, pp. 157–174.
14. *McEliece R. J.* A public-key cryptosystem based on algebraic coding theory. DSN Progress Report, 1978, vol. 42–44, pp. 114–116.
15. *Niederreiter H.* Knapsack-type cryptosystems and algebraic coding theory. Problems Control Inform. Theory, 1986, vol. 15, no. 2, pp. 159–166.
16. *Dallot L.* Towards a concrete security proof of Courtois, Finiasz and Sendrier signature scheme. LNCS, 2008, vol. 4945, pp. 65–77.
17. *Debris-Alazard T., Sendrier N., and Tillich J.-P.* Wave: a new family of trapdoor one-way preimage sampleable functions based on codes. LNCS, 2019, vol. 11921, pp. 21–51.
18. *Fiat A. and Shamir A.* How to prove yourself: practical solutions to identification and signature problems. LNCS, 1987, vol. 263, pp. 186–194.
19. *Stern J.* A new identification scheme based on syndrome decoding. LNCS, 1994, vol. 773, pp. 13–21.
20. *Jain A., Krenn S., Pietrzak K., and Tentes A.* Commitments and efficient zero-knowledge proofs from learning parity with noise. LNCS, 2012, vol. 7658, pp. 663–680.
21. *Cayrel P.-L., Véron P., and El Y. A. S. M.* A zero-knowledge identification scheme based on the q-ary SD problem. LNCS, 2010, vol. 6544, pp. 171–186.
22. *Lyubashevsky V.* Lattice signatures without trapdoors. LNCS, 2012, vol. 7237, pp. 738–755.

23. *Aragon N., Blazy O., Gaborit P., et al.* Durandal: a rank metric based signature scheme. LNCS, 2019, vol. 11478, pp. 728–758.
24. *Overbeck R. and Sendrier N.* Code-based cryptography. Post-Quantum Cryptography, 2009, pp. 95–145.
25. *Roy P. S., Morozov K., Fukushima K., and Kiyomoto S.* Evaluation of Code-Based Signature Schemes. Cryptology ePrint Archive, 2019, Report 2019/544, `https://eprint.iacr.org/`
26. *El Y. A. S. M., Cayrel P.-L., El B. R., and Hoffmann G.* Code-based identification and signature schemes in software. LNCS, 2013, vol. 8128, pp. 122–136.
27. *Pointcheval D. and Stern J.* Security proofs for signature schemes. LNCS, 1996, vol. 1070, pp. 387–398.
28. *Berlekamp E., McEliece R., and van Tilborg H.* On the inherent intractability of certain coding problems (Corresp.). IEEE Trans. Inform. Theory, 1978, vol. 24, no. 3, pp. 384–386.
29. *Both L. and May A.* Decoding linear codes with high error rate and its impact for LPN security. LNCS, 2018, vol. 10786, pp. 25–46.
30. *Lebedev P. A.* Comparison of old and new cryptographic hash function national standards of Russian Federation on CPUs and NVIDIA GPUs. Mat. Vopr. Kriptogr., 2013, vol. 4, no. 2, pp. 73–80.

МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

УДК 510.52 DOI 10.17223/20710410/57/6

О ГЕНЕРИЧЕСКОЙ СЛОЖНОСТИ ОГРАНИЧЕННОЙ ПРОБЛЕМЫ КЛАСТЕРИЗАЦИИ ГРАФОВ[1]

А. Н. Рыбалов

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

E-mail: alexander.rybalov@gmail.com

Изучается генерическая сложность проблемы кластеризации графов с ограничениями на число кластеров. В этой проблеме структура взаимосвязей объектов задаётся с помощью графа, вершины которого соответствуют объектам, а рёбра соединяют похожие объекты. Требуется разбить множество объектов на ограниченное число попарно непересекающихся групп (кластеров) так, чтобы минимизировать число связей между кластерами и число недостающих связей внутри кластеров. Строится подпроблема этой проблемы, для которой, при условии $\mathrm{P} \neq \mathrm{NP}$ и $\mathrm{P} = \mathrm{BPP}$, не существует полиномиального генерического алгоритма.

Ключевые слова: *генерическая сложность, кластеризация графа.*

THE GENERIC COMPLEXITY OF THE BOUNDED PROBLEM OF GRAPHS CLUSTERING

A. N. Rybalov

Sobolev Institute of Mathematics, Omsk, Russia

Generic-case approach to algorithmic problems studies behavior of an algorithm on typical (almost all) inputs and ignores the rest of inputs. In this paper, we study the generic complexity of the bounded problem of graphs clustering. In this problem the structure of objects relations is presented as a graph: vertices correspond to objects, and edges connect similar objects. It is required to divide the set of objects into bounded disjoint groups (clusters) to minimize the number of connections between clusters and the number of missing links within clusters. We have constructed a subproblem of this problem, for which there is no polynomial generic algorithm provided $\mathrm{P} \neq \mathrm{NP}$ and $\mathrm{P} = \mathrm{BPP}$. To prove the theorem, we use the method of generic amplification, which allows to construct generically hard problems from the problems hard in the classical sense. The main component of this method is the cloning technique, which merges the inputs of a problem together into sufficiently large sets of equivalent inputs. Equivalence is understood in the sense that the problem for them is solved in a similar way.

Keywords: *generic complexity, graph clustering.*

[1]Работа поддержана грантом РНФ № 22-11-20019.

Введение

Одной из важных проблем машинного обучения является проблема кластеризации графов. В этой задаче структура взаимосвязей объектов задаётся с помощью графа, вершины которого соответствуют объектам, а рёбра соединяют похожие объекты. Требуется разбить множество объектов на попарно непересекающиеся группы (кластеры) так, чтобы минимизировать число связей между кластерами и число недостающих связей внутри кластеров. В работах [1 – 7] доказана NP-трудность проблемы кластеризации графа для различных её постановок. Таким образом, при условии $\mathrm{P} \neq \mathrm{NP}$ не существует полиномиального алгоритма для решения этой задачи, а при условии совпадения классов P и BPP (класс проблем, решаемых за полиномиальное время вероятностными алгоритмами) для неё не существует и полиномиальных вероятностных алгоритмов. Имеются серьёзные доводы в пользу равенства $\mathrm{P} = \mathrm{BPP}$. В частности, доказано [8], что это равенство следует из весьма правдоподобных гипотез о вычислительной сложности некоторых трудных проблем.

Генерический подход [9] — это один из подходов к изучению алгоритмических проблем для «почти всех» входов. В рамках этого подхода алгоритмическая проблема рассматривается не на всём множестве входов, а на некотором подмножестве «почти всех» входов. Такие входы образуют так называемое генерическое множество. Понятие «почти все» формализуется введением естественной меры на множестве входных данных. С точки зрения практики алгоритмы, решающие быстро проблему на генерическом множестве, так же хороши, как и быстрые алгоритмы для всех входов. Отметим, что похожий подход для изучения проблем оптимизации был предложен ранее в [10].

Большой интерес как с теоретической точки зрения, так и с точки зрения практических приложений представляют алгоритмические проблемы, которые остаются неразрешимыми или трудноразрешимыми и в генерическом случае. Например, в современной криптографии интересны такие проблемы, которые, являясь (гипотетически) трудными в классическом смысле, остаются трудными и в генерическом смысле, т.е. для почти всех входов. Это объясняется тем, что при случайной генерации ключей в криптографическом алгоритме происходит генерация входа некоторой трудной алгоритмической проблемы, лежащей в основе алгоритма. Если проблема будет генерически легкоразрешимой, то для почти всех таких входов ее можно будет быстро решить и ключи почти всегда будут нестойкими. Поэтому проблема должна быть трудной для почти всех входов. Например, таким поведением обладают классические алгоритмические проблемы криптографии: проблема распознавания квадратичных вычетов, проблема дискретного логарифма, проблема извлечения корня в группах вычетов (проблема обращения функции RSA).

Метод генерической амплификации [11] позволяет строить проблемы, неразрешимые или трудноразрешимые для почти всех входов. С его помощью была доказана генерическая неразрешимость и трудноразрешимость многих алгоритмических проблем: проблема остановки для машин Тьюринга [12], проблема равенства в некоторых конечно определенных полугруппах [11], проблема разрешимости элементраных теорий, неразрешимых в классическом случае [11], арифметика Пресбургера [13], десятая проблема Гильберта [14], проблема выполнимости булевых формул [15], проблема кластеризации графов [16], проблема распознавания гамильтоновых графов [17].

Данная работа посвящена изучению генерической сложности ограниченной проблемы кластеризации графов. Эта задача является проблемой кластеризации графов, в которой число кластеров ограничено заранее заданной константой $k \geqslant 2$. В [3] была

доказана NP-трудность такой задачи для любого $k \geqslant 2$. В данной работе строится подпроблема этой проблемы, для которой, при условии $\mathrm{P} \neq \mathrm{NP}$ и $\mathrm{P} = \mathrm{BPP}$, не существует полиномиального генерического алгоритма. Отметим, что в [16] была изучена генерическая сложность проблемы кластеризации графов без ограничения на число кластеров.

1. Предварительные сведения

Пусть I — некоторое множество входов, а I_n — подмножество входов размера n. Для подмножества $S \subseteq I$ определим последовательность

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

где $S_n = S \cap I_n$ — множество входов из S размера n. Заметим, что $\rho_n(S)$ — это вероятность попасть в S при случайной и равновероятной генерации входов из I_n. *Асимптотической плотностью* S назовём предел

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

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

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

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

Генерический алгоритм $\mathcal{A}$ вычисляет функцию $f : I \to J$, если для всех $x \in I$ выполнено

$$(\mathcal{A}(x) = y \in J) \Rightarrow (f(x) = y).$$

Ситуация $\mathcal{A}(x) = ?$ означает, что $\mathcal{A}$ не может вычислить функцию f на аргументе x. Но условие 2 гарантирует, что $\mathcal{A}$ корректно вычисляет f на почти всех входах (входах из генерического множества).

Напомним также некоторые понятия классической теории сложности вычислений. *Время работы* $t_M(x)$ машины Тьюринга M на входе $x \in I$ – это число шагов машины от начала работы до остановки. Если M на x не останавливается, полагаем $t_M(x) = \infty$. Машина Тьюринга M *полиномиальна*, если существует полином $p(n)$, такой, что для любого $x \in I$ имеет место $t_M(x) < p(|x|)$. Класс P состоит из подмножеств I, распознаваемых полиномиальными машинами Тьюринга.

Вероятностная машина Тьюринга — это машина Тьюринга, в программе которой допускаются пары правил вида

$$(q_i, a) \to (q_j, b, S_1),$$
$$(q_i, a) \to (q_k, c, S_2).$$

В процессе работы такой машины с вероятностью $1/2$ выбирается первое правило и с вероятностью $1/2$ — второе. Обозначим через $P(M(x) = y)$ вероятность того, что машина M на входе x выдаёт ответ y. Время работы $t_M(x, \tau)$ вероятностной машины Тьюринга на входе x зависит от вычислительного пути (последовательности выполненных команд) τ. Проблема $S \subseteq I$ принадлежит *классу* BPP, если существуют вероятностная машина Тьюринга M и полином $p(n)$, такие, что

1) Для любого x и для любого вычислительного пути τ машины M на x имеет место $t_M(x,\tau) < p(|x|)$.
2) Если $x \in S$, то $P(M(x) = 1) > 2/3$.
3) Если $x \notin S$, то $P(M(x) = 0) > 2/3$.

Вероятностные машины Тьюринга формализуют понятие алгоритма, использующего генератор случайных чисел. Класс BPP — это класс проблем, эффективно решаемых такими вероятностными алгоритмами. Большинство специалистов по теоретической информатике сейчас считает, что имеет место равенство P = BPP. Это равенство означает, что любой полиномиальный вероятностный алгоритм можно эффективно дерандомизировать, т. е. построить полиномиальный детерминированный алгоритм, решающий ту же задачу. Хотя равенство пока не доказано, имеются серьёзные результаты в пользу него [8].

2. Ограниченная проблема кластеризации графов

Будем рассматривать неориентированные графы без петель и кратных рёбер. Граф называется *кластерным*, если каждая его компонента связности является полным графом. Обозначим через $\mathcal{M}_{\leqslant k}(V)$ множество всех кластерных графов на множестве вершин V, имеющих не более k компонент связности. Если $G_1 = (V, E_1)$ и $G_2 = (V, E_2)$ — графы на одном и том же множестве вершин V, то *расстояние* $\rho(G_1, G_2)$ между ними есть число несовпадающих рёбер в графах G_1 и G_2, то есть

$$\rho(G_1, G_2) = |E_1 \triangle E_2| = |E_1 \setminus E_2| + |E_2 \setminus E_1|.$$

Ограниченная проблема кластеризации графов $\mathcal{GC}_{\leqslant k}$ состоит в следующем. Задан граф $G = (V, E)$ и целое число $k \geqslant 2$. Найти такой граф $M^* \in \mathcal{M}_{\leqslant k}(V)$, что

$$\rho(G, M^*) = \min_{M \in \mathcal{M}_{\leqslant k}(V)} \rho(G, M).$$

В [3] доказана NP-трудность этой проблемы для любого $k \geqslant 2$. В частности, это означает, что существует полиномиально эквивалентная ей проблема распознавания из класса NP, которая является NP-полной.

Рассмотрим бесконечную последовательность графов $\gamma = \{G_1, G_2, \ldots, G_n, \ldots\}$, такую, что G_n имеет n вершин для любого n. Напомним, что два графа G_1 и G_2 называются изоморфными (обозначается $G_1 \cong G_2$), если существует биекция π между множествами вершин G_1 и G_2, такая, что для любых вершин v, u графа G_1 v и u соединены ребром в G_1 тогда и только тогда, когда $\pi(v)$ и $\pi(u)$ соединены ребром в G_2. Биекция π, осуществляющая изоморфизм, является перестановкой множества вершин графов G_1, G_2, если вершины обоих графов занумерованы числами $\{1, 2, \ldots, n\}$. Будем обозначать также $G_2 = \pi(G_1)$.

Для каждой последовательности графов γ определим проблему $\mathcal{GC}_{\leqslant k}(\gamma)$ как ограничение проблемы $\mathcal{GC}_{\leqslant k}$ на множество входов $\{G : G \cong G_n, G_n \in \gamma\}$. Заметим, что множество всех входов размера n в проблеме $\mathcal{GC}_{\leqslant k}(\gamma)$ состоит из всевозможных графов G с произвольным графом G, изоморфным фиксированному графу G_n из последовательности γ. Очевидно, что проблема $\mathcal{GC}_{\leqslant k}(\gamma)$ является подпроблемой ограниченной проблемы кластеризации графов. Следующее утверждение говорит о том, что эта проблема для некоторых последовательностей γ может быть так же трудна, как и ограниченная проблема кластеризации графов.

Лемма 1. Если не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{GC}_{\leqslant k}$, то найдется последовательность графов γ, такая, что не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{GC}_{\leqslant k}(\gamma)$.

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

3. Основной результат

Для изучения генерической сложности проблемы кластеризации графов будем использовать представление графов с помощью матриц смежности. Под размером графа будем понимать число вершин.

Теорема 1. Пусть γ — произвольная последовательность графов. Если существует генерический полиномиальный алгоритм, решающий проблему $\mathcal{GC}_{\leqslant k}(\gamma)$, то существует вероятностный полиномиальный алгоритм, решающий эту проблему на всём множестве входов.

Доказательство. Допустим, что существует генерический полиномиальный алгоритм $\mathcal{A}$, решающий проблему ограниченной кластеризации графов $\mathcal{GC}_{\leqslant k}(\gamma)$. Построим вероятностный полиномиальный алгоритм $\mathcal{B}$, решающий эту проблему на всём множестве входов. На графе G с n вершинами алгоритм $\mathcal{B}$ работает следующим образом:

1) Запускает алгоритм $\mathcal{A}$ на G.
2) Если $\mathcal{A}(G) \neq ?$, то $\mathcal{B}$ выдаёт ответ $\mathcal{A}(G)$ и останавливается, иначе идёт на шаг 3.
3) Генерирует случайно и равномерно перестановку π на вершинах $\{1, \ldots, n\}$ и вычисляет граф $G' = \pi(G)$.
4) Запускает алгоритм $\mathcal{A}$ на (G').
5) Если $\mathcal{A}(G') = ?$, выдаёт полный граф на всех вершинах (возможно, неправильный ответ).
6) Пусть $\mathcal{A}(G') = M^*$ — решение задачи ограниченной кластеризации для графа G' и $M^* \in \mathcal{M}_{\leqslant k}(\{1, \ldots, n\})$. Тогда если $M^* = C_1 \cup C_2 \cup \ldots C_m$, где C_i — непересекающиеся кластеры на множестве вершин $\{1, \ldots, n\}$, то легко видеть, что

$$\pi^{-1}(M^*) = \pi^{-1}(C_1) \cup \pi^{-1}(C_2) \cup \ldots \pi^{-1}(C_m)$$

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

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

Непосредственным следствием теоремы 1 является следующее утверждение:

Теорема 2. Если P $\neq$ NP и P = BPP, то существует последовательность графов γ, такая, что для решения ограниченной проблемы кластеризации $\mathcal{GC}_{\leqslant k}(\gamma)$ не существует генерического полиномиального алгоритма.

Доказательство. Покажем сначала, что, при условиях $\mathrm{P} \neq \mathrm{NP}$ и $\mathrm{P} = \mathrm{BPP}$, не существует полиномиального вероятностного алгоритма для решения проблемы $\mathcal{GC}_{\leqslant k}$. Действительно, пусть такой алгоритм существует. Так как проблема $\mathcal{GC}_{\leqslant k}$ является NP-трудной, то существует полиномиально эквивалентная ей NP-полная проблема распознавания A. Из полиномиального вероятностного алгоритма для $\mathcal{GC}_{\leqslant k}$ легко получается полиномиальный вероятностный алгоритм для решения проблемы A. А так как $\mathrm{P} = \mathrm{BPP}$, то существует и детерминированный полиномиальный алгоритм для A, откуда $\mathrm{P} = \mathrm{NP}$. Противоречие.

Теперь нужное утверждение следует из леммы 1 и теоремы 1. ■

Автор выражает благодарность рецензенту за полезные замечания и предложения по улучшению текста статьи.

ЛИТЕРАТУРА

1. *Křivanek M. and Morávek J.* NP-hard problems in hierarchical-tree clustering // Acta Informatica. 1986. V. 23. P. 311–323.
2. *Bansal N., Blum A., and Chawla S.* Correlation clustering // Machine Learning. 2004. V. 56. P. 89–113.
3. *Shamir R., Sharan R., and Tsur D.* Cluster graph modification problems // Discrete Appl. Math. 2004. V. 144. No. 1–2. P. 173–182.
4. *Агеев А. А., Ильев В. П., Кононов А. В., Талевнин А. С.* Вычислительная сложность задачи аппроксимации графов // Дискретный анализ и исследование операций. Сер. 1. 2006. Т. 13. № 1. С. 3–11.
5. *Ильев В. П., Ильева С. Д.* О задачах кластеризации графов // Вестник Омского университета. 2016. № 2. С. 16–18.
6. *Ильев А. В., Ильев В. П.* Об одной задаче кластеризации графа с частичным обучением // Прикладная дискретная математика. 2018. № 42. С. 66–75.
7. *Талевнин А. С.* О сложности задачи аппроксимации графов // Вестник Омского университета. 2004. № 4. С. 22–24.
8. *Impagliazzo R. and Wigderson A.* $\mathrm{P} = \mathrm{BPP}$ unless E has subexponential circuits: Derandomizing the XOR Lemma // Proc. 29th STOC. El Paso: ACM, 1997. P. 220–229.
9. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks // J. Algebra. 2003. V. 264. No. 2. P. 665–694.
10. *Гимади Э. Х., Глебов Н. И., Перепелица В. А.* Алгоритмы с оценками для задач дискретной оптимизации // Проблемы кибернетики. 1975. Т. 31. С. 35–42.
11. *Myasnikov A. G. and Rybalov A.N.* Generic complexity of undecidable problems // J. Symbolic Logic. 2008. V. 73. No. 2. P. 656–673.
12. *Rybalov A. N.* On the strongly generic undecidability of the Halting Problem // Theor. Comput. Sci. 2007. V. 377. P. 268–270.
13. *Rybalov A. N.* Generic complexity of Presburger arithmetic // Theory Comput. Systems. 2010. V. 46. No. 1. P. 2–8.
14. *Rybalov A. N.* Generic complexity of the Diophantine problem // Groups Complexity Cryptology. 2013. V. 5. No. 1. P. 25–30.
15. *Rybalov A. N.* Generic hardness of the Boolean satisfiability problem // Groups Complexity Cryptology. 2017. V. 9. No. 2. P. 151–154.
16. *Рыбалов А. Н.* О генерической сложности проблемы кластеризации графов // Прикладная дискретная математика. 2019. № 46. С. 72–77.

17. *Рыбалов А. Н.* О генерической сложности проблемы распознавания гамильтоновых путей // Прикладная дискретная математика. 2021. № 53. С. 120–126.

REFERENCES

1. *Křivanek M. and Morávek J.* NP-hard problems in hierarchical-tree clustering. Acta Informatica, 1986, vol. 23, pp. 311–323.
2. *Bansal N., Blum A., and Chawla S.* Correlation clustering. Machine Learning, 2004, vol. 56, pp. 89–113.
3. *Shamir R., Sharan R., and Tsur D.* Cluster graph modification problems. Discrete Appl. Math., 2004, vol. 144, no. 1–2, pp. 173–182.
4. *Ageev A. A., Il'ev V. P., Kononov A. V., and Talevnin A. S.* Computational complexity of the graph approximation problem. J. Appl. Ind. Math., 2007, vol. 1, no. 1, pp. 1–8.
5. *Il'ev V. P. and Il'eva S. D.* O zadachah klasterizacii grafov [On problems of graph clustering]. Vestnik Omskogo Universiteta, 2016, no. 2, pp. 16–18. (in Russian)
6. *Il'ev A. V. and Il'ev V. P.* Ob odnoy zadache klasterizatsii grafa s chastichnym obucheniem [On a semi-superwized graph clustering problem]. Prikladnaya Diskretnaya Matematika, 2018, no. 42, pp. 66–75. (in Russian)
7. *Talevnin A. S.* O slozhnosti zadachi approksimacii grafov [On the complexity of the graph approximation problem]. Vestnik Omskogo Universiteta, 2004, no. 4, pp. 22–24. (in Russian)
8. *Impagliazzo R. and Wigderson A.* P = BPP unless E has subexponential circuits: Derandomizing the XOR Lemma. Proc. 29th STOC, El Paso, ACM, 1997, pp. 220–229.
9. *Kapovich I., Miasnikov A., Schupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks. J. Algebra, 2003, vol. 264, no. 2, pp. 665–694.
10. *Gimadi E. H., Glebov N. I., and Perepelitsa V. A.* Algoritmy s ocenkami dlya zadach diskretnoi optimizacii [Algorithms with bounds for problems of discrete optimization]. Problemy Kibernetiki, 1975, V. 31, pp. 35–42. (in Russian)
11. *Myasnikov A. G. and Rybalov A. N.* Generic complexity of undecidable problems. J. Symbolic Logic, 2008, vol. 73, no. 2, pp 656–673.
12. *Rybalov A. N.* On the strongly generic undecidability of the Halting Problem. Theor. Comput. Sci., 2007, vol. 377, pp. 268–270.
13. *Rybalov A. N.* Generic complexity of Presburger arithmetic. Theory Comput. Systems, 2010, vol. 46, no. 1, pp. 2–8.
14. *Rybalov A. N.* Generic complexity of the Diophantine problem. Groups Complexity Cryptology, 2013, vol. 5, no. 1, pp. 25–30.
15. *Rybalov A. N.* Generic hardness of the Boolean satisfiability problem. // Groups Complexity Cryptology, 2017, vol. 9, no. 2, pp. 151–154.
16. *Rybalov A. N.* O genericheskoy slozhnosti problemy klasterizatsii grafov [On generic complexity of the graph clustering problem]. Prikladnaya Diskretnaya Matematika, 2019, no. 46, pp. 72–77. (in Russian)
17. *Rybalov A. N.* O genericheskoy slozhnosti problemy raspoznavaniya gamil'tonovykh putey [The general complexity of the problem to recognize Hamiltonian paths]. Prikladnaya Diskretnaya Matematika, 2021, no. 53, pp. 120–126. (in Russian)

ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ

УДК 519.81 DOI 10.17223/20710410/57/7

МЕДИАНА ДЛЯ НЕЧЁТНОГО ЧИСЛА ОТНОШЕНИЙ ЛИНЕЙНОГО ПОРЯДКА И ЕЁ ИСПОЛЬЗОВАНИЕ В ЗАДАЧАХ ГРУППОВОГО ВЫБОРА

В. Н. Нефедов

Московский авиационный институт, г. Москва, Россия

E-mail: nefedovvn@yandex.ru

Ищется медиана для нечётного числа отношений линейного порядка, заданных на конечном множестве A, также являющаяся линейным порядком на A. К подобной задаче приходим при исследовании некоторых задач группового выбора. В рассматриваемом случае бинарное отношение $\tilde{\rho}$, имеющее минимальное суммарное расстояние (по Хеммингу) до заданного набора бинарных отношений, является медианой для этих отношений, и притом единственной. Однако эта медиана не всегда является линейным порядком (или даже квазипорядком) и в этих случаях не может быть взята в качестве решения поставленной задачи. Тем не менее бинарное отношение $\tilde{\rho}$ обязательно принадлежит множеству $LA[n]$ (линейных асимметричных бинарных отношений на A), которому, в частности, принадлежат и все линейные порядки на A. Исследуются некоторые свойства бинарных отношений из $LA[n]$. Вводятся понятия «почти оптимального» и Δ-оптимального отношений, являющихся линейными порядками и одновременно точными решениями поставленной задачи. Приводятся алгоритмы их нахождения, основанные на полученных утверждениях относительно бинарных отношений из $LA[n]$ и имеющие полиномиальную вычислительную сложность. Рассматривается отношение эквивалентности на множестве $LA[n]$, позволяющее разбивать это множество на классы эквивалентности, количество которых K_n намного меньше числа элементов в $LA[n]$. Например, $|LA[5]| = 1024$, $K_5 = 12$. Таким образом, каждое бинарное отношение из $LA[n]$ эквивалентно в точности одному из K_n представителей классов эквивалентности, а следовательно, обладает его основными свойствами. Но тогда исследование широкого класса задач может быть сведено к рассмотрению сравнительно небольшого их набора. Процесс нахождения указанного набора представителей классов эквивалентности иллюстрируется для $n = 2, 3, 4, 5$. Приводится также метод решения поставленной задачи, использующий представление бинарных отношений в виде графов (метод выделения минимальных множеств представителей контуров в медиане $\tilde{\rho}$), имеющий экспоненциальную вычислительную сложность.

Ключевые слова: *медиана отношений, отношение линейного порядка, линейные отношения, асимметричные отношения, расстояние Хемминга, классы эквивалентности, задача группового выбора.*

MEDIAN FOR AN ODD NUMBER OF LINEAR ORDER RELATIONS AND ITS USE IN GROUP CHOICE PROBLEMS

V. N. Nefedov

Moscow Aviation Institute, Moscow, Russia

We consider the problem of constructing a median for an odd set of linear order relations defined on a finite set $A = \{a_1, a_2, \ldots, a_n\}$, which is also sought in the class of linear order relations. We arrive at a similar problem when considering some group choice problems. The distance between binary relations is the Hamming distance between their adjacency matrices. In the case under consideration, the binary relation $\tilde{\rho}$, which has the minimum total distance to the given set of binary relations, is the median for these relations and, moreover, is unique. However, this median is not always transitive (and in this case is neither linear order, nor even a quasi-order), and therefore cannot be taken as a solution to a given problem. However, the median $\tilde{\rho}$ necessarily belongs to the set $LA[n]$ (of linear asymmetric binary relations on A), to which, in particular, all linear orders on A also belong. Some properties of binary relations from $LA[n]$ are investigated. The concepts of "almost optimal" and Δ-optimal relations are introduced, which are linear orders and, at the same time, exact solutions of the stated problem. Algorithms for finding them are given, based on the obtained statements about binary relations from $LA[n]$ and having polynomial computational complexity. An equivalence relation on the set $LA[n]$ is considered, which allows one to divide this set into equivalence classes, the number of which K_n is much less than the number of elements in $LA[n]$. For example, $|LA[5]| = 1024$, $K_5 = 12$. Thus, each binary relation from $LA[n]$ is equivalent to exactly one of the K_n representatives of the equivalence classes and, therefore, has its main properties. But then the study of a wide class of problems can be reduced to considering a relatively small set of them. The process of finding the specified set of equivalence class representatives is illustrated for $n = 2, 3, 4, 5$. A method for solving the problem posed is also given, using the representation of binary relations in the form of graphs (the method of selecting the minimum sets of contour representatives in the median $\tilde{\rho}$), which has exponential computational complexity.

Keywords: *median of relations, linear order relation, linear relations, asymmetric relations, Hamming distance, equivalence classes, group choice problem.*

Введение

Рассматривается задача нахождения медианы для нечётного числа отношений линейного порядка, которая ищется также в классе отношений линейного порядка. К подобной постановке приходим, например, при рассмотрении задачи группового выбора [1 – 15], когда профиль экспертных предпочтений определяется бинарными отношениями $\rho_1, \ldots, \rho_m$, заданными на конечном множестве альтернатив $A = \{a_1, a_2, \ldots, a_n\}$. В этом случае для обеспечения согласованности агрегированного отношения с профилем экспертных предпочтений часто ищется (см., например, [1 – 3]) бинарное отношение ρ, которое имеет минимальное суммарное расстояние $D(\rho)$ до бинарных отношений, задающих экспертные предпочтения, и которое принято называть *медианой для этих отношений* (*точной медианой*). Расстояние между бинарными отношениями определяется в настоящей работе (аналогично [1]) по формуле Хемминга: сумма модулей разности элементов матриц смежности этих отношений. При этом в случае, когда экспертные предпочтения задаются линейными порядками при нечётном числе

экспертов, медиана определяется однозначно по правилу «большинства» [1]. Тем не менее часто медиана не обладает свойством транзитивности, т. е. не является ни линейным порядком, ни даже квазипорядком (т. е. не является строгой или нестрогой ранжировкой). Но тогда в случае, когда агрегированное отношение ищется в виде ранжировки, эта медиана не может быть взята в качестве окончательного решения. Одним из возможных выходов из этой ситуации является нахождение *линейно упорядоченной медианы*, которая соответствует случаю, когда минимальное суммарное расстояние до бинарных отношений, задающих экспертные предпочтения, ищется не на множестве всех возможных бинарных отношений, а на множестве отношений линейного порядка. Такой подход рассматривается во многих работах (см., например, [1] вместе с ссылками на другие работы). И если при этом разница между двумя получаемыми таким образом минимальными суммарными расстояниями невелика, то в качестве искомого агрегированного отношения можно взять линейно упорядоченную медиану (или наиболее предпочтительную, если их несколько). Поскольку для нахождения линейно упорядоченной медианы используется точная медиана и она в нашем случае единственная, будем её обозначать через $\tilde{\rho}$. Следует оговориться, что в общем случае используемый подход применим лишь при небольшом количестве альтернатив (по-видимому, не больше 30), т. е. для достаточного узкого класса задач. В случаях, когда агрегированное отношение ищется при более слабых предположениях (например, в виде нестрогой ранжировки), следует рассматривать иные, более простые и эффективные (для этих случаев) подходы, в частности представленные в [15].

Множество линейно упорядоченных бинарных отношений, т. е. рефлексивных, антисимметричных, транзитивных и с попарно сравнимыми элементами (линейных) на конечном множестве, состоящем из n элементов, обозначим $LO[n]$. Как уже отмечалось, в случае, когда $\rho_1, \ldots, \rho_m \in LO[n]$ и m нечётно, бинарное отношение $\tilde{\rho}$ может не принадлежать $LO[n]$, однако обязательно принадлежит $LA[n]$ — множеству всех рефлексивных антисимметричных линейных бинарных отношений на n-элементном множестве. Множество $LA[n]$ является предметом детального исследования в п. 1–4.

Таким образом, основной задачей, рассматриваемой в работе, является нахождение отношений $\rho \in LO[n]$, доставляющих минимум величине $D(\rho)$ — сумме расстояний от ρ до заданных $\rho_1, \ldots, \rho_m \in LO[n]$, в частности задаваемых экспертами в случае задачи группового выбора, где m нечётно. Такая задача может быть решена методом простого перебора. Однако при больших n такой подход трудно реализуем. Чтобы увеличить допустимое значение n, можно использовать метод ветвей и границ. Одна из возможных схем применения этого метода указана в [13, с. 3]. В проведённых численных экспериментах при использовании метода ветвей и границ удалось даже на маломощном компьютере найти численное решение задачи при $n = 26$.

Возможен и другой подход, описанный в п. 2. В этом подходе исследуем множества $\{\rho \in LO[n] : D(\rho) = D(\tilde{\rho}) + \Delta\}$ для значений $\Delta \in \{0, 2, 4, 6, \ldots\}$ (значение $D(\rho)$ всегда чётно) и ищем минимальное Δ, при котором множество не пусто. Решения, соответствующие этому Δ, в случае $\Delta > 0$ назовём Δ-*оптимальными*, а при $\Delta = 2$ — *почти оптимальными*. Проверка выполнения условия $\{\rho \in LO[n] : D(\rho) = D(\tilde{\rho})\} \neq \varnothing$ является алгоритмически простой, и в случае его выполнения $\tilde{\rho} \in LO[n]$, т. е. является единственным решением поставленной задачи.

Следующий по важности случай соответствует выполнению условий $\tilde{\rho} \notin LO[n]$, $\{\rho \in LO[n] : D(\rho) = D(\tilde{\rho}) + 2\} \neq \varnothing$. В этом случае существует бинарное отношение ρ, являющееся линейным порядком и удовлетворяющее условию $D(\rho) = D(\tilde{\rho}) + 2$, т. е. имеющее «почти» минимальное суммарное расстояние до бинарных отношений

$\rho_1, \ldots, \rho_m$, а именно с минимально возможным отклонением от $D(\tilde{\rho})$, равным 2. Этот случай подробно рассматривается в п. 2. Приводятся легко проверяемые необходимые и достаточные условия его выполнения, а также алгоритм выделения всех возможных решений (с вычислительной сложностью $O(n^2)$). Кроме того, описан простой алгоритм (с полиномиальной вычислительной сложностью) нахождения всех возможных решений для каждого последующего множества $\{\rho \in LO[n] : D(\rho) = D(\tilde{\rho}) + \Delta\}$, где $\Delta = 4, 6, \ldots$, если предыдущие оказались пустыми.

Отметим: множество Δ-оптимальных решений даёт все множество $\operatorname{Arg}\min\limits_{\rho \in LO[n]} D(\rho)$, т. е. является точным решением поставленной задачи. Термин «Δ-оптимальность» говорит об отклонении на Δ величины $\min\limits_{\rho \in LO[n]} D(\rho)$ от $D(\tilde{\rho})$.

Одним из двух необходимых условий для «почти оптимальности» отношения $\tilde{\rho} \notin \notin LO[n]$ является условие $\tilde{\rho} \in LA^{(2)}[n]$, где $LA^{(2)}[n]$ — множество всех бинарных отношений из $LA[n] \setminus LO[n]$, имеющих расстояние по Хеммингу до ближайшего линейного порядка, равное 2. В п. 2 и 3 приводятся утверждения о структуре множества $LA^{(2)}[n]$, в частности получена формула для $|LA^{(2)}[n]|$, дающая возможность определить долю этого множества в $LA[n] \setminus LO[n]$. Например, $|LA^{(2)}[5]| = 480$, $|LA[5] \setminus LO[5]| = 904$.

В п. 3 исследуется также структура множества $LA[n]$. Рассматривается отношение эквивалентности на $LA[n]$, позволяющее разбивать это множество на классы эквивалентности, количество которых K_n намного меньше числа элементов в $LA[n]$. Например, $|LA[5]| = 1024$, $K_5 = 12$. В частности, $\tilde{\rho}$ эквивалентно в точности одному из K_n представителей классов эквивалентности, а следовательно, обладает его основными свойствами. Таким образом, исследование широких классов бинарных отношений может быть сведено к рассмотрению сравнительно небольшого набора их представителей. Процесс нахождения указанного семейства представителей классов эквивалентности иллюстрируется для $n = 2, 3, 4, 5$.

В п. 4 приводятся некоторые методы сведения исходной задачи к одной или нескольким задачам меньшей размерности.

В п. 5 представлен метод решения поставленной задачи, использующий представление бинарных отношений в виде графов (метод выделения минимальных множеств представителей контуров в медиане $\tilde{\rho}$) и имеющий экспоненциальную вычислительную сложность.

Полученные результаты могут быть использованы при построении для единственного эксперта линейного порядка на множестве альтернатив, максимальным образом учитывающего результаты их попарного сравнения. В [1, 12] приведён следующий пример: изучались гастрономические предпочтения собаки на совокупности из шести видов пищи. Каждое утро перед собакой выставляли два вида пищи; считалось, что она отдаёт предпочтение тому виду, с которого начинает завтрак. Результаты сравнений можно свести в матрицу, являющуюся матрицей смежности некоторого бинарного отношения $\rho_1 \in LA[6]$. Требуется найти $\operatorname{Arg}\min\limits_{\rho \in LO[n]} D(\rho)$, где $D(\rho)$ в данном случае совпадает с $d(\rho, \rho_1)$ — расстоянием Хемминга между ρ, ρ_1. Эту задачу можно обобщить на случай, когда исследуются предпочтения нечётного числа испытуемых, для каждого из которых на основании попарных сравнений строится бинарное отношение $\rho_i \in LA[n]$, $i = 1, \ldots, m$. В этом случае, как и ранее, при решении задачи группового выбора приходим к поиску $\operatorname{Arg}\min\limits_{\rho \in LO[n]} D(\rho)$, где $D(\rho)$ имеет тот же вид, что и в случае $\rho_i \in LO[n]$. Оказывается, что в случае $\rho_i \in LA[n]$ также выполняется

$\tilde{\rho} \in LA[n]$ и, более того, все изложенные в работе результаты переносятся и на этот случай, что учитывается в условиях приводимых утверждений.

Замечание 1. Возможны и другие способы задания расстояния между бинарными отношениями, например по формуле Кемени — Снелла [14, 15]. Эта формула используется для нахождения расстояния между ранжировками. При этом в случае линейных порядков (строгих ранжировок) значения расстояний по Хеммингу и по Кемени — Снеллу совпадают (совпадение этих значений будет и в случае, когда отношения принадлежат $LA[n]$).

Замечание 2. Условие нечётности m может показаться весьма ограничительным. Однако в случае чётного m, если, например, выбрать из $\rho_1, \ldots, \rho_m$ бинарное отношение ρ_i с минимальным значением $D_i(\rho) = \sum\limits_{t=1}^{m} d(\rho_i, \rho_t)$ (т. е. с минимальным суммарным расстоянием до остальных отношений) и продублировать его, то будем иметь нечётное число отношений.

1. Основные понятия, определения и вспомогательные утверждения

Пусть $A = \{a_1, a_2 \ldots, a_n\}$. Будем рассматривать бинарные отношения ρ на множестве A, т. е. множества упорядоченных пар $\rho \subseteq A^2$, или кратко $\rho \in 2^{A^2}$. Матрицу смежности бинарного отношения $\rho \in 2^{A^2}$ будем обозначать $R(\rho) = [r_{ij}]_{n\times n}$, где $r_{ij} = 1 \Leftrightarrow (a_i, a_j) \in \rho$. Расстоянием (по Хеммингу) между отношениями $\rho, \rho^{'} \in 2^{A^2}$ назовём величину $d(\rho, \rho^{'}) = \sum\limits_{i=1}^{n}\sum\limits_{j=1}^{n} |r_{ij} - r^{'}_{ij}|$.

Пусть имеется набор бинарных отношений $\rho_1, \rho_2, \ldots, \rho_m \in 2^{A^2}$ с матрицами смежности $R^{(1)}, R^{(2)}, \ldots, R^{(m)}$, где $R^{(t)} = R(\rho_t) = \left[r_{ij}^{(t)}\right]_{n\times n}$, $t = 1, 2, \ldots, m$. Поставим ему в соответствие матрицы $P = \sum\limits_{t=1}^{m} R^{(t)} = [p_{ij}]_{n\times n}$, $L = P - P^{\mathrm{T}} = [l_{ij}]_{n\times n}$ (T — операция транспонирования матриц), т. е. $p_{ij} = \sum\limits_{t=1}^{m} r_{ij}^{(t)}$, $l_{ij} = p_{ij} - p_{ji}$, $i, j = 1, 2, \ldots, n$, а также функцию $D(\rho) = \sum\limits_{t=1}^{m} d(\rho, \rho_t)$, где $\rho \in 2^{A^2}$, $R(\rho) = [r_{ij}]_{n\times n}$.

Рассмотрим для данного набора бинарных отношений $\rho_1, \rho_2, \ldots, \rho_m \in 2^{A^2}$ *мажоритарное отношение* $\tilde{\rho} \in 2^{A^2}$, однозначно определяемое по «правилу большинства» $\tilde{r}_{ij} = 1 \Leftrightarrow l_{ij} \geqslant 0$, где $R(\tilde{\rho}) = [\tilde{r}_{ij}]_{n\times n}$. Будем, кроме того, использовать *нестрогий мажоритарный граф* $\tilde{G} = (A, \tilde{\rho})$ с множеством вершин A и множеством дуг $\tilde{\rho}$. Мажоритарный граф $\tilde{G}$ можно считать нагруженным, если поставить в соответствие каждой дуге $(a_i, a_j) \in \tilde{\rho}$ вес l_{ij}. Определим матрицу весов $C = [c_{ij}]_{n\times n}$ этого графа:

$$c_{ij} = \begin{cases} l_{ij}, & \text{если} (a_i, a_j) \in \tilde{\rho}, \\ \infty & \text{в противном случае.} \end{cases}$$

Нам понадобится

Теорема 1 [1]. Пусть $\rho_1, \rho_2, \ldots, \rho_m$ — произвольные бинарные отношения на $A = \{a_1, a_2, \ldots, a_n\}$. Тогда суммарное расстояние $D(q) = \sum\limits_{t=1}^{m} d(q, \rho_t)$ принимает минимальное значение относительно всех бинарных отношений $q \in 2^{A^2}$ тогда и только тогда, когда для $R(q) = [q_{ij}]_{n\times n}$ выполняется: $q_{ij} = 1 \Leftrightarrow r_{ij}^{t} = 1$ не менее чем для $m/2$ зна-

чений $t \in \{1,\dots,m\}$, где $i,j \in \{1,\dots,n\}$. При этом в случае $\sum\limits_{t=1}^{m} r_{ij}^t = m/2$ (т. е. при чётном m) $D(q)$ остаётся минимальным при любом выборе $q_{ij} \in \{0,1\}$.

В настоящей работе предполагается, что $\rho_t \in LA[n]$ (в частном случае $\rho_t \in LO[n]$), $t = 1,\dots,m$. Заметим в этой связи, что если $\rho \in LA[n], R(\rho) = [r_{ij}]_{n\times n}$, то из определения $LA[n]$ следует, что

$$\forall i,j \in \{1,\dots,n\}\ \ r_{ii} = 1,\ \ i \neq j \Rightarrow r_{ij} + r_{ji} = 1, \tag{1}$$

а следовательно, в силу сделанного предположения

$$\forall i,j \in \{1,\dots,n\}\ \ i \neq j \Rightarrow p_{ij} + p_{ji} = \sum_{t=1}^{m} \left(r_{ij+}^{(t)} r_{ji}^{(t)}\right) = \sum_{t=1}^{m} 1 = m. \tag{2}$$

Используя (1) и (2), нетрудно показать, что справедливо

Утверждение 1 [13]. Пусть $\rho_1,\dots,\rho_m \in LA[n]$ и m нечётно. Тогда для мажоритарного отношения $\tilde\rho$ выполняется $\tilde\rho \in LA[n]$, при этом $\tilde r_{ij} = 1 \Leftrightarrow r_{ij}^t = 1$ для большинства значений $t \in \{1,\dots,m\}$.

Следствие 1. Пусть $\rho_1,\dots,\rho_m \in LA[n]$ и m нечётно. Тогда из теоремы 1 и утверждения 1 получаем, что мажоритарное отношение $\tilde\rho$ принадлежит $LA[n]$, при этом $\operatorname{Arg}\min\limits_{q\in 2^{A^2}} D(q) = \{\tilde\rho\}$.

Приведём некоторые вспомогательные утверждения относительно $LO[n]$, $LA[n]$. Из определений следует, что $LO[n] \subseteq LA[n]$; $\forall \rho \in LA[n]$ $\big(\rho \in LO[n] \Leftrightarrow \rho$ транзитивно на $A\big)$. Кроме того, из определения $LA[n]$ получаем

Утверждение 2. Пусть $\rho_1, \rho_2 \in LA[n]$. Тогда $d(\rho_1,\rho_2)$ чётно.

Следствие 2. Пусть $\rho_1,\dots,\rho_m \in LA[n]$. Тогда $D(\rho)$ чётно для всех $\rho \in LA[n]$.

Из утверждения 1 и следствий 1 и 2 получаем

Утверждение 3. Пусть $\rho_1,\dots,\rho_m \in LA[n]$ и m нечётно. Тогда $\min\limits_{q\in 2^{A^2}} D(q)$ чётно.

Приведём простое условие проверки того, что $\rho \in LO[n]$, для $\rho \in LA[n]$. Для этого понадобятся некоторые обозначения.

Пусть $n \geqslant 2$, $\rho \in 2^{A^2}$, $R(\rho) = [r_{ij}]_{n\times n}$. Обозначим $\kappa_i(\rho) = \sum\limits_{j=1}^{n} r_{ij}$ — количество единиц в i-й строке матрицы $R(\rho)$, $i \in \{1,\dots,n\}$; $I(\rho) = \langle i_1,\dots,i_n\rangle$, где $1 \leqslant i_1 \leqslant \dots \leqslant i_n \leqslant n$, $i_1,\dots,i_n \in \{\kappa_1(\rho),\dots,\kappa_n(\rho)\}$, причём для всех $j \in \{1,\dots,n\}$ компонента i_j входит в $I(\rho)$ столько раз, в скольких строках количество единиц равно i_j.

Пример 1. Пусть бинарные отношения $\rho_1,\rho_2,\rho_3 \in 2^{A^2}$ при $n = 4$ заданы матрицами $R(\rho_i)$, $i = 1,2,3$:

$$R(\rho_1) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & \mathbf{1} & 0 \\ 0 & \mathbf{1} & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}, \quad R(\rho_2) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, \quad R(\rho_3) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Тогда $I(\rho_1) = \langle 2,3,3,3\rangle$, $I(\rho_2) = \langle 1,2,3,4\rangle$, $I(\rho_3) = \langle 2,2,3,3\rangle$, $d(\rho_1,\rho_2) = 5$. При этом $\rho_1 \notin LA[4]$ (см. выделенные элементы в $R(\rho_1)$), $\rho_2 \in LO[4]$, $\rho_3 \in LA[4] \setminus LO[4]$.

Нетрудно показать [13], что справедливо

Утверждение 4. Пусть $\rho \in LA[n]$. Тогда $\rho \in LO[n] \Leftrightarrow I(\rho) = \langle 1, 2, \ldots, n\rangle$.

Замечание 3. Условие $I(\rho) = \langle 1, 2, \ldots, n\rangle$, где $\rho \in 2^{A^2}$, проверяется сложением n элементов каждой из n строк матрицы $R(\rho)$, т. е. имеет вычислительную сложность $O(n^2)$. Проверка условия $\rho \in LA[n]$, которая сводится к проверке выполнения (1), также имеет вычислительную сложность $O(n^2)$. Следовательно, проверка условия $\rho \in LO[n]$ имеет вычислительную сложность $O(n^2)$.

Таким образом, в случае, когда $\rho_1, \ldots, \rho_m \in LA[n]$ и m нечётно, мажоритарное отношение $\tilde{\rho}$, определяемое по «правилу большинства» из матриц P, L, принадлежит $LA[n]$, но может не принадлежать $LO[n]$. Поскольку мы ищем множество $\operatorname{Arg}\min\limits_{\rho\in LO[n]} D(\rho)$, нас интересуют отношения $\rho \in LO[n]$ со значениями $D(\rho)$, имеющими минимальное отклонение от $D(\tilde{\rho})$. В связи с этим можно рассматривать два типа задач: «полную» и «частичную». Решением «полной» задачи является безусловное (т. е. без дополнительных ограничений) нахождение множества всех отношений из $\operatorname{Arg}\min\limits_{\rho\in LO[n]} D(\rho)$. Такая задача в общем случае является экспоненциально сложной. Некоторыми методами её решения являются перебор, в том числе с использованием метода ветвей и границ (см. введение), а также метод (см. п. 5), основанный на исследовании графа $\tilde{G} = (A, \tilde{\rho})$. «Частичная» задача заключается в предварительном выяснении существования для заданного $\Delta_0 \in \{0, 2, 4, \ldots\}$ (см. следствие 2) отношения $\rho \in LO[n]$ с $D(\rho) \leqslant D(\tilde{\rho}) + \Delta_0$, и только в случае положительного ответа — в нахождении $\operatorname{Arg}\min\limits_{\rho\in LO[n]} D(\rho)$ (в случае отрицательного ответа считаем, что приемлемого отношения не существует). Как показано далее, такая задача является полиномиально сложной.

Замечание 4. Функцию $D(\rho) = \sum\limits_{t=1}^{m} d(\rho, \rho_t)$ можно модифицировать, добавив коэффициенты при $d(\rho, \rho_t)$. Возможный часто используемый подход заключается в следующем: в рассмотрение вводятся величины $D(\rho_t)$, $t = 1, \ldots, m$. В случае задачи выбора эти величины характеризуют «согласованность» i-го эксперта с другими экспертами: чем эта величина больше, тем «хуже» согласованность. Далее рассматриваются коэффициенты $\mu_t = \mu/D(\rho_t)$, где $\mu = \left[\sum\limits_{i=1}^{n} 1/D(\rho_i)\right]^{-1}$, $\sum\limits_{t=1}^{m} \mu_t = 1$. Таким образом, наряду с $D(\rho)$ можно использовать функцию $\tilde{D}(\rho) = \sum\limits_{t=1}^{m} \mu_t d(\rho, \rho_t)$. Можно рассматривать $\tilde{D}(\rho)$ как основную функцию (вместо $D(\rho)$), минимизация которой обеспечивает согласованность агрегированного отношения с профилем экспертных предпочтений, либо как вспомогательную — в случае неоднозначности выбора из $\operatorname{Arg}\min\limits_{\rho\in LO[n]} D(\rho)$ (для двухэтапной минимизации). Существует также немало процедур, позволяющих задавать коэффициенты предпочтительности α_i для альтернатив (процедура Коупленда и др. [5–8]), с учётом которых приходим к функции ещё одного вида:

$$\tilde{\tilde{D}}(\rho) = \sum_{t=1}^{m} \mu_t \tilde{d}(\rho, \rho_t) = \sum_{t=1}^{m} \mu_t \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \left| r_{ij} - r_{ij}^{(t)} \right|.$$

2. Условия существования и алгоритмы нахождения всех «почти оптимальных» решений

Получим сначала необходимое и достаточное условие существования «почти оптимального» (или 2-оптимального) отношения $\rho \in LO[n]$.

Теорема 2. Пусть $\rho_1, \ldots, \rho_m \in LA[n]$ и m нечётно, $\tilde{\rho}$ — мажоритарное отношение, $R(\tilde{\rho}) = [\tilde{r}_{ij}]_{n\times n}$, $\tilde{\rho} \notin LO[n]$. Тогда для существования «почти оптимального» отношения

необходимо и достаточно, чтобы для некоторого $\lambda \in LO[n]$ выполнялись следующие условия:

1) $R(\lambda) = [r_{ij}^{\lambda}]_{n\times n}$, $d(\lambda, \tilde{\rho}) = 2$, т. е. существуют такие $i_0, j_0 \in \{1, \ldots, n\}$, что $i_0 \neq j_0$, $r_{i_0,j_0}^{\lambda} = 1$, $\tilde{r}_{i_0,j_0} = 0$, $r_{j_0,i_0}^{\lambda} = 0$, $\tilde{r}_{j_0,i_0} = 1$, $r_{ij}^{\lambda} = \tilde{r}_{ij}$ при всех $\langle i, j\rangle \in J_0 = \{1, 2, \ldots, n\}^2 \setminus \{\langle i_0, j_0\rangle, \langle j_0, i_0\rangle\}$;
2) $p_{i_0,j_0} = (m-1)/2$, $p_{j_0,i_0} = (m+1)/2$.

Пример 2. Пусть $n = 4$, $m = 5$, $\rho_1 = \rho_2 : a_3 \prec a_2 \prec a_1 \prec a_4$, $\rho_3 : a_1 \prec a_2 \prec a_4 \prec a_3$, $\rho_4 : a_1 \prec a_4 \prec a_3 \prec a_2$, $\rho_5 : a_2 \prec a_1 \prec a_4 \prec a_3$. Тогда $R^{(t)} = R(\rho_t) = [r_{ij}^{(t)}]_{n\times n}$, $t = 1, 2, 3, 4, 5$, имеют следующий вид:

$$R^{(1)} = R^{(2)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R^{(3)} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad R^{(4)} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix},$$

$$R^{(5)} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix};$$

$$P = \sum_{t=1}^{5} R^{(t)} = [p_{ij}]_{4\times 4} = \begin{bmatrix} 5 & 2 & 3 & 5 \\ 3 & 5 & \mathbf{2} & 4 \\ 2 & \mathbf{3} & 5 & 2 \\ 0 & 1 & 3 & 5 \end{bmatrix}, \quad R(\tilde{\rho}) = [\tilde{r}_{ij}]_{4\times 4} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & \mathbf{0} & 1 \\ 0 & \mathbf{1} & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

Пусть $\lambda : a_2 \prec a_1 \prec a_4 \prec a_3$, $[r_{ij}^{\lambda}]_{4\times 4} = R(\lambda) = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & \mathbf{1} & 1 \\ 0 & \mathbf{0} & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$. Тогда при $i_0 = 2$, $j_0 = 3$ условия 1 и 2 теоремы 2 выполняются (для наглядности элементы $r_{2,3}^{\lambda}$, $\tilde{r}_{2,3}$, $r_{3,2}^{\lambda}$, $\tilde{r}_{3,2}$, $p_{2,3}$, $p_{3,2}$ выделены): $r_{2,3}^{\lambda} = 1$, $\tilde{r}_{2,3} = 0$, $r_{3,2}^{\lambda} = 0$, $\tilde{r}_{3,2} = 1$; $r_{ij}^{\lambda} = \tilde{r}_{ij}$ при всех $\langle i, j\rangle \in J_0 = \{1, 2, 3, 4\}^2 \setminus \{\langle 2, 3\rangle, \langle 3, 2\rangle\}$; $p_{2,3} = (m-1)/2 = 2$, $p_{3,2} = (m+1)/2 = 3$, при этом $D(\tilde{\rho}) = 18$, $D(\lambda) = 20 = D(\tilde{\rho}) + 2$.

Для доказательства теоремы 2 используется

Утверждение 5 [13]. Пусть $\rho, \ldots, \rho_m \in LA[n]$ и m нечётно, $\tilde{\rho}$ — мажоритарное отношение, $R(\tilde{\rho}) = [\tilde{r}_{ij}]_{n\times n}$, $\rho \in LA[n]$, $R(\rho) = [r_{ij}]_{n\times n}$, $J_\rho = \{\langle i, j\rangle \in \{1, \ldots, n\}^2 : \tilde{r}_{ij} \neq r_{ij}\}$. Тогда

$$D(\rho) = D(\tilde{\rho}) + \sum_{\langle i,j\rangle \in J_\rho} (\max\{p_{ij}, p_{ji}\} - \min\{p_{ij}, p_{ji}\}) = D(\tilde{\rho}) + 2 \sum_{\langle i,j\rangle \in J_\rho, c_{ij} \neq \infty} c_{ij} \geqslant$$
$$\geqslant D(\tilde{\rho}) + |J_\rho| = D(\tilde{\rho}) + d(\rho, \tilde{\rho}). \tag{3}$$

Приведем также следующее

Утверждение 6 [13]. Пусть $\rho \in LA[n]$. Тогда $\rho \in LO[n]$ в том и только в том случае, когда в графе $G(\rho) = (A, \rho)$, рассматриваемом без петель, нет контуров.

Следствие 3. Пусть, в условиях теоремы 2, для некоторого $\lambda \in LO[n]$ выполняются условия 1 и 2 этой теоремы. Тогда $c_{i_0,j_0} = \infty$, $c_{j_0,i_0} = 1$ и дуга $\langle a_{i_0}, a_{j_0}\rangle$ входит во все контуры графа $G(\tilde{\rho}) = (A, \tilde{\rho})$.

Доказательство. Поскольку $\tilde{\rho} \notin LO[n]$, то в силу утверждения 6 в $\tilde{G}$ имеется контур. В λ в силу того же утверждения контуров нет и λ получается из $\tilde{\rho}$ заменой одной дуги $\langle a_{i_0}, a_{j_0}\rangle$ на противоположную, поэтому эта замена должна «разрушить» все контуры в $\tilde{G}$, т. е. эта дуга входит во все контуры графа $\tilde{G}$. ■

Утверждение 6 и следствие 3 позволяют выяснить геометрический смысл «почти оптимального» решения. Оно существует тогда и только тогда, когда в нагруженном графе $G(\tilde{\rho}) = (A, \tilde{\rho})$ (рассматриваемом без петель) найдётся дуга $\langle a_i, a_j\rangle$ минимально возможной длины $c_{ij} = 1$, замена которой на противоположную приводит к графу без контуров. На рис. 1 представлен граф $\tilde{G} = (A, \tilde{\rho})$ для $\tilde{\rho}$ из примера 2; замена дуги $\langle a_3, a_2\rangle$ на $\langle a_2, a_3\rangle$ «разрушает» все три контура в $\tilde{G}$.

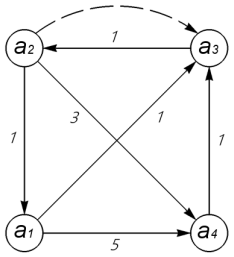

Рис. 1

Используя теорему 2 и следствие 3, опишем простой алгоритм 1 определения существования «почти оптимального» решения и нахождения всех таких решений.

Алгоритм 1. Нахождение всех «почти оптимальных» решений

1: Находим в нагруженном графе $\tilde{G}$ все дуги длины 1.
2: Для каждой из найденных в п. 1 дуг рассматриваем граф, полученный из $\tilde{G}$ заменой этой дуги на противоположную. Если в полученном графе отсутствуют контуры, то одно из возможных «почти оптимальных» решений найдено. Действуя так, получим их все.
3: Если после перебора в п. 2 всех дуг длины 1 не нашли ни одного «почти оптимального» решения, то такого решения не существует.

Алгоритм 1 использует полиномиально сложную процедуру проверки графа на наличие контуров, применяемую не более $n(n-1)/2$ (количество дуг в $\tilde{G}$, отличных от петель) раз, т. е. также имеет полиномиальную сложность вычислений. Проверку графа $\tilde{G}' = (A, \tilde{\rho}')$ на наличие контуров, где $\tilde{\rho}' \in LA[n]$ — бинарное отношение, полученное заменой одной из пар в $\tilde{\rho}$ длины 1 на противоположную, можно в силу утверждения 6 заменить на проверку $\tilde{\rho}' \in LO[n]$, равносильную проверке $I(\tilde{\rho}') = \langle 1, 2, \ldots, n\rangle$ (см. утверждение 4), имеющей сложность $O(n^2)$. Таким образом, алгоритм 1 имеет вычислительную сложность $O(n^4)$.

Если дуг длины 1 в $\tilde{G}$ много, а дуг, входящих в контуры, сравнительно мало, то можно воспользоваться также алгоритмом 2.

Алгоритм 2. Нахождение всех «почти оптимальных» решений

1: Находим в $\tilde{G}$ все дуги, входящие в контуры. Выделяем среди них дуги длины 1.
2: Далее действуем согласно пп. 2 и 3 алгоритма 1.

Если Δ-оптимального решения при $\Delta = 2$ не оказалось, то можем попытаться найти его при $\Delta = 4$ (это имеет смысл делать, если величина $4/D(\tilde{\rho})$ достаточно мала). В этом случае, согласно второму равенству в (3), проверяем на наличие контуров графы, полученные из $\tilde{G}$ заменой на противоположные ровно двух дуг длины 1.

Если Δ-оптимального решения при $\Delta \in \{2, 4\}$ не оказалось, то можем попытаться найти его при $\Delta = 6$ (это имеет смысл делать, если величина $6/D(\tilde{\rho})$ достаточно мала). В этом случае, согласно второму равенству в (3), проверяем на наличие контуров графы, полученные из $\tilde{G}$ заменой на противоположные ровно трёх дуг длины 1 или ровно одной дуги длины 3 (очевидно, что если $c_{ij} \neq \infty$, $i \neq j$, то $c_{ij} = p_{ij} - p_{ji} \in \{1, 3, 5, \ldots, m\}$), и т. д. При этом алгоритм остаётся полиномиальным, если ограничиваем величину Δ сверху некоторой константой.

Опишем процедуру проверки $\tilde{\rho}$ на «почти оптимальность», использующую простой подсчёт количества единиц в строках матрицы $R(\tilde{\rho})$ (подобно условию $I(\rho) = \langle 1, 2, \ldots, n\rangle$ для проверки $\tilde{\rho} \in LO[n]$). Кроме того, выясним структуру множества всех $\tilde{\rho} \in LA[n]$, для которых выполняется условие 1 теоремы 2.

Из определения $LA[n]$ следует, что для всех $\rho \in LA[n]$ для $I(\rho) = \langle i_1, \ldots, i_n\rangle$ выполняется

$$1 \leqslant i_1 \leqslant \ldots \leqslant i_n \leqslant n, \quad i_1 + \ldots + i_n = n(n+1)/2. \tag{4}$$

Замечание 5. Отметим, что не всякому целочисленному решению $\langle i_1, \ldots, i_n\rangle$ системы (4) соответствует некоторое $\rho \in LA[n]$, для которого $I(\rho) = \langle i_1, \ldots, i_n\rangle$. Например, вектор $\langle 1, 1, 4, 4\rangle$ является решением системы (4) при $n = 4$, но не существует $\rho \in LA[4]$ с $I(\rho) = \langle 1, 1, 4, 4\rangle$. Действительно, существование в $R(\rho)$ строки с некоторым номером $j \in \{1, \ldots, n\}$ с одной единицей (на главной диагонали) приводит к тому, что в j-м столбце будут только единицы, а следовательно, в $R(\rho)$ не может быть ещё одной строки с одним единичным элементом. Таким образом, из множества целочисленных решений системы (4), соответствующих $\langle i_1, \ldots, i_n\rangle = I(\rho)$, где $\rho \in LA[n]$, следует удалить все решения с $i_1 = i_2 = 1$ и (по аналогичным соображениям) все решения с $i_{n-1} = i_n = n$. Этим не исчерпываются ограничения на целочисленные решения системы (4) для $I(\rho) = \langle i_1, \ldots, i_n\rangle$, где $\rho \in LA[n]$. Например, как будет видно из дальнейшего, не существует $\rho \in LA[6]$ с $I(\rho) = \langle 1, 2, 2, 5, 5, 6\rangle$, хотя для этого вектора условия (4) при $n = 6$ выполняются.

Обозначим для всех $n \geqslant 2$:

$$J_n = \{\langle i_1, \ldots, i_n\rangle : \exists \rho \in LA[n]\,(I(\rho) = \langle i_1, \ldots, i_n\rangle)\},$$
$$\forall \langle i_1, \ldots, i_n\rangle \in J_n \quad LA_{\langle i_1, \ldots, i_n\rangle}[n] = \{\rho \in LA[n] : I(\rho) = \langle i_1, \ldots, i_n\rangle\}.$$

Для каждого $\rho \in LA[n]$ введём в рассмотрение вектор

$$\overline{I}(\rho) = I(\rho) - \langle 1, 2, \ldots, n\rangle = \langle i_1 - 1, i_2 - 2, \ldots, i_n - n\rangle$$

и множество $\overline{J}_n = J_n - \langle 1, 2, \ldots, n\rangle = \{\overline{I}(\rho) : \rho \in LA[n])\}$. Нам понадобится следующее множество V_n целочисленных векторов с элементами из $\{0, 1, -1\}$, где $n \geqslant 3$:

$$V_n = \{\langle \underbrace{0, \ldots, 0}_{s_1}, 1, \underbrace{0, \ldots, 0}_{s_2}, -1, \underbrace{0, \ldots, 0}_{s_3}\rangle \in \{0, 1, -1\}^n : s_1 \geqslant 0, s_2 \geqslant 1, s_3 \geqslant 0, s_1 + s_2 + s_3 = n - 2\}.$$

Например,

$$V_3 = \{\langle 1, 0, -1\rangle\}, \quad V_4 = \{\langle 1, 0, -1, 0\rangle, \langle 1, 0, 0, -1\rangle, \langle 0, 1, 0, -1\rangle\},$$
$$V_5 = \{\langle 1, 0, -1, 0, 0\rangle, \langle 1, 0, 0, -1, 0\rangle, \langle 1, 0, 0, 0, -1\rangle, \langle 0, 1, 0 - 1, 0\rangle, \langle 0, 1, 0, 0, -1\rangle, \langle 0, 0, 1, 0, -1\rangle\}.$$

Утверждение 7 [13]. $|V_n| = (n-1)(n-2)/2$.

Далее покажем, что для всех $\rho \in LA[n] \setminus LO[n]$ для существования $\lambda \in LO[n]$, такого, что $d(\lambda, \rho) = 2$, необходимо выполнение условия $\overline{I}(\rho) \in V_n$. Нам потребуются некоторые свойства векторов из $\overline{J}_n$. Справедливо

Утверждение 8 [13]. Пусть $n \geqslant 3$, $\rho \in LA[n] \setminus LO[n]$. Тогда $\overline{I}(\rho)$ содержит не менее двух ненулевых компонент, первая из которых положительная, а последняя — отрицательная. Кроме того, после положительной компоненты не может сразу следовать отрицательная (между ними должна быть хотя бы одна нулевая компонента).

Доказательство утверждения 8 основано на рассуждениях, аналогичных приведённым в замечании 3.

Введём в рассмотрение два преобразования бинарных отношений ρ на A, которые не выводят из $LA[n]$. Пусть S_n — множество всех биективных отображений (подстановок) вида $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$. Обозначим для $A = \{a_1, a_2, \ldots, a_n\}$, $\rho \in 2^{A^2}$, $\pi \in S_n$

$$\pi\rho = \left\{\langle a_{\pi(i)}, a_{\pi(j)}\rangle : \langle a_i, a_j\rangle \in \rho\right\}$$

(π-преобразование бинарного отношения ρ на A), а в случае $A = \{1, 2, \ldots, n\}$

$$\pi\rho = \left\{\langle \pi(i), \pi(j)\rangle : \langle i, j\rangle \in \rho\right\}.$$

Пример 3. Пусть $n = 4$, $\rho = \{\langle a_1, a_1\rangle, \langle a_1, a_2\rangle, \langle a_1, a_3\rangle, \langle a_2, a_4\rangle, \langle a_3, a_4\rangle\}$, $\pi = (1, 2, 4) \in S_4$. Тогда $\pi\rho = \{\langle a_2, a_2\rangle, \langle a_2, a_4\rangle, \langle a_2, a_3\rangle, \langle a_4, a_1\rangle, \langle a_3, a_1\rangle\}$.

При таком определении, очевидно, выполняется следующее условие:

$$\forall \rho, \rho_1 \in 2^{A^2} \;\; \forall \pi, \pi_1, \pi_2 \in S_n \;\; \big((\pi_1\pi_2)\rho = \pi_1(\pi_2\rho), \;\; \rho_1 = \pi\rho \Rightarrow \rho = \pi^{-1}\rho_1\big),$$

где $(\pi_1\pi_2)(i) = \pi_1(\pi_2(i))$, $i \in \{1, \ldots, n\}$. Заметим далее, что для всех $\rho \in 2^{A^2}$ для $R(\rho) = [r_{ij}]_{n\times n}$ справедливо

$$\forall \pi \in S_n \; \big(R(\pi\rho) = [\tilde{r}_{ij}]_{n\times n} \;\Rightarrow\; \tilde{r}_{ij} = r_{\pi^{-1}(i),\pi^{-1}(j)}\big),$$

т. е. если в i-й строке матрицы $R(\rho)$ содержится l единиц и $n - l$ нулей, где $i, l \in \in \{1, \ldots, n\}$, то в $\pi(i)$-й строке матрицы $R(\pi\rho)$ также содержится l единиц и $n - l$ нулей. Следовательно, справедливо

Утверждение 9. Пусть $n \geqslant 2$, $\langle i_1, \ldots, i_n\rangle \in J_n$, $\rho \in LA_{\langle i_1,\ldots,i_n\rangle}[n]$. Тогда для всех $\pi \in S_n$ выполняется $\pi\rho \in LA_{\langle i_1,\ldots,i_n\rangle}[n]$.

Будем использовать ещё одно преобразование. Обозначим для всех $\rho \in 2^{A^2}$ и $\langle i_0, j_0\rangle \in \left\{\langle i, j\rangle \in \{1, 2, \ldots, n\}^2 : i < j\right\}$

$$\delta_{i_0,j_0}\rho = \{\langle a_i, a_j\rangle \in \rho : \langle i, j\rangle \notin \{\langle i_0, j_0\rangle, \langle j_0, i_0\rangle\}\} \cup$$
$$\cup \left\{\langle a_{i_0}, a_{j_0}\rangle : \langle a_{j_0}, a_{i_0}\rangle \in \rho\right\} \cup \left\{\langle a_{j_0}, a_{i_0}\rangle : \langle a_{i_0}, a_{j_0}\rangle \in \rho\right\}.$$

В соответствии с этим определением для всех $\rho \in 2^{A^2}$ для $R(\rho) = [r_{ij}]_{n\times n}$ выполняется

$$\forall \langle i_0, j_0\rangle \in \left\{\langle i, j\rangle \in \{1, \ldots, n\}^2 : i < j\right\} \;\; R(\delta_{i_0,j_0}\rho) = [r'_{ij}]_{n\times n},$$

где $r'_{i_0,j_0} = r_{j_0,i_0}$; $r'_{j_0,i_0} = r_{i_0,j_0}$; $\forall \langle i, j\rangle \in \{1, \ldots, n\}^2 \setminus \{\langle i_0, j_0\rangle, \langle j_0, i_0\rangle\}$ ($r'_{ij} = r_{ij}$), т. е. преобразование δ_{i_0,j_0} меняет ровно два элемента r_{i_0,j_0}, r_{j_0,i_0} местами. Из определения этого преобразования следует, что если $\rho \in LA[n]$, то

$$\forall \langle i_0, j_0\rangle \in \left\{\langle i, j\rangle \in \{1, 2, \ldots, n\}^2 : i < j\right\} \;\; \rho' = \delta_{i_0,j_0}\rho \in LA[n],$$

и при этом вектор $I(\rho) - I(\rho')$ содержит ровно один элемент «1», ровно один элемент «-1» и $n-2$ элементов «0», т. е. $I(\rho) \neq I(\rho')$, а следовательно, $\rho' \notin LA_{\langle i_1,\ldots,i_n\rangle}[n]$, где $\langle i_1,\ldots,i_n\rangle = I(\rho)$.

Утверждение 10. Пусть $\rho \in LA[n] \setminus LO[n]$, $d(\rho, LO[n]) = 2$. Тогда существуют такие $\lambda \in LO[n]$, $\langle i_0, j_0\rangle \in \left\{\langle i,j\rangle \in \{1,2,\ldots,n\}^2 : i<j\right\}$, что верны следующие условия:

1) $\rho = \delta_{i_0,j_0}\lambda$, $\lambda = \delta_{i_0,j_0}\rho$;
2) $\overline{I}(\rho) \in V_n$, т. е. в векторе $\overline{I}(\rho)$ ровно две ненулевые компоненты 1 и -1, находящиеся в v_1-й и v_2-й позициях соответственно, где $v_1 \leqslant v_2 - 2$ (см. утверждение 8);
3) $i_0 = \min\{i'_0, j'_0\}$, $j_0 = \max\{i'_0, j'_0\}$, $i'_0 \in \left\{i \in \{1,\ldots,n\} : \kappa_i(\rho) = \sum\limits_{j=1}^{n} r_{ij} = v_1 + 1\right\}$, $j'_0 \in \left\{i \in \{1,\ldots,n\} : \kappa_i(\rho) = \sum\limits_{j=1}^{n} r_{ij} = v_2 - 1\right\}$, $r_{i'_0,j'_0} = 1, r_{j'_0,i'_0} = 0$.

Утверждение 10 является очевидным следствием того, что:

а) по его условиям для некоторого $\lambda \in LO[n]$ матрицы $R(\rho) = [r_{ij}]_{n\times n}$, $R(\lambda) = [r^{\lambda}_{ij}]_{n\times n}$ имеют различие ровно в двух позициях (симметричных относительно главной диагонали, поскольку $\rho, \lambda \in LA[n]$): $\langle i_0, j_0\rangle$ и $\langle j_0, i_0\rangle$, где $i_0 < j_0$;

б) $I(\lambda) = \langle 1,2,\ldots,n\rangle$, а следовательно, в силу свойства а, $\overline{I}(\rho) \in V_n$ (поскольку в $R(\rho)$ по сравнению с $R(\lambda)$ в i_0-й строке на одну единицу меньше (или больше), а в j_0-й строке соответственно на одну единицу больше (или меньше));

в) если $r^{\lambda}_{i_0,j_0} = 0$, то для $v_1 = \sum\limits_{j=1}^{n} r^{\lambda}_{i_0,j}$, $v_2 = \sum\limits_{j=1}^{n} r^{\lambda}_{j_0,j}$ в векторе $\overline{I}(\rho) \in V_n$ компоненты 1 и -1 находятся в v_1-й и v_2-й позициях соответственно, а если $r^{\lambda}_{i_0,j_0} = 1$, то это выполняется для $v_1 = \sum\limits_{j=1}^{n} r^{\lambda}_{j_0,j}$, $v_2 = \sum\limits_{j=1}^{n} r^{\lambda}_{i_0,j}$.

Замечание 6. Условия 1–3 утверждения 10, очевидно, являются достаточными для существования для некоторого $\rho \in LA[n] \setminus LO[n]$ бинарного отношения $\lambda \in LO[n]$, такого, что $d(\rho, \lambda) = 2$.

Используя утверждение 10, нетрудно описать следующий алгоритм 3.

Алгоритм 3. Нахождение для $\rho \in LA[n] \setminus LO[n]$ всех $\lambda \in LO[n]$, таких, что $d(\rho, \lambda) = 2$

1: Находим по матрице $R(\rho) = [r_{ij}]_{n\times n}$ векторы $I(\rho) = \langle i_1, i_2, \ldots, i_n\rangle$, $\overline{I}(\rho) = I(\rho) - \langle 1,2,\ldots,n\rangle$. Проверяем выполнение $\overline{I}(\rho) \in V_n$. Если это условие не выполняется, то в силу утверждения 10 искомых $\lambda \in LO[n]$ не существует и работа алгоритма заканчивается. В противном случае переходим к п. 2.

2: Находим номера v_1, v_2 компонент вектора $\overline{I}(\rho) \in V_n$, равных 1 и -1 соответственно. Тогда $v_1 \leqslant v_2 - 2$. Определяем множества

$$I_1 = \left\{i \in \{1,\ldots,n\} : v_1 + 1 = i_{v_1} = \sum_{j=1}^{n} r_{ij}\right\},\ I_2 = \left\{i \in \{1,\ldots,n\} : v_2 - 1 = i_{v_2} = \sum_{j=1}^{n} r_{ij}\right\},$$

где i_{v_1}, i_{v_2} — компоненты вектора $I(\rho) = \langle i_1,\ldots,i_n\rangle$.

3: Находим все упорядоченные пары $\langle i_0, j_0\rangle$, такие, что $i_0 \in I_1$, $j_0 \in I_2$, $i_0 \neq j_0$, $r_{i_0,j_0} = 1$. Если таких пар нет, то в силу утверждения 10 искомого $\lambda \in LO[n]$ не существует и работа алгоритма заканчивается. В противном случае для каждой найденной пары $\langle i_0, j_0\rangle$ одним из искомых линейных порядков в случае $(i_0 < j_0)$ является $\lambda = \delta_{i_0,j_0}\rho$, а в случае $(j_0 < i_0)$ — $\lambda = \delta_{j_0,i_0}\rho$.

Замечание 7. Очевидно, что на шаге 2 алгоритма 3, если $v_1 = v_2 - 2$ (т. е. в векторе $\overline{I}(\rho) \in V_n$ ровно один нуль между 1 и -1), то $I_1 = I_2$ и $|I_1| = |I_2| = 3$ (например, в случае $\overline{I}(\rho) = \langle 0, 1, 0, -1, 0\rangle$ выполняется $I(\rho) = \langle 1, 3, 3, 3, 5\rangle$), а если $v_1 \leqslant v_2 - 3$ (т. е. в векторе $\overline{I}(\rho) \in V_n$ более одного нуля между 1 и -1), то $I_1 \cap I_2 = \varnothing$ и $|I_1| = |I_2| = 2$ (например, в случае $\overline{I}(\rho) = \langle 0, 1, 0, 0, -1\rangle$ выполняется $I(\rho) = \langle 1, 3, 3, 4, 4\rangle$). В первом случае для любого $n \geqslant 3$ на шаге 3 проверяем шесть элементов r_{i_0,j_0}, где $i_0, j_0 \in \in I_1 = I_2$, $i_0 \neq j_0$, для трёх из которых $r_{i_0,j_0} = 1$, т. е. всегда имеем ровно три решения. Во втором случае проверяем четыре элемента r_{i_0,j_0}, где $i_0 \in I_1$, $j_0 \in I_2$ (т. е. решений в этом случае не более четырёх, и как показано ниже, оно единственное). Нетрудно видеть, что вычислительная сложность алгоритма 3 равна $O(n^2)$ (аналогично проверке выполнения условия $I(\rho) = \langle 1, 2, \ldots, n\rangle$).

Пример 4. Применим алгоритм 3 к $\rho_1, \rho_2 \in LA[5] \setminus LO[5]$, заданными матрицами $R(\rho_1) = [r^{(1)}_{ij}]_{5\times5}$, $R(\rho_2) = [r^{(2)}_{ij}]_{5\times5}$:

$$R(\rho_1) = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ \mathbf{0} & \mathbf{0} & 0 & 1 & 1 \\ \mathbf{1} & \mathbf{0} & 0 & 0 & 1 \end{bmatrix}, \quad R(\rho_2) = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ \mathbf{0} & \mathbf{0} & 1 & 1 & 0 \\ \mathbf{0} & \mathbf{0} & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Заметим, что

$$I(R(\rho_1)) = I(R(\rho_2)) = \langle 2, 2, 3, 4, 4\rangle,\ \overline{I}(R(\rho_1)) = \overline{I}(R(\rho_2)) = \langle 1, 0, 0, 0, -1\rangle \in V_5.$$

Рассмотрим сначала ρ_1. Согласно п. 2 алгоритма 3, определяем $v_1 = 1$, $v_2 = 5$, $i_{v_1} = 2$, $i_{v_2} = 4$,

$$I_1 = \left\{ i \in \{1, 2, 3, 4, 5\} : v_1 + 1 = i_{v_1} = 2 = \sum_{j=1}^{5} r^{(1)}_{ij} \right\} = \{4, 5\},$$

$$I_2 = \left\{ i \in \{1, 2, 3, 4, 5\} : v_2 - 1 = i_{v_2} = 4 = \sum_{j=1}^{5} r^{(1)}_{ij} \right\} = \{1, 2\}.$$

Далее перебираем все значения $r^{(1)}_{i_0,j_0}$, где $i_0 \in I_1 = \{4, 5\}$, $j_0 \in I_2 = \{1, 2\}$ (выделены в матрице $R(\rho_1)$): $r^{(1)}_{4,1} = 0$, $r^{(1)}_{5,1} = 1$, $r^{(1)}_{4,2} = 0$, $r^{(1)}_{5,2} = 0$. Таким образом, нашлась единственная пара $\langle 5, 1\rangle$, такая, что $r^{(1)}_{5,1} = 1$, а следовательно, существует единственный линейный порядок $\lambda_1 = \delta_{1,5}\rho$, такой, что $d(\rho_1, \lambda_1) = 2$:

$$R(\lambda_1) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Рассмотрим ρ_2. Согласно п. 2 алгоритма 3, определяем $v_1 = 1$, $v_2 = 5$, $i_{v_1} = 2$, $i_{v_2} = 4$,

$$I_1 = \left\{ i \in \{1, 2, 3, 4, 5\} : v_1 + 1 = i_{v_1} = 2 = \sum_{j=1}^{5} r^{(1)}_{ij} \right\} = \{3, 4\},$$

$$I_2 = \left\{ i \in \{1, 2, 3, 4, 5\} : v_2 - 1 = i_{v_2} = 4 = \sum_{j=1}^{5} r^{(1)}_{ij} \right\} = \{1, 2\}.$$

Далее перебираем все значения $r^{(2)}_{i_0,j_0}$, где $i_0 \in I_1 = \{3,4\}$, $j_0 \in I_2 = \{1,2\}$ (выделены в матрице $R(\rho_2)$): $r^{(2)}_{3,1} = 0$, $r^{(2)}_{4,1} = 0$, $r^{(2)}_{3,2} = 0$, $r^{(2)}_{4,2} = 0$. Не нашлось ни одной пары $\langle i_0, j_0\rangle$, такой, что $i_0 \in I_1, j_0 \in I_2$, $r^{(2)}_{i_0,j_0} = 1$, а следовательно, в силу утверждения 10, искомого $\lambda \in LO[5]$ с $d(\rho_2, \lambda) = 2$ не существует и работа алгоритма заканчивается. Нетрудно показать, что ближайшим к ρ_2 линейным порядком также является λ_1, но при этом $d(\rho_2, \lambda_1) = 4$.

Цель дальнейших рассуждений — показать, что множество бинарных отношений $LA^{(2)}[n] = \left\{\rho \in LA[n] \setminus LO[n] : \min\limits_{\lambda \in LO[n]} d(\lambda, \rho) = 2\right\}$ является достаточно широким, а также описать структуру этого множества. Далее установлено, что в случае $n = 4$ выполняется $LA^{(2)}[4] = LA[4] \setminus LO[4]$, но уже для $n = 5$ ситуация меняется и множество $LA^{(2)}[5]$ составляет лишь некоторую (хотя и достаточно широкую) часть множества $LA[5] \setminus LO[5]$. Проиллюстрируем рассуждения для $n = 5$. Пусть $\lambda^{(n)}_1$ — линейный порядок, определяемый условиями $a_1 \prec a_2 \prec \ldots \prec a_n$ (в случае $n = 5$ с матрицей $R(\lambda^{(5)}_1)$):

$$R(\lambda^{(5)}_1) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Перечислим все возможные $\rho \in LA[n] \setminus LO[n]$, такие, что $d(\lambda^{(n)}_1, \rho) = 2$. Очевидно, что любое из таких бинарных отношений получается из $\lambda^{(n)}_1$ заменой ровно одной пары с разными элементами на противоположную, т. е. является бинарным отношением вида $\rho = \delta_{i,j}\lambda^{(n)}_1$, где $\langle i, j\rangle \in \{\langle i, j\rangle \in \{1, \ldots, n\}^2 : i < j\}$. Исключением являются случаи с $j = i + 1$, поскольку $\delta_{i,i+1}\lambda^{(n)}_1$ (где $i = 1, 2, \ldots, n-1$) — линейные порядки вида $a_1 \prec a_2 \prec \ldots \prec a_{i-1} \prec a_{i+1} \prec a_i \prec a_{i+2} \prec \ldots \prec a_n$. Например, $\lambda^{(5)}_2 = \delta_{1,2}\lambda^{(5)}_1 : a_2 \prec a_1 \prec \prec a_3 \prec a_4 \prec a_5$,

$$R(\lambda^{(5)}_2) = \begin{bmatrix} 1 & \mathbf{0} & 1 & 1 & 1 \\ \mathbf{1} & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Во всех остальных $(n-1)(n-2)/2$ случаях (их шесть при $n = 5$) получаем попарно различные бинарные отношения из $LA^{(2)}[n]$. В случае $n = 5$ имеем

$$\rho^{(5)}_{1,3} = \delta_{1,3}\lambda^{(5)}_1,\ I(\rho^{(5)}_{1,3}) = \langle 1,2,4,4,4\rangle,\ \overline{I}(\rho^{(5)}_{1,3}) = \langle 0,0,1,0,-1\rangle;$$
$$\rho^{(5)}_{1,4} = \delta_{1,4}\lambda^{(5)}_1,\ I(\rho^{(5)}_{1,4}) = \langle 1,3,3,4,4\rangle,\ \overline{I}(\rho^{(5)}_{1,4}) = \langle 0,1,0,0,-1\rangle;$$
$$\rho^{(5)}_{1,5} = \delta_{1,5}\lambda^{(5)}_1,\ I(\rho^{(5)}_{1,5}) = \langle 2,2,3,4,4\rangle,\ \overline{I}(\rho^{(5)}_{1,5}) = \langle 1,0,0,0,-1\rangle;$$
$$\rho^{(5)}_{2,4} = \delta_{2,4}\lambda^{(5)}_1,\ I(\rho^{(5)}_{2,4}) = \langle 1,3,3,3,5\rangle,\ \overline{I}(\rho^{(5)}_{2,4}) = \langle 0,1,0,-1,0\rangle;$$
$$\rho^{(5)}_{2,5} = \delta_{2,5}\lambda^{(5)}_1,\ I(\rho^{(5)}_{2,5}) = \langle 2,2,3,3,5\rangle,\ \overline{I}(\rho^{(5)}_{2,5}) = \langle 1,0,0,-1,0\rangle;$$
$$\rho^{(5)}_{3,5} = \delta_{3,5}\lambda^{(5)}_1,\ I(\rho^{(5)}_{3,5}) = \langle 2,2,2,4,5\rangle,\ \overline{I}(\rho^{(5)}_{3,5}) = \langle 1,0,-1,0,0\rangle.$$

Матрица $R(\rho^{(5)}_{1,3})$ имеет следующий вид:

$$R(\rho^{(5)}_{1,3}) = \begin{bmatrix} 1 & 1 & \mathbf{0} & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ \mathbf{1} & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

В общем случае с произвольным $n \geqslant 5$:

$$\overline{I}(\delta_{l,k}\lambda^{(n)}_1) = \langle \underbrace{0,\ldots,0}_{n-k}, 1, \underbrace{0,\ldots,0}_{k-l-1}, -1, \underbrace{0,\ldots,0}_{l-1} \rangle, \quad l = 1,\ldots,n-2, \quad k = l+2,\ldots,n. \tag{5}$$

Зная $\overline{I}(\delta_{l,k}\lambda^{(n)}_1)$, можно легко найти $I(\delta_{l,k}\lambda^{(n)}_1) = \overline{I}(\delta_{l,k}\lambda^{(n)}_1) + \langle 1,2,\ldots,n\rangle$.

Обозначим $\Lambda^{(n)} = \{\delta_{l,k}\lambda^{(n)}_1 : l = 1,\ldots,n-2,\ k = l+2,\ldots,n\}$. Из (5) следует, что элементы в $\Lambda^{(n)}$ попарно различны. Таким образом, справедливо

Утверждение 11. Пусть $n \geqslant 3$. Тогда

$$\Lambda^{(n)} = \{\delta_{l,k}\lambda^{(n)}_1 : l = 1,\ldots,n-2,\ k = l+2,\ldots,n\} = \{\rho \in LA[n] \setminus LO[n] : d(\lambda^{(n)}_1, \rho) = 2\}.$$

Найденный набор бинарных отношений вида $\delta_{l,k}\lambda^{(n)}_1$, $l = 1,\ldots,n-2$, $k = l+2,\ldots,n$, из искомого множества $LA^{(2)}[n]$, содержащий $(n-1)(n-2)/2$ попарно различных бинарных отношений, на самом деле является набором представителей попарно не пересекающихся подмножеств этого множества, совокупность которых является разбиением $LA^{(2)}[n]$. Для описания этого разбиения понадобятся некоторые дополнительные свойства рассмотренных ранее π-преобразований бинарных отношений на A.

Назовём бинарные отношения $\rho_1, \rho_2 \in 2^{A^2}$ π-эквивалентными, если $\rho_1 = \pi\rho_2$ для некоторой подстановки $\pi \in S_n$ (пишем $\rho_1 \sim \rho_2$). Очевидно, что это бинарное отношение на 2^{A^2} является рефлексивным, симметричным и транзитивным, т. е. является отношением эквивалентности на всём множестве 2^{A^2}, а следовательно, и на любом его подмножестве.

Для любого бинарного отношения $\rho \in 2^{A^2}$ обозначим $\Pi\rho = \{\pi\rho : \pi \in S_n\}$, $H(\rho) = \{h \in S_n : h\rho = \rho\}$ — подгруппа группы S_n. Очевидно, что $\Pi\rho$ — класс эквивалентности бинарного отношения ρ относительно π-эквивалентности. Цель ближайших рассуждений — доказать равенство

$$LA^{(2)}[n] = \{\rho \in LA[n] \setminus LO[n] : \min_{\lambda \in LO[n]} d(\lambda, \rho) = 2\} = \bigcup\nolimits_{\rho \in \Lambda^{(n)}} \Pi\rho,$$

где $\Pi\rho_1 \cap \Pi\rho_2 = \varnothing$ для любых неравных $\rho_1, \rho_2 \in \Lambda^{(n)}$. Кроме того, докажем утверждения, позволяющие определять $|\Pi\rho|$, где $\rho \in LA[n]$, а тем самым и $|LA^{(2)}[n]| = \sum\limits_{\rho \in \Lambda^{(n)}} |\Pi\rho|$.

Утверждение 12 [13]. Пусть $\rho_1, \rho_2 \in 2^{A^2}$, $\pi \in S_n$. Тогда $d(\rho_1, \rho_2) = d(\pi\rho_1, \pi\rho_2)$.

Пусть $\rho \in 2^{A^2}$. Заметим, что $\pi_1\rho \neq \pi_2\rho \Leftrightarrow \pi_1 H(\rho) \cap \pi_2 H(\rho) = \varnothing$ для всех $\pi_1, \pi_2 \in S_n$, т. е. количество попарно различных бинарных отношений в $\Pi\rho$ совпадает с количеством левых смежных классов группы S_n по подгруппе $H(\rho)$. Таким образом, справедливо

Утверждение 13. Для всех $\rho \in 2^{A^2}$

$$|\Pi\rho| = |S_n / H(\rho)| = \frac{|S_n|}{|H(\rho)|} = \frac{n!}{|H(\rho)|}.$$

Для дальнейшего потребуются некоторые свойства подгруппы $H(\rho)$.

Утверждение 14 [13]. Пусть $\rho \in LA[n]$, $\pi \in S_n$ и в разложении π в произведение независимых циклов присутствует хотя бы один цикл чётной длины. Тогда $\pi\rho \neq \rho$, т. е. $\pi \notin H(\rho)$.

Утверждение 15. Пусть $n \geqslant 2$, $\rho \in 2^{A^2}$, $\pi \in H(\rho)$, $i, j \in \{1, 2, \ldots, n\}$, $i \neq j$, $\pi(i) = j$. Тогда в i-й и j-й строках матрицы $R(\rho)$ одинаковое число единиц (нулей), т. е. $\kappa_i(\rho) = \kappa_j(\rho)$.

Для удобства применения утверждения 15 воспользуемся следующими обозначениями. Пусть $n \geqslant 2$, $\rho \in 2^{A^2}$. Введем в рассмотрение множества

$$A_1(\rho) = \{a_i \in A : \kappa_i(\rho) = 1\}, \quad \ldots, \quad A_n(\rho) = \{a_i \in A : \kappa_i(\rho) = n\}.$$

Они образуют разбиение множества A. Из утверждения 15 следует

Утверждение 16. Пусть $n \geqslant 2$, $\rho \in 2^{A^2}$, $\pi \in H(\rho)$ и $\sigma = (i_1, i_2, \ldots, i_k)$ — цикл длины $k \geqslant 2$ из разложения подстановки π в произведение независимых циклов. Тогда найдётся номер $j \in \{1, \ldots, n\}$, такой, что $\{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\} \subseteq A_j(\rho)$.

Следствие 4. Из утверждений 14 и 16 получаем, что если $\rho \in LA[n]$ и $|A_j(\rho)| \leqslant 2$ для $j = 1, \ldots, n$, то $H(\rho) = \{e\}$, где e — тождественная подстановка.

Используя следствие 4, а также утверждения 4, 9 и 13, получаем

Утверждение 17. Пусть $n \geqslant 2$, $\lambda \in LO[n]$. Тогда $H(\lambda) = \{e\}$, $\Pi\lambda = \{\pi\lambda : \pi \in S_n\} = LO[n]$.

Теперь можем показать,что справедлива

Теорема 3. $LA^{(2)}[n] = \{\rho \in LA[n] \setminus LO[n] : \min\limits_{\lambda \in LO[n]} d(\lambda, \rho) = 2\} = \bigcup\limits_{\rho \in \Lambda^{(n)}} \Pi\rho$ и для всех $\rho_1, \rho_2 \in \Lambda^{(n)} = \{\delta_{l,k}\lambda_1^{(n)} : l = 1, \ldots, n-2, k = l+2, \ldots, n\}$ если $\rho_1 \neq \rho_2$, то $\Pi\rho_1 \cap \Pi\rho_2 = \varnothing$.

Доказательство.

1) Докажем, что $\bigcup\limits_{\rho \in \Lambda^{(n)}} \Pi\rho \subseteq LA^{(2)}[n]$. Пусть $\rho \in \bigcup\limits_{\rho' \in \Lambda^{(n)}} \Pi\rho'$. Тогда $\rho = \pi\rho'$ для некоторых $\rho' \in \Lambda^{(n)}$, $\pi \in S_n$. По определению $\Lambda^{(n)}$ имеем $\rho' = \delta_{l.k}\lambda_1^{(n)}$ для некоторых $l \in \{1, \ldots, n-2\}$, $k \in \{l+2, \ldots, n\}$. Но тогда $d(\rho', \lambda_1^{(n)}) = 2$, и в силу утверждений 9, 12 и 17 получаем $\rho = \pi\rho' \in LA[n] \setminus LO[n]$, $\pi\lambda_1^{(n)} \in LO[n]$, $d(\rho, \pi\lambda_1^{(n)}) = d(\pi\rho', \pi\lambda_1^{(n)}) = d(\rho', \lambda_1^{(n)}) = 2$, т. е. $\rho \in LA^{(2)}[n]$.

2) Докажем, что $LA^{(2)}[n] \subseteq \bigcup\limits_{\rho \in \Lambda^{(n)}} \Pi\rho$. Пусть $\rho \in LA^{(2)}[n]$. Тогда $\rho \in LA[n] \setminus LO[n]$ и найдётся $\lambda \in LO[n]$, такое, что $d(\lambda, \rho) = 2$. Из утверждения 17 получаем, что существует $\pi \in S_n$, для которой $\pi\lambda = \lambda_1^{(n)}$, откуда по утверждению 12 следует $d(\pi\rho, \lambda_1^{(n)}) = d(\pi\rho, \pi\lambda) = d(\rho, \lambda) = 2$, а значит (см.утверждение 11), $\pi\rho = \rho' \in \Lambda^{(n)}$, $\rho = \pi^{-1}\rho' \in \Pi\rho'$ и $\rho \in \bigcup\limits_{\rho' \in \Lambda^{(n)}} \Pi\rho'$. Таким образом, равенство $LA^{(2)}[n] = \bigcup\limits_{\rho \in \Lambda^{(n)}} \Pi\rho$ установлено и для завершения доказательства осталось воспользоваться равенством (5) и утверждением 9. ■

3. Система представителей классов π-эквивалентности из $LA[n]$

В п. 2 выделена система представителей классов π-эквивалентности из $LA^{(2)}[n]$, т. е. множество $\Lambda^{(n)} = \{\delta_{l,k}\lambda_1^{(n)} : l = 1, \ldots, n-2, \ k = l+2, \ldots, n\}$, состоящее из

$(n-1)(n-2)/2$ элементов. Они конструктивным образом определяются и их сравнительно немного. Более того, далее приведено утверждение (лемма 1 «о редукции»), в силу которого при каждом новом n существует единственный класс π-эквивалентности из $LA^{(2)}[n]$, несводимый к соответствующему (редуцированному) классу эквивалентности из $LA^{(2)}[n-1]$. Представителем нового класса является $\rho^{(n)}_{1,n} = \delta_{1,n}\lambda^{(n)}_1$, где $I(\rho^{(n)}_{1,n}) = \langle 2,2,3,\ldots,n-2,n-1,n-1\rangle$, $\bar{I}(\rho^{(n)}_{1,n}) = \langle 1,0,\ldots,0,-1\rangle \in \mathbb{Z}^n$. При этом при $n \geqslant 5$ справедливо $\Pi\rho^{(n)}_{1,n} \neq LA_{I(\rho^{(n)}_{1,n})}[n]$, т. е. выполнение для некоторого $\rho \in LA[n]$ условия $\bar{I}(\rho) = \langle 1,0,\ldots,0,-1\rangle$ не гарантирует выполнение $\rho \in LA^{(2)}[n]$, но требует дополнительной проверки (например, с помощью алгоритма 3).

При небольших $n \geqslant 2$ можно найти систему всех представителей классов π-эквивалентности для всего множества $LA[n]$. Оценим их количество K_n снизу. Поскольку в каждом классе эквивалентности не более $n!$ элементов, количество классов эквивалентности больше или равно $|LA[n]|/n! = 2^{n(n-1)/2}/n!$. Как будет видно из дальнейшего, $K_2 = 1$, $K_3 = 2$, $K_4 = 4$, $K_5 = 12$ и, исходя из приведённой нижней оценки, $K_6 \geqslant 2^{15}/720 \simeq 46$, $K_7 \geqslant 2^{21}/5040 \simeq 416$, $K_8 \geqslant 2^{28}/40320 \simeq 53261$ и т. д. С учётом редукции (см. лемму 1) для каждого очередного n «новых» представителей (несводимых к представителям меньшей размерности) будет меньше. Например, при $n = 5$ из 12 представителей «новых» только 6.

Таким образом, системой представителей классов π-эквивалентности для множества $LA[n]$ является конечный набор $\rho_i{}^{(n)} \in LA[n], i = 1,\ldots,K_n$, такой, что

$$\bigcup_{i=1}^{K_n} \Pi\rho^{(n)}_i = LA[n]; \quad i \neq j \Rightarrow \Pi\rho^{(n)}_i \cap \Pi\rho^{(n)}_j = \varnothing. \tag{6}$$

Одним из таких представителей будет любой линейный порядок из $LO[n]$, например $\lambda^{(n)}_1$, при этом $\Pi\lambda^{(n)}_1 = LO[n]$ (см. утверждение 17). Другие примеры представителей — $\rho \in \Lambda^{(n)} = \{\delta_{l,k}\lambda^{(n)}_1 : l = 1,\ldots,n-2,\ k = l+2,\ldots,n\}$. Интересен ответ на вопрос, какую долю составляет $LA^{(2)}[n] = \bigcup\limits_{\rho\in\Lambda^{(n)}} \Pi\rho$ от всего множества $LA[n]$. Далее получены системы представителей для $LA[n]$ при $n = 2,3,4,5$, а также описан подход к получению этих систем, восходящий от достигнутого n к $n+1$.

Прежде чем сформулировать необходимую для дальнейшего лемму 1, рассмотрим следующий пример:

Пример 5. Пусть $\rho_1, \rho_2, \rho_3 \in LA[5]$ заданы матрицами смежности

$$R(\rho_1) = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & 1 & 1 & 0 & 1 \\ \mathbf{0} & 0 & 1 & 1 & 0 \\ \mathbf{0} & 1 & 0 & 1 & 1 \\ \mathbf{0} & 0 & 1 & 0 & 1 \end{bmatrix},\ R(\rho_2) = \begin{bmatrix} 1 & 1 & 0 & 0 & \mathbf{1} \\ 0 & 1 & 1 & 0 & \mathbf{1} \\ 1 & 0 & 1 & 1 & \mathbf{1} \\ 1 & 1 & 0 & 1 & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix},\ R(\rho_3) = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & 1 & 1 & 0 & \mathbf{1} \\ \mathbf{0} & 0 & 1 & 1 & \mathbf{1} \\ \mathbf{0} & 1 & 0 & 1 & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}.$$

Тогда $I(\rho_1) = \langle 2,2,3,3,5\rangle$, при этом подматрица матрицы $R(\rho_1)$, полученная вычёркиванием первой строки и первого столбца, соответствует бинарному отношению $\rho'_1 \in LA[4]$, где $I(\rho'_1) = \langle 2,2,3,3\rangle$, которое естественно назвать отношением, *редуцированным* из бинарного отношения ρ_1:

$$R(\rho'_1) = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

Это связано с тем, что последней компонентой в $I(\rho_1) = \langle 2,2,3,3,5\rangle$ является $n=5$, следовательно, в матрице $R(\rho_1)$ присутствует строка из единиц и симметричный ей столбец из нулей (за исключением диагонального элемента). В общем случае это i-я строка и i-й столбец, где $i \in \{1,\ldots,5\}$, вычеркивая которые, получаем подматрицу, соответствующую редуцированному из исходного отношения из $LA[5]$ отношению из $LA[4]$. В случае произвольного $n \geqslant 2$ подобная редукция возможна для любого $\rho \in LA[n]$ при $I(\rho) = \langle i_1,\ldots,i_{n-1},n\rangle$. При этом для $\rho' \in LA[n-1]$, редуцированного из ρ, выполняется $I(\rho') = \langle i_1,\ldots,i_{n-1}\rangle$.

Аналогичная ситуация возникает, когда $I(\rho) = \langle 1,i_2,\ldots,i_n\rangle$. Этот случай иллюстрируют ρ_2 и ρ_2':

$$R(\rho_2') = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}.$$

Редуцированное бинарное отношение получаем вычеркиванием столбца из единиц (он может быть только единственным, см. замечание 5) и симметричной ему строки из нулей (за исключением диагонального элемента). Для отношения $\rho' \in LA[n-1]$, редуцированного из отношения $\rho \in LA[n]$, выполняется $I(\rho') = \langle i_2-1,\ldots,i_n-1\rangle$. Например, $I(\rho_2) = \langle 1,3,3,4,4\rangle$, $I(\rho_2') = \langle 2,2,3,3\rangle$.

Наконец, в случае, когда $I(\rho) = \langle 1,i_2,\ldots,i_{n-1},n\rangle$, возможна редукция размерности сразу на 2. Для отношения $\rho' \in LA[n-2]$, редуцированного из отношения $\rho \in LA[n]$, выполняется $I(\rho') = \langle i_2-1,\ldots,i_{n-1}-1\rangle$. Например, $I(\rho_3) = \langle 1,3,3,3,5\rangle$, $I(\rho_3') = \langle 2,2,2\rangle$:

$$R(\rho_3') = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}.$$

В общем случае справедлива

Лемма 1 (о редукции). Пусть $n \geqslant 3$, $\rho \in LA[n]$, $k,l \in \{0,1,\ldots,n-2\}$, $k+l \leqslant n-3$, $I(\rho) = \langle 1,2,\ldots,k,i_{k+1},\ldots,i_{n-l-1},i_{n-l},n-l+1,\ldots,n\rangle$, $i_{k+1} \geqslant k+2$, $i_{n-l} \leqslant n-l-1$. Пусть, далее, $A_k(\rho) = \{a_{v_1}\}$, …, $A_1(\rho) = \{a_{v_k}\}$, $A_n(\rho) = \{a_{j_1}\}$, …, $A_{n-l+1}(\rho) = \{a_{j_l}\}$, $A' = A \setminus \{a_{v_1},\ldots,a_{v_k},a_{j_1},\ldots,a_{j_l}\}$, $\rho' = \rho \cap (A')^2$. Тогда $\rho' \in 2^{(A')^2}$, $\rho' \in LA[n-k-l]$, $I(\rho') = \langle i_{k+1}-k,\ldots,i_{n-l}-k\rangle$ и матрица $R(\rho')$ является результатом вычёркивания из $R(\rho)$ строк и столбцов с номерами $v_1,\ldots,v_k,j_1,\ldots,j_l$ (предполагается, что оставшиеся элементы в A' расположены в том же порядке, что и в A).

Как и ранее, отношение $\rho' \in LA[n-k-l]$ из леммы 1 будем называть *редуцированным* из отношения $\rho \in LA[n]$.

Замечание 8. Заметим, что в условиях леммы 1, если $A' = \{a_{\iota_1},\ldots,a_{\iota_{n-k-l}}\} = A \setminus \{a_{v_1},\ldots,a_{v_k},a_{j_1},\ldots,a_{j_l}\}$ и λ' — линейный порядок на A' вида $a_{\iota_1} \prec \ldots \prec a_{\iota_{n-k-l}}$, то для линейного порядка

$$\lambda : a_{j_1} \prec \ldots \prec a_{j_l} \prec \underbrace{a_{\iota_1} \prec \ldots \prec a_{\iota_{n-k-l}}}_{\lambda'} \prec a_{v_1} \prec \ldots \prec a_{v_k}$$

выполняется $d(\lambda,\rho) = d(\lambda',\rho')$, т. е. множество $\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda,\rho)$ легко определяется по $\operatorname{Arg}\min\limits_{\lambda' \in LO[n-k-l]} d(\lambda',\rho')$ и количества элементов в этих множествах совпадают. Действительно, элементы матриц $R(\lambda), R(\rho)$ могут отличаться только на множествах своих

элементов, находящихся на пересечении строк и столбцов, соответствующих A', т. е. на элементах матриц $R(\lambda'), R(\rho')$. Кроме того, нетрудно показать [13], что, в условиях леммы 1, $H(\rho) = H(\rho')$, $|\Pi\rho| = n!/|H(\rho)| = n!/|H(\rho')|$.

Используем эти результаты для рекурсивного нахождения системы представителей $\rho_i^{(n)} \in LA[n]$, $i = 1, \ldots, K_n$, классов π-эквивалентности для множества $LA[n]$, из систем представителей при меньших значениях n: последовательно получим системы представителей, начиная с минимально возможного n.

С л у ч а й $n = 2$: $|LA[2]| = 2^{n(n-1)/2} = 2^1 = 2$, $|LO[2]| = 2! = 2$, т. е. $LA[2] = LO[2]$, а следовательно, представителем единственного класса π-эквивалентности для множества $LA[2]$ является линейный порядок $\lambda_1^{(2)}$, соответствующий последовательности $a_1 \prec a_2$ (очевидно, что для второго линейного порядка $\lambda_2^{(2)}$, соответствующего последовательности $a_2 \prec a_1$, выполняется $\lambda_2^{(2)} = (1,2)\lambda_1^{(2)}$).

С л у ч а й $n = 3$: $|LA[3]| = 2^{n(n-1)/2} = 2^3 = 8$, $LO[3] = 3! = 6$. Единственным представителем класса π-эквивалентности $LO[3]$ является линейный порядок $\lambda_1^{(3)}$, соответствующий последовательности $a_1 \prec a_2 \prec a_3$. Для оставшихся отношений $\rho_1^{(3)}$, $\rho_2^{(3)}$ приведём матрицы $R(\rho_1^{(3)})$, $R(\rho_2^{(3)})$:

$$R(\rho_1^{(3)}) = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}, R(\rho_2^{(3)}) = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}.$$

Заметим, что $I(\rho_1^{(3)}) = I(\rho_2^{(3)}) = \langle 2,2,2\rangle$, $\overline{I}(\rho_1^{(3)}) = \overline{I}(\rho_2^{(3)}) = \langle 1,0,-1\rangle$, $\rho_2^{(3)} = (1\ 2)(\rho_1^{(3)})$ (т. е. $\rho_2^{(3)} \in \Pi\rho_1^{(3)}$), при этом $\rho_1^{(3)} = \delta_{1,3}\lambda_1^{(3)}$ является единственным элементом множества $\Lambda^{(n)} = \left\{\delta_{l,k}\lambda_1^{(n)} : l = 1, \ldots, n-2,\ k = l+2, \ldots, n\right\} = \Lambda^{(3)} = \{\delta_{1,3}\lambda_1^{(3)}\}$. Заметим также: $(1,2,3)\rho_1^{(3)} = \rho_1^{(3)}$, а следовательно, $H(\rho_1^{(3)}) = \langle (1,2,3)\rangle = \{e, (1,2,3), (1,3,2)\}$, $|H(\rho_1^{(3)})| = 3$, $|\Pi\rho_1^{(3)}| = 2 = 3!/|H(\rho_1^{(3)})| = 6/3$ (в силу утверждения 14, разложение любой подстановки из $H(\rho_1^{(3)})$ в произведение независимых циклов может содержать единственный цикл длины 3). Таким образом, имеем систему из двух представителей классов π-эквивалентности для $LA[3]$: $\lambda_1^{(3)}, \rho_1^{(3)}$, при этом $LA[3] = \Pi\lambda_1^{(3)} \cup \Pi\rho_1^{(3)}$, $\Pi\lambda_1^{(3)} = LO[3]$, $\Pi\rho_1^{(3)} = LA[3] \setminus LO[3] = LA^{(2)}[3]$, $|\Pi\lambda_1^{(3)}| = 6$, $|\Pi\rho_1^{(3)}| = 2$.

С л у ч а й $n = 4$: $|LA[4]| = 2^{n(n-1)/2} = 2^6 = 64$, $|LO[4]| = 4! = 24$, $|LA[4] \setminus LO[4]| = 64 - 24 = 40$. Единственным представителем класса π-эквивалентности $LO[4]$ является линейный порядок $\lambda_1^{(4)}$, соответствующий последовательности $a_1 \prec a_2 \prec a_3 \prec a_4$. Для перечисления представителей классов π-эквивалентности из $LA^{(2)}[4] \subseteq LA[4] \setminus LO[4]$ воспользуемся множеством $\Lambda^{(4)} = \{\rho_{1,3}^{(4)} = \delta_{1,3}\lambda_1^{(4)}, \rho_{1,4}^{(4)} = \delta_{1,4}\lambda_1^{(4)}, \rho_{2,4}^{(4)} = \delta_{2,4}\lambda_1^{(4)}\}$:

$$R(\rho_{1,3}^{(4)}) = \begin{bmatrix} 1 & 1 & 0 & \mathbf{1} \\ 0 & 1 & 1 & \mathbf{1} \\ 1 & 0 & 1 & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}, R(\rho_{1,4}^{(4)}) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}, R(\rho_{2,4}^{(4)}) = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & 1 & 1 & 0 \\ \mathbf{0} & 0 & 1 & 1 \\ \mathbf{0} & 1 & 0 & 1 \end{bmatrix}.$$

При этом $I(\rho_{1,3}^{(4)}) = \langle 1,3,3,3\rangle$, $\bar{I}(\rho_{1,3}^{(4)}) = \langle 0,1,0,-1\rangle$, $I(\rho_{1,4}^{(4)}) = \langle 2,2,3,3\rangle$, $\bar{I}(\rho_{1,4}^{(4)}) = \langle 1,0,0,-1\rangle$, $I(\rho_{2,4}^{(4)}) = \langle 2,2,2,4\rangle$, $\bar{I}(\rho_{2,4}^{(4)}) = \langle 1,0,-1,0\rangle$. Поскольку $I(\rho_{1,3}^{(4)}) = \langle 1,3,3,3\rangle$, выполнены условия леммы 1. Редуцированным отношением для $\rho_{1,3}^{(4)}$ является отношение $\rho_1^{(3)}$ (после вычёркивания в $R(\rho_{1,3}^{(4)})$ столбца с единицами и симметричной ему строки получаем $R(\rho_1^{(3)})$), а следовательно, в силу утверждения 13 и замечания 8,

$|H(\rho^{(4)}_{1,3})| = |H(\rho^{(3)}_1)| = 3$, $|\Pi\rho^{(4)}_{1,3}| = 4!/|H(\rho^{(4)}_{1,3})| = 24/3 = 8$. Аналогично, редуцированным отношением для $\rho^{(4)}_{2,4}$ является $\rho^{(3)}_1$, откуда $|H(\rho^{(4)}_{2,4})| = |H(\rho^{(3)}_1)| = 3$, $|\Pi\rho^{(4)}_{2,4}| = 8$. Поскольку $I(\rho^{(4)}_{1,4}) = \langle 2,2,3,3\rangle$, то, в силу следствия 4, $|H(\rho^{(4)}_{1,4})| = \{e\} \Rightarrow |\Pi\rho^{(4)}_{1,4}| = 24$. Таким образом, $|\Pi\rho^{(4)}_{1,3}|+|\Pi\rho^{(4)}_{1,4}|+|\Pi\rho^{(4)}_{2,4}| = 8+24+8 = 40 = |LA[4]\setminus LO[4]|$, а следовательно, $LA[4]\setminus LO[4] = \Pi\rho^{(4)}_{1,3}\cup\Pi\rho^{(4)}_{1,4}\cup\Pi\rho^{(4)}_{2,4} = LA^{(2)}[4]$, $\Pi\rho^{(4)}_{1,3} = LA_{\langle 1,3,3,3\rangle}[4]$, $\Pi\rho^{(4)}_{1,4} = LA_{\langle 2,2,3,3\rangle}[4]$, $\Pi\rho^{(4)}_{2,4} = LA_{\langle 2,2,2,4\rangle}[4]$, т. е. имеем систему из четырёх представителей классов π-эквивалентности для $LA[4]$: $\lambda^{(4)}_1 \in LO[4]$, $\rho^{(4)}_{1,3} \in \Lambda^{(4)}$, $\rho^{(4)}_{1,4} \in \Lambda^{(4)}$, $\rho^{(4)}_{2,4} \in \Lambda^{(4)}$. Тогда в силу теоремы 3 справедливо $LA[4]\setminus LO[4] = LA^{(2)}[4]$, а следовательно, для любого $\rho \in LA[4]$ либо $\rho \in LO[4]$, либо существует $\lambda \in LO[4]$, такое, что $d(\lambda,\rho) = 2$. Как увидим далее, при $n \geqslant 5$ подобное свойство не выполняется.

С л у ч а й $n = 5$: $|LA[5]| = 2^{n(n-1)/2} = 2^{10} = 1024$, $|LO[5]| = 5! = 120$, $|LA[5]\setminus LO[5]| = 1024 - 120 = 904$. Единственным представителем класса π-эквивалентности $LO[5]$ является, например, линейный порядок $\lambda^{(5)}_1 : a_1 \prec a_2 \prec a_3 \prec a_4 \prec a_5$. Для перечисления представителей классов π-эквивалентности из $LA^{(2)}[5] \subseteq LA[5]\setminus LO[5]$ воспользуемся множеством $\Lambda^{(5)} = \{\rho^{(5)}_{1,3} = \delta_{1,3}\lambda^{(5)}_1, \rho^{(5)}_{1,4} = \delta_{1,4}\lambda^{(5)}_1, \rho^{(5)}_{1,5} = \delta_{1,5}\lambda^{(5)}_1, \rho^{(5)}_{2,4} = \delta_{2,4}\lambda^{(5)}_1, \rho^{(5)}_{2,5} = \delta_{2,5}\lambda^{(5)}_1, \rho^{(5)}_{3,5} = \delta_{3,5}\lambda^{(5)}_1\}$, где

$$
\begin{gathered}
I(\rho^{(5)}_{1,3}) = \langle 1,2,4,4,4\rangle, \quad \bar{I}(\rho^{(5)}_{1,3}) = \langle 0,0,1,0,-1\rangle, \quad I(\rho^{(5)}_{1,4}) = \langle\ 1,3,3,4,4\rangle,\\
\bar{I}(\rho^{(5)}_{1,4}) = \langle 0,1,0,0,-1\rangle, \quad I(\rho^{(5)}_{1,5}) = \langle 2,2,3,4,4\rangle, \quad \bar{I}(\rho^{(5)}_{1,5}) = \langle 1,0,0,0,-1\rangle,\\
I(\rho^{(5)}_{2,4}) = \langle 1,3,3,3,5\rangle, \quad \bar{I}(\rho^{(5)}_{2,4}) = \langle 0,1,0,-1,0\rangle, \quad I(\rho^{(5)}_{2,5}) = \langle 2,2,3,3,5\rangle,\\
\bar{I}(\rho^{(5)}_{2,5}) = \langle 1,0,0,-1,0\rangle, \quad I(\rho^{(5)}_{3,5}) = \langle 2,2,2,4,5\rangle, \quad \bar{I}(\rho^{(5)}_{3,5}) = \langle 1,0,-1,0,0\rangle.
\end{gathered}
$$

Приведём матрицы, соответствующие некоторым отношениям:

$$
R(\rho^{(5)}_{1,3}) = \begin{bmatrix} 1 & 1 & 0 & \mathbf{1} & \mathbf{1}\\ 0 & 1 & 1 & \mathbf{1} & \mathbf{1}\\ 1 & 0 & 1 & \mathbf{1} & \mathbf{1}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1}\end{bmatrix}, \quad R(\rho^{(5)}_{2,4}) = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1}\\ \mathbf{0} & 1 & 1 & 0 & \mathbf{1}\\ \mathbf{0} & 0 & 1 & 1 & \mathbf{1}\\ \mathbf{0} & 1 & 0 & 1 & \mathbf{1}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1}\end{bmatrix}, \quad R(\rho^{(5)}_{3,5}) = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1}\\ \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1}\\ \mathbf{0} & \mathbf{0} & 1 & 1 & 0\\ \mathbf{0} & \mathbf{0} & 0 & 1 & 1\\ \mathbf{0} & \mathbf{0} & 1 & 0 & 1\end{bmatrix}.
$$

Поскольку $I(\rho^{(5)}_{1,3}) = \langle 1,2,4,4,4\rangle$, выполнены условия леммы 1. Редуцированным отношением для $\rho^{(5)}_{1,3}$ является $\rho^{(3)}_1$, а следовательно, в силу утверждения 13 и замечания 8, $|H(\rho^{(5)}_{1,3})| = |H(\rho^{(3)}_1)| = 3$, $|\Pi\rho^{(5)}_{1,3}| = 5!/3 = 40$. Аналогично, редуцированным отношением для $\rho^{(5)}_{2,4}$, $\rho^{(5)}_{3,5}$ также является $\rho^{(3)}_1$, а следовательно, $|\Pi\rho^{(5)}_{2,4}| = |\Pi\rho^{(5)}_{3,5}| = 5!/3 = 40$.

Поскольку $I(\rho^{(5)}_{1,4}) = \langle 1,3,3,4,4\rangle$, $I(\rho^{(5)}_{1,5}) = \langle 2,2,3,4,4\rangle$, $I(\rho^{(5)}_{2,5}) = \langle 2,2,3,3,5\rangle$, в силу следствия 4 получаем $H(\rho^{(5)}_{1,4}) = H(\rho^{(5)}_{1,5}) = H(\rho^{(5)}_{2,5}) = \{e\}$, а следовательно (см. утверждение 13), $|H(\rho^{(5)}_{1,4}) = |H(\rho^{(5)}_{1,5})| = |H(\rho^{(5)}_{2,5})| = 1$, $|\Pi\rho^{(5)}_{1,4}| = |\Pi\rho^{(5)}_{1,5}| = |\Pi\rho^{(5)}_{2,5}| = 5! = 120$.

Таким образом, $|LA^{(2)}[n]| = 120{\cdot}3{+}40{\cdot}3 = 480$. Поскольку $|LO[n]|+|LA^{(2)}[n]| = 120+{}+480 = 600$, остаётся ещё 404 отношения из множества $LA[5]\setminus\{LO[5]\cup LA^{(2)}[5]\}$. Кроме того, ниже показано, что $LA_{\langle 1,2,4,4,4\rangle}[5] = \Pi\rho^{(5)}_{1,3}$, $LA_{\langle 1,3,3,4,4\rangle}[5] = \Pi\rho^{(5)}_{1,4}$, $LA_{\langle 1,3,3,3,5\rangle}[5] = {}= \Pi\rho^{(5)}_{2,4}$, $LA_{\langle 2,2,3,3,5\rangle}[5] = \Pi\rho^{(5)}_{2,5}$, $LA_{\langle 2,2,2,4,5\rangle}[5] = \Pi\rho^{(5)}_{3,5}$ (только $LA_{\langle 2,2,3,4,4\rangle}[5] \neq \Pi\rho^{(5)}_{1,5}$).

Рассмотрим другие возможные случаи для $\rho \in LA[5]$. Для $I(\rho) = \langle 3,3,3,3,3\rangle$ имеем $\overline{I}(\rho) = \langle 2,1,0,-1,-2\rangle$ и, например, $I(\rho_1^{(5)}) = \langle 3,3,3,3,3\rangle$, где

$$R(\rho_1^{(5)}) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Нетрудно доказать, что $H(\rho_1^{(5)}) = \langle\sigma\rangle = \{e, \sigma, \sigma^2, \sigma^3, \sigma^4\}$, где $\sigma = (1,2,3,4,5)$, откуда (см. утверждение 13) $|\Pi\rho_1^{(5)}| = 5!/5 = 24$. Кроме того, далее показано, что $LA_{\langle 3,3,3,3,3\rangle}[5] = \Pi\rho_1^{(5)}$, т. е. $\rho_1^{(5)}$ может служить единственным представителем класса $LA_{\langle 3,3,3,3,3\rangle}[5]$.

Рассмотрим случай $I(\rho) = \langle 2,3,3,3,4\rangle$, $\overline{I}(\rho) = \langle 1,1,0,-1,-1\rangle$. Пусть $i,j \in \{1,2,3,4,5\}$, $A_2(\rho) = \{a_i\}$, $\langle a_i, a_j\rangle \in \rho$, $i \neq j$. Если $\rho' = (j,4)(i,5)\rho$, то $R(\rho')$ имеет следующий вид (элементы матрицы $R(\rho')$, соответствующие знаку $-$, определяются далее с учётом того, что $\rho' \in LA_{\langle 2,3,3,3,4\rangle}[5]$):

$$R(\rho') = \begin{bmatrix} 1 & - & - & - & 1 \\ - & 1 & - & - & 1 \\ - & - & 1 & - & 1 \\ - & - & - & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

Три возможных варианта для $R(\rho')$ в случае $\rho' \in LA_{\langle 2,3,3,3,4\rangle}[5]$ следующие:

$$R(\rho_2^{(5)}) = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \quad R(\rho_3^{(5)}) = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}, \quad R(\rho_4^{(5)}) = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

Нетрудно показать [13], что $\Pi\rho_2^{(5)} \cap \Pi\rho_3^{(5)} = \varnothing$, $\Pi\rho_2^{(5)} \cap \Pi\rho_4^{(5)} = \varnothing$, $\Pi\rho_3^{(5)} \cap \Pi\rho_4^{(5)} = \varnothing$; далее установим, что $LA_{\langle 2,3,3,3,4\rangle}[5] = \Pi\rho_2^{(5)} \cup \Pi\rho_3^{(5)} \cup \Pi\rho_4^{(5)}$.

Заметим, что если для некоторого $\pi \in S_5$ выполняется $\pi\rho_2^{(5)} = \rho_2^{(5)}$, то, в силу утверждения 15, $\pi(4) = 4$, $\pi(5) = 5$, т. е. π действует на множестве $\{1,2,3\}$, откуда $\pi \in S_3$, и из матрицы $R(\rho_2^{(5)})$ видно, что $\pi\rho_2^{(5)} = \rho_2^{(5)} \Leftrightarrow \pi\rho_2^{(3)} = \rho_2^{(3)}$ (см. случай $n = 3$), т. е. $H(\rho_2^{(5)}) = H(\rho_2^{(3)}) = \langle(1,2,3)\rangle$, следовательно, $|\Pi\rho_2^{(5)}| = 5!/3 = 40$.

Если для некоторого $\pi \in S_5$ выполняется $\pi\rho_3^{(5)} = \rho_3^{(5)}$, то, в силу утверждения 15, $\pi(1) = 1$, $\pi(5) = 5$ (поскольку первая строка матрицы $R(\rho_3^{(5)})$ — единственная с четырьмя единицами, а пятая — единственная с двумя единицами). Далее из первой строки матрицы $R(\rho_3^{(5)})$ заключаем, что $\pi(3) \neq 2$, $\pi(4) \neq 2$, откуда $\pi(3), \pi(4) \in \{3,4\}$, и поскольку случай $\pi = (3,4)$ невозможен (см. утверждение 14), то $\pi = e$. Таким образом, $H(\rho_3^{(5)}) = \{e\}$, а следовательно, $|\Pi\rho_3^{(5)}| = 5!/1 = 120$. Аналогично доказывается, что $H(\rho_4^{(5)}) = \{e\}$, $|\Pi\rho_4^{(5)}| = 5!/1 = 120$.

Рассмотрим теперь случай $I(\rho) = \langle 2,2,3,4,4\rangle$, $\overline{I}(\rho) = \langle 1,0,0,0,-1\rangle$. Заметим, что $\rho_{1,5}^{(5)} \in LA_{\langle 2,2,3,4,4\rangle}[5]$, т. е. один из возможных представителей классов π-эквивалентности из $LA_{\langle 2,2,3,4,4\rangle}[5]$ уже найден. Покажем, что он не является единственным,

т. е. что $\Pi\rho^{(5)}_{1,5} \neq LA_{\langle 2,2,3,4,4\rangle}[5]$. Мы уже сталкивались с подобной ситуацией в случае $LA_{\langle 2,3,3,3,4\rangle}[5]$, где оказалось три класса π-эквивалентности. Рассмотрим бинарное отношение $\rho^{(5)}_5$ с $R(\rho^{(5)}_5)$:

$$R(\rho^{(5)}_5) = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Тогда $\rho^{(5)}_5 \in LA_{\langle 2,2,3,4,4\rangle}[5]$. Нетрудно показать [13], что не существует $\lambda \in LO[5]$, такого, что $d(\rho^{(5)}_5, \lambda) = 2$, и тем самым $\rho^{(5)}_5 \notin \Pi\rho^{(5)}_{1,5} \subset LA^{(2)}[5]$. В силу следствия 4, $H(\rho^{(5)}_5) = \{e\}$, а следовательно, $|\Pi\rho^{(5)}_4| = 5!/1 = 120$.

Перечислим всех найденных представителей классов π-эквивалентности для $LA[5]$:

1) $\lambda^{(5)}_1 \in LO[5]$, $\Pi\lambda^{(5)}_1 = LO[5]$, $|\Pi\lambda^{(5)}_1| = |LO[5]| = 5! = 120$;
2) $\rho^{(5)}_{1,3} = \delta_{1,3}\lambda^{(5)}_1 \in \Lambda^{(5)}$, $I(\rho^{(5)}_{1,3}) = \langle 1,2,4,4,4\rangle$, $\Pi\rho^{(5)}_{1,3} \subseteq LA_{\langle 1,2,4,4,4\rangle}[5]$, $|\Pi\rho^{(5)}_{1,3}| = 40$;
3) $\rho^{(5)}_{1,4} = \delta_{1,4}\lambda^{(5)}_1 \in \Lambda^{(5)}$, $I(\rho^{(5)}_{1,4}) = \langle 1,3,3,4,4\rangle$, $\Pi\rho^{(5)}_{1,4} \subseteq LA_{\langle 1,3,3,4,4\rangle}[5]$, $|\Pi\rho^{(5)}_{1,4}| = 120$;
4) $\rho^{(5)}_{1,5} = \delta_{1,5}\lambda^{(5)}_1 \in \Lambda^{(5)}$, $I(\rho^{(5)}_{1,5}) = \langle 2,2,3,4,4\rangle$, $\Pi\rho^{(5)}_{1,5} \subset LA_{\langle 2,2,3,4,4\rangle}[5]$, $|\Pi\rho^{(5)}_{1,5}| = 120$;
5) $\rho^{(5)}_{2,4} = \delta_{2,4}\lambda^{(5)}_1 \in \Lambda^{(5)}$, $I(\rho^{(5)}_{2,4}) = \langle 1,3,3,3,5\rangle$, $\Pi\rho^{(5)}_{2,4} \subseteq LA_{\langle 1,3,3,3,5\rangle}[5]$, $|\Pi\rho^{(5)}_{2,4}| = 40$;
6) $\rho^{(5)}_{2,5} = \delta_{2,5}\lambda^{(5)}_1 \in \Lambda^{(5)}$, $I(\rho^{(5)}_{2,5}) = \langle 2,2,3,3,5\rangle$, $\Pi\rho^{(5)}_{2,5} \subseteq LA_{\langle 2,2,3,3,5\rangle}[5]$, $|\Pi\rho^{(5)}_{2,5}| = 120$;
7) $\rho^{(5)}_{3,5} = \delta_{3,5}\lambda^{(5)}_1 \in \Lambda^{(5)}$, $I(\rho^{(5)}_{3,5}) = \langle 2,2,2,4,5\rangle$, $\Pi\rho^{(5)}_{3,5} \subseteq LA_{\langle 2,2,2,4,5\rangle}[5]$, $|\Pi\rho^{(5)}_{3,5}| = 40$;
8) $\rho^{(5)}_1 \in LA_{\langle 3,3,3,3,3\rangle}[5]$, $\overline{I}(\rho^{(5)}_1) = \langle 2,1,0,-1,-2\rangle$, $\Pi\rho^{(5)}_1 \subseteq LA_{\langle 3,3,3,3,3\rangle}[5]$, $|\Pi\rho^{(5)}_1| = 24$;
9) $\rho^{(5)}_2 \in LA_{\langle 2,3,3,3,4\rangle}[5]$, $\overline{I}(\rho^{(5)}_2) = \langle 1,1,0,-1,-1\rangle$, $\Pi\rho^{(5)}_2 \subset LA_{\langle 2,3,3,3,4\rangle}[5]$, $|\Pi\rho^{(5)}_2| = 40$;
10) $\rho^{(5)}_3 \in LA_{\langle 2,3,3,3,4\rangle}[5]$, $\overline{I}(\rho^{(5)}_3) = \langle 1,1,0,-1,-1\rangle$, $\Pi\rho^{(5)}_3 \subset LA_{\langle 2,3,3,3,4\rangle}[5]$, $|\Pi\rho^{(5)}_3| = 120$;
11) $\rho^{(5)}_4 \in LA_{\langle 2,3,3,3,4\rangle}[5]$, $\overline{I}(\rho^{(5)}_4) = \langle 1,1,0,-1,-1\rangle$, $\Pi\rho^{(5)}_4 \subset LA_{\langle 2,3,3,3,4\rangle}[5]$, $|\Pi\rho^{(5)}_4| = 120$;
12) $\rho^{(5)}_5 \in LA_{\langle 2,2,3,4,4\rangle}[5]$, $\overline{I}(\rho^{(5)}_5) = \langle 1,0,0,0,-1\rangle$, $\Pi\rho^{(5)}_5 \subset LA_{\langle 2,2,3,4,4\rangle}[5]$, $|\Pi\rho^{(5)}_5| = 120$.

Таким образом, получены представители классов π-эквивалентности для $LA[5]$, при этом

$$\begin{gathered} |\Pi\lambda^{(5)}_1| + |\Pi\rho^{(5)}_{1,3}| + |\Pi\rho^{(5)}_{1,4}| + |\Pi\rho^{(5)}_{1,5}| + |\Pi\rho^{(5)}_{2,4}| + |\Pi\rho^{(5)}_{2,5}| + |\Pi\rho^{(5)}_{3,5}| + |\Pi\rho^{(5)}_1| + |\Pi\rho^{(5)}_2| + \\ + |\Pi\rho^{(5)}_3| + |\Pi\rho^{(5)}_4| + |\Pi\rho^{(5)}_5| = 120 + 40 + 120 + 120 + 40 + 120 + 40 + \\ + 24 + 40 + 120 + 120 + 120 = 1024 = |LA[5]|, \end{gathered} \tag{7}$$

следовательно, найденная система представителей является полной.

Нам понадобится следующее очевидное

Утверждение 18. Пусть X — конечное непустое множество, $|X| = N \geqslant 2$, $X_1, \ldots, X_k$ — непустые множества, образующие разбиение множества X. Пусть, далее, $Y_1, \ldots, Y_k$ — конечные множества, такие, что $Y_1 \subseteq X_1, \ldots, Y_k \subseteq X_k$. Тогда в случае $\sum\limits_{i=1}^{k} |Y_i| = N$ выполняется: $Y_1 = X_1, \ldots, Y_k = X_k$.

Используя утверждение 18, из (7), учитывая классы эквивалентности 1–12, получаем

$$\Pi\lambda^{(5)}_1 = LO[5],\ \Pi\rho^{(5)}_{1,3} = LA_{\langle 1,2,4,4,4\rangle}[5],\ \Pi\rho^{(5)}_{1,4} = LA_{\langle 1,3,3,4,4\rangle}[5], \Pi\rho^{(5)}_{1,5} = LA_{\langle 2,2,3,4,4\rangle}[5],$$
$$\Pi\rho^{(5)}_{2,4} = LA_{\langle 1,3,3,3,5\rangle}[5],\ \Pi\rho^{(5)}_{2,5} = LA_{\langle 2,2,3,3,5\rangle}[5],\ \Pi\rho^{(5)}_{3,5} = LA_{\langle 2,2,2,4,5\rangle}[5],$$
$$\Pi\rho^{(5)}_1 = LA_{\langle 3,3,3,3,3\rangle}[5],\ \Pi\rho^{(5)}_2 \cup \Pi\rho^{(5)}_3 \cup \Pi\rho^{(5)}_4 = LA_{\langle 2,3,3,3,4\rangle}[5],\ \Pi\rho^{(5)}_{1,5} \cup \Pi\rho^{(5)}_5 = LA_{\langle 2,2,3,4,4\rangle}[5].$$

Результаты для $\rho \in LA[n]$ при $n = 3, 4, 5$ позволяют с помощью леммы 1 сформулировать некоторые свойства для произвольного $n \geqslant 3$. Особенно просто они получаются для $\rho \in LA^{(2)}[n]$. Например, для всех $\rho \in LA^{(2)}[n]$ можно по вектору $\bar{I}(\rho) \in V_{(n)}$ (см. утверждение 10) определить:

1) количество элементов в $\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda, \rho)$, которое равно 1 или 3 (см. утверждение 20);
2) количество элементов в $\Pi\rho$: $|\Pi\rho| = n!/3$ либо $|\Pi\rho| = n!$ (см. утверждение 20);
3) точную формулу для $|LA^{(2)}[n]|$ (см. следствие 5);
4) случаи выполнения равенства $LA_{I(\rho)}[n] = \Pi\rho$ для $\rho \in \Lambda^{(n)}$(см. утверждение 21);
5) множество $\bar{V}_{(n)} \subseteq V_{(n)}$ (масксимально широкое), такое, что при любом $\rho \in LA[n]$ выполняется: $I(\rho) \in \bar{V}_{(n)} \Rightarrow \rho \in LA^{(2)}[n]$ (см. утверждение 22).

Для этого понадобятся следующие утверждения:

Утверждение 19. Пусть $n \geqslant 4$, $\rho^{(n)}_{1,n} = \delta_{1,n}\lambda^{(n)}_1 \in \Lambda^{(n)}$. Тогда

$$I(\rho^{(n)}_{1,n}) = \langle 2, 2, 3, \ldots, n-2, n-1, n-1\rangle,\ \bar{I}(\rho^{(n)}_{1,n}) = \langle 1, 0, \ldots, 0, -1\rangle \in V_n,$$

1) $\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda, \rho^{(n)}_{1,n}) = \{\lambda^{(n)}_1\}$;
2) для всех $\rho \in LA^{(2)}[n]$ в случае $\bar{I}(\rho) = \langle 1, 0, \ldots, 0, -1\rangle \in V_n$ выполняется $|\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda, \rho)| = 1,\ H(\rho) = \{e\}$;
3) в случае $n \geqslant 5$ для $\rho^{(n)}_* = \delta_{3,n}\delta_{1,3}\lambda^{(n)}_1$ верно: $\min\limits_{\lambda \in LO[n]} d(\lambda, \rho^{(n)}_*) = d(\lambda^{(n)}_1, \rho^{(n)}_*) = 4$, $\bar{I}(\rho^{(n)}_*) = \langle 1, 0, \ldots, 0, -1\rangle$, а следовательно, $\rho^{(n)}_* \notin \Pi\rho^{(n)}_{1,n}$, $LA_{I(\rho^{(n)}_{1,n})}[n] \neq \Pi\rho^{(n)}_{1,n}$.

Доказательство.

1) Заметим, что $\min\limits_{\lambda \in LO[n]} d(\lambda, \rho^{(n)}_{1,n}) = 2$, и при $n \geqslant 4$, в отличие от $n = 3$ (см. $\rho^{(3)}_1 = \delta_{1,3}\lambda^{(3)}_1$), имеем единственную возможность для $\delta_{i,j}$, где $i < j$, $i, j \in \{1, \ldots, n\}$, чтобы $\delta_{i,j}\rho^{(n)}_{1,n} \in LO[n]$, а именно: $\delta_{i,j} = \delta_{1,n}$ (например, в случае $\rho^{(4)}_{1,4} = \delta_{1,4}\lambda^{(4)}_1$); другие $\delta_{i,j}$ не приводят $\rho^{(n)}_{1,n}$ к линейному порядку.

2) Следует из п. 1, а также того, что из условий $\rho \in LA^{(2)}[n]$, $\bar{I}(\rho) = \langle 1, 0, \ldots, 0, -1\rangle \in V_n$, в силу теоремы 3, получаем, что найдётся подстановка $\pi \in S_n$, такая, что $\rho = \pi\rho^{(n)}_{1,n}$, откуда $\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda, \rho) = \{\pi\lambda^{(n)}_1\}$ (см. утверждение 12). Равенство $H(\rho) = \{e\}$ получаем из следствия 4.

3) Доказательство несложно провести для $\rho^{(5)}_5$: заметим, что $\rho^{(5)}_5 = \delta_{3,5}\delta_{1,3}\lambda^{(5)}_1$, $\bar{I}(\rho^{(5)}_5) = \langle 1, 0, 0, 0, -1\rangle$. В общем случае рассуждения аналогичны. ■

Утверждение 20. Пусть $n \geqslant 4$, $\rho \in LA^{(2)}[n]$,

$$\bar{I}(\rho) = \langle 0, \ldots, 0, 1, \underbrace{0, \ldots, 0}_{s}, -1, 0, \ldots, 0\rangle \in V_n,\ s \geqslant 1.$$

Тогда

1) если $s = 1$, то $|\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda, \rho)| = 3$, $|\Pi\rho| = n!/3$;
2) если $s \geqslant 2$, то $H(\rho) = \{e\}$, $|\operatorname{Arg}\min\limits_{\lambda \in LO[n]} d(\lambda, \rho)| = 1$, $|\Pi\rho| = n!$.

Доказательство. Для $s = 1$, используя лемму 1, получаем, что редуцированное бинарное отношение принадлежит $LA_{\langle 2,2,2\rangle}[3] = \{\rho^{(3)}_1, (1, 2)\rho^{(3)}_1\}$ (см. случай $n = 3$), а

следовательно, имеет ровно три линейных порядка, удалённых на расстояние 2 от него (см. $R(\rho_1^{(3)})$), что в силу замечания 8 переносится и на само бинарное отношение ρ. Воспользовавшись замечанием 8, а также равенством $|H(\rho_1^{(3)})| = |H((1,2)\rho_1^{(3)})| = 3$, получаем второе равенство в п. 1. В случае $s \geqslant 2$ для редуцированного отношения выполняются условия утверждения 19, из которого следуют равенства в п. 2. ■

Следствие 5. Пусть выполнены условия утверждения 20. Тогда в случае $s = 1$ имеем $n-2$ варианта для $\bar{I}(\rho) = \langle 0,\dots,0,1,\underbrace{0,\dots,0}_{s},-1,0,\dots,0\rangle \in V_n$, в случае $s = 2$ имеем $n-3$ варианта и т. д., в случае $s = n-2$ — один вариант, следовательно, в силу утверждения 20, справедлива формула

$$|LA^{(2)}[n]| = (n-2)n!/3 + [(n-2)(n-3)/2]n! = (n-2)(3n-7)n!/6.$$

Например, при $n = 5$ по этой формуле имеем $LA^{(2)}[5] = 3\cdot 8\cdot 120/6 = 480$, что совпадает с полученным ранее результатом.

Используя лемму 1, свойства отношений из $LA[3], LA[4]$, а также утверждение 19, нетрудно показать [13], что справедливы следующие утверждения:

Утверждение 21. Пусть $n \geqslant 4$, $\rho^* \in \Lambda^{(n)}$,

$$\bar{I}(\rho^*) = \langle 0,\dots,0,1,\underbrace{0,\dots,0}_{s},-1,0,\dots,0\rangle \in V_n,\ s \geqslant 1.$$

Тогда если $s \in \{1,2\}$, то $LA_{I(\rho^*)}[n] = \Pi\rho^*$, а если $s \geqslant 3$, то $LA_{I(\rho^*)}[n] \neq \Pi\rho^*$.

Утверждение 22. Пусть $n \geqslant 4$, $\rho \in LA[n]$, $\bar{I}(\rho) \in V_n$, при этом $\bar{I}(\rho) = \langle 0,\dots,0,1,0,-1,0,\dots,0\rangle$ или $\bar{I}(\rho) = \langle 0,\dots,0,1,0,0,-1,0,\dots,0\rangle$. Тогда $\rho \in LA^{(2)}[n]$.

4. Методы сведения исходной задачи к задаче меньшей размерности

Приведём некоторые методы сведения исходной задачи нахождения $\operatorname{Arg}\min\limits_{\rho\in LO[n]} D(\rho)$ в случае, когда $\rho_1,\dots,\rho_m \in LO[n]$ и m нечётно, к задаче меньшей размерности, основанные на использовании леммы 1, а также на разбиении графа $\tilde{G} = (A,\tilde{\rho})$ на компоненты сильной связности, где $\tilde{\rho}$ — мажоритарное отношение, введённое в п. 1. Для описания метода, основанного на лемме 1, понадобятся следующие обозначения:

$$J_{(n)} = \{\langle i,j\rangle : i,j \in \{1,\dots,n\}, i<j\},$$
$$\forall\langle i,j\rangle \in J_{(n)},\ \rho \in 2^{A^2}\quad D_{ij}(\rho) = \sum_{t=1}^{m}|r_{ij}-r_{ij}^{(t)}| + \sum_{t=1}^{m}|r_{ji}-r_{ji}^{(t)}|.$$

Тогда для всех $\langle i,j\rangle \in J_{(n)}$, $\rho \in LA[n]$, используя то, что $r_{ij}+r_{ji}=1$, $p_{ij}+p_{ji}=m$, получаем следующее равенство [13]:

$$\begin{aligned} D_{ij}(\rho) &= 2\min\{p_{ij}, m-p_{ij}\} + 2|r_{ij}-\tilde{r}_{ij}|[\max\{p_{ij},m-p_{ij}\} - \min\{p_{ij},m-p_{ij}\}] = \\ &= 2\min\{p_{ij},m-p_{ij}\} + 2|r_{ij}-\tilde{r}_{ij}|[m-2\min\{p_{ij},m-p_{ij}\}]. \end{aligned} \tag{8}$$

Для $J \subseteq \{1,\dots,n\}^2$ обозначим $D_J(\rho) = \sum\limits_{\langle i,j\rangle\in J}\sum\limits_{t=1}^{m}|r_{ij}-r_{ij}^{(t)}|$. Пусть выполнены условия леммы 1 и $\rho'_t = \rho_t \cap (A')^2$, $t = 1,\dots,m$, $I_1 = \{v_1,\dots,v_k,j_1,\dots,j_l\}$, $I' = \{\iota_1,\dots,\iota_{n-k-l}\} = I\setminus I_1$, $D'(\rho) = \sum\limits_{t=1}^{m} d(\rho\cap(A')^2, \rho'_t) = D_{(I')^2}(\rho)$. Тогда, учитывая (8),

для всех $\rho \in LO[n]$ имеем

$$D(\rho) = D_{(I')^2}(\rho) + D_{\{1,\dots,n\}^2\setminus(I')^2}(\rho) = D'(\rho) + \sum_{\langle i,j\rangle\in J_{(n)}\setminus(I')^2} 2\min\{p_{ij}, m-p_{ij}\} + \\ + \sum_{\langle i,j\rangle\in J_{(n)}\setminus(I')^2} 2|r_{ij}-\tilde r_{ij}|[\max\{p_{ij}, m-p_{ij}\} - \min\{p_{ij}, m-p_{ij}\}]. \tag{9}$$

Из равенства (9) следует (см. далее замечание 9), что

$$\operatorname{Arg}\min_{\rho\in LO[n]} D(\rho) = \{\rho : \rho\cap(A')^2 \in \operatorname{Arg}\min_{\lambda'\in LO[n-k-l]} D'(\lambda'), \\ r_{ij} = \tilde r_{ij}, \langle\, i,j\rangle \in \{1,\dots,n\}^2\setminus(I')^2\},$$

при этом $\min\limits_{\rho\in LO[n]} D(\rho) = \min\limits_{\lambda'\in LO[n-k-l]} D'(\lambda') + \sum\limits_{\langle i,j\rangle\in J^1_{(n)}} 2\min\{p_{ij}, m-p_{ij}\}$, т. е. исходную задачу свели к задаче меньшей размерности — $(n-k-l)$.

Замечание 9. В выражении в правой части равенства (9) второе слагаемое является постоянной величиной, а минимизацию первого и третьего можно проводить независимо. При этом если минимум первого слагаемого достигается на некотором линейном порядке $\lambda' : a_{\iota_1} \prec \dots \prec a_{\iota_{n-k-l}}$, т. е. при $\rho\cap(A')^2 = \lambda'$, где $A' = \{a_{\iota_1}, \dots, a_{\iota_{n-k-l}}\}$, то выполнение $r_{ij} = \tilde r_{ij}$, $\langle i,j\rangle \in \{1,\dots,n\}^2\setminus(I')^2$, приведёт, с одной стороны, к точному минимуму, равному 0, третьего (неотрицательного) слагаемого в (9), а с другой — к тому, что получаемое таким образом бинарное отношение ρ будет линейным порядком вида $a_{j_1} \prec \dots \prec a_{j_l} \prec \underbrace{a_{\iota_1} \prec \dots \prec a_{\iota_{n-k-l}}}_{\lambda'} \prec a_{v_1} \prec \dots \prec a_{v_k}$.

В другом подходе используется разложение $\tilde G = (A, \tilde\rho)$ на компоненты сильной связности. Этот подход использован в [1] для решения более общей задачи, в условиях которой не гарантировано выполнение $\tilde\rho \in LA[n]$; в нашем случае это условие будет использовано. Под сильной связностью графа понимается взаимная достижимость любых двух его вершин. Под компонентой сильной связности графа понимается его максимальный сильносвязный подграф (т. е. не являющийся собственным подграфом никакого другого сильносвязного подграфа). Задача выделения указанных компонент имеет полиномиальную вычислительную сложность. Пусть $\tilde G_1 = (\tilde A_1, \tilde\rho_1)$, …, $\tilde G_p = (\tilde A_p, \tilde\rho_p)$ — компоненты сильной связности графа $\tilde G = (A, \tilde\rho)$, где $\{\tilde A_1, \dots, \tilde A_p\}$ — разбиение множества A. Рассмотрим для $\tilde G = (A, \tilde\rho)$ «граф конденсации» $G_0 = (A_0, \rho_0)$, вершинами которого являются компоненты $\tilde G_1, \dots, \tilde G_p$ (т. е. $A_0 = \{\tilde G_1, \dots, \tilde G_p\}$) с дугами, определяемыми по следующему правилу: $\langle \tilde G_i, \tilde G_j\rangle \in \rho_0 \Leftrightarrow \exists a' \in \tilde A_i, a'' \in \tilde A_j \big(\langle a', a''\rangle \in \tilde\rho\big)$. Тогда из условия $\tilde\rho \in LA[n]$ следует, что для бинарных отношений $\rho_0 \in 2^{A_0^2}$, $\tilde\rho_1 \in 2^{\tilde A_1^2}$, …, $\tilde\rho_p \in 2^{\tilde A_p^2}$ справедливо: $\rho_0 \in LO[p]$, $\tilde\rho_1 \in LA[n_1]$, …, $\tilde\rho_p \in LA[n_p]$, где $n_1 = |\tilde A_1|$, …, $n_p = |\tilde A_p|$, $n_1 + \dots + n_p = n$. Пусть для простоты обозначений для линейного порядка ρ_0 выполняется $\tilde A_1 \prec \tilde A_2 \prec \dots \prec \tilde A_p$. Тогда для нахождения всех линейных порядков $\lambda \in \operatorname{Arg}\min\limits_{\rho\in LO[n]} D(\rho)$ составляем всевозможные линейные порядки вида $\lambda_1 \prec \lambda_2 \prec \dots \prec \lambda_p$ (это условие с учётом транзитивности однозначно определяет линейный порядок на A), где $\lambda_i \in \operatorname{Arg}\min\limits_{\rho\in LO[n_i]} D_{I_i^2}(\rho)$, $I_i = \{i \in \{1,\dots,n\} : a_i \in \tilde A_i\}$. Таким образом, произведена декомпозиция исходной задачи на p аналогичных задач меньшей размерности. Нетрудно видеть, что метод, основанный на лемме 1, является следствием этого более общего метода.

5. Использование методов теории графов для точного решения задачи

Из п. 4 следует, что для решения поставленной задачи достаточно ограничиться случаем, когда $\tilde{G} = (A, \tilde{\rho})$ — сильносвязный граф. Всюду далее рассматриваем графы без петель.

Приведём сначала общие утверждения относительно произвольного сильносвязного графа $G = (A, \rho)$, где $\rho \in 2^{A^2}$. Пронумеруем все пары (дуги графа G), входящие в ρ, т. е. пусть $\rho = \{x_1, \ldots, x_N\}$. Пусть $\eta_1, \ldots, \eta_k$ — множество всех простых контуров (через каждую вершину в простом контуре проходим ровно один раз) в G, записываемых в виде последовательностей (слов) вида $x_{i_1}x_{i_2}\ldots x_{i_l}$, перечисленных в лексикографическом порядке (в этом порядке их алгоритмически легче перечислить). Если дуга $x \in \rho$ входит в контур η, то кратко пишем $x \in \eta$. Множество дуг $U \subseteq \rho$ назовём *множеством представителей контуров* $\eta_1, \ldots, \eta_k$, если $\forall \eta \in \{\eta_1, \ldots, \eta_k\}\, \exists x \in U\, (x \in \eta)$. Оно называется *минимальным*, если отбрасывание любой дуги из U приводит к множеству дуг, не являющемуся множеством представителей контуров $\eta_1, \ldots, \eta_k$. Одним из возможных методов нахождения семейства минимальных представителей контуров $\eta_1, \ldots, \eta_k$ является известный метод Магу. Пусть U — некоторое множество дуг графа G. Введём булевы переменные X_i, соответствующие дугам $x_i, i = 1, \ldots, N$: $X_i = 1 \Leftrightarrow x_i \in U$. Тогда для того, чтобы U являлось множеством представителей контуров $\eta_1, \ldots, \eta_k$, необходимо и достаточно, чтобы

$$F(X_1, \ldots, X_N) = \bigwedge_{i=1}^{k} \left(\bigvee_{x_j \in \eta_i} X_j \right) = 1. \tag{10}$$

Согласно методу Магу, для нахождения семейства минимальных представителей контуров $\eta_1, \ldots, \eta_k$ приводим $F(X_1, \ldots, X_N)$ к дизъюнктивной нормальной форме (днф), а затем к сокращённой днф $F_1(X_1, \ldots, X_N)$, используя «закон поглощения» $C \vee (C\&D) \equiv C$ (пока это возможно). Тогда каждому дизъюнктивному члену $X_{i_1}\& \ldots \& X_{i_s}$ из $F_1(X_1, \ldots, X_N)$ соответствует $\{x_{i_1}, \ldots, x_{i_s}\}$ — минимальное множество представителей контуров $\eta_1, \ldots, \eta_k$. Действуя таким образом, получаем их все.

Пример 6. Пусть $G = (A, \rho)$, $\rho \in LA[4]$, $R(\rho) = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}$. Тогда (без учёта рефлексивных пар) $\rho \supseteq \{x_1 = \langle a_1, a_2\rangle, x_2 = \langle a_1, a_3\rangle, x_3 = \langle a_2, a_3\rangle, x_4 = \langle a_2, a_4\rangle, x_5 = \langle a_3, a_4\rangle, x_6 = \langle a_4, a_1\rangle\}$. Перечислим контуры в G (в лексикографическом порядке): $\eta_1 = x_1x_3x_5x_6$, $\eta_2 = x_1x_4x_6$, $\eta_3 = x_2x_5x_6$. Составим формулу $F(X_1, \ldots, X_6)$, согласно (10):

$$F(X_1, \ldots, X_6) = (X_1 \vee X_3 \vee X_5 \vee X_6)\&(X_1 \vee X_4 \vee X_6)\&(X_2 \vee X_5 \vee X_6) \equiv$$
$$\equiv X_6 \vee X_1X_2 \vee X_1X_5 \vee X_2X_3X_4 \vee X_4X_5$$

(преобразуем, заменяя & на $\cdot$ и используя равносильность $C \vee (D\&E) \equiv (C \vee D)\&(C \vee E)$). Таким образом, семействами минимальных представителей контуров η_1, η_2, η_3 являются множества дуг $\{x_6\}$, $\{x_1, x_2\}$, $\{x_1, x_5\}$, $\{x_2, x_3, x_4\}$, $\{x_4, x_5\}$.

Утверждение 23. Пусть U — минимальное множество представителей контуров $\eta_1, \ldots, \eta_k$. Тогда для всех $x \in U$ существует контур $\eta(x) \in \eta_1, \ldots, \eta_k$, такой, что x — единственный представитель в $\eta(x)$ из U, т. е. $\forall x' \in U \setminus \{x\}\, \big(x' \notin \eta(x)\big)$.

Доказательство. Если предположить противное, то для некоторого $x \in U$ имеет место: $\forall \eta \in \eta_1, \ldots, \eta_k \ \exists x'(\eta) \in U \setminus \{x\} \big(x'(\eta) \in \eta\big)$, откуда $U \setminus \{x\}$ — множество представителей контуров $\eta_1, \ldots, \eta_k$, а это противоречит минимальности U.■

Утверждение 24. Пусть $x = \langle a_i, a_j \rangle \in \rho$, $a_i \neq a_j$, $U = \{x\}$ — минимальное множество представителей контуров $\eta_1, \ldots, \eta_k$. Тогда граф G', получаемый из G заменой дуги $\langle a_i, a_j \rangle$ на противоположную — $\langle a_j, a_i \rangle$, является бесконтурным.

Доказательство. Пусть для простоты обозначений $x = \langle a_1, a_2 \rangle$. Предположим, что в G' имеется контур η. Не теряя общности, считаем, что он простой. Заметим, что $\langle a_2, a_1 \rangle \in \eta$, так как в противном случае η — контур в G, в который не входит дуга $x = \langle a_1, a_2 \rangle$, что противоречит условиям. Заметим, что η_1 — контур в G, проходящий через $x = \langle a_1, a_2 \rangle$. Заменяя в η_1 дугу $\langle a_1, a_2 \rangle$ на цепь $a_1, a_{i_1}, \ldots, a_{i_k}, a_2$, получаемую из η удалением дуги $\langle a_2, a_1 \rangle$ (очевидно, являющуюся простой цепью в G), получаем контур в G, из которого, в свою очередь, можно выделить простой контур, в который не входит дуга $x = \langle a_1, a_2 \rangle$, что противоречит условиям. ■

Утверждение 25. Пусть $U = \{x_{i_1}, \ldots, x_{i_s}\}$ — минимальное множество представителей контуров $\eta_1, \ldots, \eta_k$. Тогда граф G', получаемый из G заменой всех дуг из U на противоположные, является бесконтурным.

Доказательство. Обозначим для $x_i \in U$ через x'_i дугу, противоположную x_i, $U' = \{x'_{i_1}, \ldots, x'_{i_s}\}$. Предположим, что в G' имеется контур η. Не теряя общности, считаем, что он простой. Заметим, что η содержит по крайней мере одну дугу из U', так как в противном случае η — контур в G, в который не входят дуги из U, что противоречит условиям. Поэтому в η входит по крайней мере одна дуга из U. Случай, когда эта дуга является единственной, рассмотрен при доказательстве утверждения 24. Случай, когда их несколько, рассматривается аналогично с использованием утверждения 23, из которого следует, что мы можем последовательно заменять в η дуги x'_i из U' на соответствующие цепи, получающиеся из контуров $\eta(x_i) \in \{\eta_1, \ldots, \eta_k\}$, в каждый из которых входит дуга x_i и не входят другие дуги из U, и в результате получить контур в G, в который не вошла ни одна дуга из U, т. е. прийти к противоречию с тем, что U — множество представителей всех контуров графа G. ■

Из приведённых рассуждений получаем следующий алгоритм 4.

Алгоритм 4. Нахождение множества $\operatorname{Arg} \min\limits_{\rho \in LO[n]} D(\rho)$

1: Находим все простые контуры $\eta_1, \ldots, \eta_k$ в мажоритарном графе $\tilde{G} = (A, \tilde{\rho})$. Для однозначности перечисляем их в лексикографическом порядке.

2: Определяем $U_1, \ldots, U_T$ — минимальные множества представителей контуров (например, методом Магу).

3: Для каждого $U = \{x_{i_1}, \ldots, x_{i_s}\} \in \{U_1, \ldots, U_T\}$ определяем $c(U) = c(x_{i_1}) + \ldots + c(x_{i_s})$. Замена в $\tilde{G} = (A, \tilde{\rho})$ всех дуг из U на противоположные приводит к бесконтурному графу (см. утверждение 25) $G_U = (A, \rho_U)$ (и к бинарному отношению $\rho_U \in LO[n]$, такому, что $D(\rho_U) = D(\tilde{\rho}) + 2c(U)$ (см. утверждение 5)).

4: Находим $c_{\min} = \min\{c(U_i) : i = 1, \ldots, T\}$. Тогда

$$\operatorname{Arg} \min_{\rho \in LO[n]} D(\rho) = \{\rho_U : U \in \{U_1, \ldots, U_T\}, \ c(U) = c_{\min}\}.$$

Замечание 10. Если $\lambda_1^{(n-1)} \in LO[n-1]$ — линейный порядок вида $a_1 \prec a_2 \prec \ldots \prec \prec a_{n-1}$, то для $\rho = \lambda_1^{(n-1)} \cup \{\langle a_n, a_1\rangle, \langle a_2, a_n\rangle, \langle a_3, a_n\rangle, \ldots, \langle a_n, a_n\rangle\}$ выполняется: $\rho \in \in LA[n]$ и граф $G = (A, \rho)$ содержит все контуры вида

$$\langle a_1, a_{i_1}\rangle, \langle a_{i_1}, a_{i_2}\rangle, \ldots, \langle a_{i_{k-1}}, a_{i_k}\rangle, \langle a_{i_k}, a_n\rangle, \langle a_n, a_1\rangle,$$

где $2 \leqslant i_1 < i_2 < \ldots < i_k \leqslant n-1$. Их столько же, сколько непустых подмножеств множества $\{2, 3, \ldots, n-1\}$, т. е. $2^{n-2}-1$. Кроме того, в сильносвязном графе все дуги входят в контуры. Таким образом, для формулы (10), по которой определяется $F(X_1, \ldots, X_N)$ в методе Магу для примера 4, имеем $k = 2^{n-2} - 1$, $N = (n-1)(n-2)/2$, что приводит к очень громоздкому при больших n выражению. Эта же величина $k = 2^{n-2} - 1$ фигурирует и в алгоритме 4, что говорит об ограниченной применимости рассматриваемого в настоящем пункте метода (в отличие, например, от метода нахождения Δ-оптимального решения при фиксированном Δ в п. 2).

Заключение

Перечислим основные вопросы, затронутые в работе.

1. Исследуется класс бинарных отношений $LA[n]$. К этому классу, в частности, приводят некоторые задачи группового выбора. Изучается задача построения медианы для нечётного числа отношений линейного порядка (т. е. из $LO[n]$, или, в более общем случае, — из $LA[n]$), которая также ищется в классе отношений линейного порядка. Рассматриваются два типа задач: «полная» и «частичная». Решением «полной» задачи является безусловное (т. е. без каких-либо дополнительных ограничений) нахождение множества всех отношений из $\operatorname{Arg}\min\limits_{\rho \in LO[n]} D(\rho)$. Такая задача в общем случае является экспоненциально сложной. Некоторыми методами её решения являются перебор, в том числе с использованием метода ветвей и границ, а также метод, основанный на исследовании графа $\tilde{G} = (A, \tilde{\rho})$ (п. 5). «Частичная» задача заключается в предварительном выяснении существования для заданного $\Delta_0 \in \{0, 2, 4, \ldots\}$ (следствие 2) отношения $\rho \in LO[n]$ с $D(\rho) \leqslant D(\tilde{\rho}) + \Delta_0$ и только в случае положительного ответа на этот вопрос — в нахождении $\operatorname{Arg}\min\limits_{\rho \in LO[n]} D(\rho)$ (в случае отрицательного ответа считаем, что приемлемого отношения не существует). Эта задача имеет полиномиальную сложность. В любом случае ищется точное множество $\operatorname{Arg}\min\limits_{\rho \in LO[n]} D(\rho)$.

2. Подробно рассматривается случай (достаточно частый), когда $\Delta_0 = 2$. Доказывается теорема о необходимом и достаточном условии существования решений и приводится простой алгоритм их нахождения, вычислительная сложность которого составляет $O(n^2)$.

3. При решении основной задачи часто возникает вспомогательная задача, заключающаяся в определении для некоторого $\rho \in LA[n]/LO[n]$ наличия «рядом» с ρ какого-нибудь линейного порядка $\lambda(\rho) \in LO[n]$, находящегося на минимально возможном расстоянии (равном 2) от ρ, и нахождении этого «ближайшего» к ρ линейного порядка (или всех таковых). Множество всех ρ, для которых указанное $\lambda(\rho)$ существует, обозначается $LA^{(2)}[n]$. Приводятся необходимое и некоторые достаточные легко проверяемые условия для выполнения $\rho \in LA^{(2)}[n]$, а также алгоритм нахождения соответствующих $\lambda(\rho)$.

4. Рассматривается задача разбиения $LA[n]$ на классы π-эквивалентности (аналог изоморфизма в графах). Для $LA^{(2)}[n]$ эта задача решается исчерпывающим образом и описывается алгоритм перечисления всех представителей классов π-эквивалентности

для этого множества. Для случаев $n = 3, 4$ и 5 эта задача также решается полностью. Приведены утверждения (см. лемму 1, а также п. 4), позволяющие при решении задачи перечисления для каждого очередного n использовать решения для предыдущих n. Знание системы представителей позволяет некоторые исследования, проводимые для $\rho \in LA[n]$ (в частности, при использовании $\tilde{\rho}$ в методе из п. 5), сводить к нахождению представителя класса π-эквивалентности, которому принадлежит ρ (при условии, что свойства этого представителя уже изучены).

5. Получен ряд результатов для классов π-эквивалентности множества $LA^{(2)}[n]$ (количество элементов в этих классах, число ближайших линейных порядков для заданного $\rho \in LA^{(2)}[n]$ и некоторые другие). Выведена формула $|LA^{(2)}[n]| = (n-2)(3n-7)n!/6$, показывающая, что при больших n число $|LA^{(2)}[n]|$ во много раз превышает $|LO[n]|$. Например, $|LA^{(2)}[10]| \approx 29{,}95|LO[10]|$, $|LA^{(2)}[15]| \approx 82{,}33|LO[15]|$.

ЛИТЕРАТУРА

1. *Миркин Б. Г.* Проблема группового выбора. М.: Наука, 1974. 256 с.
2. *Мулен Э.* Кооперативное принятие решений: Аксиомы и модели. М.: Мир, 1991. 464 с.
3. *Young H. P.* Condorcet's theory of voting // Amer. Political Sci. Rev. 1988. No. 82. P. 1231–1244.
4. *Осипова В. А., Подиновский В. В., Яшина Н. П.* О непротиворечивом расширении отношений предпочтения в задачах принятия решений // Журн. вычисл. матем. и матем. физ. 1984. Т. 24. № 6. С. 831–839.
5. *Schulze M.* A new monotonic, clone-independent, reversal symmetric, and Condorcet-consistent single-winner election method // Social Choice and Welfare. 2011. No. 36. P. 267–303.
6. *Нефедов В. Н., Осипова В. А., Смерчинская С. О., Яшина Н. П.* Непротиворечивое агрегирование отношений строгого порядка // Изв. вузов. Математика. 2018. № 5. С. 71–85.
7. *Нефедов В. Н., Смерчинская С. О., Яшина Н. П.* Непротиворечивое агрегирование отношений квазипорядка // Прикладная дискретная математика. 2019. № 45. С. 113–126.
8. *Жуков М. С., Орлов А. И.* Задача исследования и итогового ранжирования мнений группы экспертов с помощью медианы Кемени // Научный журнал КубГАУ. 2016. № 122. С. 784–805.
9. *Алексеев Н. С., Осипова В. А.* Методика построения полной порядковой классификации при наличии информации о сравнительной важности критериев // Научно-технич. вестник Поволжья. 2020. № 11. С. 7–11.
10. *Смерчинская С. О., Яшина Н. П.* Агрегирование предпочтений с учетом важности критериев // Труды МАИ. 2015. № 84. `https://mai.ru/upload/iblock/c72/smerchinskaya_yashina_rus.pdf`.
11. *Smerchinskaya S. O. and Yashina N. P.* An algorithm for pairwise comparison of alternatives in multi-criteria problems // Intern. J. Modeling Simulation Scientific Computing. 2018. V. 9. No. 1. `https://www.worldscientific.com/doi/10.1142/S179396231850006X`.
12. *Берж К.* Теория графов и ее применение. М.: ИЛ, 1962.
13. *Нефедов В. Н.* Некоторые свойства линейно упорядоченной медианы для нечетного числа линейных асимметричных отношений. МАИ. Деп. в ВИНИТИ. № 62–В2021 от 08.11.2021. 50 с.
14. *Кемени Дж., Снелл Дж.* Кибернетическое моделирование. М.: Сов. радио, 1972.
15. *Корнеенко В. П.* Методы многокритериального оценивания объектов с многоуровневой структурой показателей эффективности. М.: МАКС Пресс, 2018. 296 с.

REFERENCES

1. *Mirkin B. G.* Group Choice. V.H. Winston and Sons Publ., 1979. 252 p.
2. *Moulin E.* Axioms of Cooperative Decision Making. Cambridge, Cambridge University Press, 1988.
3. *Young H. P.* Condorcet's theory of voting. Amer. Political Sci. Rev., 1988, no. 82, pp. 1231–1244.
4. *Osipova V. A., Podinovskiy V. V., and Yashina N.P.* On non-contradictory extension of preference relations in decision-making problems. USSR Comput. Math. and Math. Physics, 1984, vol. 24, iss. 3, pp. 128–134.
5. *Schulze M.* A new monotonic, clone-independent, reversal symmetric, and Condorcet-consistent single-winner election method. Social Choice and Welfare, 2011, no. 36, pp.267–303.
6. *Nefedov V. N., Osipova V. A., Smerchinskaya S. O., and Yashina N. P.* Non-contradictory aggregation of strict order relations. Russian Math., 2018, vol. 62, pp. 61–73.
7. *Nefedov V. N., Smerchinskaya S. O., and Yashina N. P.* Neprotivorechivoe agregirovanie otnosheniy kvaziporyadka [Non-contradictory aggregation of quasi-order relations]. Prikladnaya Diskretnaya Matematika, 2019, no. 45, pp. 113–126. (in Russian)
8. *Zhukov M. S. and Orlov A. I.* Zadacha issledovaniya i itogovogo ranzhirovaniya mneniy gruppy ekspertov s pomoshch'yu mediany Kemeni [The problem of research of final ranking for group of experts by means of Kemeny median]. Scientific J. KubSAU, 2016, vol. 122, pp. 784–805. (in Russian)
9. *Alekseev N. S. and Osipova V. A.* Metodika postroeniya polnoy poryadkovoy klassifikatsii pri nalichii informatsii o sravnitel'noy vazhnosti kriteriev [Methodology for building a full order classification with information about comparative importance of criteria]. Nauchno-Tekhnich. Vestnik Povolzh'ya, 2020, no. 11, pp. 7–11. (in Russian)
10. *Smerchinskaya S. O. and Yashina N. P.* Agregirovanie predpochteniy s uchetom vazhnosti kriteriev [Aggregation of preferences taking into account the importance of criteria]. Proc. MAI, 2015, no. 84. `https://mai.ru/upload/iblock/c72/smerchinskaya_yashina_rus.pdf`. (in Russian)
11. *Smerchinskaya S. O. and Yashina N. P.* An algorithm for pairwise comparison of alternatives in multi-criteria problems. Intern. J. Modeling Simulation Scientific Computing, 2018, vol. 9, no. 1. `https://www.worldscientific.com/doi/10.1142/S179396231850006X`.
12. *Berge C.* Theorie des Graphes et ses Applications. Paris, DUNOD, 1958.
13. *Nefedov V. N.* Nekotorye svoystva lineyno uporyadochennoy mediany dlya nechetnogo chisla lineynykh asimmetrichnykh otnosheniy [Some properties of a linearly ordered median for an odd number of linear asymmetric binary relations]. MAI. Dep. VINITI no. 62–B2021, 08.11.2021. 50 p. (in Russian)
14. *Kemeny J. and Snell J.* Mathematical Models in the Social Sciences. N.Y., Blaisdell Publ. Comp., 1963.
15. *Korneenko V. P.* Metody mnogokriterial'nogo otsenivaniya ob"ektov s mnogourovnevoy strukturoy pokazateley effektivnosti [Methods for Multi-Criteria Evaluation of Objects with a Multi-Level Structure of Performance Indicators]. Moscow, MAKS Press , 2018. (in Russian)

СВЕДЕНИЯ ОБ АВТОРАХ

ВЫСОЦКАЯ Виктория Владимировна — аспирантка кафедры информационной безопасности Московского государственного университета им. М. В. Ломоносова, специалист-исследователь лаборатории криптографии АО НПК «Криптонит», г. Москва. E-mail: **v.vysotskaya@kryptonite.ru**

КОВРИЖНЫХ Мария Антоновна — выпускница Национального исследовательского университета «Высшая школа экономики», г. Москва.
E-mail: **makovrizhnykh@gmail.com**

КОСОЛАПОВ Юрий Владимирович — кандидат технических наук, доцент Южного федерального университета, г. Ростов-на-Дону. E-mail: **yvkosolapov@sfedu.ru**

ЛЕЛЮК Евгений Андреевич — аспирант Южного федерального университета, г. Ростов-на-Дону. E-mail: **lelukevgeniy@mail.ru**

НЕФЕДОВ Виктор Николаевич — кандидат физико-математических наук, доцент, доцент Московского авиационного института, г. Москва.
E-mail: **nefedovvn@yandex.ru**

ПОЛИЩУК Павел Александрович — аспирант Белорусского государственного университета, г. Минск. E-mail: **poulpvp101@gmail.com**

РЫБАЛОВ Александр Николаевич — кандидат физико-математических наук, старший научный сотрудник лаборатории комбинаторных и вычислительных методов алгебры и логики Института математики им. С. Л. Соболева СО РАН, г. Омск.
E-mail: **alexander.rybalov@gmail.com**

ТРИФОНОВ Дмитрий Игоревич — ведущий научный сотрудник Академии криптографии Российской Федерации, г. Москва. E-mail: **d.arlekino@gmail.com**

ФОМИН Денис Бонифлавович — старший преподаватель Национального исследовательского университета «Высшая школа экономики», г. Москва.
E-mail: **dfomin@hse.ru**

ЧЕРЕМУШКИН Александр Васильевич — доктор физико-математических наук, академик Академии криптографии Российской Федерации, г. Москва.
E-mail: **avc238@mail.ru**

ЧИЖОВ Иван Владимирович — кандидат физико-математических наук, доцент кафедры информационной безопасности Московского государственного университета им. М. В. Ломоносова, заместитель руководителя лаборатории криптографии по научной работе АО НПК «Криптонит», старший научный сотрудник института проблем информатики Федерального исследовательского центра «Информатика и управление» РАН, г. Москва. E-mail: **i.chizhov@kryptonite.ru**