МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА

Научный журнал

2024 № 63

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*
Редактор-переводчик *Т. В. Бутузова*
Верстка *И. А. Панкратовой*

Подписано к печати 01.03.2024. Формат $60 \times 84\frac{1}{8}$. Усл. п. л. 15,2. Тираж 300 экз.
Заказ № 5798. Цена свободная. Дата выхода в свет 07.03.2024.

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

СОДЕРЖАНИЕ

CONTENTS

ПАМЯТИ ВИТАЛИЯ АНАТОЛЬЕВИЧА РОМАНЬКОВА

13 декабря 2023 г. ушел из жизни Виталий Анатольевич Романьков, выдающийся математик, доктор физико-математических наук, главный научный сотрудник лаборатории комбинаторных и вычислительных методов алгебры и логики Омского филиала Института математики им. С. Л. Соболева Сибирского отделения Российской академии наук, профессор Омского государственного университета им. Ф. М. Достоевского.

Виталий Анатольевич родился 12 февраля 1948 г. в Усть-Каменогорске (ныне Республика Казахстан). Уже в школьные годы проявлял большие математические способности, участвовал в республиканских и всесоюзных математических олимпиадах. Увлекался шахматами. В 1966 г. поступил на мехмат Новосибирского государственного университета, который с отличием окончил.

В 1974 г. успешно защитил кандидатскую диссертацию под руководством профессора В. Н. Ремесленникова, а затем принял приглашение М. И. Каргаполова и стал одним из участников «научного десанта», отправившегося в Омск для открытия в 1978 г. Омского Комплексного отдела Института математики СО РАН и преподавания в недавно образованном Омском государственном университете (ОмГУ). В этот «научный десант», помимо Виталия Анатольевича, входили молодые перспективные новосибирские алгебраисты: Г. П. Кукин, Г. А. Носков, В. Я. Беляев, А. Г. Мясников, А. Н. Гришков, А. В. Боровик, А. Н. Зубков. В дальнейшем все они станут крупными учеными, а полученные ими результаты сделают Омск одним из признанных мировых алгебраических центров. Во главе «десанта» стоял В. Н. Ремесленников, уже тогда математик с мировым именем.

В 1992 г. Виталий Анатольевич защитил докторскую диссертацию по специальности 01.01.06 «Математическая логика, алгебра и теория чисел». В 1995 г. получил звание профессора и возглавил кафедру информационных систем в ОмГУ. В настоящее

время после объединения с кафедрой математической логики эта кафедра носит название кафедры компьютерной математики и программирования. Был награжден знаком «Почетный работник высшего профессионального образования Российской Федерации» (2002), стал лауреатом премии «Достижение года» ОмГУ им. Ф. М. Достоевского (2014). Под его руководством были успешно защищены 12 кандидатских диссертаций и две Ph.D. диссертации.

Виталий Анатольевич — автор более 100 научных работ, 10 монографий и учебных пособий. Член редакционных коллегий журналов «Прикладная дискретная математика», «Вестник Омского университета», «Groups-Complexity-Cryptology». Руководил грантами Российского фонда фундаментальных исследований (1995–2014), грантами Российского научного фонда (2015–2023). Работал по приглашению за рубежом: в США, Англии, Канаде, Испании, Турции. Выступал с пленарными докладами на крупных международных конференциях в Нью-Йорке, Линкольне, Олбани (США), на известных семинарах по теории групп Нью-Йоркского городского университета, семинарах Принстона и Ратгерса (США), Бирмингема, Лидса и Манчестера (Англия), Виннипега, Оттавы и Монреаля (Канада). Им также был прочитан ряд курсов за рубежом: курс дифференциальной геометрии в Институте технологий Стивенса (Хобокен, США), несколько курсов в университетах Усть-Каменогорска (Республика Казахстан).

Основной областью научных исследований Виталия Анатольевича была теория бесконечных групп. Здесь ещё в молодые годы ему удалось решить несколько классических задач, поставленных А. И. Мальцевым, М. И. Каргаполовым, Б. Нейманом, Э. Хрущевским и др. Отличительной чертой научного творчества Виталия Анатольевича уже тогда являлся оригинальный, нестандартный подход к трудным проблемам, дающий в итоге решение, как в боксе — сильный неожиданный удар, который приводит к нокауту. В качестве примера можно привести доказательство алгоритмической неразрешимости проблемы решения уравнений в нильпотентных группах достаточно большого ранга. На тот момент (конец 1970-х годов) основным методом доказательства алгоритмической неразрешимости в алгебре было моделирование машин Тьюринга и сведение связанных с ними неразрешимых проблем к изучаемым алгебраическим проблемам. Виталию Анатольевичу же удалось в нильпотентных группах смоделировать диофантовы уравнения над целыми числами и свести знаменитую неразрешимую десятую проблему Гильберта к проблеме решения уравнений в нильпотентных группах, тем самым доказав неразрешимость последней. В дальнейшем метод Романькова был использован многими математиками, а статья, где этот результат получен, является одной из его самых цитируемых работ.

Научное творчество Виталия Анатольевича опровергает стереотип о том, что математика — это дело молодых. За последние пять лет жизни им были решены проблема Ольшанского — Микаэляна о вложениях в разрешимые группы, проблема о вложении в подмоноид нильпотентных групп, проблема Поста для свободных групп. Эти результаты были признаны важнейшими результатами Института математики им. С. Л. Соболева СО РАН в 2021 и 2023 годах. Как говорил сам Виталий Анатольевич, после шестидесяти лет он написал больше работ, чем до шестидесяти. Если смотреть на наукометрические показатели, то стоит отметить высокий (особенно для математика) индекс Хирша: по РИНЦ он равен 21, по SCOPUS — 12.

Другой научной страстью Виталия Анатольевича была алгебраическая криптография. Здесь им открыт принципиально новый метод криптографического анализа схем алгебраической криптографии — метод разложения. Первая версия метода оперирует с линейным, а вторая — с нелинейным разложением. Впоследствии Виталием Анато-

льевичем и его учениками было установлено, что несколько десятков известных криптографических схем не защищены относительно атаки с использованием этого метода. Передаваемое в схеме сообщение эффективно раскрывается без вычисления закрытых параметров шифрования. Кроме того, им были заложены основы диофантовой криптографии, где в качестве платформы для криптографических алгоритмов используются полиномиальные уравнения над целыми числами. С 2006 г. Виталий Анатольевич принимал активное участие в конференциях SIBECRYPT, делал там доклады с новыми результатами, читал образовательные лекции.

Помимо своих научных заслуг, Виталий Анатольевич был прекрасным наставником. Он всегда неравнодушно обсуждал научные дела со своими коллегами, с готовностью делился знаниями с молодыми учеными, своими учениками.

Виталий Анатольевич был очень интересным собеседником, увлекающимся человеком. В список его увлечений входили многие виды спорта, история России, поэзия. Он мог наизусть долго читать стихи многих поэтов — от классиков до современников. Для него было важно то, что связывало поэзию и алгебру — очевидная красота и скрытая гармония. Он ценил красоту науки и всегда стремился к ней в своих работах. Его страсть к математике вдохновляла многих исследователей.

Светлая память о Виталии Анатольевиче Романькове навсегда останется в наших сердцах.

Редакционная коллегия

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

УДК 519.7 DOI 10.17223/20710410/63/1

КОНСТРУКЦИЯ УРАВНОВЕШЕННЫХ ФУНКЦИЙ С ВЫСОКОЙ НЕЛИНЕЙНОСТЬЮ И ДРУГИМИ КРИПТОГРАФИЧЕСКИМИ СВОЙСТВАМИ[1]

А. С. Шапоренко

Новосибирский государственный университет, г. Новосибирск, Россия

E-mail: shaporenko.alexandr@gmail.com

Предлагается новая итеративная конструкция, которую можно применить для построения уравновешенных функций с высокой нелинейностью. Показано, как данная конструкция может быть использована для построения уравновешенных функций от чётного числа $n \geqslant 18$ переменных без линейных структур с нелинейностью $2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7})$. Приведены дополнительные условия, при которых функции, полученные с помощью итеративной конструкции, будут корреляционно-иммунными. Получены результаты, связанные с проблемой разложения булевых функций в сумму двух бент-функций.

Ключевые слова: *уравновешенные булевы функции, нелинейные булевы функции, бент-функции.*

CONSTRUCTION OF BALANCED FUNCTIONS WITH HIGH NONLINEARITY AND OTHER CRYPTOGRAPHIC PROPERTIES

A. S. Shaporenko

Novosibirsk State University, Novosibirsk, Russia

We present an iterative construction that can be used to construct balanced functions with high nonlinearity. Using this construction, we obtained Boolean functions in an even number $n \geqslant 18$ of variables which have no linear structures with nonlinearity $2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7})$. Additional conditions are given under which the functions obtained using the construction will be correlation immune. We also present results concerning "bent sum decomposition problem".

Keywords: *balanced Boolean functions, nonlinear Boolean functions, bent functions.*

Введение

Нелинейность является важным криптографическим свойством булевых функций. Шифры, которые используют функции с высокой нелинейностью в качестве своих

[1]Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования Российской Федерации № 075-15-2022-282.

компонент, являются более стойкими к линейному криптоанализу [1], так как их тяжелее всего приблизить аффинными функциями. Булевы функции от чётного числа переменных называются бент-функциями, если они имеют наибольшее значение нелинейности [2]. Бент-функции использовались в построении блочного шифра CAST [3], поточного шифра Grain [4] и хэш-функции HAVAL [5]. Бент-функции также связаны с некоторыми объектами теории кодирования, алгебры и комбинаторики [6, 7].

Известно, что бент-функции не обладают другим важным криптографическим свойством — они не уравновешены. Данная работа посвящена построению уравновешенных функций с высокой нелинейностью. Мы приводим итеративный способ построения уравновешенных булевых функций, которые при дополнительных условиях могут обладать такими криптографическими свойствами, как высокая нелинейность, отсутствие линейных структур и корреляционная иммунность.

Структура работы следующая: в п. 1 приведены основные определения и вспомогательные факты, которые используются при доказательстве основных результатов. Пункт 2 посвящён итеративной конструкции булевых функций, производная которых по некоторому ненулевому направлению имеет хотя бы одну линейную переменную. В п. 3 рассматривается частный случай — конструкции функций, которые имеют аффинные производные. Приводятся достаточные условия, при которых функции, полученные с помощью итеративной конструкции, обладают такими криптографическими свойствами, как уравновешенность, отсутствие линейных структур и корреляционная иммунность. В п. 4 описан способ получения уравновешенных функций от чётного числа $n \geqslant 18$ переменных без линейных структур с нелинейностью $2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7})$. В п. 5 приведены результаты, связанные с проблемой разложения произвольной булевой функции в сумму двух бент-функций.

1. Определения и необходимые утверждения

1.1. Булевы функции

Пусть $\mathbb{Z}_2 = \{0, 1\}$. Векторное пространство двоичных векторов длины n обозначается $\mathbb{Z}_2^n$. Пусть $\oplus$ обозначает сложение по модулю 2. Для $x, y \in \mathbb{Z}_2^n$ будем использовать следующее произведение:

$$\langle x, y \rangle = x_1 y_1 \oplus \cdots \oplus x_n y_n,$$

где x_i — i-я координата x, $i = 1, \ldots, n$.

Функция $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ называется *булевой функцией* от n переменных. Множество всех булевых функций от n переменных обозначим $\mathcal{F}_n$. С каждой булевой функцией f от n переменных можно связать её *носитель*:

$$\operatorname{supp}(f) = \{x \in \mathbb{Z}_2^n : f(x) = 1\}.$$

Весом Хэмминга $\operatorname{wt}(f)$ функции $f \in \mathcal{F}_n$ называется количество ненулевых значений f: $|\{x \in \mathbb{Z}_2^n : f(x) = 1\}|$. Функция $f \in \mathcal{F}_n$ называется *уравновешенной*, если $\operatorname{wt}(f) = 2^{n-1}$.

Расстояние Хэмминга $\mathrm{d}(f, g)$ между двумя булевыми функциями $f, g \in \mathcal{F}_n$ вычисляется следующим образом:

$$\mathrm{d}(f, g) = |\{x \in \mathbb{Z}_2^n : f(x) \neq g(x)\}|.$$

Каждую булеву функцию f от n переменных можно единственным образом представить в виде *алгебраической нормальной формы* (*АНФ*), или *полинома Жегалкина*:

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

где при каждом k индексы $i_1,\dots,i_k$ различны и в совокупности пробегают все k-элементные подмножества $\{1,\dots,n\}$, а коэффициенты $a_{i_1,\dots,i_k}, a_0$ принимают значения 0 или 1.

Алгебраической степенью (*степенью*) $\deg(f)$ функции f называется количество переменных в самом длинном слагаемом её АНФ, при котором коэффициент не равен нулю. Функция степени не выше 1 называется *аффинной*. Аффинную функцию от n переменных можно представить в виде $\ell=\langle x,a\rangle\oplus b$, где $a\in\mathbb{Z}_2^n$ и $b\in\mathbb{Z}_2$. Множество всех аффинных функций от n переменных обозначим $\mathcal{A}_n$.

Булевы функции $f,g\in\mathcal{F}_n$ *аффинно эквивалентны*, если существуют невырожденная квадратная двоичная матрица A порядка $n\times n$ и вектор $b\in\mathbb{Z}_2^n$, такие, что $g(x)=f(Ax\oplus b)$.

Производной булевой функции $f\in\mathcal{F}_n$ называется функция $D_yf(x)=f(x)\oplus f(x\oplus y)$, где вектор $y\in\mathbb{Z}_2^n$ является *направлением*, по которому берётся производная. Легко убедиться, что $D_y\big(f\oplus g\big)=D_yf\oplus D_yg$.

Следующий факт представлен в [8] без доказательства. Для полноты приведём его с доказательством.

Лемма 1 (Н. Н. Токарева [8]). Булева функция $f\in\mathcal{F}_n$ является производной некоторой булевой функции $g\in\mathcal{F}_n$ по ненулевому направлению $y\in\mathbb{Z}_2^n$ тогда и только тогда, когда $f(x)\oplus f(x\oplus y)=0$ для всех $x\in\mathbb{Z}_2^n$.

Доказательство.

Необходимость. Пусть $D_yg(x)=f(x)$. Можно заметить, что $D_yg(x)=g(x)\oplus$ $\oplus\, g(x\oplus y)=D_yg(x\oplus y)$ для всех $x\in\mathbb{Z}_2^n$. Значит, $f(x)=f(x\oplus y)$ для всех $x\in\mathbb{Z}_2^n$.

Достаточность. Пусть i — первая ненулевая координата y и $g(x)=x_if(x)$ для всех $x\in\mathbb{Z}_2^n$. Тогда

$$D_yg(x)=x_if(x)\oplus(x_i\oplus1)f(x\oplus y)=f(x)\text{ для всех }x\in\mathbb{Z}_2^n.$$

Следовательно, f — производная g по направлению y. ■

Для каждого $y\in\mathbb{Z}_2^n$ *коэффициентом Уолша — Адамара* $W_f(y)$ булевой функции $f\in\mathcal{F}_n$ называется величина, определяемая равенством

$$W_f(y)=\sum_{x\in\mathbb{Z}_2^n}(-1)^{f(x)\oplus\langle x,y\rangle}.$$

Нам также понадобятся следующие хорошо известные факты:

Лемма 2. Булева функция $f\in\mathcal{F}_n$ является уравновешенной тогда и только тогда, когда $W_f(0)=0$.

Лемма 3. Пусть $f\in\mathcal{F}_n$, $\ell\in\mathcal{A}_n$ и $\ell(x)=\langle a,x\rangle\oplus b$, где $a\in\mathbb{Z}_2^n, b\in\mathbb{Z}_2$. Тогда для любого $c\in\mathbb{Z}_2^n$ справедливо $W_{f\oplus\ell}(c)=(-1)^b\,W_f(a\oplus c)$.

1.2. Бент-функции

Нелинейностью N_f булевой функции $f \in \mathcal{F}_n$ называется расстояние Хэмминга от данной функции до множества всех аффинных функций:

$$N_f = \mathrm{d}(f, \mathcal{A}_n) = \min_{a \in \mathbb{Z}_2^n, b \in \mathbb{Z}_2} \mathrm{d}(f, \ell_{a,b}),$$

где $\ell_{a,b}(x) = \langle a, x \rangle \oplus b$.

Лемма 4 (О. Ротхаус [2]). Пусть $f \in \mathcal{F}_n$. Тогда

$$N_f = 2^{n-1} - \frac{1}{2} \max_{a \in \mathbb{Z}_2^n} |W_f(a)|.$$

Булева функция от чётного числа переменных n называется *бент-функцией*, если её нелинейность достигает наибольшего возможного значения $2^{n-1} - 2^{n/2-1}$. Обозначим через $\mathcal{B}_n$ множество всех бент-функций от n переменных.

Бент-функции были определены О. Ротхаусом в 60-х годах прошлого века, хотя его работа [2] была опубликована только в 1976 г. Однако известно, что с конца 1950-х годов в Советском Союзе исследовались булевы функции с аналогичными свойствами, которые называли «минимальными функциями». В 1961 г. математики В. А. Елисеев и О. П. Степченков описали класс функций, который является аналогом класса Мэйорана — МакФарланда, представленного в 1973 г. Бент-функции также связаны с другими математическими объектами. Так, например, Р. Л. МакФарланд [9] и Дж. Диллон [10] исследовали бент-функции в терминах разностных множеств.

Нам понадобятся следующие хорошо известные факты:

Лемма 5. Пусть $f \in \mathcal{B}_n$ и $n \geqslant 4$. Тогда $\deg(f) \leqslant n/2$.

Лемма 6. Пусть $f \in \mathcal{B}_n$. Тогда $\mathrm{wt}(f) = 2^{n-1} \pm 2^{n/2-1}$.

Следовательно, бент-функции никогда не являются уравновешенными.

Лемма 7 (О. Ротхаус [2]). Булева функция $f \in \mathcal{F}_n$ является бент-функцией тогда и только тогда, когда $W_f(y) = \pm 2^{n/2}$ для любого $y \in \mathbb{Z}_2^n$.

Лемма 8 (О. Ротхаус [2]). Пусть $f \in \mathcal{B}_n$. Тогда:

1) любая булева функция, аффинно эквивалентная f, является бент-функцией;
2) функция $f \oplus \ell$ является бент-функцией от n переменных для любой аффинной функции ℓ.

Для бент-функции f от n переменных *дуальная функция* $\tilde{f}$ определяется с помощью равенств $W_f(y) = 2^{n/2}(-1)^{\tilde{f}(y)}$ для всех $y \in \mathbb{Z}_2^n$. Отметим, что $\tilde{f}$ также является бент-функцией [7].

Лемма 9 (О. Ротхаус [2]). Булева функция $f \in \mathcal{F}_n$ является бент-функцией тогда и только тогда, когда для любого ненулевого направления y её производная $D_y f(x) = f(x) \oplus f(x \oplus y)$ является уравновешенной.

Приведём один из самых известных классов бент-функций — класс Мэйорана — МакФарланда, который был впервые определён в [10] и основан на работах Дж. А. Майорана и Р. Л. МакФарланда 1971–1973 гг.

Лемма 10 (Дж. Диллон [10]). Пусть $x, y \in \mathbb{Z}_2^n$, π — взаимно однозначное отображение на $\mathbb{Z}_2^n$, $g \in \mathcal{F}_n$ — произвольная функция. Тогда функция

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

— бент-функция от $2n$ переменных.

1.3. Линейные структуры и корреляционная иммунность

Переменная булевой функции называется *линейной*, если она входит в АНФ функции линейно. Если переменная не входит в АНФ булевой функции, то эта переменная называется *фиктивной*. Булева функция f *имеет линейную структуру*, если существует ненулевое направление $y \in \mathbb{Z}_2^n$, такое, что $D_y f(x) \equiv \mathrm{const}$. Следующий факт показывает, что функции, которые имеют линейные структуры, эквивалентны функциям с простым строением.

Лемма 11 (О. А. Логачев и др. [11]). Пусть $f \in \mathcal{F}_n$ имеет линейную структуру. Тогда существует функция $g \in \mathcal{F}_n$, которая аффинно эквивалентна f и имеет линейную или фиктивную переменную.

Булева функция $f \in \mathcal{F}_n$ называется *корреляционно-иммунной порядка* r, $1 \leqslant r \leqslant n$, если для любой её подфункции $g = f^{a_1,\ldots,a_r}_{i_1,\ldots,i_r}$, полученной из f подстановкой констант $a_1, \ldots, a_r$ вместо переменных $x_{i_1}, \ldots, x_{i_r}$, выполняется $\mathrm{wt}(g) = \mathrm{wt}(f)/2^r$. Требование корреляционной иммунности функции связано с противостоянием корреляционной атаке [12].

Лемма 12 (Т. Зигенталер [12]). Функция $f \in \mathcal{F}_n$ является корреляционно-иммунной порядка r, если и только если $W_f(a) = 0$ для всех векторов $a \in \mathbb{Z}_2^n$, таких, что $1 \leqslant \mathrm{wt}(a) \leqslant r$.

1.4. Уравновешенные функции с высокой нелинейностью

Как уже отмечалось, бент-функции не являются уравновешенными, что вызывает статистическую корреляцию между открытым и зашифрованным текстами.

Максимальная нелинейность уравновешенных функций неизвестна для $n > 7$. В работе [13] приведена следующая верхняя оценка нелинейности уравновешенных функций от чётного числа переменных.

Утверждение 1 (Дж. Себерри и др. [13]). Пусть $n \geqslant 4$ — чётное число и f — уравновешенная булева функция от n переменных. Тогда $N_f \leqslant 2^{n-1} - 2^{n/2-1} - 2$.

Одним из способов построения уравновешенных функций с высокой нелинейностью является преобразование бент-функций с целью получения уравновешенных булевых функций, которые сохраняют высокие значения нелинейности [14, 15]. Уравновешенным функциям с высокой нелинейностью посвящены также работы [13, 16–18].

2. Конструкция булевых функций, производные которых имеют линейную переменную

Опишем конструкцию булевых функций, производная которых по некоторому ненулевому направлению имеет хотя бы одну линейную переменную. Данная конструкция имеет управляемую производную и позволяет строить все булевы функции, имеющие в качестве своей производной по некоторому ненулевому направлению функцию хотя бы с одной линейной переменной. Для $n = 4$ и 6 покажем, что множество функций, которые можно построить с помощью данной конструкции, содержит уравновешенные функции с высокой нелинейностью.

Теорема 1. Пусть $n \geqslant 2$ — чётное число, $g_1, g_2, h_1 \in \mathcal{F}_n$, $(y, 1, y_{n+2}) \in \mathbb{Z}_2^{n+2}$ и $h(x, x_{n+1}, x_{n+2}) = (D_y h_1(x) \oplus y_{n+2}) x_{n+1} \oplus h_1(x) \oplus x_{n+2}$. Тогда функция $f \in \mathcal{F}_{n+2}$, построенная следующим образом:

$$\begin{aligned} f(x, x_{n+1}, x_{n+2}) = \big((D_y g_1(x) \oplus 1) h(x, x_{n+1}, x_{n+2}) \oplus D_y g_2(x)\big) x_{n+1} \oplus \\ \oplus g_1(x) h(x, x_{n+1}, x_{n+2}) \oplus g_2(x), \end{aligned} \tag{1}$$

имеет h своей производной по направлению $(y, 1, y_{n+2})$. При этом для вектора $(a, a_{n+1}, a_{n+2}) \in \mathbb{Z}_2^{n+2}$ и $c = \langle a, y\rangle \oplus a_{n+1} \oplus a_{n+2}y_{n+2}$ справедливо

$$W_f(a, a_{n+1}, a_{n+2}) = (-1)^{c\cdot a_{n+2}} \cdot 2 \cdot W_{cg_1(x)\oplus g_2(x)\oplus a_{n+2}h_1(x)}(a).$$

Доказательство. Заметим, что $D_{(y,1,y_{n+2})}h(x, x_{n+1}, x_{n+2}) = 0$ для любого $(x, x_{n+1}, x_{n+2}) \in \mathbb{Z}_2^{n+2}$. Из леммы 1 следует, что h является производной булевой функции по направлению $(y, 1, y_{n+2})$. Для любой функции $f \in \mathcal{F}_{n+2}$, которая имеет h своей производной по направлению $(y, 1, y_{n+2})$, справедливо

$$f(x, x_{n+1}, x_{n+2}) \oplus f(x \oplus y, x_{n+1} \oplus 1, x_{n+2} \oplus y_{n+2}) = h(x, x_{n+1}, x_{n+2}). \tag{2}$$

Поскольку $h(x, x_{n+1}, x_{n+2}) = h(x \oplus y, x_{n+1} \oplus 1, x_{n+2} \oplus y_{n+2})$, получаем, что

$$h(x, x_{n+1}, x_{n+2}) = 1 \iff h(x \oplus y, x_{n+1} \oplus 1, x_{n+2} \oplus y_{n+2}) = 1. \tag{3}$$

Если $h(x, x_{n+1}, x_{n+2}) = 1$, то, поскольку h зависит линейно от переменной x_{n+2}, имеем $h(x, x_{n+1}, x_{n+2} \oplus 1) = 0$. Таким образом, справедливо, что

$$\{x : \exists x_{n+2} \in \mathbb{Z}_2\,(h(x, 0, x_{n+2}) = 1)\} = \{x : \exists x_{n+2} \in \mathbb{Z}_2\,(h(x, 0, x_{n+2}) = 0)\} = \mathbb{Z}_2^n. \tag{4}$$

Из (2)–(4) следует, что любая булева функция f от $(n + 2)$ переменных, для которой $D_{(y,1,y_{n+2})}f(x, x_{n+1}, x_{n+2}) = h(x, x_{n+1}, x_{n+2})$, имеет следующее представление:

$$\begin{cases} f(x, 0, x_{n+2}) = f_1(x), & \text{если } h(x, 0, x_{n+2}) = 1, \\ f(x \oplus y, 1, x_{n+2} \oplus y_{n+2}) = f_1(x) \oplus 1, & \text{если } h(x \oplus y, 1, x_{n+2} \oplus y_{n+2}) = 1, \\ f(x, 0, x_{n+2}) = f_2(x), & \text{если } h(x, 0, x_{n+2}) = 0, \\ f(x \oplus y, 1, x_{n+2} \oplus y_{n+2}) = f_2(x), & \text{если } h(x \oplus y, 1, x_{n+2} \oplus y_{n+2}) = 0, \end{cases} \tag{5}$$

где f_1 и f_2 — произвольные функции от n переменных. Следовательно, перебирая все возможные f_1 и f_2, мы получим все булевы функции от $(n + 2)$ переменных, которые имеют $h(x, x_{n+1}, x_{n+2})$ своими производными по направлению $(y, 1, y_{n+2})$.

Положим, что $g_1 = f_1 \oplus f_2$ и $g_2 = f_2$. Тогда формула (1) для функции f следует из представления (5).

Отметим, что для $(x, x_{n+1}, x_{n+2}) \in \mathbb{Z}_2^{n+2}$ выполняется

$$x_{n+2} = h(x, x_{n+1}, x_{n+2}) \oplus (D_y h_1(x) \oplus y_{n+2})x_{n+1} \oplus h_1(x). \tag{6}$$

Теперь проверим, чему равны коэффициенты Уолша — Адамара функции f для каждого $(a, a_{n+1}, a_{n+2}) \in \mathbb{Z}_2^{n+2}$. Заметим, что

$$\langle (x, x_{n+1}, x_{n+2}), (a, a_{n+1}, a_{n+2})\rangle = \langle a, x\rangle \oplus a_{n+1}x_{n+1} \oplus a_{n+2}x_{n+2}.$$

Тогда из (2) следует, что

$$\begin{aligned} W_f(a, a_{n+1}, a_{n+2}) &= \sum_{(x,x_{n+1},x_{n+2})\in\mathbb{Z}_2^{n+2}} (-1)^{f(x,x_{n+1},x_{n+2})\oplus\langle(x,x_{n+1},x_{n+2}),(a,a_{n+1},a_{n+2})\rangle} = \\ &= \sum_{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}} \Big((-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}} + \\ &+ (-1)^{f(x\oplus y,1,x_{n+2}\oplus y_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}\oplus\langle a,y\rangle\oplus a_{n+1}\oplus a_{n+2}y_{n+2}}\Big) = \end{aligned}$$

$$= \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=1}} \Big((-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}} + $$

$$+(-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}\oplus\langle a,y\rangle\oplus a_{n+1}\oplus a_{n+2}y_{n+2}\oplus 1}\Big) +$$

$$+ \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=0}} \Big((-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}} + (-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}\oplus\langle a,y\rangle\oplus a_{n+1}\oplus a_{n+2}y_{n+2}}\Big).$$

Допустим, что $\langle a,y\rangle \oplus a_{n+1} \oplus a_{n+2}y_{n+2} = 0$. Тогда

$$W_f(a,a_{n+1},a_{n+2}) = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=0}} (-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}}.$$

Из (4) и (5) следует, что если $a_{n+2} = 0$, то

$$W_f(a,a_{n+1},0) = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=0}} (-1)^{f_2(x)\oplus\langle a,x\rangle} = 2\sum_{x\in\mathbb{Z}_2^n} (-1)^{f_2(x)\oplus\langle a,x\rangle} = 2W_{f_2}(a) = 2W_{g_2}(a).$$

Если $a_{n+2} = 1$, то из (5) и (6) следует, что

$$W_f(a,a_{n+1},1) = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=0}} (-1)^{f_2(x)\oplus\langle a,x\rangle\oplus x_{n+2}} = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=0}} (-1)^{f_2(x)\oplus\langle a,x\rangle\oplus h_1(x)}.$$

Тогда, согласно (4), справедливо

$$W_f(a,a_{n+1},1) = 2\sum_{x\in\mathbb{Z}_2^n} (-1)^{f_2(x)\oplus h_1(x)\oplus\langle a,x\rangle} = 2\,W_{f_2\oplus h_1}(a) = 2\,W_{g_2\oplus h_1}(a).$$

Теперь пусть $\langle a,y\rangle \oplus a_{n+1} \oplus a_{n+2}y_{n+2} = 1$. Тогда

$$W_f(a,a_{n+1},a_{n+2}) = \sum_{(x,x_{n+1},x_{n+2})\in\mathbb{Z}_2^{n+2}} (-1)^{f(x,x_{n+1},x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+1}x_{n+1}\oplus a_{n+2}x_{n+2}} =$$

$$= 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=1}} (-1)^{f(x,0,x_{n+2})\oplus\langle a,x\rangle\oplus a_{n+2}x_{n+2}}.$$

Из (4) и (5) следует, что если $a_{n+2} = 0$, то

$$W_f(a,a_{n+1},0) = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=1}} (-1)^{f_1(x)\oplus\langle a,x\rangle} = 2\sum_{x\in\mathbb{Z}_2^n} (-1)^{f_1(x)\oplus\langle a,x\rangle} = 2\,W_{f_1}(a) = 2\,W_{g_1\oplus g_2}(a).$$

Если $a_{n+2} = 1$, то из (5) и (6) следует, что

$$W_f(a,a_{n+1},1) = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=1}} (-1)^{f_1(x)\oplus\langle a,x\rangle\oplus x_{n+2}} = 2 \sum_{\substack{(x,0,x_{n+2})\in\mathbb{Z}_2^{n+2}\\ h(x,0,x_{n+2})=1}} (-1)^{f_1(x)\oplus h_1(x)\oplus\langle a,x\rangle\oplus 1}.$$

Тогда, согласно (4), справедливо

$$W_f(a,a_{n+1},1) = 2\sum_{x\in\mathbb{Z}_2^n} (-1)^{f_1(x)\oplus h_1(x)\oplus\langle a,x\rangle\oplus 1} = -2\,W_{f_1\oplus h_1}(a) = -2\,W_{g_1\oplus g_2\oplus h_1}(a).$$

Теорема 1 доказана. ■

Отметим, что произвольная функция $h \in \mathcal{F}_{n+2}$, которая имеет хотя бы одну линейную переменную, может быть представлена следующим образом: $h(x, x_{n+1}, x_{n+2}) = h_2(x)x_{n+1} \oplus h_1(x) \oplus x_{n+2}$, где $h_1, h_2 \in \mathcal{F}_n$ и $x \in \mathbb{Z}_2^n$. Тогда по лемме 1 функция h является производной некоторой функции по направлению $(y, 1, y_{n+2})$ тогда и только тогда, когда $D_{(y,1,y_{n+2})}h(x, x_{n+1}, x_{n+2}) = 0$. Отсюда нетрудно получить, что $h_2(x) = D_y h_1(x) \oplus y_{n+2}$. Таким образом, теорема 1 позволяет построить все функции от n переменных, производная которых по некоторому ненулевому направлению имеют хотя бы одну линейную переменную.

Полным перебором проверено, что для $n = 4$ множество всех функций, производная которых по некоторому ненулевому направлению имеет хотя бы одну линейную переменную, состоит из 28 896 функций. Это множество содержит все 896 бент-функций от четырёх переменных. Кроме того, все 10 920 уравновешенных функций от четырёх переменных, которые имеют нелинейность 4 (максимально возможную для уравновешенных функций), имеют производную по некоторому ненулевому направлению хотя бы с одной линейной переменной. Более того, все уравновешенные функции, производная которых по некоторому ненулевому направлению имеет хотя бы одну линейную переменную, имеют нелинейность 4.

Булева функция от шести переменных

$$x_3x_4x_5 \oplus x_2x_4x_5 \oplus x_3x_4x_6 \oplus x_3x_5x_6 \oplus x_4x_5x_6 \oplus x_2x_5x_6 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus x_3x_4$$

является уравновешенной и имеет нелинейность 24, тогда как верхняя оценка нелинейности для уравновешенных функций от чётного числа переменных (утверждение 1) даёт 26. Её производная по направлению $(1, 0, \ldots, 0)$ является аффинной. Отметим, что оценка 26 нелинейности уравновешенных функций от 6 переменных достижима [13].

Таким образом, для $n = 6$ существует уравновешенная функция, производная которой по некоторому ненулевому направлению имеет хотя бы одну линейную переменную, с нелинейностью $2^{n-1} - 2^{n/2-1} - 4$. Более того, доказана следующая

Теорема 2. Пусть $f \in \mathcal{F}_{n+2}$ — уравновешенная функция от чётного $n \geqslant 6$ числа переменных, производная которой по некоторому ненулевому направлению имеет хотя бы одну линейную переменную. Тогда $N_f \leqslant 2^{n+1} - 2^{n/2} - 4$.

Доказательство. Из теоремы 1 известно, что f имеет форму (1), при этом g_2 из (1) является уравновешенной функцией от n переменных. Тогда из утверждения 1 верна следующая оценка: $N_{g_2} \leqslant 2^{n-1} - 2^{n/2-1} - 2$. Таким образом, из леммы 4 следует $\max\limits_{a \in \mathbb{Z}_2^n} |W_{g_2}| \geqslant 2^{n/2} + 4$. Тогда из теоремы 1 заключаем, что $N_f = 2^{n+1} - \max\limits_{a \in \mathbb{Z}_2^n, g \in M} |W_g(a)|$, где $M = \{g_2, g_1 \oplus g_2, g_2 \oplus h_1, g_1 \oplus g_2 \oplus h_1\}$, и, следовательно, $N_f \leqslant 2^{n+1} - 2^{n/2} - 4$. ■

3. Криптографические свойства булевых функций, которые имеют аффинные производные

Рассмотрим частный случай конструкции из теоремы 1 — итеративную конструкцию функций, которые имеют аффинные производные, и приведём достаточные условия, при которых функции, полученные с помощью этой конструкции, обладают такими криптографическими свойствами, как уравновешенность, отсутствие линейных структур и корреляционная иммунность.

Утверждение 2. Пусть $n \geqslant 2$ — чётное число, $g_1, g_2 \in \mathcal{F}_n$, $(y, 1, y_{n+2}) \in \mathbb{Z}_2^{n+2}$ и $b \in \mathbb{Z}_2^n$ такие, что $\langle b, y \rangle = y_{n+2}$, и

$$h(x, x_{n+1}, x_{n+2}) = \langle b, x \rangle \oplus c \oplus x_{n+2}, \text{ где } c \in \mathbb{Z}_2.$$

Тогда $f \in \mathcal{F}_{n+2}$ из (1) является уравновешенной функцией от $n+2$ переменных, если и только если g_2 — уравновешенная функция от n переменных. При этом

$$N_f = 2^{n+1} - \max_{a \in \mathbb{Z}_2^n, g \in \{g_2, g_1 \oplus g_2\}} |W_g(a)|.$$

Доказательство. Пусть $\ell_1(x) = \langle b, x \rangle \oplus c$, где $x \in \mathbb{Z}_2^n$. Можно убедиться, что $D_y \ell_1(x) = y_{n+2}$ для любого $x \in \mathbb{Z}_2^n$ и

$$h(x, x_{n+1}, x_{n+2}) = (D_y \ell_1(x) \oplus y_{n+2}) x_{n+1} \oplus \ell_1(x) \oplus x_{n+2}.$$

Из теоремы 1 и леммы 3 для $f \in \mathcal{F}_{n+2}$ из (1) следует, что

$$|W_f(a, a_{n+1}, a_{n+2})| = \begin{cases} 2\,|W_{g_2}(a)|, & \text{если } \langle a, y \rangle = a_{n+1} \text{ и } a_{n+2} = 0, \\ 2\,|W_{g_2}(a \oplus b)|, & \text{если } \langle a, y \rangle = a_{n+1} \oplus y_{n+2} \text{ и } a_{n+2} = 1, \\ 2\,|W_{g_1 \oplus g_2}(a)|, & \text{если } \langle a, y \rangle = a_{n+1} \oplus 1 \text{ и } a_{n+2} = 0, \\ 2\,|W_{g_1 \oplus g_2}(a \oplus b)|, & \text{если } \langle a, y \rangle = a_{n+1} \oplus y_{n+2} \oplus 1 \text{ и } a_{n+2} = 1. \end{cases}$$

Тогда $W_f(\mathbf{0}) = W_{g_2}(0)$ и первое утверждение следует из леммы 2. Второе утверждение следует из леммы 4. ■

3.1. Функции без линейных структур

Приведём достаточные условия того, что функции из утверждения 2 не имеют линейных структур.

Теорема 3. Пусть $n \geqslant 2$ — чётное число, g_1, g_2 — булевы функции от n переменных, $(y, 1, y_{n+2}) \in \mathbb{Z}_2^{n+2}$ и $\ell_1 \in \mathcal{A}_n$ такие, что $D_y \ell_1(x) = y_{n+2}$ и $h(x) = \ell_1(x) \oplus x_{n+2}$. Тогда если g_2 и $g_1 \oplus g_2$ являются уравновешенной функцией и бент-функцией от n переменных соответственно, то булева функция $f \in \mathcal{F}_{n+2}$ из (1) является уравновешенной и не имеет линейных структур.

Доказательство. Рассмотрим производную функции f из (1) по направлению (z, z_{n+1}, z_{n+2}), где $z \in \mathbb{Z}_2^n$ и $z_{n+1}, z_{n+2} \in \mathbb{Z}_2$:

$$\begin{gathered} D_{(z, z_{n+1}, z_{n+2})} f(x, x_{n+1}, x_{n+2}) = \big((D_y g_1(x) \oplus 1)(\ell_1(x) \oplus x_{n+2}) \oplus D_y g_2(x)\big) x_{n+1} \oplus \\ \oplus g_1(x)(\ell_1(x) \oplus x_{n+2}) \oplus g_2(x) \oplus \\ \oplus \big((D_y g_1(x \oplus z) \oplus 1)(\ell_1(x \oplus z) \oplus x_{n+2} \oplus z_{n+2}) \oplus D_y g_2(x \oplus z)\big) x_{n+1} \oplus \\ \oplus \big((D_y g_1(x \oplus z) \oplus 1)(\ell_1(x \oplus z) \oplus x_{n+2} \oplus z_{n+2}) \oplus D_y g_2(x \oplus z)\big) z_{n+1} \oplus \\ \oplus g_1(x \oplus z)(\ell_1(x \oplus z) \oplus x_{n+2} \oplus z_{n+2}) \oplus g_2(x \oplus z). \end{gathered}$$

Пусть $\ell_1(x \oplus z) = \ell_1(x) \oplus d$, где $d \in \mathbb{Z}_2$. Заметим, что если $z = \mathbf{0}$, то $d = 0$. Тогда

$$\begin{gathered} D_{(z, z_{n+1}, z_{n+2})} f(x, x_{n+1}, x_{n+2}) = x_{n+1} x_{n+2} \big(D_z D_y g_1(x)\big) \oplus \\ \oplus x_{n+1} \big(\ell_1(x) D_z D_y g_1(x) \oplus (z_{n+2} \oplus d) D_y g_1(x \oplus z) \oplus D_z D_y g_2(x) \oplus z_{n+2} \oplus d\big) \oplus \\ \oplus x_{n+2} \big(D_z g_1(x) \oplus z_{n+1}(D_y g_1(x \oplus z) \oplus 1)\big) \oplus \ell_1(x) D_z g_1(x) \oplus \\ \oplus z_{n+1}(D_y g_1(x \oplus z) \oplus 1) \ell_1(x) \oplus z_{n+1} d (D_y g_1(x \oplus z) \oplus 1) \oplus \\ \oplus z_{n+1} D_y g_2(x \oplus z) \oplus z_{n+1} z_{n+2}(D_y g_1(x \oplus z) \oplus 1) \oplus (z_{n+2} \oplus d) g_1(x \oplus z) \oplus D_z g_2(x). \end{gathered}$$

Докажем, что для любого ненулевого направления (z, z_{n+1}, z_{n+2}) функция $D_{(z, z_{n+1}, z_{n+2})} f$ не является константой. Предположим обратное. Пусть $D_{(z, z_{n+1}, z_{n+2})} f \equiv \text{const}$ для $(z, z_{n+1}, z_{n+2}) \neq (0, \ldots, 0)$.

Пусть $z_{n+1} = 0$. Тогда $D_z g_1(x) = 0$. Если $z_{n+2} = d$, то $z \neq \mathbf{0}$ и $D_{(z,0,d)}f$ имеет слагаемое $D_z g_2(x) = D_z\big(g_1(x) \oplus g_2(x)\big)$, которое не является константой, согласно лемме 9.

Если $z_{n+2} = d \oplus 1$, то $D_{(z,0,d\oplus 1)}f$ имеет слагаемое $g_1(x \oplus z) \oplus g_2(x \oplus z) \oplus g_2(x)$, которое для любого z не является константой, поскольку $g_2(x)$ уравновешенная, а $g_1(x \oplus z) \oplus g_2(x \oplus z)$ является бент-функцией, согласно лемме 8.

Пусть $z_{n+1} = 1$. Тогда

$$D_z g_1(x) = D_y g_1(x \oplus z) \oplus 1.$$

Заметим, что если $y = z$, то равенство не выполняется.

Если $z_{n+2} = d$, то $D_{(z,1,d)}f$ имеет слагаемое

$$\begin{aligned} D_y g_2(x \oplus z) \oplus D_z g_2(x) &= D_y\big(g_1(x \oplus z) \oplus g_2(x \oplus z)\big) \oplus D_z\big(g_1(x) \oplus g_2(x)\big) \oplus 1 = \\ &= D_{y\oplus z}\big(g_1(x) \oplus g_2(x)\big) \oplus 1, \end{aligned}$$

которое для $y \neq z$ не является константой, согласно лемме 9.

Если $z_{n+2} = d \oplus 1$, то $D_{(z,1,d\oplus 1)}f$ имеет слагаемое

$$g_1(x \oplus y \oplus z) \oplus D_y g_2(x \oplus z) \oplus D_z g_2(x) \oplus 1 = g_1(x \oplus y \oplus z) \oplus g_2(x \oplus y \oplus z) \oplus g_2(x) \oplus 1,$$

которое для любого z не является константой, поскольку $g_2(x)$ и $g_2(x) \oplus 1$ являются уравновешенными, а $g_1(x \oplus y \oplus z) \oplus g_2(x \oplus y \oplus z)$ — бент-функция, согласно лемме 8. Таким образом, $D_{(z,z_{n+1},z_{n+2})}f \not\equiv \mathrm{const}$ для любого $(z, z_{n+1}, z_{n+2}) \neq (0, \ldots, 0)$.

Пусть $\ell_1(x) = \langle b, x\rangle \oplus c$, где $b \in \mathbb{Z}_2^n$ и $c \in \mathbb{Z}_2$. Так как $D_y\ell_1(x) = y_{n+2}$, то $\langle b, y\rangle = y_{n+2}$. Из утверждения 2 следует, что f уравновешенная. ■

3.2. Корреляционно-иммунные функции

Приведём достаточные условия того, что функции из утверждения 2 являются корреляционно-иммунными.

Утверждение 3. Пусть $n \geqslant 2$ — чётное число, $g_1, g_2 \in \mathcal{F}_n$, $(y, 1, y_{n+2}) \in \mathbb{Z}_2^{n+2}$ и $b \in \mathbb{Z}_2^n$ такие, что $\langle b, y\rangle = y_{n+2}$ и $h(x, x_{n+1}, x_{n+2}) = \langle b, x\rangle \oplus c \oplus x_{n+2}$ для $c \in \mathbb{Z}_2$. Тогда если функции g_2 и $g_1 \oplus g_2$ являются корреляционно-иммунными порядка r, то функция $f \in \mathcal{F}_{n+2}$ из (1) корреляционно-иммунна порядка r. Если при этом g_2 уравновешенная, то f также уравновешенная.

Доказательство. Пусть $\ell_1(x) = \langle b, x\rangle \oplus c$, где $x \in \mathbb{Z}_2^n$. Можно убедиться, что $D_y\ell_1(x) = y_{n+2}$ для любого $x \in \mathbb{Z}_2^n$ и

$$h(x, x_{n+1}, x_{n+2}) = (D_y\ell_1(x) \oplus y_{n+2})x_{n+1} \oplus \ell_1(x) \oplus x_{n+2}.$$

Из теоремы 1 и леммы 3 для $f \in \mathcal{F}_{n+2}$ из (1) следует, что

$$|W_f(a, a_{n+1}, a_{n+2})| = \begin{cases} 2\,|W_{g_2}(a)|, & \text{если } \langle a, y\rangle = a_{n+1} \text{ и } a_{n+2} = 0, \\ 2\,|W_{g_2}(a \oplus b)|, & \text{если } \langle a, y\rangle = a_{n+1} \oplus y_{n+2} \text{ и } a_{n+2} = 1, \\ 2\,|W_{g_1\oplus g_2}(a)|, & \text{если } \langle a, y\rangle = a_{n+1} \oplus 1 \text{ и } a_{n+2} = 0, \\ 2\,|W_{g_1\oplus g_2}(a \oplus b)|, & \text{если } \langle a, y\rangle = a_{n+1} \oplus y_{n+2} \oplus 1 \text{ и } a_{n+2} = 1. \end{cases}$$

Тогда первое утверждение следует из леммы 12, а второе — из утверждения 2. ■

4. Построение уравновешенных функций с высокой нелинейностью

Используем итеративную конструкцию из теоремы 3 и уравновешенную функцию от 16 переменных с высокой нелинейностью, представленную в [18], для построения уравновешенных функций от чётного числа переменных $n \geqslant 18$ без линейных структур с нелинейностью

$$2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7}).$$

Сравним полученные значения нелинейности уравновешенных функций с верхней оценкой нелинейности из утверждения 1, а также со значениями нелинейности уравновешенных функций, полученных в других работах.

В [18] показано, как построить уравновешенную функцию от 16 переменных с нелинейностью 32 598. Мы использовали её в качестве уравновешенной функции g_2 из теоремы 3, чтобы получить уравновешенную булеву функцию от 18 переменных с высокой нелинейностью.

Пусть $\sigma_{2,16}$ — булева функция от 16 переменных, которая содержит все квадратичные слагаемые и только их, и $f_{16} = \sigma_{2,16} \oplus \bigoplus\limits_{i=1}^{n/2} x_i$. Тогда g_2 можно задать с помощью её носителя: $\operatorname{supp}(g_2) = \operatorname{supp}(f_{16}) \cup S$, где

$$S = \{8256, 2080, 4112, 2049, 36912, 5264, 34840, 10264, 49169, 38400, 1632, 3075, 2570, 16800, 16908, 1569, 24612, 12417, 29504, 17825, 37413, 18965, 41410, 16613, 5028, 35122, 21656, 61968, 42122, 8000, 24873, 9546, 21541, 10763, 35881, 57372, 45256, 42033, 37524, 19529, 7237, 16446, 17888, 20881, 26817, 49539, 14964, 54452, 51612, 22981, 20723, 989, 46868, 50830, 11884, 1518, 5363, 36553, 43729, 39321, 50459, 55401, 37771, 52359, 5965, 8511, 18551, 58538, 14987, 53799, 44090, 10156, 29283, 27057, 58443, 61497, 35782, 44047, 22940, 7540, 19865, 43961, 15221, 62179, 43927, 57240, 59741, 61867, 14190, 62511, 44665, 3067, 8107, 61937, 51161, 42937, 31835, 44725, 30435, 14324, 30381, 31964, 56506, 54652, 59951, 61206, 43993, 14310, 58959, 32494, 24443, 32381, 62451, 60915, 60381, 44990, 62845, 36351, 32508, 61147, 56309, 32351, 48503, 57215, 32751, 63483, 64510, 65535\}$$

и каждому числу из S ставится в соответствие вектор его двоичного представления длины 16.

В качестве $g_1 \oplus g_2$ мы взяли бент-функцию $\bigoplus\limits_{i=1}^{8} x_i x_{i+8} \oplus \prod\limits_{i=1}^{8} x_i$, которая, согласно лемме 10, принадлежит классу Мэйорана — МакФарланда. Пусть $\ell(x) = \bigoplus\limits_{i=2}^{18} x_i$ и $y = (1, 0, 0, \ldots, 0)$.

Итоговая уравновешенная булева функция f от 18 переменных, которая получается с помощью конструкции из теоремы 3, имеет степень 16, нелинейность $N_f = 130\,732$ и не имеет линейных структур. Стоит отметить, что конструкция из теоремы 3 позволяет получить больше одной функции от 18 переменных с указанной нелинейностью. Достаточно рассмотреть другие направления y, число которых равно $2^{17} - 1$ [19]. Эти направления — ненулевые векторы y, такие, что $\langle (0, 1, \ldots, 1), y \rangle = 0$. В качестве функции $g_1 \oplus g_2$ можно взять бент-функцию, полученную с помощью других известных конструкций бент-функций.

В свою очередь, полученную функцию f от 18 переменных можно использовать, чтобы построить уравновешенную функцию от 20 переменных с нелинейностью

523 608, так как из леммы 4 следует, что $\max\limits_{a\in\mathbb{Z}_2^{18}} W_f(a) = 680 = 2^{18/2} + 2^{18/2-2} + 2^{18/2-4} + + 2^{18/2-6}$. Кроме того, если в качестве $g_1 \oplus g_2$ снова взять бент-функцию, например, из класса Мэйорана — МакФарланда, то по теореме 3 полученная функция не будет иметь линейных структур.

Таким образом, итеративная конструкция из теоремы 3 позволяет строить уравновешенные функции f от чётного числа переменных $n \geqslant 18$ без линейных структур с нелинейностью

$$N_f = 2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7}), \tag{7}$$

поскольку $\max\limits_{a\in\mathbb{Z}_2^{n-2}} W_{g_2}(a) = 2^{(n-2)/2} + 2^{(n-2)/2-2} + 2^{(n-2)/2-4} + 2^{(n-2)/2-6}$.

В табл. 1 приведены значения нелинейности функций, которые можно получить с помощью теоремы 3, и значения нелинейности уравновешенных функций, полученных в работах К. Ху и др. [16] и К. Карле и др. [17]. Отметим, что в [16] рассматриваются значения $n \leqslant 28$, а нелинейность (7) имеют уравновешенные функции от чётного числа $n \geqslant 18$ переменных.

Таблица 1

n	$2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7})$	N_f [16]	N_f [17]
18	130 732	130 504	130 688
20	523 608	523 154	Не приводится
22	2 095 792	2 094 980	Не приводится
24	8 385 888	8 384 490	Не приводится
26	33 548 992	33 545 992	Не приводится
28	134 206 848	134 201 460	Не приводится

Из табл. 1 видно, что значения нелинейности уравновешенных функций, которые могут быть получены с помощью теоремы 3, превосходят значения из работ [16, 17].

В табл. 2 приводятся значения нелинейности функций, которые можно получить с помощью теоремы 3, и верхние оценки нелинейности уравновешенных функций из утверждения 1 для $18 \leqslant n \leqslant 28$.

Таблица 2

n	$2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7})$	$2^{n-1} - 2^{n/2-1} - 2$
18	130 732	130 814
20	523 608	523 774
22	2 095 792	2 096 126
24	8 385 888	8 386 558
26	33 548 992	33 550 334
28	134 206 848	134 209 534

5. Проблема разложения булевых функций в сумму двух бент-функций

Докажем верхнюю оценку степени функции $(f_1 \oplus f_2)h$, где h — булева функция от n переменных, f_1 и f_2 — бент-функции от n переменных, причём $h \oplus f_1$ и $h \oplus f_2$ также являются бент-функциями.

Вопрос о разложении булевых функций в сумму двух бент-функций поставлен Н. Н. Токаревой в работе [20].

Гипотеза 1 (Н. Н. Токарева [20]). Пусть n — чётное число. Тогда любая булева функция от n переменных степени не больше $n/2$ может быть разложена в сумму двух бент-функций от n переменных.

В [20] гипотеза 1 проверена с помощью полного перебора для $n \leqslant 6$. Согласно [20], если гипотеза 1 верна, то справедлива следующая нижняя оценка для числа бент-функций от n переменных:

$$|\mathcal{B}_n| \geqslant 2^{2^{n-2}+\binom{n}{n/2}/4}.$$

В [19] показана связь этой гипотезы со следующей проблемой о производных бент-функций: любая сбалансированная функция f от чётного числа переменных n степени не больше $n/2-1$, такая, что $f(x) = f(x \oplus y)$ для любого $x \in \mathbb{Z}_2^n$ и некоторого ненулевого $y \in \mathbb{Z}_2^n$, является производной бент-функции от n переменных. Эта связь также следует из теоремы 1.

Утверждение 4. Пусть $n \geqslant 2$ — чётное число, $g_1, g_2, h_1 \in \mathcal{F}_n$, вектор $(y, 1, y_{n+2}) \in \in \mathbb{Z}_2^{n+2}$ такой, что

$$h(x, x_{n+1}, x_{n+2}) = (D_y h_1(x) \oplus y_2)x_{n+1} \oplus h_1(x) \oplus x_{n+2}.$$

Тогда $f \in \mathcal{F}_{n+2}$ из (1) имеет функцию h своей производной по направлению $(y, 1, y_{n+2})$ и является бент-функцией от $n+2$ переменных тогда и только тогда, когда g_2, $g_1 \oplus g_2$, $g_2 \oplus h_1$, $g_1 \oplus g_2 \oplus h_1$ являются бент-функциями от n переменных.

Доказательство. Из леммы 4 и теоремы 1 следует, что

$$N_f = 2^{n+1} - \frac{1}{2} \max_{a \in \mathbb{Z}_2^{n+2}} |W_f(a)| = 2^{n+1} - 2^{n/2}$$

тогда и только тогда, когда для любого $b \in \mathbb{Z}_2^n$ справедливо

$$|W_{g_1 \oplus g_2}(b)| = |W_{g_2}(b)| = |W_{g_1 \oplus g_2 \oplus h_1}(b)| = |W_{g_2 \oplus h_1}(b)| = 2^{n/2}.$$

Из леммы 7 следует, что g_2, $g_1 \oplus g_2$, $g_2 \oplus h_1$, $g_1 \oplus g_2 \oplus h_1$ являются бент-функциями от n переменных. ■

Приведём два вспомогательных утверждения.

Утверждение 5 (Н. Н. Токарева [20]). Пусть f_1, f_2, f_3 — бент-функции от n переменных. Тогда функция f, определённая следующим образом:

$$\begin{aligned} f(0,0,x) &= f_1(x), & f(0,1,x) &= f_2(x), \\ f(1,0,x) &= f_3(x), & f(1,1,x) &= f_4(x), \end{aligned}$$

является бент-функцией от $n+2$ переменных тогда и только тогда, когда f_4 — бент-функция от n переменных и $\tilde{f}_1 \oplus \tilde{f}_2 \oplus \tilde{f}_3 \oplus \tilde{f}_4 = 1$.

Утверждение 5 является упрощённой версией результата из работы А. Канто и П. Шарпин 2003 г. [21]. В [21] доказано также

Утверждение 6 (А. Канто и П. Шарпин [21]). Пусть $f_1, f_2, f_3, f_4 \in \mathcal{F}_n$ и функция f, определённая следующим образом:

$$\begin{aligned} f(0,0,x) &= f_1(x), & f(0,1,x) &= f_2(x), \\ f(1,0,x) &= f_3(x), & f(1,1,x) &= f_4(x), \end{aligned}$$

является бент-функцией от $n+2$ переменных. Тогда f_1 — бент-функция, если и только если f_2, f_3, f_4 — бент-функции от n переменных.

Теорема 4. Пусть $n \geqslant 2$ — чётное число, $h \in \mathcal{F}_n$ и $f_1, f_2, h \oplus f_1, h \oplus f_2 \in \mathcal{B}_n$. Тогда:

1) $\deg((f_1 \oplus f_2)h) \leqslant (n+2)/2$;
2) следующие утверждения эквивалентны:
   а) $\varphi_1(x) = \big(f_1(x) \oplus f_2(x)\big)h(x) \oplus f_1(x) \in \mathcal{B}_n$;
   б) $\varphi_2(x) = \big(f_1(x) \oplus f_2(x)\big)h(x) \oplus f_2(x) \in \mathcal{B}_n$;
   в) $\varphi_3(x) = \big(f_1(x \oplus y) \oplus f_2(x \oplus y)\big)h(x \oplus y) \oplus f_2(x \oplus y) \oplus h(x \oplus y) \in \mathcal{B}_n$, где $y \in \mathbb{Z}_2^n$;
   г) $\varphi_4(x) = \big(f_1(x \oplus y) \oplus f_2(x \oplus y)\big)h(x \oplus y) \oplus f_1(x \oplus y) \oplus h(x \oplus y) \in \mathcal{B}_n$, где $y \in \mathbb{Z}_2^n$;
   д) $\widetilde{\varphi_1} \oplus \widetilde{\varphi_2} \oplus \widetilde{\varphi_3} \oplus \widetilde{\varphi_4} \equiv 0$.

Доказательство. Пусть $y \in \mathbb{Z}_2^n$. Тогда $g \in \mathcal{F}_{n+2}$, такая, что

$$g(x, x_{n+1}, x_{n+2}) = \big(h(x) \oplus h(x \oplus y)\big)x_{n+1} \oplus h(x) \oplus x_{n+2},$$

удовлетворяет условию утверждения 4 для направления $(y, 1, 0)$. Следовательно, булева функция от $n+2$ переменных

$$f(x, x_{n+1}, x_{n+2}) = \big((D_y g_1(x) \oplus 1)g(x, x_{n+1}, x_{n+2}) \oplus D_y g_2(x)\big)x_{n+1} \oplus \\ \oplus g_1(x)g(x, x_{n+1}, x_{n+2}) \oplus g_2(x),$$

где $g_1 = f_1 \oplus f_2$ и $g_2 = f_2$, является бент-функцией от $n+2$ переменных. Из леммы 5 следует, что $\deg(f) \leqslant (n+2)/2$ и $\deg(g_1 h) = \deg((f_1 \oplus f_2)h) \leqslant (n+2)/2$.

Легко убедиться в том, что

$$f(x, 0, 0) = g_1(x)h(x) \oplus g_2(x) = \varphi_2(x),$$
$$f(x, 0, 1) = g_1(x)h(x) \oplus g_1(x) \oplus g_2(x) = \varphi_1(x),$$
$$f(x, 1, 0) = g_1(x \oplus y)h(x \oplus y) \oplus g_2(x \oplus y) \oplus h(x \oplus y) = \varphi_3(x),$$
$$f(x, 1, 1) = g_1(x \oplus y)h(x \oplus y) \oplus g_1(x \oplus y) \oplus g_2(x \oplus y) \oplus h(x \oplus y) \oplus 1 = \varphi_4(x) \oplus 1.$$

Тогда из утверждений 5 и 6 следует второе утверждение теоремы. ■

Следствие 1. Пусть $h, g \in \mathcal{F}_n$ и $\deg(hg) > (n+2)/2$. Тогда если $f_1, f_2 \in \mathcal{B}_n$ и $h \equiv f_1 \oplus f_2$, то хотя бы одна из функций $g \oplus f_1$ или $g \oplus f_2$ не является бент-функцией.

Следствие 2. Пусть $h \in \mathcal{F}_n$ и $f_1, f_2, h \oplus f_1, h \oplus f_2 \in \mathcal{B}_n$. Тогда если $(f_1 \oplus f_2)h \equiv 0$ или $(f_1 \oplus f_2)h \equiv h$, при этом $f_3(x) = h(x \oplus y) \oplus f_1(x \oplus y)$ и $f_4(x) = h(x \oplus y) \oplus f_2(x \oplus y)$, где $y \in \mathbb{Z}_2^n$, то справедливо, что $\widetilde{f_1} \oplus \widetilde{f_2} \equiv \widetilde{f_3} \oplus \widetilde{f_4}$.

В обозначениях теоремы 4 приведём пример того, как верхняя оценка степени функции $(f_1 \oplus f_2)h$ может быть использована для описания бент-функций f_1 и f_2.

Пусть $h(x) = x_1 x_2$ — булева функция от четырёх переменных, $x \in \mathbb{Z}_2^4$, и $f_1, f_2 \in \mathcal{B}_4$ такие, что $h \oplus f_1$ и $h \oplus f_2$ являются бент-функциями. Положим, что АНФ функции f_1 содержит моном $x_3 x_4$, а АНФ функции f_2 его не содержит. Тогда $\deg((f_1 \oplus f_2)h) = 4 > 3 = (n+2)/2$. Таким образом, либо каждая бент-функция из разложения функции $h(x)$ в сумму двух бент-функций имеет моном $x_3 x_4$ в своей АНФ, либо каждая из них его не имеет. Пример достижения оценки можно получить для следующих функций от четырёх переменных: $h(x) = x_3$, $f_1(x) = x_1 x_2 \oplus x_3 x_4$ и $f_2(x) = x_1 x_3 \oplus x_2 x_4$.

Заключение

Работа посвящена вопросу построения уравновешенных булевых функций с высокими значениями нелинейности. Приведена итеративная конструкция уравновешенных функций от чётного числа переменных, с помощью которой получена булева функция от 18 переменных со значением нелинейности 130 732. Эта функция может быть

использована для итеративного построения уравновешенных функций от чётного числа $n \geqslant 20$ переменных с нелинейностью $2^{n-1} - (2^{n/2-1} + 2^{n/2-3} + 2^{n/2-5} + 2^{n/2-7})$.

Приведены достаточные условия того, что функции, полученные с помощью конструкции, обладают такими свойствами, как отсутствие линейных структур и корреляционная иммунность. Интерес представляет также изучение дополнительных условий, при которых получаемые функции будут, например, удовлетворять строгому лавинному критерию (SAC) или критерию распространения $\mathrm{PC}(k)$ порядка k.

ЛИТЕРАТУРА

1. *Matsui M.* Linear cryptanalysis method for DES cipher // LNCS. 1994. V. 765. P. 386–397.
2. *Rothaus O. S.* On bent functions // J. Comb. Theory. Ser. A. 1976. V. 20. No. 3. P. 300–305.
3. *Adams C. M.* Constructing symmetric ciphers using the CAST design procedure // Des. Codes Cryptogr. 1997. V. 12. No. 3. P. 283–316.
4. *Hell C., Johansson T., Maximov A., and Meier W.* A stream cipher proposal: Grain-128 // IEEE Intern. Symp. Inform. Theory. Seattle, WA, USA, 2006. P. 1614–1618.
5. *Zheng Y., Pieprzyk J., and Seberry J.* Haval — a one-way hashing algorithm with variable length of output (extended abstract) // LNCS. 1993. V. 718. P. 83–104.
6. *Helleseth T. and Kholosha A.* Bent functions and their connections to combinatorics / S. R. Blackburn, S. Gerke, and M. Wildon (eds.). Surveys in Combinatorics. London Math. Soc. Lecture Note Ser. 2013. V. 409. Cambridge: Cambridge University Press, 2013. P. 91–126.
7. *Tokareva N.* Bent Functions: Results and Applications to Cryptography. London: Acad. Press, 2015.
8. *Токарева Н. Н.* О множестве производных булевой бент-функции // Прикладная дискретная математика. Приложение. 2016. № 9. С. 327–350.
9. *McFarland R. L.* A family of difference sets in non-cyclic groups // J. Combin. Theory. Ser. A. 1973. V. 15. P. 1–10.
10. *Dillon J. F.* Elementary Hadamard Difference Sets. PhD. Thesis. Univ. of Maryland, 1974.
11. *Логачев О. А., Сальников А. А., Смышляев С. В., Ященко В. В.* Булевы функции в теории кодирования и криптологии. М.: МЦНМО, 2012.
12. *Siegentaler T.* Correlation-immunity of nonlinear combining functions for cryptographic applications // IEEE Trans. Inform. Theory. 1984. V. 30. No. 5. P. 776–780.
13. *Seberry J., Zhang X-M., and Zheng Y.* Nonlinearly balanced Boolean functions and their propagation characteristics // LNCS. 1994. V. 773. P. 49–60.
14. *Dobbertin H.* Construction of bent functions and balanced Boolean functions with high nonlinearity // LNCS. 1994. V. 1008. P. 61–74.
15. *Dobbertin H. and Leander G.* Cryptographer's Toolkit for Construction of 8-bit Bent Functions. Cryptology ePrint Archive. Report 2005/089. 2005.
16. *Hu X., Yang B., and Huang M.* A construction of highly nonlinear Boolean functions with optimal algebraic immunity and low hardware implementation cost // Discr. Appl. Math. 2020. V. 285. P. 407–422.
17. *Carlet C., Djurasevic M., Jakobovic D., et al.* Evolving Constructions for Balanced, Highly Nonlinear Boolean Functions. `https://arxiv.org/abs/2202.08743`. 2022.
18. *Gini A. and Meaux P.* Weightwise perfectly balanced functions and nonlinearity // LNCS. 2023. V. 13874. P. 386–397.
19. *Shaporenko A.* Derivatives of bent functions in connection with the bent sum decomposition problem // Des. Codes Cryptogr. 2023. V. 91. P. 1607–1625.
20. *Tokareva N. N.* On the number of bent functions from iterative constructions: lower bounds and hypotheses // Adv. Math. Commun. 2011. V. 5. No. 4. P. 609–621.

21. *Canteaut A. and Charpin P.* Decomposing bent functions // IEEE Trans. Inform. Theory. 2003. V. 49. No. 8. P. 2004–2019.

REFERENCES

1. *Matsui M.* Linear cryptanalysis method for DES cipher. LNCS, 1994, vol. 765, pp. 386–397.
2. *Rothaus O. S.* On bent functions. J. Comb. Theory. Ser. A, 1976, vol. 20, no. 3, pp. 300–305.
3. *Adams C. M.* Constructing symmetric ciphers using the CAST design procedure. Des. Codes Cryptogr., 1997, vol. 12, no. 3, pp. 283–316.
4. *Hell C., Johansson T., Maximov A., and Meier W.* A stream cipher proposal: Grain-128. IEEE Intern. Symp. Inform. Theory, Seattle, WA, USA, 2006, pp. 1614–1618.
5. *Zheng Y., Pieprzyk J., and Seberry J.* Haval — a one-way hashing algorithm with variable length of output (extended abstract). LNCS, 1993, vol. 718, pp. 83–104.
6. *Helleseth T. and Kholosha A.* Bent functions and their connections to combinatorics. S. R. Blackburn, S. Gerke, and M. Wildon (eds.). Surveys in Combinatorics. London Math. Soc. Lecture Note Ser., 2013, vol. 409, Cambridge, Cambridge University Press, 2013, pp. 91–126.
7. *Tokareva N.* Bent Functions: Results and Applications to Cryptography. London, Acad. Press, 2015.
8. *Tokareva N. N.* O mnozhestve proizvodnykh bulevoy bent-funktsii [On the set of derivatives of a Boolean bent function]. Prikladnaya diskretnaya matematika. Prilozhenie, 2016, no. 9, pp. 327–350. (in Russian)
9. *McFarland R. L.* A family of difference sets in non-cyclic groups. J. Combin. Theory, Ser. A, 1973, vol. 15, pp. 1–10.
10. *Dillon J. F.* Elementary Hadamard Difference Sets. PhD. Thesis, Univ. of Maryland, 1974.
11. *Logachev O. A., Salnikov A. A., and Yashchenko V. V.* Boolean Functions in Coding Theory and Cryptography. AMS, 2012. 334 p.
12. *Siegentaler T.* Correlation-immunity of nonlinear combining functions for cryptographic applications. IEEE Trans. Inform. Theory, 1984, vol. 30, no. 5. pp. 776–780.
13. *Seberry J., Zhang X-M., and Zheng Y.* Nonlinearly balanced boolean functions and their propagation characteristics. LNCS, 1994, vol. 773, pp. 49–60.
14. *Dobbertin H.* Construction of bent functions and balanced Boolean functions with high nonlinearity. LNCS, 1994, vol. 1008, pp. 61–74.
15. *Dobbertin H. and Leander G.* Cryptographer's Toolkit for Construction of 8-bit Bent Functions. Cryptology ePrint Archive, report 2005/089, 2005.
16. *Hu X., Yang B., and Huang M.* A construction of highly nonlinear Boolean functions with optimal algebraic immunity and low hardware implementation cost. Discrete Appl. Math., 2020, vol. 285, pp. 407–422.
17. *Carlet C., Djurasevic M., Jakobovic D., et al.* Evolving Constructions for Balanced, Highly Nonlinear Boolean Functions. `https://arxiv.org/abs/2202.08743`. 2022.
18. *Gini A. and Meaux P.* Weightwise perfectly balanced functions and nonlinearity. LNCS, 2023, vol. 13874, pp. 386–397.
19. *Shaporenko A.* Derivatives of bent functions in connection with the bent sum decomposition problem. Des. Codes Cryptogr., 2023, vol. 91, pp. 1607–1625.
20. *Tokareva N. N.* On the number of bent functions from iterative constructions: lower bounds and hypotheses. Adv. Math. Commun., 2011, vol. 5, no. 4, pp. 609–621.
21. *Canteaut A. and Charpin P.* Decomposing bent functions. IEEE Trans. Inform. Theory, 2003, vol. 49, no. 8, pp. 2004–2019.

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 519.7 DOI 10.17223/20710410/63/2

КЛЮЧЕВОЙ КРИПТОАЛГОРИТМ ПО СХЕМЕ «СЭНДВИЧ» НА ОСНОВЕ ХЕШ-ФУНКЦИИ «СТРИБОГ»

В. А. Кирюхин*,**, А. М. Сергеев*

**ООО «СФБ Лаб», г. Москва, Россия*
***АО «ИнфоТеКС», г. Москва, Россия*

E-mail: vitaly.kiryukhin@sfblaboratory.ru, andrey.sergeev@sfblaboratory.ru

Предложен способ преобразования хеш-функции «Стрибог» в ключевой криптоалгоритм, условно называемый Стрибог-С («сэндвич» — ключ в начале и ключ в конце) и не требующий изменений в самой хеш-функции. Особенности криптоалгоритма упрощают реализацию мер защиты от атак по побочным каналам. Доказано, что Стрибог-С, а также HMAC-Стрибог и Стрибог-К являются стойкими псевдослучайными функциями (PRF) и алгоритмами имитозащиты (MAC) в условиях, когда (почти все) их внутренние состояния становятся известными противнику. От функции сжатия, итеративно применяемой внутри хеш-функции, в таких условиях требуются дополнительные свойства — стойкость к атакам на построение коллизий и прообраза.

Ключевые слова: *Стрибог, PRF, HMAC, доказуемая стойкость.*

"SANDWICH"-LIKE KEYED ALGORITHM BASED ON THE "STREEBOG" HASH FUNCTION

V. A. Kiryukhin*,**, A. M. Sergeev*

**LLC "SFB Lab", Moscow, Russia*
***JSC "InfoTeCS", Moscow, Russia*

We propose a keyed cryptographic algorithm based on the "Streebog" hash function. We do not make any structural changes to the hash function itself, but only introduce a special type of padding. As a result, the key appears on both sides of the message in so-called "sandwich" manner — hence the name Streebog-S for our construction. "Sandwich" properties make it possible to simplify defenses against side-channel attacks while maintaining their effectiveness. We prove that Streebog-S and other algorithms based on "Streebog", HMAC-Streebog and Streebog-K, remain secure as pseudorandom functions (PRF) and message authentication codes (MAC) even when almost all internal states are leaked to the adversary. This leakage resistance requires additional properties from the underlying compression function, namely collision- and preimage-resistance.

Keywords: *Streebog, PRF, HMAC, provable security.*

Введение

Российская бесключевая хеш-функция «Стрибог» (ГОСТ 34.11-2018) [1] основана на модифицированной схеме Меркла — Дамгарда (МД) [2, 3]. Хешируемое сообщение M дополняется специальным образом и разбивается на блоки по $n = 512$ бит, затем к n-битному состоянию хеш-функции h и блоку сообщения m итеративно применяется функция сжатия $\mathsf{g}(h, m) = h'$. Начальное состояние хеш-функции является предопределённой константой, последнее состояние — результат хеширования (хеш-значение).

Примечательной особенностью отечественной хеш-функции, отличающей её от оригинальной схемы МД, является использование контрольной суммы (КС). Последний хешируемый блок — сумма всех блоков сообщения M по модулю 2^n (операция «$\boxplus$»).

Указанный механизм играет важную роль, когда «Стрибог» (H) используется в качестве основы для *ключевых* криптоалгоритмов. Примерами таковых могут служить HMAC-Стрибог [4] (двойное хеширование) и Стрибог-К [5] (ключ K перед сообщением — $\mathsf{H}(K||M)$). Эти алгоритмы являются стойкими псевдослучайными функциями (PRF) [5, 6] и применяются для защиты целостности (имитозащиты), т. е. служат для выработки кодов аутентификации сообщений (MAC — Message Authentication Code).

Поясним влияние КС на ключевые схемы. Пусть секретный ключ K является одним из хешируемых блоков, тогда последним блоком (назовём его финализирующим ключом) будет $K \boxplus \sigma$. Значение σ может выбираться противником, так как в типовой модели угроз у атакующего есть возможность выбора текста M. Сообщения $M_1, \ldots, M_q$ дают КС $\sigma_1, \ldots, \sigma_q$ и соответственно *связанные* ключи $K \boxplus \sigma_1, \ldots, K \boxplus \sigma_q$.

Первое следствие этого — от функции сжатия g требуется [5, 6] быть стойкой PRF в условиях атак со связанными ключами (PRF-RKA). Конструктивные исследования [7, 8] показывают, что g соответствует этим требованиям — на текущий момент не найдено атак лучше универсальных. Проблема в том, что в общем случае вероятность успеха универсального метода (тотального опробования) при r связанных ключах в r раз больше, чем при их отсутствии. До определённого объёма обрабатываемых данных (нагрузки на ключ), как показано в [6], специфика хеш-функции позволяет уйти от указанной проблемы, но значительное превышение этой границы приводит к эффективным атакам [9, 5]. Известны методы определения секретного ключа для HMAC-Стрибог со сложностью порядка $2^{4n/5}$ по времени и данным [9], аналогично и для Стрибог-К [5].

Второе следствие носит позитивный характер. Финализацию нельзя осуществить без ключа K (пусть и связанного). Если в результате некоторой утечки, например по побочным каналам, противнику становится известно внутреннее состояние криптоалгоритма (состояние хеш-функции после обработки первых блоков некоторого сообщения), то из-за ключа K в последнем блоке противник не сможет непосредственно вычислить имитовставку (хеш-значение). Здесь и далее считаем, что КС не является частью состояния, а ключ K складывается с σ однократно при вычислении последней функции сжатия. При «наивной» реализации хэш-функции частью состояния является блок вида $K \boxplus m_1 \boxplus \ldots \boxplus m_j$ (сумма ключа и первых j блоков сообщения), утечка которого немедленно приводит к раскрытию ключа.

Третье следствие — для защиты от атак по побочным каналам часто используется *маскирование*. В таких условиях попеременное использование операций «$\boxplus$» (при вычислении КС) и «$\oplus$» (сложение по модулю 2 внутри самой функции сжатия) приводит к дополнительным накладным расходам, требует применения специальных алгоритмов [10, 11], что снижает скорость работы и усложняет реализацию.

В настоящей работе предлагается простой способ построения ключевого криптоалгоритма, сохраняющий положительные свойства контрольной суммы и устраняющий отрицательные. К сообщению M приписывается специальный блок C так, чтобы КС от их конкатенации $M||C$ была равна нулю. При хешировании $\mathsf{H}(K||M||C)$ ключ K будет и первым блоком (в силу явного расположения), и последним (из-за КС). В саму хеш-функцию, как видно, не вносится никаких изменений. Получившийся криптоалгоритм по своей идее схож со схемой «Сэндвич-MAC» [12], отсюда условное наименование «Стрибог-С(эндвич)». Значение C по сути является «альтернативной» КС, не зависит от ключа, может вычисляться по ходу хеширования сообщения.

Стойкость схемы «сэндвич» требует более слабых предположений о функции сжатия (PRF вместо PRF-RKA), чем требуются для Стрибог-К и для HMAC-Стрибог. Эвристические оценки преобладания противника в задаче различения «криптоалгоритм или случайная функция» (а равно, оценки вероятности навязывания [13]) у трёх упомянутых криптоалгоритмов почти одинаковы.

Отсутствие связанных ключей позволяет доказать, что даже при утечке внутреннего состояния (но без утечки дополнительных сведений о ключе[1]) единственным эффективным способом определения секретного ключа для Стрибог-С является тотальное опробование (в предположении, что ключ функции сжатия также нельзя определить эффективнее тотального опробования). Для Стрибог-К и HMAC-Стрибог аналогичное утверждение верно только при длине ключа $k \leqslant n/2 = 256$ бит. При $k \leqslant n$ и q выбранных сообщениях определение ключа в таких условиях потребует порядка $(2^k/q + 2^{n/2})$ операций [5].

При раскрытии внутреннего состояния три рассматриваемых криптоалгоритма остаются стойкими PRF и стойкими к атакам на ключ (модель KR — Key Recovery). В настоящей работе описываются соответствующие формальные модели PRF-LEAK и KR-LEAK. От функции сжатия в таких условиях дополнительно требуется стойкость к атакам на построение коллизий (модель CR — Collision Resistance) и прообраза к зафиксированному значению (модель TPR — Target Preimage Resistance). Отдельное рассмотрение моделей KR (и KR-LEAK) обусловлено возможностью получения более точных оценок.

Реализация маскирования для Стрибог-С упрощается — последним обрабатываемым блоком является ключ K, а не сумма $K \boxplus \sigma$. При этом, в силу стойкости к утечке состояния, защищать необходимо только последний вызов функции сжатия, а точнее, используемый в ней LPSX-шифр. Соответствующие способы хорошо известны и рассматриваются, например, в [14–16].

Изложение результатов построено следующим образом: в п. 1 вводятся обозначения и приводятся общие сведения о математическом аппарате теории доказуемой стойкости. Пункт 2 содержит описание хеш-функции «Стрибог» и ключевых криптоалгоритмов, построенных на её основе. В п. 3 даётся формальное описание вычислительно сложных базовых задач, к которым сводится стойкость анализируемых алгоритмов. Пункт 4 посвящён описанию схемы «сэндвич», её основным эксплуатационным и криптографическим свойствам. Пункты 5 и 6 содержат соответственно описание моделей угроз PRF-LEAK и KR-LEAK. Для рассматриваемых криптоалгоритмов приводятся доказательства стойкости и соответствующие эвристические оценки. В п. 7 с учётом результатов анализа в моделях PRF-LEAK и KR-LEAK обсуждаются общие подходы

[1] Дополнительные сведения о ключе, например из побочных каналов, могут позволить действовать эффективнее тотального опробования.

к обеспечению защиты от атак по побочным каналам. Показывается, какие сведения о процессе вычислений должны быть защищены от утечек, а какие могут быть раскрыты противнику.

1. Обозначения и общие сведения

Обозначим:

— n, k, τ — битовый размер состояния/блока, ключа и выхода соответственно ($n = 512$, $k \leqslant n$, $\tau \leqslant n$);
— V^n — множество всех n-битных строк (элементы V^n могут естественным образом интерпретироваться как целые числа и наоборот);
— $V^{<2^n}$ — множество битовых строк длины менее 2^n;
— 0^u — строка из u нулевых бит;
— $||$ — конкатенация битовых строк;
— $\mathsf{msb}_u(X)$ и $\mathsf{lsb}_u(X)$ — усечение строки $X \in V^n$ до u старших и младших бит соответственно;
— $\oplus$ — сложение по модулю 2;
— $\boxplus$ и $\boxminus$ — сложение и вычитание по модулю 2^n соответственно;
— $\mathrm{Func}(\mathbf{X}, \mathbf{Y})$ — множество всех функций, отображающих конечное множество $\mathbf{X}$ в конечное множество $\mathbf{Y}$;
— $X \xleftarrow{\mathrm{R}} \mathbf{X}$ — случайный и равновероятный выбор элемента X из множества $\mathbf{X}$;
— $\mathsf{F} : \mathbf{X} \to \mathbf{Y}$ — детерминированный алгоритм, отображение из множества входов $\mathbf{X}$ в множество выходов $\mathbf{Y}$, $\mathsf{F} \in \mathrm{Func}(\mathbf{X}, \mathbf{Y})$.

Под противником будем понимать интерактивный вероятностный алгоритм $\mathcal{A}$, взаимодействующий с другими алгоритмами (оракулами) [17]. В рамках модели угроз TM для криптоалгоритма Alg количественную характеристику успешности противника $\mathcal{A}$ обозначаем $\mathrm{Adv}^{TM}_{\mathsf{Alg}}(\mathcal{A})$. В зависимости от модели TM значение $\mathrm{Adv}^{TM}_{\mathsf{Alg}}$ может определяться как преобладание в задаче различения реальной криптосхемы от идеальной или как вероятность реализации угрозы (пример — восстановление ключа). Вероятностное пространство каждой рассматриваемой модели угроз определяется равновероятным выбором заполнения случайной ленты у вероятностного алгоритма $\mathcal{A}$, ключа и «идеальных алгоритмов», соответствующий выбор обозначается символом «$\xleftarrow{\mathrm{R}}$».

Максимум значения $\mathrm{Adv}^{TM}_{\mathsf{Alg}}(\mathcal{A})$ среди противников, возможности которых ограничены вычислительно (числом операций t в некоторой модели вычислений) и информационно (числом запросов/ответов к оракулам и другими определяемыми моделью TM и алгоритмом Alg параметрами $prms$) обозначаем $\mathrm{Adv}^{TM}_{\mathsf{Alg}}(t, prms)$.

Результатом вычислений $\mathcal{A}$ после взаимодействия с оракулом $\mathcal{O}$ является значение x, обозначаем это $\mathcal{A}^{\mathcal{O}} \Rightarrow x$. Для моделей угроз, в которых целью противника является различение «реального» алгоритма от «идеального», результат работы $\mathcal{A}$ — бит $b \in \{0, 1\}$, где «реальному» алгоритму соответствует 1 ($\mathcal{A}^{\mathcal{O}} \Rightarrow 1$), а «идеальному» — 0 ($\mathcal{A}^{\mathcal{O}} \Rightarrow 0$). Противник $\mathcal{A}$ может формировать запросы к оракулу адаптивным образом, i-й запрос к оракулу может зависеть от ответа на запросы с номерами 1, 2, …, $i-1$, $1 \leqslant i \leqslant q$ (осуществляется атака с адаптивно выбираемыми сообщениями). Содержимое запроса полностью определяется противником, ограничение задаётся только на максимальную длину запроса. Без потери общности полагаем, что противник $\mathcal{A}$ всегда использует максимально возможное число запросов и среди них нет совпадающих (нет «бессмысленных действий»). Считаем, что размер описания алгоритма $\mathcal{A}$ (его ис-

ходного кода) ограничен некоторым малым значением, что позволяет исключить из рассмотрения отдельные атаки, основанные на «бесплатных» предвычислениях [18].

Неформально называем Alg стойким в модели угроз TM (TM-стойким), если $\mathrm{Adv}_{\mathsf{Alg}}^{TM}(t, prms) \leqslant \varepsilon$, где ε не превосходит некоторого малого значения, определяемого требованиями к стойкости криптосистемы, а ресурсы t и $prms$ сопоставимы с доступными противнику на практике.

Символ «$\lessapprox$» используем с целью демонстрации практической значимости результатов, подразумевая под ним: «меньше или равно, если соответствующие эвристические предположения истинны».

Приведём здесь определения часто упоминаемых моделей угроз, остальные дадим по тексту работы.

Определение 1. Преобладанием противника $\mathcal{A}$ в модели PRF (PRF-CMA — неотличимость от случайной функции при атаке с выбранными сообщениями) для ключевой функции $\mathsf{F} : \mathbf{K} \times \mathbf{X} \to \mathbf{Y}$ назовём

$$\mathrm{Adv}_{\mathsf{F}}^{\mathrm{PRF}}(\mathcal{A}) = \Pr\Big[K \xleftarrow{\mathrm{R}} \mathbf{K} : \mathcal{A}^{\mathsf{F}_K} \Rightarrow 1\Big] - \Pr\Big[\mathsf{R} \xleftarrow{\mathrm{R}} \mathrm{Func}(\mathbf{X}, \mathbf{Y}) : \mathcal{A}^{\mathsf{R}} \Rightarrow 1\Big].$$

Противник $\mathcal{A}$ делает к оракулу q запросов. Если $\mathbf{X} = V^{<2^n}$, то в ресурсы $\mathcal{A}$ включается l — максимальная длина запроса (в n-битных блоках).

Определение 2. Характеристикой успешности противника $\mathcal{A}$ в модели KR (KR-CPA, Key Recovery — восстановление ключа в условиях атаки с выбранными открытыми текстами) для ключевого алгоритма $\mathsf{F} : \mathbf{K} \times \mathbf{X} \to \mathbf{Y}$ назовём вероятность восстановления ключа

$$\mathrm{Adv}_{\mathsf{F}}^{\mathrm{KR}}(\mathcal{A}) = \Pr\Big[K \xleftarrow{\mathrm{R}} \mathbf{K} : \mathcal{A}^{\mathsf{F}_K(\cdot)} \Rightarrow K',\, K = K'\Big].$$

Ресурсы противника определяются как в модели PRF.

Хорошо известна связь этих моделей [17]:

$$\mathrm{Adv}_{\mathsf{F}}^{\mathrm{KR}}(t, q) \leqslant \mathrm{Adv}_{\mathsf{F}}^{\mathrm{PRF}}(t', q + u) + |\mathbf{Y}|^{-u}, \quad t' = t + q + u,$$

т. е. преобладание в задаче различения является верхней оценкой вероятности восстановления ключа (здесь и далее считаем, что вычислительная сложность передачи одного блока между противником и оракулом не превосходит сложности вычисления алгоритма F).

2. Хеш-функция «Стрибог»

Приведём описание хеш-функции «Стрибог» [1], пользуясь часто применяемым эквивалентным представлением [19], подробности можно найти также в [5].

Хешируемое двоичное сообщение $M \in V^{<2^n}$ дополняется битовой строкой $10\ldots0$, чтобы длина текста была кратна n: $M' = M||10\ldots0$. Дополнение выполняется, даже если длина M кратна n.

Далее M' разбивается на $(l+1)$ блоков по $n = 512$ бит, $M' = m_0||m_1||m_2||\ldots||m_l$. С помощью функции сжатия $\mathsf{g} : V^n \times V^n \to V^n$ выполняется их последовательная обработка:

$$h_{i+1} = \mathsf{g}(h_i, m_i) \oplus \Delta_i,\ 0 \leqslant i \leqslant l-1,$$
$$h_{l+1} = \mathsf{g}(h_l, m_l) \oplus \widetilde{\Delta}_l.$$

Здесь $\Delta_i, \widetilde{\Delta}_l \in V^n$ — некоторые константы, при этом $\Delta_i \neq \widetilde{\Delta}_l$, $i = 0, \ldots, l-1$. Начальное состояние хеш-функции $h_0 = IV_\tau \in V^n$ зависит от длины выхода $\tau \in \{256, 512\}$.

После обработки блоков из M' выполняется финализация, обрабатывается блок L (битовая длина сообщения M) и блок контрольной суммы $\Sigma = \mathsf{sum}_{\boxplus}(M) = m_0 \boxplus \ldots \boxplus m_l$:

$$H = \mathsf{g}(\mathsf{g}(h_{l+1}, L), \Sigma).$$

Результатом хеширования является $\mathsf{H}_\tau(M) = \mathsf{msb}_\tau(H)$, при $\tau = 512$ усечение фактически не выполняется. Если длина выхода не играет роли, то соответствующий индекс в обозначениях опускаем.

Функция сжатия реализована с помощью 12-раундового блочного шифра LPSX-типа $\mathsf{E} : V^n \times V^n \to V^n$ с использованием конструкциии Миагучи — Пренеля:

$$\mathsf{g}(h_i, m_i) = \mathsf{E}(h_i, m_i) \oplus h_i \oplus m_i = h_{i+1}.$$

Блочный шифр E состоит из 12 полных раундов и одного усечённого (всего 13 раундовых ключей)

$$\mathsf{E}(h, m) = \mathsf{X}_{13}\,\mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_{12} \ldots \mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_2\mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_1(m),$$

а каждый раунд содержит четыре операции:

$\mathsf{X_j}$ — сложение блока по модулю 2 с j-м раундовым ключом $h^{(j)}$, $1 \leqslant j \leqslant 13$;

S — параллельное применение зафиксированной подстановки к каждому байту;

P — перестановка байт в блоке;

L — параллельное применение линейного преобразования к 64-битным подблокам.

Раундовые ключи формируются схожим образом:

$$h^{(1)} = \mathsf{LPS}(h),\ h^{(j+1)} = \mathsf{LPS}(h^{(j)} \oplus rc^{(j)}),\ 1 \leqslant j \leqslant 12.$$

Здесь $rc^{(j)} \in V^n$ — раундовые константы.

На основе хеш-функции «Стрибог» в [4] определено ключевое преобразование

$$\mathsf{HMAC}\text{-Стрибог-}\tau(K, M) = \mathsf{H}_\tau(\overline{K} \oplus \mathtt{opad} \,||\, \mathsf{H}_\tau(\overline{K} \oplus \mathtt{ipad} \,||\, M)),$$

где ключ $\overline{K} = K \,||\, 0^{n-k}$, $k \leqslant n$; $\mathtt{opad}$ и $\mathtt{ipad}$ — различные ненулевые константы.

Особенности хеш-функции позволяют построить ключевое преобразование более простым способом [5] за счёт однократного хеширования:

$$\text{Стрибог-K}(K, M) = \mathsf{H}(\overline{K} \,||\, M).$$

Для упрощения обозначений положим $\overline{K} = m_0$ и $M' = m_1||\ldots||m_l$ и определим каскадное преобразование следующим образом:

$$\mathsf{Csc}(K_{\mathsf{Csc}}, M) = \mathsf{g}(\ldots \mathsf{g}(\mathsf{g}(K_{\mathsf{Csc}} \oplus \Delta_0, m_1) \oplus \Delta_1, m_2) \ldots \oplus \widetilde{\Delta}_l, L).$$

Здесь каскадный ключ $K_{\mathsf{Csc}} \in V^n$. Полагаем, что входные данные $M \in V^{<2^n}$ дополняются в Csc строкой $10\ldots0$, а длина L увеличивается на n из-за приписывания ключа. Ключевая хеш-функция примет следующий вид (рис. 1):

$$\mathsf{H}(\overline{K}||M) = \mathsf{g}(\mathsf{Csc}(\mathsf{g}(IV, \overline{K}), M), \overline{K} \boxplus \sigma),$$
$$\sigma = \mathsf{sum}_{\boxplus}(M) = m_1 \boxplus m_2 \boxplus \ldots \boxplus m_l.$$

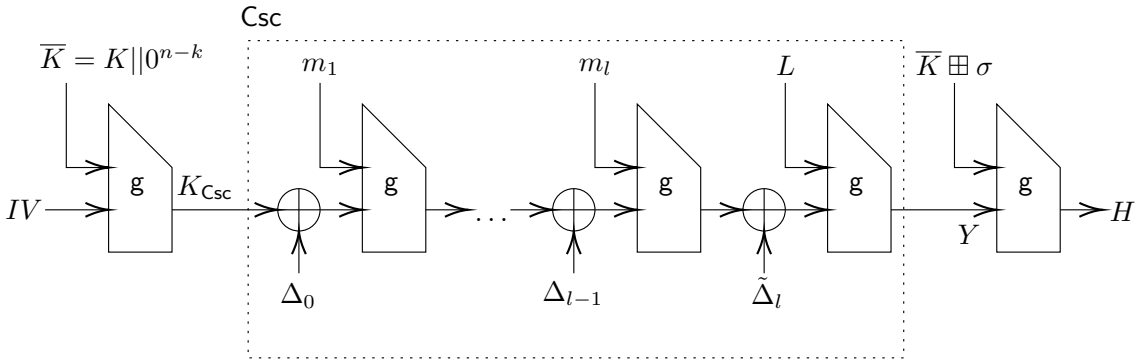

Рис. 1. Ключевая хеш-функция

Для краткости записи иногда сокращаем наименование Стрибог-K до KH, а HMAC-Стрибог — до HMAC; их областью определения считаем $V^k \times V^{<2^n} \to V^\tau$.

3. Базовые задачи

Стойкость рассматриваемых ключевых криптоалгоритмов сводится к ряду вычислительно сложных базовых задач, зависящих от функции сжатия g в соответствующих моделях угроз. Результаты конструктивного криптоанализа свидетельствуют, что g действительно является стойкой, а базовые задачи, следовательно, сложными. Приведём определения формальных моделей и дадим эвристические оценки преобладания противника, атакующего g.

Определение 3. Преобладанием противника $\mathcal{A}$ в модели PRF-RKA$_\circledast$ для ключевого криптоалгоритма $\mathsf{F} : \mathbf{K} \times \mathbf{X} \to \mathbf{Y}$ назовём

$$\mathrm{Adv}_{\mathsf{F}}^{\text{PRF-RKA}_\circledast}(\mathcal{A}) = \Pr\Big[K \xleftarrow{\mathrm{R}} \overline{\mathbf{K}} : \mathcal{A}^{\mathsf{F}_{K\circledast\cdot}(\cdot)} \Rightarrow 1\Big] -$$
$$- \Pr\Big[K \xleftarrow{\mathrm{R}} \overline{\mathbf{K}},\, \mathsf{R}_i \xleftarrow{\mathrm{R}} \mathrm{Func}(\mathbf{X}, \mathbf{Y}),\, \forall i \in \mathbf{K} : \mathcal{A}^{\mathsf{R}_{K\circledast\cdot}(\cdot)} \Rightarrow 1\Big],$$

где $\mathbf{K}, \mathbf{X}, \mathbf{Y}$ — множества ключей, входов и выходов соответственно; $\overline{\mathbf{K}} \subseteq \mathbf{K}$. Символом «$\circledast$» обозначена w-арная операция, определённая над $\mathbf{K}$ и являющаяся параметром модели. Запрос от $\mathcal{A}$ состоит из входа $x \in \mathbf{X}$ и «связи» $\kappa \in \mathbf{K}^{w-1}$. Ответом является значение $y = \mathsf{F}_{K\circledast\kappa}(x)$ (соответственно $y = \mathsf{R}_{K\circledast\kappa}(x)$). Противник $\mathcal{A}$ делает к оракулу q запросов, их содержимое ограничено по числу «связей» (r) и по максимальному числу различных «связей» (d), с которыми преобразуется одно и то же значение x $(d \leqslant r \leqslant q)$.

Определение 4. Характеристикой успешности противника $\mathcal{A}$ в модели KR-RKA$_\circledast$ для ключевого алгоритма $\mathsf{F} : \mathbf{K} \times \mathbf{X} \to \mathbf{Y}$ назовём вероятность восстановления ключа

$$\mathrm{Adv}_{\mathsf{F}}^{\text{KR-RKA}_\circledast}(\mathcal{A}) = \Pr\Big[K \xleftarrow{\mathrm{R}} \overline{\mathbf{K}} :\, \mathcal{A}^{\mathsf{F}_{K\circledast\cdot}(\cdot)} \Rightarrow K',\, K = K'\Big].$$

Параметры модели и ресурсы противника определяются как в PRF-RKA$_\circledast$.

Отметим, что при $\mathbf{K} = \overline{\mathbf{K}}$ и использовании в качестве «$\circledast$» унарного тождественного преобразования модель PRF-RKA$_\circledast$ эквивалентна модели PRF, а модель KR-RKA$_\circledast$ — модели KR. При одинаковых параметрах («$\circledast$» и $\overline{\mathbf{K}}$) модели PRF-RKA$_\circledast$ и KR-RKA$_\circledast$ соотносятся так же, как PRF и KR.

В качестве «$\circledast$» обычно используется бинарная операция $\circledast \in \{\boxplus, \oplus\}$. Анализ криптоалгоритма HMAC-Стрибог требует рассмотрения тернарной операции, определяемой композицией $\boxplus \circ \oplus$, в этом случае подразумеваем, что операндом для $\oplus$ служат

только два различных значения (`ipad` и `opad`). Множество $\overline{\mathbf{K}}$ введено исключительно из-за технического соображения — ключ может быть короче n-битного блока и в таком случае дополняется нулями.

Секретным ключом функции сжатия может выступать как блок сообщения, так и блок состояния, обозначаем это соответственно

$$\mathsf{g}^{\triangledown}_{\overline{K}}(\cdot) = \mathsf{g}(\cdot, \overline{K}),\ k \leqslant n,\ \overline{K} \in \overline{\mathbf{K}} = \{K \,||\, 0^{n-k} :\ K \in V^k\},$$
$$\mathsf{g}^{\triangleright}_{h}(\cdot) = \mathsf{g}(h, \cdot),\ h \in \overline{\mathbf{K}} = \mathbf{K} = V^n.$$

С учётом конструктивных результатов анализа [7, 8, 20] в работе [6] для $\mathsf{g}^{\triangledown}$ и $\mathsf{g}^{\triangleright}$ приводятся следующие эвристические оценки (по вероятности успеха универсальных атак):

$$\mathrm{Adv}^{\text{PRF-RKA}_{\boxplus}}_{\mathsf{g}^{\triangledown}}(t, q, r, d) \lessapprox \frac{t \cdot d}{2^k} \leqslant \frac{t \cdot r}{2^k} \leqslant \frac{t \cdot q}{2^k}; \tag{1}$$

$$\mathrm{Adv}^{\text{PRF-RKA}_{\oplus}}_{\mathsf{g}^{\triangleright}}(t, q, r = 2) \lessapprox \frac{2t}{2^n} + \frac{q(q-1)}{2^{n+1}}. \tag{2}$$

Оценку (1) используем в таком же виде для модели KR-RKA$_{\boxplus}$. Оценку для $\mathsf{g}^{\triangleright}$ в аналогичном случае можно несколько уточнить, так как второе слагаемое в (2) соответствует различителю по парадоксу дней рождения, а не атаке на ключ. Одной операцией (из t возможных) здесь и далее считается вычисление функции сжатия.

Предположение о стойкости $\mathsf{g}^{\triangleright}$ в модели PRF-RKA$_{\oplus}$ нужно для оценки каскадного преобразования [5]:

$$\mathrm{Adv}^{\text{PRF}}_{\mathsf{Csc}}(t, q, l) \leqslant q \cdot l' \cdot \mathrm{Adv}^{\text{PRF-RKA}_{\oplus}}_{\mathsf{g}^{\triangleright}}(t', q, r = 2), \quad t' = t + ql, \quad l' = l + 1.$$

В п. 5 показано, что анализируемые ключевые хеш-функции остаются PRF-стойкими даже при реализации угроз, связанных с утечкой внутреннего состояния криптоалгоритма (модель PRF-LEAK). От функции сжатия в таких условиях необходимо потребовать стойкости к атакам CR и TPR.

Определение 5. Характеристикой успешности противника $\mathcal{A}$ в модели CR для хеш-функции $\mathsf{F} : \mathbf{S} \times \mathbf{X} \to \mathbf{Y}$ назовём вероятность построения коллизии

$$\mathrm{Adv}^{\text{CR}}_{\mathsf{F}}(\mathcal{A}) = \Pr\Big[S \xleftarrow{\text{R}} \mathbf{S} :\ \mathcal{A}(S) \Rightarrow (M, M'),\ \mathsf{F}(S, M) = \mathsf{F}(S, M')\ \&\ M \neq M'\Big].$$

Определение 6. Характеристикой успешности противника $\mathcal{A}$ в модели TPR для хеш-функции $\mathsf{F} : \mathbf{S} \times \mathbf{X} \to \mathbf{Y}$ назовём вероятность построения прообраза

$$\mathrm{Adv}^{\text{TPR}}_{\mathsf{F}}(\mathcal{A}) = \Pr\Big[S \xleftarrow{\text{R}} \mathbf{S} :\ \mathcal{A}(S) \Rightarrow M,\ \mathsf{F}(S, M) \in \mathbf{T}\Big];$$

множество целевых образов $\mathbf{T}$ является параметром модели.

Причина введения множества $\mathbf{S}$ носит формальный характер [21]. Необходимо вывести из рассмотрения алгоритмы, в которых «зашита» найденная заранее пара сообщений (M, M'), порождающих коллизию. Такому алгоритму достаточно лишь предъявить эту пару. Указанный алгоритм, разумеется, существует, обладает малым описанием и требует пренебрежимо мало вычислительных ресурсов, но если смотреть с практической точки зрения, требует огромного объёма предварительно выполняемых операций для своего явного построения. Введение случайного выбора S делает подобные предвычисления бессмысленными. При этом реальные алгоритмы построения

коллизий обычно не используют существенным образом конкретное значение S и эффективны при любом выборе. Аналогичные соображения верны для модели TPR.

Для g множеством входов является $\mathbf{X} = (V^n \times V^n)$ — множество пар (блок состояния, блок сообщения). Множеством $\mathbf{S}$ можно условно считать все значения, которые могут принимать раундовые константы $rc^{(1)}, \ldots, rc^{(12)}$, $\mathsf{g} : \mathbf{S} \times (V^n \times V^n) \to V^n$.

С учётом отсутствия специфических методов построения коллизий в открытой литературе [22–26] (атакуется не более 9,5 из 12 полных раундов [25]), эвристическая оценка равна

$$\mathrm{Adv}^{\mathrm{CR}}_{\mathsf{g}}(t) \lessapprox \frac{t^2}{2^{n+1}} \tag{3}$$

— по вероятности успеха универсального метода — ρ-метода Полларда [27] (по парадоксу дней рождения).

Для полнораундовой функции сжатия не представлены нетривиальные методы построения прообраза [23, 24, 28–30] (атакуется не более 8,5 раундов [30]), поэтому эвристическая оценка вероятности успеха

$$\mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{g}}(t) \lessapprox \frac{t \cdot |\mathbf{T}|}{2^n} = \frac{t}{2^n} \tag{4}$$

даётся по методу полного перебора, в качестве $\mathbf{T}$ используется только одноэлементное множество целей (образов), $\mathbf{T} = \{IV\}$.

Для каскадного преобразования, которое здесь для формальности определяется как $\mathsf{Csc} : \mathbf{S} \times (V^n \times V^{<2^n}) \to V^n$, верны следующие неравенства:

$$\mathrm{Adv}^{\mathrm{CR}}_{\mathsf{Csc}}(t) \leqslant \mathrm{Adv}^{\mathrm{CR}}_{\mathsf{g}}(t'); \tag{5}$$

$$\mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{Csc}}(t) \leqslant \mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{g}}(t'), \quad t' = t + 1. \tag{6}$$

Первое является классическим результатом [2, 3]. Последний хешируемый блок в Csc содержит длину сообщения L (МД-усиление), а значит, коллизия каскада — это коллизия и для функции сжатия. Требование к g может быть ослаблено с CR-стойкости до стойкости к коллизиям специального вида «Constrained CR» [31], но для упрощения изложения ограничимся оценкой (5).

Неравенство (6) для модели TPR следует из простого наблюдения. Пусть тройка (S, K_{Csc}, M) такова, что $\mathsf{Csc}(S, (K_{\mathsf{Csc}}, M)) \in \mathbf{T}$, тогда $\mathsf{g}(S, (x, L)) \in \mathbf{T}$, где x — состояние перед последним вызовом функции сжатия, которое легко вычислить, зная вход Csc.

Везде далее подразумеваем случайный выбор S и множество $\mathbf{S}$, опуская их явное обозначение.

4. Схема «сэндвич»

Стрибог-C (обозначаем для краткости SH) дополняет хешируемый текст M так, чтобы в последнем хешируемом блоке был именно ключ $\overline{K} = K \,||\, 0^{n-k}$, а не сумма $\overline{K} \boxplus \sigma$. Таким образом, ключ $\overline{K}$ является и первым хешируемым блоком, и последним — отсюда наименование «сэндвич». Напомним, что у хэш-функции «Стрибог» двумя последними хэшируемыми блоками являются битовая длина и контрольная сумма хэшируемых данных (рис. 2).

Различные варианты схемы «сэндвич» были предложены в [12] для «простой» конструкции Меркла — Дамгарда (т. е. без контрольной суммы). Сами схемы анализировались только в модели PRF, доказательства опирались на результат [32], практическая

неприменимость которого показана в [33]. Отметим также, что в предложенных вариантах схемы [12] первый и последний хешируемые блоки содержали ключ, но были заведомо различны.

Для хеш-функции «Стрибог» схема «сэндвич» реализуется за счёт специального блока $C \in V^n$:

$$\mathsf{SH}(K, M) = \mathsf{H}(\overline{K} \,||\, M \,||\, C), \quad C = \mathsf{cs}(M).$$

Алгоритм $\mathsf{cs} : V^{<2^n} \to V^n$ не зависит от секретного ключа и может выполняться по ходу обработки сообщения M.

Определим необходимую «поправку» к контрольной сумме

$$\tilde{C} \boxplus \mathsf{sum}_{\boxplus}(M) = 0, \quad \tilde{C} = 0 \boxminus \mathsf{sum}_{\boxplus}(M) = 0 \boxminus m_1 \boxminus m_2 \boxminus \ldots \boxminus m_l,$$

при расчёте $\mathsf{sum}_{\boxplus}$ текст M дополняется p-битной строкой $10\ldots0$ $(1 \leqslant p \leqslant n)$.

Если $p = n$, то C располагается в одном n-битном блоке, $C = \tilde{C}$, а если $p < n$, то биты $\tilde{C}$ попадают в два блока — старшие (msb) в последний, а младшие (lsb) — в предпоследний (рис. 2):

$$C = C_{lsb} \,||\, C_{msb} = \mathsf{lsb}_p(\tilde{C}) || \mathsf{msb}_{n-p}(\tilde{C}),$$

случай $p = n$ также описывается этой формулой: $C = \mathsf{lsb}_n(\tilde{C}) = \tilde{C} = C_{msb} \,||\, C_{lsb}$.

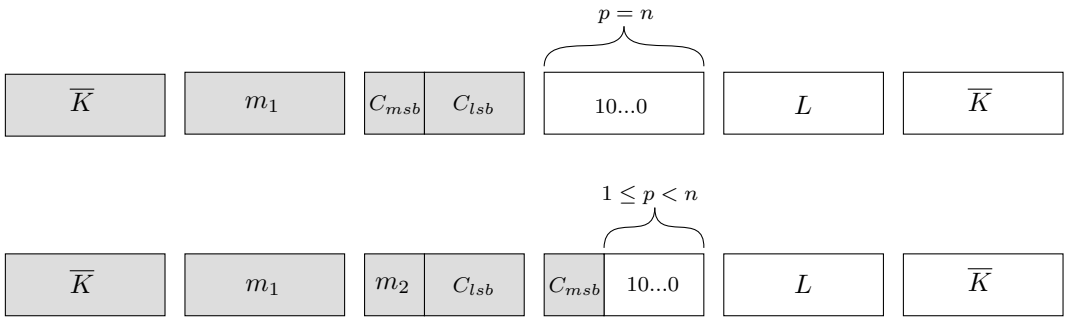

Рис. 2. Блоки на входе функции сжатия для случаев $p = n$ и $1 \leqslant p < n$. Серым выделены биты, которые являются входом хеш-функции, белым — биты, формируемые хеш-функцией. L – битовая длина входа хэш-функции

Формирование блока C не сказывается существенно на производительности криптоалгоритма. Вычисление КС в любом случае выполняется в рамках хеш-функции, Стрибог-С лишь заменяет сложение по модулю 2^n на вычитание — $\boxplus$ на $\boxminus$. Значение $\tilde{C}$ можно вычислить без непосредственного применения операции $\boxminus$ — за счёт побитовой инверсии (обозначаем «$\sim$») и сложения с единицей: $\tilde{C} = \sim \mathsf{sum}_{\boxplus}(M) \boxplus 1$.

Таким образом, на вычислительную эффективность существенно влияет лишь необходимость одного дополнительного обращения к функции сжатия. В таблице сравнивается число обращений к g для случая, когда длина сообщения M меньше n бит. Под предвычислениями понимается возможность заранее вычислить и хранить каскадный ключ $K_{\mathsf{Csc}} = \mathsf{g}(IV, \overline{K})$ (для HMAC-Стрибог соответственно два каскадных ключа, $K^I_{\mathsf{Csc}} = \mathsf{g}(IV, \overline{K} \oplus \mathtt{ipad})$ и $K^O_{\mathsf{Csc}} = \mathsf{g}(IV, \overline{K} \oplus \mathtt{opad})$).

Как можно видеть, «сэндвич» порождает меньше накладных расходов, чем двойное хеширование, но требует на один вызов g больше, чем Стрибог-К. Референсные реализации перечисленных в таблице криптоалгоритмов представлены в репозитории [34].

Минимальное число вызовов g, необходимое для обработки сообщения

Предвычисления	Стрибог-K	Стрибог-C	HMAC-Стрибог-256	HMAC-Стрибог-512
Нет	4	5	8	9
Есть	3	4	6	7

Необходимость использования «альтернативной» КС вызвана желанием оставить саму хеш-функцию «Стрибог» без каких-либо изменений, что, как представляется, позволяет использовать существующие реализации алгоритмов и в целом упрощает процесс внедрения. Безусловно, ключевую хеш-функцию по схеме «сэндвич» можно построить «с нуля» так, чтобы рассматриваемые криптографические свойства оставались такими же, а минимальное число обращений к g составляло 3 (ключ — блок с дополнением — ключ) и 2 (с предвычислениями).

Отметим, что аналогичный способ преобразования в схему «сэндвич» можно применить к хеш-функции ГОСТ Р 34.11-94 [35].

В [5, 6] показано, что Стрибог-K и HMAC-Стрибог являются PRF-стойкими, в частности, доказана

Теорема 1 [6]. Преобладание любого противника, атакующего Стрибог-K в модели PRF, ограничено следующим образом:

$$\mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{KH}}(t,q,l) \leqslant \mathrm{Adv}^{\text{PRF-RKA}_{\boxplus}}_{\mathsf{g}^{\triangledown}}(t',q',r,d) + \mathrm{Adv}^{PRF}_{\mathsf{Csc}}(t',q,l') + \frac{q^2+q}{2^{n+1}}. \tag{7}$$

Здесь $t' = t + ql$; $r = q' = q+1$; $l' = l+1$; $d = 1$.

Эвристическая оценка преобладания противника

$$\mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{KH}}(t',q,l) \lessapprox \frac{t'}{2^k} + \frac{2\cdot t'\cdot q\cdot l'}{2^n} + \frac{q^2+q}{2^n}, \quad t' \approx t, \quad l' = l+1, \tag{8}$$

получена в [6] в предположении об отсутствии для функции сжатия атак, которые были бы лучше универсальных. Оценка (8) является точной — каждому слагаемому оценки соответствует атака, действующая с сопоставимой вероятностью успеха: тотальное опробование ключа; атака за счёт деградации каскада; атака по парадоксу дней рождения.

PRF-стойкость схемы «сэндвич» является прямым следствием теоремы 1. В самом деле, из равенства

$$\mathsf{SH}(K,M) = \mathsf{KH}(K, M \,||\, \mathsf{cs}(M)) = \mathsf{H}(\overline{K} \,||\, M \,||\, C)$$

следует, что любой запрос к SH может быть выражен через запрос к KH. Пусть существует алгоритм $\mathcal{A}$, эффективно атакующий SH в модели PRF. Построим алгоритм $\mathcal{B}$, который будет атаковать KH с такой же эффективностью. На любой запрос M от $\mathcal{A}$, алгоритм $\mathcal{B}$ вычисляет $C = \mathsf{cs}(M)$, передаёт своему оракулу $\mathcal{O} \in \{\mathsf{KH}, \mathsf{R}\}$ запрос $M||C$, ответ оракула возвращает алгоритму $\mathcal{A}$. Результат работы алгоритма $\mathcal{B}$ равен результату работы $\mathcal{A}$. Алгоритм $\mathcal{B}$ идеально симулирует для $\mathcal{A}$ оракул SH или R, следовательно, в силу произвольности алгоритма $\mathcal{A}$ верно неравенство

$$\mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{SH}}(t,q,l) \leqslant \mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{KH}}(t',q,l+1), \quad t' = t + ql.$$

Кроме этого, у SH все блоки на входе функции $\mathsf{g}^{\triangledown}$ обрабатываются при ключе $\overline{K}$, а не при различных связанных ключах. Значение r снижается с $q+1$ до 1 — вместо модели $\text{PRF-RKA}_{\boxplus}$ фактически используется модель PRF.

Следствие 1. Преобладание любого противника, атакующего Стрибог-С в модели PRF, ограничено:

$$\mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{SH}}(t,q,l) \leqslant \mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{g}^{\triangledown}}(t',q') + \mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{Csc}}(t',q,l') + \frac{q^2+q}{2^{n+1}}. \tag{9}$$

Здесь $t' = t + ql$; $q' = q + 1$; $l' = l + 2$.

Оценка (8) является точной и для схемы «сэндвич», для которой могут быть применены три соответствующие атаки.

5. Стойкость при раскрытии состояния

Рассматриваемые ключевые криптоалгоритмы (SH, KH, HMAC) обладают важным и полезным на практике свойством — они остаются стойкими псевдослучайными функциями, даже если противник получает доступ к внутреннему состоянию хеш-функции. Сам ключ, разумеется, не считается частью состояния. Контрольная сумма не зависит от ключа и сама по себе известна противнику (также не является частью состояния). Сложение КС с ключом происходит в KH однократно при вычислении последней функции сжатия (в HMAC соответственно двукратно), а в SH не происходит вообще.

Определение 7. Преобладанием противника $\mathcal{A}$ в модели PRF-LEAK для ключевого криптоалгоритма $\mathsf{F} : \mathbf{K} \times \mathbf{X} \to \mathbf{Y}$ при утечке, определяемой функциями $\mathsf{leak}_1 : \mathbf{K} \to \mathbf{L}_1$ и $\mathsf{leak}_2 : \mathbf{K} \times \mathbf{X} \to \mathbf{L}_2$, назовём

$$\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{F},\mathsf{leak}_1,\mathsf{leak}_2}(\mathcal{A}) = \Pr\Big[K \xleftarrow{\mathrm{R}} \mathbf{K} : \mathcal{A}^{\mathcal{L}}(\mathsf{leak}_1(K)) \Rightarrow 1\Big] -$$
$$- \Pr\Big[\mathsf{R} \xleftarrow{\mathrm{R}} \mathrm{Func}(\mathbf{X}, \mathbf{Y} \times \mathbf{L}_2),\, lk \xleftarrow{\mathrm{R}} \mathbf{L}_1 : \mathcal{A}^{\mathsf{R}}(lk) \Rightarrow 1\Big].$$

Оракул $\mathcal{L}$ на запрос $X \in \mathbf{X}$ возвращает ответ $(\mathsf{F}(K,X), \mathsf{leak}_2(K,X))$, состоящий из результата работы криптоалгоритма F и утечки, зависящей от ключа K и запроса X.

Нетрудно видеть, что при любом виде утечки модель PRF-LEAK расширяет модель PRF. При $\mathbf{L}_1 = \mathbf{L}_2 = \varnothing$ модели PRF и PRF-LEAK эквивалентны.

Функции leak_1 и leak_2 являются детерминированными, хотя в общем случае вместо них можно использовать вероятностные алгоритмы. В пользу противника считаем тем самым, что он получает достоверную «незашумлённую» информацию.

Согласно модели, раскрываемые промежуточные состояния должны быть вычислительно неотличимы от случайных. Это достаточно строгое требование, которому тем не менее соответствуют анализируемые криптоалгоритмы.

5.1. Стрибог-С и Стрибог-К

Для Стрибог-С и Стрибог-К в качестве leak_1 выберем $\mathsf{g}^{\triangledown}_{K}(IV)$ — до начала взаимодействия с оракулом раскрывается каскадный ключ K_{Csc}. Противник, таким образом, может вычислить любое состояние хеш-функции, исключение составляет последний вызов функции сжатия. Значение $H = \mathsf{g}^{\triangledown}_{\overline{K} \boxplus \sigma}(Y)$ даже при известных $Y = \mathsf{Csc}(K_{\mathsf{Csc}}, M)$ и $\sigma = \mathsf{sum}_{\boxplus}(M)$ ($\sigma = 0$ для SH) вычислить без знания $\overline{K}$ нельзя. Функция leak_2 тривиальна, $\mathbf{L}_2 = \varnothing$, так как знание K_{Csc} уже даёт противнику дополнительные возможности, релевантные для нашего анализа.

Покажем, что информация о промежуточных состояниях, полученная противником, не увеличивает существенным образом его преобладание в задаче различения, а следовательно, вероятность навязывания подделки и вероятность восстановления ключа остаются малыми.

Теорема 2. Преобладание любого противника, атакующего Стрибог-K или Стрибог-C в модели PRF-LEAK, при утечке каскадного ключа $K_{\mathsf{Csc}} = \mathsf{g}^{\triangledown}_{\overline{K}}(IV)$ ограничено соответственно

$$\begin{gathered}\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{KH},K_{\mathsf{Csc}}}(t,q,l) \leqslant \mathrm{Adv}^{\text{PRF-RKA}_{\boxplus}}_{\mathsf{g}^{\triangledown}}(t',q',r,d) + \mathrm{Adv}^{\mathrm{CR}}_{\mathsf{g}}(t') + \mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{g}}(t'), \qquad (10)\\ \mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{SH},K_{\mathsf{Csc}}}(t,q,l) \leqslant \mathrm{Adv}^{\mathrm{PRF}}_{\mathsf{g}^{\triangledown}}(t',q') + \mathrm{Adv}^{\mathrm{CR}}_{\mathsf{g}}(t') + \mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{g}}(t'),\end{gathered}$$

где $t' = t + ql$; $r = q' = q + 1$; $d = 1$.

Доказательство. Докажем утверждение для KH, делая для SH оговорки по ходу изложения.

Назовём коллизией (C) совпадение любой пары элементов в ряду

$$IV, Y_1, Y_2, \ldots, Y_q, \text{ где } Y_i = \mathsf{Csc}(K_{\mathsf{Csc}}, M_i),\ 1 \leqslant i \leqslant q.$$

Противоположное событие обозначаем символом «$\overline{\mathrm{C}}$» . Событие C фактически означает, что противник смог создать прообраз к IV или непосредственно коллизию $Y_i = Y_j$, $i \neq j$. Коллизию, возникающую в результате взаимодействия противника $\mathcal{A}$ с KH, обозначаем $\mathcal{A}^{\mathsf{KH}} \Rightarrow (b, \mathrm{C})$. Значение бита $b \in \{0,1\}$ является результатом, возвращаемым противником, а коллизия — неявным «побочным эффектом» его вычислений и взаимодействий. Если символ b не указан в обозначениях, то подразумевается, что значение бита может быть любым, т. е. имеет место равенство

$$\Pr\bigl[\mathcal{A}^{\mathsf{KH}} \Rightarrow \mathrm{C}\bigr] = \Pr\bigl[\mathcal{A}^{\mathsf{KH}} \Rightarrow (1,\mathrm{C})\bigr] + \Pr\bigl[\mathcal{A}^{\mathsf{KH}} \Rightarrow (0,\mathrm{C})\bigr].$$

Представленные далее рассуждения повторяют ход доказательства теоремы о PRF-стойкости KH [6]. Главное отличие заключается в том, что при секретном каскадном ключе вероятность коллизии ограничивается за счёт PRF-стойкости каскада, а при утечке каскадного ключа — за счёт его стойкости в моделях CR и TPR.

Рассмотрим преобразование

$$\widetilde{\mathsf{KH}}(M_i) = \mathsf{f}_{\overline{K}\boxplus\sigma_i}(\mathsf{Csc}(\widetilde{K}_{\mathsf{Csc}}, M_i)),\ \widetilde{K}_{\mathsf{Csc}} = \mathsf{f}_{\overline{K}}(IV),\ \sigma_i = \mathsf{sum}_{\boxplus}(M_i),$$

полученное заменой в KH первого и последнего вызовов функции сжатия $\mathsf{g}^{\triangledown}$ на семейство из 2^n случайных функций f (для SH — на случайную функцию f). Если коллизии не происходит, то алгоритм $\widetilde{\mathsf{KH}}$ неотличим от случайной функции R, т. е.

$$\Pr\bigl[\mathcal{A}^{\mathsf{R}}(lk) \Rightarrow 1\bigr] = \Pr\Bigl[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow (1,\overline{\mathrm{C}})\Bigr],$$

благодаря тому, что независимо от σ_i запрашиваемые у f значения $IV, Y_1, \ldots, Y_q$ не повторяются, а следовательно, результаты запросов к f (они же — результаты запросов $\mathcal{A}$ к $\widetilde{\mathsf{KH}}$) есть последовательность случайных значений. Значение lk случайно и равновероятно выбирается из V^n.

Согласно определению модели PRF-LEAK,

$$\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{KH},K_{\mathsf{Csc}}}(\mathcal{A}) = \Pr\bigl[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow 1\bigr] - \Pr\bigl[\mathcal{A}^{\mathsf{R}}(lk) \Rightarrow 1\bigr].$$

По формуле полной вероятности

$$\Pr\bigl[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow 1\bigr] = \Pr\bigl[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow (1,\mathrm{C})\bigr] + \Pr\bigl[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow (1,\overline{\mathrm{C}})\bigr].$$

Группируя слагаемые и используя неравенство треугольника, получим

$$\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{KH},K_{\mathsf{Csc}}}(\mathcal{A}) \leqslant \Big(\Pr\big[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow (1,\overline{\mathrm{C}})\big] - \Pr\Big[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow (1,\overline{\mathrm{C}})\Big]\Big) +$$
$$+ \Big(\Pr\big[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow \mathrm{C}\big] - \Pr\Big[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow \mathrm{C}\Big]\Big) + \Pr\Big[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow \mathrm{C}\Big] = \epsilon + \epsilon_{\mathrm{C}} + p_{\mathrm{C}}.$$

Построим алгоритм $\mathcal{B}$, атакующий $\mathsf{g}^{\triangledown}$ в модели PRF-RKA$_{\boxplus}$ (для SH — в модели PRF), преобладание которого равно сумме ϵ и ϵ_{C} (полагаем, что каждое из этих значений неотрицательно). Алгоритм $\mathcal{B}$ делает запрос IV к оракулу $\mathcal{O} \in \{\mathsf{g}^{\triangledown}, \mathsf{f}\}$, получает каскадный ключ K_{Csc}, передаёт его алгоритму $\mathcal{A}$ (симулирует leak_1). При обработке запроса M_i от $\mathcal{A}$ алгоритм $\mathcal{B}$:

— самостоятельно вычисляет значения $\sigma_i = \mathsf{sum}_{\boxplus}(M_i)$ и $Y_i = \mathsf{Csc}(K_{\mathsf{Csc}}, M_i)$;
— проверяет условие коллизии: если $Y_i \in \{IV, Y_1, \ldots, Y_{i-1}\}$, то завершает работу $\mathcal{A}$ и возвращает 1;
— делает запрос (Y_i, σ_i) к оракулу $\mathcal{O} \in \{\mathsf{g}^{\triangledown}, \mathsf{f}\}$ и возвращает его ответ $\mathcal{A}$.

Если после q запросов от $\mathcal{A}$ коллизии не произошло, то результатом работы $\mathcal{B}$ является результат работы $\mathcal{A}$.

Для SH: $Y_i = \mathsf{Csc}(K_{\mathsf{Csc}}, M_i \,||\, \mathsf{cs}(M_i))$, а запрос к $\mathcal{O} \in \{\mathsf{g}^{\triangledown}, \mathsf{f}\}$ содержит только Y_i.

К оракулу выполняется не более $q' \leqslant q+1$ запросов, совокупное число связанных ключей такое же: $r' \leqslant q+1$. До возникновения коллизии первый аргумент в запросах не повторяется ($d = 1$), так как значения в ряду $IV, Y_1, \ldots, Y_i$ различны. Кроме того, если $\mathcal{O} = \mathsf{g}^{\triangledown}$, то $\mathcal{B}$ идеально симулирует для $\mathcal{A}$ оракул KH и утечку каскадного ключа $K_{\mathsf{Csc}} = \mathsf{g}^{\triangledown}_{\overline{K}}(IV)$. Аналогично, если $\mathcal{O} = \mathsf{f}$, то для $\mathcal{A}$ идеально симулируется оракул $\widetilde{\mathsf{KH}}$ и утечка $\widetilde{K}_{\mathsf{Csc}} = \mathsf{f}_{\overline{K}}(IV)$. Преобладание алгоритма $\mathcal{B}$ равно

$$\mathrm{Adv}^{\text{PRF-RKA}_{\boxplus}}_{\mathsf{g}^{\triangledown}}(\mathcal{B}) = \Pr\Big[\mathcal{B}^{\mathsf{g}^{\triangledown}} \Rightarrow 1\Big] - \Pr\big[\mathcal{B}^{\mathsf{f}} \Rightarrow 1\big] =$$
$$= \big(\Pr\big[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow (1,\overline{\mathrm{C}})\big] + \Pr\big[\mathcal{A}^{\mathsf{KH}}(K_{\mathsf{Csc}}) \Rightarrow \mathrm{C}\big]\big) -$$
$$- \Big(\Pr\Big[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow (1,\overline{\mathrm{C}})\Big] + \Pr\Big[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow \mathrm{C}\Big]\Big) = \epsilon + \epsilon_{\mathrm{C}}.$$

Для завершения доказательства осталось оценить значение вероятности

$$p_{\mathrm{C}} = \Pr\Big[\mathcal{A}^{\widetilde{\mathsf{KH}}}(\widetilde{K}_{\mathsf{Csc}}) \Rightarrow \mathrm{C}\Big] \leqslant \Pr[Y_i = Y_j,\ 1 \leqslant i < j \leqslant q] + \Pr[IV \in \{Y_1, \ldots, Y_q\}].$$

Построим алгоритм $\mathcal{B}_{\mathrm{CR}}$, пытающийся сформировать коллизию для Csc в модели CR. Он имитирует семейство случайных функций f (для SH — случайную функцию f), формирует случайное значение $\widetilde{K}_{\mathsf{Csc}} = \mathsf{f}_{\overline{K}\boxplus 0}(IV)$, передаёт $\widetilde{K}_{\mathsf{Csc}}$ алгоритму $\mathcal{A}$. На любой запрос M_i от $\mathcal{A}$ алгоритм $\mathcal{B}_{\mathrm{CR}}$: вычисляет $Y_i = \mathsf{Csc}(\widetilde{K}_{\mathsf{Csc}}, M_i)$ и $\sigma_i = \mathsf{sum}_{\boxplus}(M_i)$; сохраняет (Y_i, M_i) в памяти; возвращает $H_i = \mathsf{f}_{\overline{K}\boxplus\sigma_i}(M_i)$ алгоритму $\mathcal{A}$ (для SH: $\widetilde{K}_{\mathsf{Csc}} = \mathsf{f}(IV)$; $Y_i = \mathsf{Csc}(\widetilde{K}_{\mathsf{Csc}}, M_i||\mathsf{cs}(M_i))$; $H_i = \mathsf{f}(M_i)$).

Если на каком-то шаге обнаружено равенство $Y_i = Y_j$, $1 \leqslant i < j \leqslant q$, то построена коллизия $((\widetilde{K}_{\mathsf{Csc}}, M_i), (\widetilde{K}_{\mathsf{Csc}}, M_j))$ и

$$\Pr[Y_i = Y_j,\ 1 \leqslant i < j \leqslant q] = \mathrm{Adv}^{\mathrm{CR}}_{\mathsf{Csc}}(\mathcal{B}_{\mathrm{CR}}).$$

Схожим образом построим алгоритм $\mathcal{B}_{\mathrm{TPR}}$, который вычисляет для Csc прообраз к значению IV:

$$\Pr[IV \in \{Y_1, \ldots, Y_q\}] = \mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{Csc}}(\mathcal{B}_{\mathrm{TPR}}).$$

Вычислительные ресурсы алгоритмов $\mathcal{B}$, $\mathcal{B}_{\mathrm{CR}}$ и $\mathcal{B}_{\mathrm{TPR}}$ превосходят ресурсы t алгоритма $\mathcal{A}$ на значение, пропорциональное объёму обрабатываемых данных: $t' = t + ql$. Преобладание $\mathcal{B}_{\mathrm{CR}}$ и $\mathcal{B}_{\mathrm{TPR}}$ в соответствующих моделях оценивается неравенствами (5) и (6) соответственно.

В силу произвольности алгоритма $\mathcal{A}$ и неравенства треугольника доказываемое утверждение верно. ■

Подстановкой в (10) эвристических оценок (1), (3) и (4) получаем

$$\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{KH},K_{\mathsf{Csc}}}(t,q,l) \lessapprox \frac{t'}{2^k} + \frac{t'^2}{2^{n+1}} + \frac{t'}{2^n}, \quad t' \approx t, \tag{11}$$

и для SH идентично. Как и для оценки (8) в модели PRF, при длине ключа, не превосходящей половины блока ($k \leqslant n/2 = 256$ бит), единственным эффективным методом нарушения свойств безопасности является тотальное опробование, которому соответствует первое слагаемое в (11).

Второе слагаемое в (11) соответствует простой атаке. Противник, знающий K_{Csc}, строит самостоятельно коллизию из сообщений вида $M = P\,||(\sigma \boxminus P)$, $M' = P'\,||(\sigma \boxminus P')$:

$$\mathsf{Csc}(K_{\mathsf{Csc}}, M) = \mathsf{Csc}(K_{\mathsf{Csc}}, M'),$$
$$\mathsf{sum}_{\boxplus}(M) = \mathsf{sum}_{\boxplus}(M') = \sigma = \mathrm{const},$$

отправляет запрос M к оракулу, получает H, предъявляет пару (M', H) в качестве подделки. Альтернативно, делает два запроса M и M' и пользуется равенством $\mathcal{O}(M) \stackrel{?}{=} \mathcal{O}(M')$ в качестве критерия различения.

Схожим образом выполняется атака за счёт построения прообраза. Противник подбирает такое сообщение $M = P\,||(0 \boxminus P)$, чтобы $\mathsf{Csc}(K_{\mathsf{Csc}}, M) = IV$ и $\mathsf{sum}_{\boxplus}(M) = 0$. Хеш-значение для такого сообщения M равно K_{Csc}, что можно использовать для формирования подделки или как критерий для различения.

Следует заметить, что, в отличие от (8), оценка (11) зависит только от вычислительных ресурсов противника и, следовательно, в общем случае её гарантии значительно слабее. Например, при длине ключа $k = 512$ и ресурсах противника $t = 2^{256}$, $q = 2^{128}$, $l = 2^{64}$ криптоалгоритмы KH и SH будут стойкими PRF, но не будут таковыми при утечке каскадного ключа.

5.2. HMAC-Стрибог

Двойное хеширование мотивирует расширить возможности противника, ему даются оба каскадных ключа, а также раскрываются промежуточные состояния после первого хеширования:

$$\mathsf{leak}_1(K) = (\mathsf{g}^{\nabla}_{\overline{K}\oplus \mathtt{ipad}}(IV), \mathsf{g}^{\nabla}_{\overline{K}\oplus \mathtt{opad}}(IV)) = (K^I_{\mathsf{Csc}}, K^O_{\mathsf{Csc}}),$$
$$\mathsf{leak}_2(K, M_i) = \mathsf{H}(\overline{K} \oplus \mathtt{ipad} \,||\, M_i) = H^I_i, \quad 1 \leqslant i \leqslant q.$$

Противник может вычислить $Y^I_i = \mathsf{Csc}(K^I_{\mathsf{Csc}}, M_i)$ и $Y^O_i = \mathsf{Csc}(K^O_{\mathsf{Csc}}, H^I_i)$, $1 \leqslant i \leqslant q$.

Теорема 3. Преобладание любого противника, атакующего HMAC-Стрибог-τ в модели PRF-LEAK, ограничено:

$$\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{HMAC},\mathsf{leak}_1,\mathsf{leak}_2}(t,q,l) \leqslant \mathrm{Adv}^{\text{PRF-RKA}_{\boxplus\circ\oplus}}_{\mathsf{g}^{\nabla}}(t',q',r,d) + \frac{q^2}{2^{\tau+1}} + \mathrm{Adv}^{\mathrm{CR}}_{\mathsf{g}}(t') + \mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{g}}(t') + 2^{-n},$$

где $t' = t + ql$, $r = q' = 2q + 2$, $d = 2$, $\tau \in \{256, 512\}$.

Доказательство. Структура доказательства аналогична [6, Theorem (PRF-security of HMAC-Streebog)]. Так же, как и для (7), вероятность коллизий вида $Y_i^I = Y_j^I$, $Y_i^O = Y_j^O$, $Y_i^I = Y_j^O$ ограничивается за счёт CR-стойкости каскада, а вероятность события $IV \in \{Y_1^I, \ldots, Y_q^O\}$ — за счёт стойкости к атакам TPR.

Обозначим значения, относящиеся к первому и ко второму хешированию, с помощью верхних индексов «I» и «O» соответственно (рис. 3):

$$K^I = \overline{K} \oplus \texttt{ipad},\ H^I = \mathsf{H}(K^I \,||\, M),\ \ \hat{H}^I = \mathsf{msb}_\tau(H^I),\ Y^I = \mathsf{Csc}(K^I_{\mathsf{Csc}}, M),\ \sigma^I = \mathsf{sum}_{\boxplus}(M),$$
$$K^O = \overline{K} \oplus \texttt{opad}, H^O = \mathsf{H}(K^O \,||\, \hat{H}^I), \hat{H}^O = \mathsf{msb}_\tau(H^O), Y^O = \mathsf{Csc}(K^O_{\mathsf{Csc}}, \hat{H}^I), \sigma^O = \mathsf{sum}_{\boxplus}(\hat{H}^I).$$

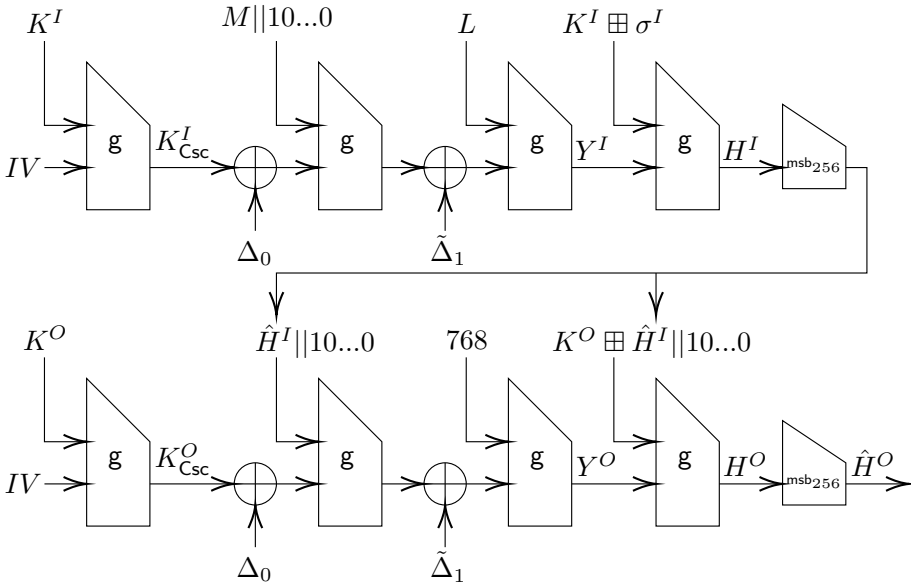

Рис. 3. HMAC-Стрибог-256, длина M равна $L < n$

Оракул $\mathcal{L}$ в ответ на запрос M возвращает $(\hat{H}^O, H^I) \in V^\tau \times V^n$ — непосредственное хеш-значение, сформированное криптоалгоритмом, и промежуточное состояние.

Рассмотрим «идеализированный» оракул $\widetilde{\mathcal{L}}$

$$\widetilde{\mathcal{L}}(M) = \left(\mathsf{msb}_\tau \mathsf{f}_{K^O \boxplus \sigma^O}\left(\ldots \mathsf{f}_{K^I \boxplus \sigma^I}(\mathsf{Csc}(\mathsf{f}_{K^I}(IV), M))\right), \mathsf{f}_{K^I \boxplus \sigma^I}(\mathsf{Csc}(\mathsf{f}_{K^I}(IV), M))\right),$$

где первое и последнее преобразование $\mathsf{g}^\triangledown$ в обоих обращениях к хеш-функции заменено на семейство из 2^n случайных функций, индексируемых связанными ключами вида $(\overline{K} \oplus \phi) \boxplus \sigma$, $\phi \in \{\texttt{ipad}, \texttt{opad}\}$.

Коллизия (C) — совпадение любой пары в ряду из $(2q+1)$ значений:

$$IV,\ Y_1^I,\ \ldots,\ Y_q^I,\ Y_1^O,\ \ldots,\ Y_q^O.$$

Если коллизия не реализуется ($\overline{\text{C}}$), то $\widetilde{\mathcal{L}}$ даже при известных противнику каскадных ключах неотличим от случайной функции $\mathsf{R} \in \mathrm{Func}(V^{<2^n}, V^\tau \times V^n)$:

$$\Pr\left[\mathcal{A}^{\mathsf{R}}(lk) \Rightarrow 1\right] = \Pr\left[\mathcal{A}^{\widetilde{\mathcal{L}}}(\mathsf{f}_{K^I}(IV), \mathsf{f}_{K^O}(IV)) \Rightarrow (1, \overline{\text{C}})\right], \quad lk \in V^n \times V^n,$$

так как все значения $(K^I_{\mathsf{Csc}}, K^O_{\mathsf{Csc}}, H^I_1, \ldots, H^I_q, \hat{H}^O_1, \ldots, \hat{H}^O_q)$, которые наблюдал противник, получены в результате различных запросов к f. Следовательно,

$$\mathrm{Adv}^{\text{PRF-LEAK}}_{\mathsf{HMAC}, \mathsf{leak}_1, \mathsf{leak}_2}(\mathcal{A}) = \left(\Pr\left[\mathcal{A}^{\mathcal{L}} \Rightarrow (1, \text{C})\right] + \Pr\left[\mathcal{A}^{\mathcal{L}} \Rightarrow (1, \overline{\text{C}})\right]\right) - \Pr\left[\mathcal{A}^{\widetilde{\mathcal{L}}} \Rightarrow (1, \overline{\text{C}})\right] \leqslant$$
$$\leqslant \left(\Pr\left[\mathcal{A}^{\mathcal{L}} \Rightarrow (1, \overline{\text{C}})\right] - \Pr\left[\mathcal{A}^{\widetilde{\mathcal{L}}} \Rightarrow (1, \overline{\text{C}})\right]\right) + \left(\Pr\left[\mathcal{A}^{\mathcal{L}} \Rightarrow \text{C}\right] - \Pr\left[\mathcal{A}^{\widetilde{\mathcal{L}}} \Rightarrow \text{C}\right]\right) + \Pr\left[\mathcal{A}^{\widetilde{\mathcal{L}}} \Rightarrow \text{C}\right] = \epsilon + \epsilon_{\text{C}} + p_{\text{C}},$$

противник $\mathcal{A}$ перед взаимодействием с оракулом $\mathcal{L}$ получает $(K^I_{\mathsf{Csc}}, K^O_{\mathsf{Csc}})$, а перед взаимодействием с $\widetilde{\mathcal{L}}$ — пару $(\mathsf{f}_{K^I}(IV), \mathsf{f}_{K^O}(IV))$.

Построим алгоритм $\mathcal{B}$, атакующий $\mathsf{g}^\triangledown$ в модели PRF-RKA$_{\boxplus\circ\oplus}$. Алгоритм $\mathcal{B}$ запрашивает каскадные ключи $K^I_{\mathsf{Csc}} = \mathcal{O}(IV, (\mathtt{ipad}, 0))$ и $K^O_{\mathsf{Csc}} = \mathcal{O}(IV, (\mathtt{opad}, 0))$ у оракула $\mathcal{O} \in \{\mathsf{g}^\triangledown, \mathsf{f}\}$, а затем запускает алгоритм $\mathcal{A}$, передавая ему $(K^I_{\mathsf{Csc}}, K^O_{\mathsf{Csc}})$.

При обработке запроса M_i от $\mathcal{A}$ алгоритм $\mathcal{B}$: вычисляет $Y^I_i = \mathsf{Csc}(K^I_{\mathsf{Csc}}, M_i)$ и $\sigma^I_i = \mathsf{sum}_{\boxplus}(M_i)$; сохраняет Y^I_i в памяти; проверяет условие коллизии; делает запрос $(Y^I_i, (\mathtt{ipad}, \sigma^I_i))$ к оракулу; получает H^I_i; вычисляет $Y^O_i = \mathsf{Csc}(K^O_{\mathsf{Csc}}, \hat{H}^I_i)$ и $\sigma^O_i = \mathsf{sum}_{\boxplus}(\hat{H}^I_i)$; сохраняет Y^O_i в памяти; проверяет условие коллизии; делает запрос $(Y^O_i, (\mathtt{opad}, \sigma^O_i))$; получает H^O_i; передаёт алгоритму $\mathcal{A}$ пару $(\hat{H}^O_i, H^I_i)$, $1 \leqslant i \leqslant q$.

Если проверка на каком-то из шагов показывает, что коллизия реализовалась, то $\mathcal{B}$ прекращает взаимодействие с $\mathcal{A}$ и возвращает 1. Если после обработки всех запросов от $\mathcal{A}$ коллизии нет, то результат работы $\mathcal{B}$ равен результату работы $\mathcal{A}$. К оракулу $\mathcal{O} \in \{\mathsf{g}^\triangledown, \mathsf{f}\}$ выполняется до $q' = (2q + 2)$ запросов, количество связанных ключей оценивается так же ($r = q'$). Только значение IV запрашивается при двух заведомо различных связанных ключах ($\mathtt{ipad} \neq \mathtt{opad}$), любой другой вход обрабатывается ровно один раз, следовательно, $d = 2$.

Преобладание алгоритма $\mathcal{B}$ равно

$$\mathrm{Adv}^{\text{PRF-RKA}_{\boxplus\circ\oplus}}_{\mathsf{g}^\triangledown}(\mathcal{B}) = \Pr\left[\mathcal{B}^{\mathsf{g}^\triangledown} \Rightarrow 1\right] - \Pr\left[\mathcal{B}^{\mathsf{f}} \Rightarrow 1\right] = \epsilon + \epsilon_{\mathrm{C}}.$$

Оценим сверху вероятность коллизии:

$$p_{\mathrm{C}} \leqslant \Pr\left[Y^I_i = Y^I_j,\ 1 \leqslant i < j \leqslant q\right] + \Pr\left[Y^O_i = Y^O_j,\ 1 \leqslant i < j \leqslant q\right] +$$
$$+ \Pr\left[Y^I_i = Y^O_j,\ 1 \leqslant i, j \leqslant q\right] + \Pr\left[IV \in \{Y^I_1, \ldots, Y^I_q, Y^O_1, \ldots, Y^O_q\}\right] = p^I_{\mathrm{C}} + p^O_{\mathrm{C}} + p^{I,O}_{\mathrm{C}} + p^{\mathrm{pre}}_{\mathrm{C}}.$$

Построим алгоритм $\mathcal{B}_{\mathrm{CR}}$, пытающийся сформировать коллизию для Csc. Он работает так же, как $\mathcal{B}$, но вместо запросов к оракулу $\mathcal{O} \in \{\mathsf{g}^\triangledown, \mathsf{f}\}$ самостоятельно имитирует семейство случайных функций f и сохраняет в памяти (Y^I_i, M^I_i) и $(Y^O_i, \hat{H}^I_i)$. Если на каком-то шаге обнаружена коллизия и при этом $IV \notin \{Y^I_1, \ldots, Y^O_q\}$, то возможны три случая:

1) $Y^I_i = Y^I_j$, $i \neq j$, даёт коллизию $((K^I_{\mathsf{Csc}}, M_i), (K^I_{\mathsf{Csc}}, M_j))$ в силу $M_i \neq M_j$;
2) $Y^O_i = Y^O_j$, $i \neq j$, даёт коллизию $((K^O_{\mathsf{Csc}}, \hat{H}^I_i), (K^O_{\mathsf{Csc}}, \hat{H}^I_j))$ при условии $\hat{H}^I_i \neq \hat{H}^I_j$;
3) $Y^I_i = Y^O_j$ даёт коллизию $((K^I_{\mathsf{Csc}}, M_i), (K^O_{\mathsf{Csc}}, \hat{H}^I_j))$, если хотя бы $K^I_{\mathsf{Csc}} \neq K^O_{\mathsf{Csc}}$.

Таким образом, вероятность успеха алгоритма $\mathcal{B}_{CR}$ оценивается как

$$\mathrm{Adv}^{\mathrm{CR}}_{\mathsf{Csc}}(\mathcal{B}_{\mathrm{CR}}) \geqslant (p^I_{\mathrm{C}} + p^O_{\mathrm{C}} + p^{I,O}_{\mathrm{C}}) - \Pr\left[K^I_{\mathsf{Csc}} = K^I_{\mathsf{Csc}}\right] - \Pr\left[\exists i \neq j\, (\hat{H}^I_i = \hat{H}^I_j)\right] \geqslant$$
$$\geqslant (p^I_{\mathrm{C}} + p^O_{\mathrm{C}} + p^{I,O}_{\mathrm{C}}) - 2^{-n} - \frac{q^2}{2^{\tau+1}}.$$

По аналогии с $\mathcal{B}_{\mathrm{CR}}$ построим алгоритм $\mathcal{B}_{\mathrm{TPR}}$, который ищет для Csc прообраз к значению IV. Искомым прообразом станет либо $(K^I_{\mathsf{Csc}}, M_i)$, либо $(K^O_{\mathsf{Csc}}, \hat{H}^I_i)$,

$$p^{\mathrm{pre}}_{\mathrm{C}} = \mathrm{Adv}^{\mathrm{TPR}}_{\mathsf{Csc}}(\mathcal{B}_{\mathrm{TPR}}).$$

Вычислительные ресурсы алгоритмов $\mathcal{B}$, $\mathcal{B}_{\mathrm{CR}}$ и $\mathcal{B}_{\mathrm{TPR}}$ превосходят ресурсы $\mathcal{A}$ на значение, пропорциональное объёму обрабатываемых данных, $t' = t + ql$. Преобладание $\mathcal{B}_{\mathrm{CR}}$

и $\mathcal{B}_{\mathrm{TPR}}$ в соответствующих моделях оценивается неравенствами (5) и (6) соответственно. Пользуясь неравенством треугольника, получаем доказываемое утверждение. ■

Не является стойким в модели PRF-LEAK, например, режим имитозащиты CMAC [36] — утечка промежуточного состояния сразу даёт противнику возможность сформировать подделку.

Раскрытие состояния приводит к потере стойкости у алгоритмов имитозащиты, основанных на однократном применении хеш-функций типа HAIFA [37], например Skein-MAC [38] и BLAKE2-MAC [39], а также у схем, построенных по схеме «губка» («sponge») — KMAC [40]. Вместе с тем из криптографической хеш-функции за счёт применения двойного хеширования или «сэндвича» обычно нетрудно сделать PRF-LEAK-стойкий криптоалгоритм.

6. Стойкость к атакам на ключ

Количественные оценки в модели PRF и PRF-LEAK одинаковы для трёх анализируемых схем. Кроме того, при $k \leqslant n/2 = 256$ единственным эффективным методом нарушения свойств безопасности является тотальное опробование ключа (пусть и в разных предположениях о функции сжатия $\mathsf{g}^{\triangledown}$).

При k, близком к n, и при отсутствии ограничений на объём обрабатываемого материала Стрибог-K и HMAC-Стрибог подвержены нетривиальным атакам на восстановление секретного ключа. Существуют методы с трудоёмкостью порядка $t \approx q \cdot l \approx 2^{4n/5}$ по времени и данным [9]. Простой подстановкой в (8) можно убедиться, что наличие таких атак не противоречит оценке, полученной с помощью доказательного подхода.

Атаки на ключ [9] дают по сути оценку снизу на вероятность успеха противника в модели KR. Получим верхние оценки в этой модели, а также в условиях утечки внутреннего состояния (KR-LEAK).

Схема «сэндвич» в моделях KR и KR-LEAK является более стойкой, чем два других криптоалгоритма.

Определение 8. Характеристикой успешности противника $\mathcal{A}$ в модели KR-LEAK для ключевого алгоритма $\mathsf{F} : \mathbf{K} \times \mathbf{X} \to \mathbf{Y}$ при утечке, определяемой функциями $\mathsf{leak}_1 : \mathbf{K} \to \mathbf{L}_1$ и $\mathsf{leak}_2 : \mathbf{K} \times \mathbf{X} \to \mathbf{L}_2$, назовём вероятность восстановления ключа:

$$\mathrm{Adv}^{\text{KR-LEAK}}_{\mathsf{F},\mathsf{leak}_1,\mathsf{leak}_2}(\mathcal{A}) = \Pr\Big[K \xleftarrow{\mathrm{R}} \mathbf{K} : \mathcal{A}^{\mathcal{L}}(\mathsf{leak}_1(K)) \Rightarrow K',\ K' = K\Big].$$

Оракул $\mathcal{L}$ на запрос $X \in \mathbf{X}$ возвращает $(\mathsf{F}(K, X), \mathsf{leak}_2(K, X))$.

При одинаковых функциях утечки модели PRF-LEAK и KR-LEAK связаны так же, как модели PRF и KR.

Утверждение 1. Вероятность успеха противника, пытающегося определить секретный ключ K криптоалгоритма Стрибог-C (при утечке K_{Csc}), или Стрибог-K (при утечке K_{Csc}), или HMAC-Стрибог (при утечке $\mathsf{leak}_1(K) = (K^I_{\mathsf{Csc}}, K^O_{\mathsf{Csc}})$ и $\mathsf{leak}_2(K, M_i) = H^I_i$, $1 \leqslant i \leqslant q$), ограничена сверху соответственно:

$$\mathrm{Adv}^{\text{KR-LEAK}}_{\mathsf{SH},K_{\mathsf{Csc}}}(t,q,l) \leqslant \mathrm{Adv}^{\text{KR}}_{\mathsf{g}^{\triangledown}}(t',q') \lessapprox \frac{t'}{2^k},$$

$$\mathrm{Adv}^{\text{KR-LEAK}}_{\mathsf{KH},K_{\mathsf{Csc}}}(t,q,l) \leqslant \mathrm{Adv}^{\text{KR-RKA}_{\boxplus}}_{\mathsf{g}^{\triangledown}}(t',q') \lessapprox \frac{t' \cdot q'}{2^k}; \tag{12}$$

$$\mathrm{Adv}^{\text{KR-LEAK}}_{\mathsf{HMAC},\mathsf{leak}_1,\mathsf{leak}_2}(t,q,l) \leqslant \mathrm{Adv}^{\text{KR-RKA}_{\boxplus\circ\oplus}}_{\mathsf{g}^{\triangledown}}(t',2q') \lessapprox \frac{2 \cdot t' \cdot q'}{2^k}, \tag{13}$$

где $t' = t + ql$, $q' = q + 1$.

Доказательство. Пусть противник $\mathcal{A}$ восстанавливает ключ алгоритма SH с вероятностью успеха p. Построим $\mathcal{B}$, атакующий $\mathsf{g}^{\triangledown}$. За счёт первого запроса получим $K_{\mathsf{Csc}} = \mathsf{g}^{\triangledown}_{K}(IV)$, передадим K_{Csc} алгоритму $\mathcal{A}$. На каждый запрос M_i от $\mathcal{A}$ к SH: самостоятельно вычисляем $Y_i = \mathsf{Csc}(K_{\mathsf{Csc}}, M_i||\mathsf{cs}(M_i))$; получаем у оракула $H_i = \mathsf{g}^{\triangledown}_{K}(Y_i)$ и возвращаем его $\mathcal{A}$. Для $\mathcal{A}$ идеально симулируется SH, а следовательно, вероятность успеха $\mathcal{B}$ равна p.

Для KH и HMAC аналогично, но $\mathsf{g}^{\triangledown}$ рассматривается относительно атак со связанными ключами. ■

Таким образом, единственным эффективным методом определения ключа схемы «сэндвич» даже при утечке K_{Csc} является метод тотального опробования (в предположении, что для функции сжатия эффективным является только тотальное опробование). Для KH и HMAC это не так, но утверждение показывает, что вероятность успеха атакующего растёт не более чем линейно с ростом числа сообщений (q) и практически не зависит от их длины (l), так как вычислительные мощности противника не меньше доступных ему данных ($t > q \cdot l$). Оптимум достигается при $t \approx q \approx 2^{k/2}$. Учитывая (11), при произвольном k получим

$$\mathrm{Adv}^{\text{KR-LEAK}}_{\mathsf{KH},K_{\mathsf{Csc}}}(t,q,l) \lessapprox \min\left(\frac{t'\cdot q'}{2^k}, \frac{t'}{2^k} + \frac{t'^2}{2^{n+1}} + \frac{t'}{2^n}\right),\ t' \approx t,\ q' = q+1,$$

и схожую оценку для HMAC.

В модели KR-LEAK оценки (12) и (13) являются точными, в [5] для схожих условий описана атака, основанная на построении мультиколлизий для каскадного преобразования. Оценки верны и в модели KR, но в этом случае говорить об их точности нельзя, атаки [9] требуют $t \approx 2^{4n/5}$, что много больше, чем $t \approx 2^{n/2}$.

7. Подходы к защите от атак по побочным каналам

Функции утечки в моделях KR-LEAK и PRF-LEAK никак не связаны с природой процессов, из-за которых противник получает дополнительные сведения. Используемый при доказательстве стойкости конкретный вид этих функций говорит о том, что полный доступ противника к содержащейся в них информации не приводит к нарушению свойств безопасности. Отсюда важное следствие: те сведения, которые не описаны функциями утечки (и не могут быть вычислены на их основе), являются критически важными и подлежат защите от раскрытия.

Конкретные меры защиты, в первую очередь маскирование [14, 41], существенным образом зависят от специфики реализации криптоалгоритма и физической модели рассматриваемых побочных каналов и не являются предметом настоящей работы. Здесь опишем части криптоалгоритмов, которые следует защищать, а также укажем на общие проблемы, возникающие при реализации соответствующей защиты.

Во всех представленных ранее результатах каскадный ключ K_{Csc} (пара ключей для HMAC-Стрибог) раскрывался противнику. Предполагаем, что каскадный ключ (ключи) вычисляется однократно и хранится в памяти.

Схема «сэндвич» и Стрибог-K схожи. Для любого сообщения M противник может вычислить:

$$\begin{aligned} \mathsf{SH}: Y &= \mathsf{Csc}(K_{\mathsf{Csc}}, M \,||\, \mathsf{cs}(M)), \quad \sigma = 0,\\ \mathsf{KH}: Y &= \mathsf{Csc}(K_{\mathsf{Csc}}, M), \qquad\quad \sigma = \mathsf{sum}_{\boxplus}(M). \end{aligned}$$

Сложение ключа $\overline{K}$ с блоками сообщения должно происходить в KH однократно: сначала вычисляется σ, затем $\overline{K} \boxplus \sigma$. В SH ключ вообще не должен складываться с блоками сообщения.

Последний вызов функции сжатия:

$$\begin{gathered} H = \mathsf{g}^{\nabla}_{\overline{K} \boxplus \sigma}(Y) = (\overline{K} \boxplus \sigma) \oplus Y \oplus \mathsf{E}(Y, \overline{K} \boxplus \sigma) = \\ = (\overline{K} \boxplus \sigma) \oplus Y \oplus \mathsf{X}_{13}\,\mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_{12} \ldots \mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_{1}(\overline{K} \boxplus \sigma), \quad \overline{K} = K \,||\, 0^{n-k}. \end{gathered} \tag{14}$$

Раундовые ключи шифра E формируются из состояния Y, а следовательно, также известны противнику. Функция сжатия сама по себе является стойкой в таких условиях [7, 8]. Однако если раскрывается промежуточное состояние, например $s^{(1)} = \mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_1(\overline{K} \boxplus \sigma)$, то противник легко вычислит ключ:

$$\begin{gathered} s^{(13)} = \mathsf{X}_{13}\,\mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_{12} \ldots \mathsf{L}\,\mathsf{P}\,\mathsf{S}\,\mathsf{X}_{2}(s^{(1)}), \\ \overline{K} = \left(H \oplus Y \oplus s^{(13)}\right) \boxminus \sigma. \end{gathered}$$

Необходима защита всех внутренних состояний шифра на протяжении 13 раундов.

Вычисления каскадного преобразования могут, следовательно, осуществляться «быстрым» незащищённым образом, а последний вызов шифра E — «медленным» защищённым, например на отдельном внешнем (по отношению к остальной вычислительной системе) модуле.

У Стрибог-С контрольная сумма всегда равна нулю ($\sigma = 0$), сложение по модулю 2^n отсутствует, что позволяет применять для защиты LPSX-преобразований (14) хорошо известные методы маскирования [14, 41, 42], в том числе использующие специфику нелинейного преобразования [16], а также методы, основанные на пороговой реализации (threshold implementation) [15, 43–45].

Для защиты криптоалгоритма Стрибог-К из-за произвольности значения σ может потребоваться применение вспомогательных алгоритмов, что снижает скорость работы и усложняет реализацию. Пусть, например, вместо ключа используется пара «маскированный ключ, маска» $(\overline{K} \boxplus W, W)$. Тогда после сложения с КС получим $(\overline{K} \boxplus W \boxplus \sigma, W)$. В шифре E используется операция $\oplus$, что потребует вычисления пары $(\overline{K} \boxplus \sigma \oplus W', W')$, например, с помощью [11]. Если одновременное хранение ключа под разными масками невозможно, то переход от $(\overline{K} \boxplus \sigma \oplus W', W')$ к $(\overline{K} \boxplus W'', W'')$ также потребует применения специального алгоритма [10].

У криптоалгоритма HMAC-Стрибог защищать требуется два вызова блочного шифра, при первом и втором хешировании. Противнику известны значения

$$\begin{aligned} Y^I &= \mathsf{Csc}(K^I_{\mathsf{Csc}}, M), & \sigma^I &= \mathsf{sum}_{\boxplus}(M), & H^I &= \mathsf{g}^{\nabla}_{\overline{K} \oplus \mathtt{ipad} \boxplus \sigma^I}(Y^I), \\ Y^O &= \mathsf{Csc}(K^O_{\mathsf{Csc}}, \mathsf{msb}_\tau(H^I)), & \sigma^O &= \mathsf{sum}_{\boxplus}(\mathsf{msb}_\tau(H^I)), & H^O &= \mathsf{g}^{\nabla}_{\overline{K} \oplus \mathtt{opad} \boxplus \sigma^O}(Y^O), \end{aligned}$$

а также раундовые ключи шифра E, формируемые из Y^I и Y^O. В общем случае $\sigma^I \neq 0$ и $\sigma^O \neq 0$, что приводит к необходимости двукратного применения таких же вспомогательных алгоритмов [10, 11], как и для Стрибог-К. Вычисление $Y^O = \mathsf{Csc}(K^O_{\mathsf{Csc}}, \ldots)$ требует два ($\tau = 256$) или три ($\tau = 512$) обращения к функции сжатия, что затрудняет реализацию E на защищённом внешнем модуле. К такому модулю потребуется либо делать два запроса, либо реализовывать вычисление Y^O внутри него.

Заключение

В работе предложен простой способ преобразования отечественной хеш-функции в ключевой криптоалгоритм по схеме «сэндвич» (Стрибог-С), к хешируемому тексту приписывается специальный блок, играющий роль «альтернативной» контрольной суммы, а в саму хеш-функцию никаких изменений не вносится.

Благодаря указанному приёму, сжимающее преобразование $\mathtt{g}^{\triangledown}$ (первое и последнее в хеш-функции) используется только с ключом K, а не со связанными ключами вида $K \boxplus \sigma$, как в алгоритмах HMAC-Стрибог и Стрибог-K.

Схема «сэндвич» является стойкой псевдослучайной функцией (PRF) и, следовательно, стойким алгоритмом имитозащиты, при этом функция $\mathtt{g}^{\triangledown}$ должна быть PRF, но стойкости к атакам со связанными ключами от неё не требуется. Аналогичное соображение позволяет показать, что при любом объёме обрабатываемых данных для Стрибог-С не существует более эффективного метода определения ключа, чем тотальное опробование, если то же самое верно для функции сжатия $\mathtt{g}^{\triangledown}$.

Предложены модели угроз PRF-LEAK и KR-LEAK, в рамках которых противник (решающий задачу различения или пытающийся восстановить ключ соответственно) получает непосредственный доступ к (почти всем) внутренним состояниям криптоалгоритма, происходит их раскрытие — утечка. Доказано, что Стрибог-С, Стрибог-K и HMAC-Стрибог являются стойкими в этих моделях при дополнительном предположении о стойкости функции сжатия к атакам на построение коллизий и прообраза. Справедливость предположения подтверждается представленными в открытой литературе конструктивными исследованиями. Стойкость в этих моделях делает указанные криптоалгоритмы предпочтительнее ряда схем, основанных на блочных шифрах, например режима имитозащиты ГОСТ 34.13-2018.

Выполненный анализ позволил выявить части хеш-функции, которые при реализации ключевых криптоалгоритмов подлежат защите от каких-либо утечек. Защищать требуется только последнюю функцию сжатия, а точнее, используемый внутри неё блочный шифр (без алгоритма развёртки ключа) и сам вход шифра (которым является секретный ключ K или $K \boxplus \sigma$). Для HMAC-Стрибог защита требуется и при первом, и при втором хешировании. Отсутствие у схемы «сэндвич» связанных ключей и, как следствие, попеременного использования операций $\boxplus$ и $\oplus$ в ряде случаев, как представляется, позволяет существенно упростить реализацию мер защиты.

ЛИТЕРАТУРА

1. ГОСТ 34.11-2018. Информационная технология. Криптографическая защита информации. Функция хэширования. М.: Стандартинформ, 2018.
2. *Damgard I.* A design principle for hash functions // LNCS. 1990. V. 435. P. 416–427.
3. *Merkle R.* One way hash functions and DES // LNCS. 1990. V. 435. P. 428–446.
4. Р 50.1.113-2016. Информационная технология. Криптографическая защита информации. Криптографические алгоритмы, сопутствующие применению алгоритмов электронной цифровой подписи и функции хэширования. М.: Стандартинформ, 2016.
5. *Kiryukhin V. A.* Keyed Streebog is a secure PRF and MAC // Матем. вопр. криптогр. 2023. Т. 14. № 2. С. 77–96.
6. *Kiryukhin V. A.* About "k-bit Security" of MACs Based on Hash Function Streebog. Cryptology ePrint Archive. Paper 2023/1305. 2023. https://eprint.iacr.org/2023/1305.
7. *Kiryukhin V. A.* Streebog compression function as PRF in secret-key settings // Матем. вопр. криптогр. 2022. Т. 13. № 2. С. 99–116.

8. *Kiryukhin V. A.* Related-key attacks on the compression function of Streebog // Матем. вопр. криптогр. 2023. Т. 14. № 2. С. 59–76.
9. *Dinur I. and Leurent G.* Improved generic attacks against hash-based MACs and HAIFA // LNCS. 2014. V. 8616. P. 149–168.
10. *Goubin L.* A Sound method for switching between Boolean and arithmetic masking // LNCS. 2001. V. 2162. P. 3–15.
11. *Coron J., Großschädl J., Tibouchi M., and Vadnala P. K.* Conversion from arithmetic to Boolean masking with Logarithmic complexity // LNCS. 2015. V. 9054. P. 130–149.
12. *Yasuda K.* "Sandwich" is indeed secure: How to authenticate a message with just one hashing // LNCS. 2007. V. 4586. P. 355–369.
13. *Bellare M., Goldreich O., and Mityagin A.* The Power of Verification Queries in Message Authentication and Authenticated Encryption. Cryptology ePrint Archive. Paper 2004/304. 2004. `https://eprint.iacr.org/2004/304`.
14. *Blömer J., Merchan J., and Krummel V.* Provably secure masking of AES // LNCS. 2004. V. 3357. P. 69–83.
15. *Nikova S., Rechberger C., and Rijmen V.* Threshold implementations against side-channel attacks and glitches // LNCS. 2006. V. 4307. P. 529–545.
16. *Lavrenteva T. A. and Matveev S. V.* Side-channel attacks countermeasure based on decomposed S-boxes for Kuznyechik // Матем. вопр. криптогр. 2021. Т. 12. № 2. С. 147–157.
17. *Bellare M. and Rogaway P.* Introduction to Modern Cryptography. 2005. `https://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf`.
18. *Bernstein D. J. and Lange T.* Non-uniform cracks in the concrete: The power of free precomputation // LNCS. 2013. V. 8270. P. 321–340.
19. *Guo J., Jean J., Leurent G., et al.* The usage of counter revisited: Second-preimage attack on new Russian standardized hash function // LNCS. 2014. V. 8781. P. 195–211.
20. *Abdelkhalek A., AlTawy R., and Youssef A. M.* Impossible differential properties of reduced round Streebog // LNCS. 2015. V. 9084. P. 274–286.
21. *Rogaway P. and Shrimpton T.* Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance // LNCS. 2004. V. 3017. P. 371–388.
22. *AlTawy R., Kircanski A., and Youssef A. M.* Rebound attacks on Stribog // LNCS. 2014. V. 8565. P. 175–188.
23. *Lin D., Xu S., and Yung M.* Cryptanalysis of the round-reduced GOST hash function // LNCS. 2014. V. 8567. P. 309–322.
24. *Ma B., Li B., Hao R., and Li X.* Improved cryptanalysis on reduced-round GOST and Whirlpool hash function // LNCS. 2014. V. 8479. P. 289–307.
25. *Wang Z., Yu H., and Wang X.* Cryptanalysis of GOST R hash function // Inform. Processing Lett. 2014. V. 114. P. 655–662.
26. *Kölbl S. and Rechberger C.* Practical attacks on AES-like cryptographic hash functions // LNCS. 2014. V. 8895. P. 259–273.
27. *Van Oorschot P. C. and Wiener M. J.* Parallel collision search with cryptanalytic applications // J. Cryptology. 1999. V. 12. No. 1. P. 1–28.
28. *AlTawy R. and Youssef A. M.* Preimage attacks on reduced-round Stribog // LNCS. 2014. V. 8469. P. 109–125.
29. *Ma B., Li B., Hao R., and Li X.* Improved (pseudo) preimage attacks on reduced-round GOST and Grostl-256 and studies on several truncation patterns for AES-like compression functions // LNCS. 2015. V. 9241. P. 79–96.

30. *Hua J., Dong X., Sun S., et al.* Improved MITM Cryptanalysis on Streebog. Cryptology ePrint Archive. Paper 2022/568. 2022. https://eprint.iacr.org/2022/568.
31. *Bellare M., Jaeger J., and Len J.* Better than advertised: Improved collision-resistance guarantees for MD-based hash functions // Proc. CCS'17. N.Y.: ACM, 2017. P. 891–906.
32. *Bellare M.* New proofs for NMAC and HMAC: Security without collision-resistance // LNCS. 2014. V. 4117. P. 602–619.
33. *Koblitz N. and Menezes A.* Another look at HMAC // J. Math. Cryptology. 2013. V. 7:3. P. 225–251.
34. Репозиторий «Ключевой Стрибог». https://gitflic.ru/project/vkir/streebog.
35. ГОСТ Р 34.11-94. Информационная технология. Криптографическая защита информации. Функция хэширования. М.: Издательство Стандартов, 1994.
36. ГОСТ 34.13-2018. Информационная технология. Криптографическая защита информации. Режимы работы блочных шифров. М.: Стандартинформ, 2018.
37. *Biham E. and Dunkelman O.* A Framework for Iterative Hash Functions — HAIFA. Cryptology ePrint Archive. Report 2007/278. 2007. https://eprint.iacr.org/2007/278.
38. *Ferguson N., Lucks S., Schneier B., et al.* The Skein Hash Function Family. 2009. https://api.semanticscholar.org/CorpusID:59739596.
39. *Aumasson J., Neves S., Wilcox-O'Hearn Z., and Winnerlein C.* BLAKE2: Simpler, Smaller, Fast as MD5. IACR Cryptology ePrint Archive. Report 2013/322. 2013. https://eprint.iacr.org/2013/322.pdf.
40. *Kelsey J., Chang S., and Perlner R.* SHA-3 Derived Functions: cSHAKE, KMAC, TupleHash and ParallelHash. NIST Special Publication 800-185. 2016. https://doi.org/10.6028/NIST.SP.800-185.
41. *Goubin L. and Patarin J.* DES and differential power analysis. The "Duplication" Method // LNCS. 1999. V. 1717. P. 158–172.
42. *Oswald E., Mangard S., Pramstaller N., and Rijmen V.* A side-channel analysis resistant description of the AES S-Box // LNCS. 2005. V. 3557. P. 413–423.
43. *Bilgin B., Nikova S., Nikov V., et al.* Threshold implementations of all 3×3 and 4×4 S-boxes // LNCS. 2012. V. 7428. P. 76–91.
44. *Daemen J.* Changing of the guards: A simple and efficient method for achieving uniformity in threshold sharing // LNCS. 2017. V. 10529. P. 137–153.
45. *Piccione E., Andreoli S., Budaghyan L., et al.* An optimal universal construction for the threshold implementation of bijective S-boxes // IEEE Trans. Inform. Theory. 2023. V. 69. No. 10. P. 6700–6710.

REFERENCES

1. GOST 34.11-2018. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Funktsiya kheshirovaniya [GOST R 34.11-2018. Information Technology. Cryptographic Data Security. Hash-function]. Moscow, Standartinform Publ., 2018. (in Russian)
2. *Damgard I.* A design principle for hash functions. LNCS, 1990, vol. 435, pp. 416–427.
3. *Merkle R.* One way wash functions and DES. LNCS, 1990, vol. 435, pp. 428–446.
4. R 50.1.113-2016. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Kriptograficheskie algoritmy, soputstvuyushchie primeneniyu algoritmov elektronnoy tsifrovoy podpisi i funktsii kheshirovaniya [R 50.1.113-2016. Information Technology. Cryptographic Data Security. Cryptographic Algorithms Accompanying the Use of Electronic Digital Signature Algorithms and Hash Functions]. Moscow, Standartinform Publ., 2016. (in Russian)
5. *Kiryukhin V. A.* Keyed Streebog is a secure PRF and MAC. Mat. Vopr. Kriptogr., 2023, vol. 14, iss. 2, pp. 77–96.

6. *Kiryukhin V. A.* About "k-bit Security" of MACs Based on Hash Function Streebog. Cryptology ePrint Archive, Paper 2023/1305, 2023, https://eprint.iacr.org/2023/1305.
7. *Kiryukhin V. A.* Streebog compression function as PRF in secret-key settings. Mat. Vopr. Kriptogr., 2022, vol. 13, iss. 2, pp. 99–116.
8. *Kiryukhin V. A.* Related-key attacks on the compression function of Streebog. Mat. Vopr. Kriptogr., 2023, vol. 14, iss. 2, pp. 59–76.
9. *Dinur I. and Leurent G.* Improved generic attacks against hash-based MACs and HAIFA. LNCS, 2014, vol. 8616, pp. 149–168.
10. *Goubin L.* A sound method for switching between Boolean and arithmetic masking. LNCS, 2001, vol. 2162, pp. 3–15.
11. *Coron J., Großschädl J., Tibouchi M., and Vadnala P. K.* Conversion from arithmetic to Boolean masking with logarithmic complexity. LNCS, 2015, vol. 9054, pp. 130–149.
12. *Yasuda K.* "Sandwich" is indeed secure: How to authenticate a message with just one hashing. LNCS, 2007, vol. 4586, pp. 355–369.
13. *Bellare M., Goldreich O., and Mityagin A.* The Power of Verification Queries in Message Authentication and Authenticated Encryption. Cryptology ePrint Archive, Paper 2004/304, 2004, https://eprint.iacr.org/2004/304.
14. *Blömer J., Merchan J., and Krummel V.* Provably secure masking of AES. LNCS, 2004, vol. 3357, pp. 69–83.
15. *Nikova S., Rechberger C., and Rijmen V.* Threshold implementations against side-channel attacks and glitches. LNCS, 2006, vol. 4307, pp. 529–545.
16. *Lavrenteva T. A. and Matveev S. V.* Side-channel attacks countermeasure based on decomposed S-boxes for Kuznyechik. Mat. vopr. kriptogr., 2021, vol. 12, iss. 2, pp. 147–157.
17. *Bellare M. and Rogaway P.* Introduction to Modern Cryptography. 2005. https://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf.
18. *Bernstein D. J. and Lange T.* Non-uniform cracks in the concrete: The power of free precomputation. LNCS, 2013, vol. 8270, pp. 321–340.
19. *Guo J., Jean J., Leurent G., et al.* The usage of counter revisited: Second-preimage attack on new Russian standardized hash function. LNCS, 2014, vol. 8781, pp. 195–211.
20. *Abdelkhalek A., AlTawy R., and Youssef A. M.* Impossible differential properties of reduced round Streebog. LNCS, 2015, vol. 9084, pp. 274–286.
21. *Rogaway P. and Shrimpton T.* Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. LNCS, 2004, vol. 3017, pp. 371–388.
22. *AlTawy R., Kircanski A., and Youssef A. M.* Rebound attacks on Stribog. LNCS, 2014, vol. 8565, pp. 175–188.
23. *Lin D., Xu S., and Yung M.* Cryptanalysis of the round-reduced GOST hash function. LNCS, 2014, vol. 8567, pp. 309–322.
24. *Ma B., Li B., Hao R., and Li X.* Improved cryptanalysis on reduced-round GOST and Whirlpool hash function. LNCS, 2014, vol. 8479, pp. 289–307.
25. *Wang Z., Yu H., and Wang X.* Cryptanalysis of GOST R hash function. Inform. Processing Lett., 2014, vol. 114, pp. 655–662.
26. *Kölbl S. and Rechberger C.* Practical attacks on AES-like cryptographic hash functions. LNCS, 2014, vol. 8895, pp. 259–273.
27. *Van Oorschot P. C. and Wiener M. J.* Parallel collision search with cryptanalytic applications. J. Cryptology, 1999, vol. 12, iss. 1, pp. 1–28.
28. *AlTawy R. and Youssef A. M.* Preimage attacks on reduced-round Stribog. LNCS, 2014, vol. 8469, pp. 109–125.

29. *Ma B., Li B., Hao R., and Li X.* Improved (pseudo) preimage attacks on reduced-round GOST and Grostl-256 and studies on several truncation patterns for AES-like compression functions. LNCS, 2015, vol. 9241, pp. 79–96.
30. *Hua J., Dong X., Sun S., et al.* Improved MITM Cryptanalysis on Streebog. Cryptology ePrint Archive, Paper 2022/568, 2022, https://eprint.iacr.org/2022/568.
31. *Bellare M., Jaeger J., and Len J.* Better than advertised: Improved collision-resistance guarantees for MD-based hash functions. Proc. CCS'17, N.Y., ACM, 2017, pp. 891–906.
32. *Bellare M.* New proofs for NMAC and HMAC: security without collision-resistance. LNCS, 2014, vol. 4117, pp. 602–619.
33. *Koblitz N. and Menezes A.* Another look at HMAC. J. Math. Cryptology, 2013, vol. 7:3, pp. 225–251.
34. Repository "Keyed Streebog", https://gitflic.ru/project/vkir/streebog.
35. GOST R 34.11-94. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Funktsiya kheshirovaniya. [GOST R 34.11-94. Information Technology. Cryptographic Data Security. Hash-function]. Moscow, Izdatelstvo Standartov, 1994. (in Russian)
36. GOST 34.13-2018. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Rezhimy raboty blochnykh shifrov [GOST 34.13-2018. Information Technology. Modes of Operation for Block Ciphers]. Moscow, Standartinform Publ., 2018. (in Russian)
37. *Biham E. and Dunkelman O.* A Framework for Iterative Hash Functions — HAIFA. Cryptology ePrint Archive, Report 2007/278. 2007, https://eprint.iacr.org/2007/278.
38. *Ferguson N., Lucks S., Schneier B., et al.* The Skein Hash Function Family. 2009, https://api.semanticscholar.org/CorpusID:59739596.
39. *Aumasson J., Neves S., Wilcox-O'Hearn Z., and Winnerlein C.* BLAKE2: Simpler, Smaller, Fast as MD5. IACR Cryptology ePrint Archive, Report 2013/322, 2013, https://eprint.iacr.org/2013/322.pdf.
40. *Kelsey J., Chang S., and Perlner R.* NIST Special Publication 800-185. SHA-3 Derived Functions: cSHAKE, KMAC, TupleHash and ParallelHash. 2016, https://doi.org/10.6028/NIST.SP.800-185.
41. *Goubin L. and Patarin J.* DES and differential power analysis. The "Duplication" method. LNCS, 1999, vol. 1717, pp. 158–172.
42. *Oswald E., Mangard S., Pramstaller N., and Rijmen V.* A side-channel analysis resistant description of the AES S-box. LNCS, 2005, vol. 3557, pp. 413–423.
43. *Bilgin B., Nikova S., Nikov V., et al.* Threshold implementations of all 3×3 and 4×4 S-boxes. LNCS, 2012, vol. 7428, pp. 76–91.
44. *Daemen J.* Changing of the guards: A simple and efficient method for achieving uniformity in threshold sharing. LNCS, 2017, vol. 10529, pp. 137–153.
45. *Piccione E., Andreoli S., Budaghyan L., et al.* An optimal universal construction for the threshold implementation of bijective S-boxes. IEEE Trans. Inform. Theory, 2023, vol. 69, iss. 10, pp. 6700–6710.

УДК 519.7 DOI 10.17223/20710410/63/3

BLIND SIGNATURE AS A SHIELD AGAINST BACKDOORS IN SMART CARDS

L. R. Akhmetzyanova, A. A. Babueva, A. A. Bozhko

CryptoPro, Moscow, Russia

E-mail: {lah, babueva, bozhko}@cryptopro.ru

The problem of signature forgery (including signature key recovery) in the presence of backdoors in the hardware or software of functional key carriers (smart cards) is considered. A new approach to solving the problem based on using blind signature schemes is proposed. It is shown that honest-signer blindness and honest-but-curious unforgeability of the blind signature schemes imply security against backdoors in smart cards. As a concrete example, we consider a blind version of the GOST signature scheme (the blind signature scheme proposed by Camenisch) and show that this scheme is resistant to backdoors under the single assumption that GOST is secure in the standard sense.

Keywords: *blind signature scheme, GOST R 34.10-2012, untrusted smart cards, backdoors.*

СХЕМЫ ПОДПИСИ ВСЛЕПУЮ КАК ЗАЩИТА ОТ ЗАКЛАДОК В СМАРТ-КАРТАХ

Л. Р. Ахметзянова, А. А. Бабуева, А. А. Божко

КриптоПро, г. Москва, Россия

Рассматривается задача обеспечения защиты от подделки подписи (в том числе за счёт восстановления ключа подписи) в условиях наличия закладок в аппаратном или программном обеспечении функциональных ключевых носителей (смарт-карт). Предлагается новый подход к решению задачи, основанный на использовании схем подписи вслепую. Показывается, что обеспечение схемой подписи вслепую свойств неотслеживаемости при условии честной генерации ключей и неподделываемости относительно «честного, но любопытного» нарушителя обеспечивает защиту от закладок в смарт-картах. В качестве конкретного примера рассматривается схема подписи вслепую на основе уравнения ГОСТ, предложенная Каменишем. Доказывается, что эта схема обеспечивает защиту от закладок при единственном предположении, что схема подписи ГОСТ обеспечивает свойство неподделываемости в стандартном смысле.

Ключевые слова: *схема подписи вслепую, ГОСТ Р 34.10-2012, недоверенные смарт-карты, закладки.*

1. Introduction

Consider an information system consisting of two components: a smart card (or token) used as a functional key storage and an application installed on a user device (desktop or handheld). The applied function of a system is to compute a signature of any document transmitted via the application with a key uploaded and stored on a smart card. The components usually interact in the following way:

1) the user opens the application, chooses the document to be signed and pushes the button "Sign";
2) the application connects to the smart card (usually by setting up a password-protected secure channel [1]) and sends it the selected document or document hash value;
3) the smart card computes the signature value of the document on its own under a stored private key and returns the computed value to the application;
4) the application verifies the received signature value and returns the signed document to the user.

The use of smart cards with unrecoverable on-board private key cryptography is considered one of the most secure approaches to key management that allows to protect against adversaries which can get physical access to key storage devices. However, it has its own disadvantages. Unlike software applications, which can be open source and therefore fully verified, self-compiled and securely installed by anyone, smart card development is a much more technically complex process that is usually carried out by companies that specialize in the field. Indeed, the signing code is often hardwired directly into smart card microchips to improve performance and, consequently, cannot be openly verified by outsiders: the users are given a ready-to-use "black-box" device. This makes it possible for unscrupulous developers to implement a malicious code.

In the paper, we address the security issues that arise when the smart card used is seen as an untrusted component and is believed to contain backdoors. In the context of systems based on ElGamal or Schnorr type signature schemes, these issues are highly crucial, since this type of signature uses one-time random values that are generated using a smart card and whose compromise immediately results in the recovery of the user's private key. For instance, malicious smart card can use low-entropy one-time values allowing an adversary (e.g., company implementing this backdoor) to perform the brute force attack and recover the user key from a correct signature.

Related work. The paper [2] is devoted to these issues. Firstly, the paper introduces two types of adversary to be considered:

External adversary: it models an honest-but-curious adversary acting on the application side; the adversary's goal is to make a new correct pair (message, signature) without interacting with a smart card or, in other words, to make a forgery. Note that this threat includes the stronger one — key recovery. Consideration of such adversaries covers the scenario where only honest user interacts with smart card through verified and trusted application, but this application is less protected from memory leaks compared to the smart card.

Remark 1. Note that this type does not cover the capabilities of active adversaries that can directly interact (e.g., using its own malicious application) with the smart card. In practice, it means that the adversary that steals the smart card cannot get access to its API. Considering only passive adversaries is justified by the fact that smart cards are usually also protected with a memorable password that should be entered by the human to get access to its API [3].

Adversary with agent: this adversary is supposed to consist of two parts. The first part is *a fully active adversary* on the smart card side but it can interact only with the trusted application, i.e., there is no other channel for data transmition from smart card. The second part collects the pairs (message, signature) computed by application and malicious smart card — this is the agent. Similar to the first type of adversary, the goal is to make a forgery.

In order to deal with these adversaries, the paper [2] proposed a solution for the GOST signature scheme [4] based on the usage of the interactive Schnorr zero-knowledge proof. This protocol is executed with the main signing algorithm and its purpose is to prove to the application that smart card is using the "correct" one-time value (for details see the original paper). This solution has the following two significant drawbacks:

1) it allows to protect against *the semi-trusted* smart card only: the crucial assumption for security is that low-level (short) arithmetic operations are implemented correctly in the smart cards. Although it is realistic assumption, there are no convenient ways to validate this on practice;
2) it is not secure if the smart card can terminate the signing process with the error on the application side. The paper [2] describes the concrete attack where the malicious smart card successfully completes the signing protocol only if certain bits of resulting signature are equal to certain bits of the signing key. One approach to protect against this attack is to delete the private signing key immediately after such errors occur. However, in practice, errors can occur not only due to the adversary's actions, but also due to technical failures, so deleting the key after each error is not a practical solution.

Our contribution. To negate the disadvantages mentioned above, we propose a new approach, the main idea of which is to use the "blind versions" of the signature schemes. The blind signature schemes firstly introduced by Chaum [5] allow one party called User to obtain a signature for an arbitrary message after interacting with another party called Signer holding a signing key in such a way that the Signer does not receive any information about either the message or the signature value (blindness property) and the User can compute only one single signature per interaction with the Signer (unforgeability property).

In the context of considered signing system, the smart card executes the Signer side and the application executes the User side. Due to the blindness property, the malicious smart card learns no information about the signature during the protocol execution and, therefore, cannot "control" the signature values, e.g., covertly transmiting bits of private key through the signature values.

In this paper, we introduce two new security notions for blind signature schemes: honest-but-curious unforgeability and backdoor resilience, which characterize the security of the proposed solution against external adversary and adversary with agent. We show that honest-signer blindness (where an adversary cannot affect the key generation algorithm) and standard unforgeability imply backdoor resilience. Moreover, for the GOST signature scheme we propose the concrete blind signature scheme for use: the Camenisch scheme [6] that provides perfect blindness (and thus honest-signer blindness) and honest-but-curious unforgeability (and thus standard unforgeability), which is implied only by the unforgeability of GOST. It means that the Camenisch blind signature scheme provides the security against both external adversary and adversary with agent under a single assumption that the GOST signature scheme provides standard security, i.e., is unforgeable under the chosen message attack.

The rest of the paper is organised as follows. In Section 2 we remind the definitions of conventional and blind signature schemes, the accompanying security notions are given. In Section 3 the formal definitions of honest-but-curious unforgeability and backdoor resilience are introduced. Section 4 is devoted to the formal analysis and Section 5 considers the Camenish blind signature scheme in details.

2. Basic definitions

(Conventional) signature schemes. The conventional signature scheme SS is determined by three algorithms:

— $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{SS.KGen}()$: a key generation algorithm that outputs a secret key sk and a public key pk;

— $\sigma \leftarrow \mathsf{SS.Sig}(\mathsf{sk}, m)$: a signature generation algorithm that takes a secret key sk and a message m and returns a signature σ;

— $b \leftarrow \mathsf{SS.Vf}(\mathsf{pk}, m, \sigma)$: a (deterministic) verification algorithm that takes a public key pk, a message m, and a signature σ, and returns 1 if σ is valid on m under pk and 0 otherwise.

Correctness. We say that SS is correct if for each message m, with probability one over the sample of parameters and the key pair $(\mathsf{sk}, \mathsf{pk})$, the equality $\mathsf{SS.Vf}(\mathsf{pk}, m, \mathsf{SS.Sig}(\mathsf{sk}, m)) = 1$ holds.

Blind signature schemes. The blind signature scheme BS is defined in the same way as the conventional signature scheme except for the signature generation algorithm which is replaced by the following protocol:

— $(b, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m)\rangle$: an interactive signing protocol that is run between a Signer with a secret key sk and a User with a public key pk and a message m; the Signer outputs $b = 1$ if the interaction completes successfully and $b = 0$ otherwise, while the User outputs σ that is either the resulting signature or an error message.

Correctness. We say that BS is correct if for each message m, with probability one over the sample of parameters and the key pair $(\mathsf{sk}, \mathsf{pk})$, the signing protocol $\langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m)\rangle$ completes with $(1, \sigma)$, $\sigma \neq \perp$, such that $\mathsf{BS.Vf}(\mathsf{pk}, m, \sigma) = 1$.

In the paper, we are interested in the blind signature schemes that are based on some conventional signature schemes. We will say that the BS scheme is *a blind version* of the SS scheme, if the KGen and Vf algorithms of these schemes coincide and for any $(\mathsf{sk}, \mathsf{pk})$, any message m, and any signature σ

$$\Pr[(1, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m)\rangle] = \Pr[\sigma \leftarrow \mathsf{SS.Sig}(\mathsf{sk}, m)],$$

where the corresponding probability spaces are determined by the randomness used in the signing protocol and signing algorithm.

Three-move blind signature schemes. For simplicity, this paper focuses on three-move blind signature schemes. For such schemes, the signing protocol can be described as follows:

$$\begin{aligned}
&(msg_{S,1}, state_S) \leftarrow \mathsf{BS.Signer}_1(\mathsf{sk}),\\
&(msg_U, state_U) \leftarrow \mathsf{BS.User}_1((\mathsf{pk}, m), msg_{S,1}),\\
&(msg_{S,2}, b) \leftarrow \mathsf{BS.Signer}_2(state_S, msg_{U,1})\\
&\sigma \leftarrow \mathsf{BS.User}_2(state_U, msg_{S,2}),
\end{aligned}$$

where $msg_{role,i}$, $role \in \{U, S\}$, is the i-th message sent by the side with role $role$ during the protocol execution. The variable $state_{role}$ is aimed to keep the internal state for using on the next protocol stage. Here the User performs the $\mathsf{BS.User}_1$ and $\mathsf{BS.User}_2$ functions, and the Signer performs the $\mathsf{BS.Signer}_1$ and $\mathsf{BS.Signer}_2$ functions during the protocol execution.

Security notions. Next, we describe security concepts using a game-based approach [7]. This approach uses the notion of "experiment" played between a challenger and an adversary. The adversary and challenger are modelled using consistent interactive

probabilistic algorithms. The challenger simulates the functioning of the analysed cryptographic scheme for the adversary and may provide him access to one or more oracles. The parameters of an adversary $\mathcal{A}$ are its computational resources (for a fixed model of computation and a method of encoding) and oracles query complexity. The query complexity usually includes the number of queries. Denote by $\mathsf{Adv}_S^{\mathrm{M}}(\mathcal{A})$ the measure of the success of the adversary $\mathcal{A}$ in realizing a certain threat, defined by the security notion M for the cryptographic scheme S.

The standard security notion for (probabilistic) signature schemes is strong unforgeability under chosen message attack (sUF-CMA). The formal definition is given below.

Definition 1. For an adversary $\mathcal{A}$ and a signature scheme SS:

$$\mathsf{Adv}_{\mathsf{SS}}^{\text{sUF-CMA}}(\mathcal{A}) = \Pr\big[\mathbf{Exp}_{\mathsf{SS}}^{\text{sUF-CMA}}(\mathcal{A}) \to 1\big],$$

where the $\mathbf{Exp}_{\mathsf{SS}}^{\text{sUF-CMA}}(\mathcal{A})$ experiment is defined in the following way:

$\mathbf{Exp}_{\mathsf{SS}}^{\text{sUF-CMA}}(\mathcal{A})$	Oracle $Sign(m)$
1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{SS.KGen}()$	1 : $\sigma \leftarrow \mathsf{SS.Sig}(\mathsf{sk}, m)$
2 : $\mathcal{L} \leftarrow \varnothing$	2 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{(m, \sigma)\}$
3 : $(m, \sigma) \leftarrow \mathcal{A}^{Sign}(\mathsf{pk})$	3 : **return** σ
4 : **if** $(m, \sigma) \in \mathcal{L}$: **return** 0	
5 : **return** $\mathsf{SS.Vf}(\mathsf{pk}, m, \sigma)$	

Remark 2. The same security notion can be applied to the blind version BS of the signature scheme SS. In this case, line 1 in the *Sign* oracle is replaced with the line $(1, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m)\rangle$. It is easy to see that for such schemes sUF-CMA-security of the SS scheme implies sUF-CMA-security of the BS scheme and vice versa.

The standard notions for blind signature schemes are one-more unforgeability (OMUF notion that considers a malicious user in the parallel setting) and blindness (Blind notion that considers a malicious signer), their formal definitions can be found in [8]. Note that the original definition of blindness proposed in [9] considers an honest signer that can not affect key generation process. In the paper, we consider only this weak notion and refer to it as "honest-signer blindness" (HS-Blind notion).

Honest-signer blindness. Informally, the blind signature scheme provides blindness if there is no way to link a (message, signature) pair to the certain execution of the signing protocol. In the context of strong notion, the adversary can fully control the Signer side. In the context of weaker HS-Blind notion, we assume that the key pair is generated honestly at the beginning of the experiment. The formal definition is given below.

Definition 2. For an adversary $\mathcal{A}$ and three-move blind scheme BS:

$$\mathsf{Adv}_{\mathsf{BS}}^{\text{HS-Blind}}(\mathcal{A}) = \Pr\Big[\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},1}(\mathcal{A}) \to 1\Big] - \Pr\Big[\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},0}(\mathcal{A}) \to 1\Big],$$

where the $\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},b}(\mathcal{A})$, $b \in \{0, 1\}$, experiments are defined in the following way:

$\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},b}(\mathcal{A})$

1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
2 : $b_0 \leftarrow b$
3 : $b_1 \leftarrow 1 - b$
4 : $b' \leftarrow \mathcal{A}^{Init, User_1, User_2}(\mathsf{sk}, \mathsf{pk})$
5 : **return** b'

Oracle $Init(m_0, m_1)$

1 : $sess_0 \leftarrow \mathsf{init}$
2 : $sess_1 \leftarrow \mathsf{init}$

Oracle $User_1(i, msg)$

1 : **if** $i \notin \{0, 1\} \vee sess_i \neq \mathsf{init}$: **return** $\perp$
2 : $sess_i \leftarrow \mathsf{open}$
3 : $(msg_i, state_i) \leftarrow \mathsf{BS.User}_1((\mathsf{pk}, m_{b_i}), msg)$
4 : **return** msg_i

Oracle $User_2(i, msg)$

1 : **if** $sess_i \neq \mathsf{open}$: **return** $\perp$
2 : $sess_i \leftarrow \mathsf{closed}$
3 : $\sigma_{b_i} \leftarrow \mathsf{BS.User}_2(state_i, msg)$
4 : **if** $sess_0 = sess_1 = \mathsf{closed}$:
5 : **if** $\sigma_{b_0} = \perp \vee \sigma_{b_1} = \perp$: **return** $(\perp, \perp)$
6 : **return** (σ_0, σ_1)
7 : **return** ε

3. New security notions for blind signatures

Here we give the formal game-based definitions of two security notions: backdoor resilience and honest-but-curious unforgeability.

Backdoor resilience/Security against adversary with agent

Consider an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ consisting of two algorithms. An algorithm $\mathcal{A}_2$ denotes the part of the adversary $\mathcal{A}$ collecting signature values for adaptively chosen messages. An algorithm $\mathcal{A}_1$ denotes the agent acting on the backdoored smart card side.

The formal definition of BDres (BackDoor resilience) for blind signature schemes is given below (see Definition 3). We parametrize this security model by the value k which determines the number of attempts by the challenger to produce a correct signature for a message (details are described below).

Definition 3. For any adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ and blind signature scheme BS:

$$\mathsf{Adv}_{\mathsf{BS}}^{\text{BDres}_k}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{\text{BDres}_k}(\mathcal{A}) \to 1\right],$$

where the $\mathbf{Exp}_{\mathsf{BS}}^{\text{BDres}_k}(\mathcal{A})$, $k \in \mathbb{N}$, experiment is defined in the following way:

$\mathbf{Exp}_{\mathsf{BS}}^{\text{BDres}_k}(\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2))$

1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
2 : $\mathcal{L} \leftarrow \varnothing$
3 : $\mathsf{lost} \leftarrow \mathsf{false}$
4 : $st \leftarrow \mathcal{A}_1(\mathsf{sk}, \mathsf{pk})$
5 : $(m, \sigma) \xleftarrow{\$} \mathcal{A}_2^{Sign}(\mathsf{pk})$
6 : **if** $((m, \sigma) \in \mathcal{L}) \vee (\mathsf{lost} = \mathsf{true})$:
7 : **return** 0
8 : **return** $\mathsf{BS.Vf}(\mathsf{pk}, m, \sigma)$

Oracle $Sign(m)$

1 : $i \leftarrow 0$
2 : **do**
3 : $(st, \sigma) \leftarrow \langle \mathcal{A}_1(st), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
4 : $i \leftarrow i + 1$
5 : **until** $(i \geqslant k) \vee (\sigma \neq \perp)$
6 : **if** $\sigma = \perp$:
7 : $\mathsf{lost} \leftarrow \mathsf{true}$
8 : **return** $\perp$
9 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{(m, \sigma)\}$
10 : **return** σ

At the experiment initialization stage (line 1), the challenger modeling an honest application generates a key pair $(\mathsf{sk}, \mathsf{pk})$ according to the key generation algorithm and sends to $\mathcal{A}_1$ a pair $(\mathsf{sk}, \mathsf{pk})$ (line 4), while to $\mathcal{A}_2$ it sends a verification key pk only (line 5). This stage models the trusted process of generating keys, issuing corresponding certificate and uploading key material onto the smart card.

The $\mathcal{A}_2$ algorithm can make queries to the challenger signing oracle $Sign$ that returns signature values σ for messages m arbitrarily chosen by the adversary. Each signature value is computed during the execution of the blind signing protocol between oracle that models the honest User side and the $\mathcal{A}_1$ algorithm modeling the malicious Signer side (line 3 in the oracle). Here the variable st denotes the internal state of $\mathcal{A}_1$ that is kept from call to call.

The $\mathcal{A}_1$ algorithm is allowed to terminate the protocol execution with an error $\perp$ on the User side (line 5 in the oracle). For this reason, for any requested message m the oracle makes k attempts to compute a correct signature, and in the case when all k attempts fail, challenger returns 0 as a game result (meaning that the adversary loses, see line 7 in the oracle). This simulates the scenario where the smart card has failed and is no longer used.

Remark 3. Note that if the algorithm $\mathcal{A}_2$ can obtain errors from the signing oracle, then there is always a trivial attack. Consider the agent $\mathcal{A}_1$ that successfully completes the signing protocol execution iff i-th bit of sk is equal to 1, where i is a sequence number of query to oracle. Having such an agent on the smart card side, the $\mathcal{A}_2$ algorithm can recover all bits of signing key and trivially make a forgery.

To break a backdoor resilience, the algorithm $\mathcal{A}_2$ is needed to make a forgery (m, σ) containing a signature σ that has not previously been returned by the oracle $Sign$ in response to a query m.

Honest-but-curious unforgeability/Security against external adversary

This notion considers only an honest-but-curious adversary acting on the User side. This adversary can adaptively choose messages to be signed by making a query m to the oracle and obtain in return a signature σ and a specific value $view$. The latter consits of all incoming messages and the values of all random parameters processed and sampled by the User side during the execution of the signing protocol. This simulates the scenario, where the adversary gets an access to the memory of trusted application.

The formal definition of HBC-UF is given below.

Definition 4. For an adversary $\mathcal{A}$ and a blind signature scheme BS:

$$\mathsf{Adv}^{\text{HBC-UF}}_{\mathsf{BS}}(\mathcal{A}) = \Pr\left[\mathbf{Exp}^{\text{HBC-UF}}_{\mathsf{BS}}(\mathcal{A}) \to 1\right],$$

where the $\mathbf{Exp}^{\text{HBC-UF}}_{\mathsf{BS}}(\mathcal{A})$ experiment is defined in the following way:

$\mathbf{Exp}^{\text{HBC-UF}}_{\mathsf{BS}}(\mathcal{A})$

1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
2 : $\mathcal{L} \leftarrow \varnothing$
3 : $(m, \sigma) \leftarrow \mathcal{A}^{Sign}(\mathsf{pk})$
4 : **if** $(m, \sigma) \in \mathcal{L}$: **return** 0
5 : **return** $\mathsf{BS.Vf}(\mathsf{pk}, m, \sigma)$

Oracle $Sign(m)$

1 : $(1, (\sigma; view)) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
2 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{(m, \sigma)\}$
3 : **return** $\sigma, view$

It is easy to see that for any blind signature scheme HBC-UF-security implies sUF-CMA-security.

4. Security analysis

4.1. Backdoor resilience / Security against adversary with agent

In this section, we prove that honest-signer blindness and standard unforgeability (sUF-CMA) imply backdoor resilience.

Theorem 1. Fix $k \in \mathbb{N}$. For any adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ in the BDres_k model with summary time complexity at most t making at most q queries to the signing oracle, there exist an adversary $\mathcal{B}$ in the sUF-CMA model making at most q queries to the signing oracle and an adversary $\mathcal{C}$ in the HS-Blind model such that

$$\mathsf{Adv}_{\mathsf{BS}}^{\mathrm{BDres}_k}(\mathcal{A}) \leqslant \mathsf{Adv}_{\mathsf{BS}}^{\text{sUF-CMA}}(\mathcal{B}) + q \cdot k \cdot \mathsf{Adv}_{\mathsf{BS}}^{\text{HS-Blind}}(\mathcal{C}).$$

Time complexities of $\mathcal{B}$ and $\mathcal{C}$ are at most t and tkq correspondingly.

Remark 4. If the blind signature scheme provides perfect blindness (i.e., $\mathsf{Adv}_{\mathsf{BS}}^{\text{HS-Blind}}(\mathcal{C}) = 0$ for any $\mathcal{C}$ with any time complexity), then the bound is transformed as follows:

$$\mathsf{Adv}_{\mathsf{BS}}^{\mathrm{BDres}_k}(\mathcal{A}) \leqslant \mathsf{Adv}_{\mathsf{BS}}^{\text{sUF-CMA}}(\mathcal{B}).$$

From the perspective of using conventional signature scheme SS, this inequality means that in order to provide backdoor resilience, it is enough for this signature scheme to have its blind version BS (with $\mathsf{Adv}_{\mathsf{BS}}^{\text{sUF-CMA}}(\mathcal{B}) = \mathsf{Adv}_{\mathsf{SS}}^{\text{sUF-CMA}}(\mathcal{B})$) and to be unforgable in the standard model. Note, that the bound does not depend on k, so this value can be chosen arbitrarily by the application developers.

Remark 5. For clarity, the proof is carried out for three-move blind signatures, but the proof does not base on any specific features of such scheme type and can be easily adapted for any-move blind signatures.

Proof. The proof consits of two parts.

Part 1. Consider the consequence of several experiments, where each next experiment slightly differs from the previous one.

Game 0. Let $\mathbf{Exp}_{\mathsf{BS}}^{0}(\mathcal{A}) = \mathbf{Exp}_{\mathsf{BS}}^{\mathrm{BDres}_k}(\mathcal{A})$.

Game 1. Consider the following modified experiment $\mathbf{Exp}_{\mathsf{BS}}^{1}(\mathcal{A})$:

$\mathbf{Exp}_{\mathsf{BS}}^{1}(\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2))$

$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
$\mathcal{L} \leftarrow \varnothing$
$\mathsf{lost} \leftarrow \mathsf{false}$
$st \leftarrow \mathcal{A}_1(\mathsf{sk}, \mathsf{pk})$
$(m, \sigma) \xleftarrow{\$} \mathcal{A}_2^{Sign}(\mathsf{pk})$
if $((m, \sigma) \in \mathcal{L}) \vee (\mathsf{lost} = \mathsf{true})$:
  return 0
return $\mathsf{BS.Vf}(\mathsf{pk}, m, \sigma)$

Oracle $Sign(m)$

$i \leftarrow 0$
do
  $(st, \sigma) \leftarrow \langle \mathcal{A}_1(st), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
  if $\sigma \neq \perp$:
    $(1, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
  $i \leftarrow i + 1$
until $(i \geqslant k) \vee (\sigma \neq \perp)$
if $\sigma = \perp$:
  $\mathsf{lost} \leftarrow \mathsf{true}$
  return $\perp$
$\mathcal{L} \leftarrow \mathcal{L} \cup \{(m, \sigma)\}$
return σ

$\mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A})$ differs from $\mathbf{Exp}^0_{\mathsf{BS}}(\mathcal{A})$ in additional lines 4 and 5 of the $Sign$ oracle code. If the oracle, interacting with the agent $\mathcal{A}_1$ as a user, completes the signing protocol with a correct signature, then the oracle recomputes a new signature honestly executing the signing protocol on its own (without interaction with the agent). The second part of the proof is devoted to estimation of winning probability difference for $\mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A})$ and $\mathbf{Exp}^0_{\mathsf{BS}}(\mathcal{A})$.

Game 2. Consider the next modification: the experiment $\mathbf{Exp}^2_{\mathsf{BS}}(\mathcal{A})$. Here the oracle always responses to requests of $\mathcal{A}_2$ with a correct honestly generated signature even in the case when $\mathcal{A}_1$ provokes errors k times in a row that sets the flag lost in $\mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A})$.

$\mathbf{Exp}^2_{\mathsf{BS}}(\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2))$

1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
2 : $\mathcal{L} \leftarrow \varnothing$
3 : $st \leftarrow \mathcal{A}_1(\mathsf{sk}, \mathsf{pk})$
4 : $(m, \sigma) \xleftarrow{\$} \mathcal{A}_2^{Sign}(\mathsf{pk})$
5 : **if** $((m, \sigma) \in \mathcal{L})$:
6 : **return** 0
7 : **return** $\mathsf{BS.Vf}(\mathsf{pk}, m, \sigma)$

Oracle $Sign(m)$

1 : $i \leftarrow 0$
2 : **do**
3 : $(st, \sigma) \leftarrow \langle \mathcal{A}_1(st), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
4 : $i \leftarrow i + 1$
5 : **until** $(i \geqslant k) \vee (\sigma \neq \perp)$
6 : $(1, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
7 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{(m, \sigma)\}$
8 : **return** σ

For this experiment:

$$\Pr\left[\mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A}) \to 1\right] = \Pr\left[\mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A}) \to 1 \wedge (\mathsf{lost} = \mathsf{false})\right] + \\ + \underbrace{\Pr\left[\mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A}) \to 1 \wedge (\mathsf{lost} = \mathsf{true})\right]}_{= 0 \text{ due to line 6 of } \mathbf{Exp}^1_{\mathsf{BS}}(\mathcal{A})} \leqslant \Pr\left[\mathbf{Exp}^2_{\mathsf{BS}}(\mathcal{A}) \to 1\right].$$

Game 3. Note that in the $\mathbf{Exp}^2_{\mathsf{BS}}(\mathcal{A})$ experiment the agent $\mathcal{A}_1$ can be thrown away, since it can no longer influence the value of the signature (see the $\mathbf{Exp}^3_{\mathsf{BS}}(\mathcal{A}_2)$ experiment below). Note that $\Pr\left[\mathbf{Exp}^2_{\mathsf{BS}}(\mathcal{A}_1, \mathcal{A}_2) \to 1\right] = \Pr\left[\mathbf{Exp}^3_{\mathsf{BS}}(\mathcal{A}_2) \to 1\right]$.

$\mathbf{Exp}^3_{\mathsf{BS}}(\mathcal{A}_2)$

1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
2 : $\mathcal{L} \leftarrow \varnothing$
3 : $(m, \sigma) \leftarrow \mathcal{A}_2^{Sign}(\mathsf{pk})$
4 : **if** $(m, \sigma) \in \mathcal{L}$:
5 : **return** 0
6 : **return** $\mathsf{BS.Vf}(\mathsf{pk}, m, \sigma)$

Oracle $Sign(m)$

1 : $(1, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
2 : $\mathcal{L} \leftarrow \mathcal{L} \cup \{(m, \sigma)\}$
3 : **return** σ

Note that $\mathbf{Exp}^3_{\mathsf{BS}}$ is exactly the experiment $\mathbf{Exp}^{\text{sUF-CMA}}_{\mathsf{BS}}$, therefore $\Pr\left[\mathbf{Exp}^2_{\mathsf{BS}}(\mathcal{A}) \to 1\right] \leqslant$ $\leqslant \mathsf{Adv}^{\text{sUF-CMA}}_{\mathsf{BS}}(\mathcal{B})$ for $\mathcal{B} = \mathcal{A}_2$.

P a r t 2 . To finalize the proof, we construct an adversary $\mathcal{C}$ breaking the blindness property. Introduce the following auxiliary experiment:

$\mathbf{Exp}^{4,b}_{\mathsf{BS}}(\mathcal{C})$

1 : $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{BS.KGen}()$
2 : $b' \xleftarrow{\$} \mathcal{C}^{Init, User_1, User_2}(\mathsf{sk}, \mathsf{pk})$
3 : **return** b'

Oracle $Init(m)$

1 : $sess \leftarrow \mathsf{init}$

Oracle $User_1(msg)$

1 : **if** $sess \neq \mathsf{init}$: **return** $\perp$
2 : $sess \leftarrow \mathsf{open}$
3 : $(msg, state) \leftarrow \mathsf{BS.User}_1((\mathsf{pk}, m), msg)$
4 : **return** msg

Oracle $User_2(msg)$

1 : **if** $sess \neq \mathsf{open}$: **return** $\perp$
2 : $\sigma \leftarrow \mathsf{BS.User}_2(state, msg)$
3 : **if** $(\sigma \neq \perp) \wedge (b = 0)$:
4 : $\quad (1, \sigma) \leftarrow \langle \mathsf{BS.Signer}(\mathsf{sk}), \mathsf{BS.User}(\mathsf{pk}, m) \rangle$
5 : **return** σ

Here an adversary can make only one query to each oracle (execute only one session). The adversary obtains a signature value generated by the oracles intacting with adversary if $b = 1$, and a signature computed according to the protocol otherwise. Note that if the adversary provokes an error in the session, then it always gets $\perp$ from the $User_2$ oracle regardless of bit b.

Using a standard technique called "hybrid argument" [10], it can be trivially shown that there exists an adversary $\mathcal{C}'$ such that

$$\Pr\left[\mathbf{Exp}^{0}_{\mathsf{BS}}(\mathcal{A}) \to 1\right] - \Pr\left[\mathbf{Exp}^{1}_{\mathsf{BS}}(\mathcal{A}) \to 1\right] = \\ = q \cdot k \left(\Pr\left[\mathbf{Exp}^{4,1}_{\mathsf{BS}}(\mathcal{C}') \to 1\right] - \Pr\left[\mathbf{Exp}^{4,0}_{\mathsf{BS}}(\mathcal{C}') \to 1\right]\right).$$

Now let construct an adversary $\mathcal{C}$ using $\mathcal{C}'$ as a black box. The adversary $\mathcal{C}$ acts in the following way:

1) The adversary $\mathcal{C}$ obtains $(\mathsf{sk}, \mathsf{pk})$ and transmits this value to $\mathcal{C}'$.
2) When $\mathcal{C}'$ makes a query m to the $Init$ oracle, $\mathcal{C}$ makes a query (m, m) to its own $Init$ oracle.
3) After starting sessions, the adversary $\mathcal{C}$ firstly executes $sess_0$ according to the protocols:
   a) it computes $(msg^0_{S,1}, state_S) \leftarrow \mathsf{BS.Signer}_1(\mathsf{sk})$ and makes a query $(0, msg^0_{S,1})$ to its own $User_1$ oracle;
   б) upon receiving $msg^0_{U,1}$, the adversary $\mathcal{C}$ computes

$$(msg^0_{S,2}, 1) \leftarrow \mathsf{BS.Signer}_2(state_S, msg_{U,1})$$

   and makes a query $(0, msg^0_{S,2})$ to its own $User_2$ oracle, receiving the ε value.
   Note that $\sigma_{b_0} \neq \perp$ due to the correctness property of the blind signature scheme.
4) Then the adversary $\mathcal{C}$ intercepts all queries of $\mathcal{C}'$ and simply passes them to $sess_1$:
   a) intercepting from $\mathcal{C}'$ a query msg_1 to the $User_1$ oracle, $\mathcal{C}$ makes a query $(1, msg^1_{S,1})$, where $msg^1_{S,1} = msg_1$, to its own $User_1$ oracle and directly transmits the response $msg^1_{U,1}$ to $\mathcal{C}'$;
   б) intercepting from $\mathcal{C}'$ a query msg_2 to the $User_2$ oracle, $\mathcal{C}$ makes a query $(1, msg^1_{S,2})$, where $msg^1_{S,2} = msg_2$, to its own $User_2$ oracle. $\mathcal{C}$ receives (σ_0, σ_1) and returns to $\mathcal{C}'$ the fist component σ_0. Note that (σ_0, σ_1) can be $(\perp, \perp)$.
5) $\mathcal{C}$ returns the same bit as $\mathcal{C}'$ returns.

If the $\mathcal{C}$ interacts with the experimentator $\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},1}$ ($\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},0}$), then $\sigma_0 = \sigma_{b_1}$ ($\sigma_0 = \sigma_{b_0}$). Moreover, $\mathcal{C}$ returns $\perp$ at stage 4 iff $\mathcal{C}'$ provokes error in $sess_1$ that perfectly coincides with $\mathbf{Exp}_{\mathsf{BS}}^{4}$. Thus,

$$\Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{4,1}(\mathcal{C}') \to 1\right] = \Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},1}(\mathcal{C}) \to 1\right],$$
$$\Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{4,0}(\mathcal{C}') \to 1\right] = \Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},0}(\mathcal{C}) \to 1\right].$$

Summing up,

$$\Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{0}(\mathcal{A}) \to 1\right] - \Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{1}(\mathcal{A}) \to 1\right] = q \cdot k \left(\Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{4,1}(\mathcal{C}') \to 1\right] - \Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{4,0}(\mathcal{C}') \to 1\right]\right) =$$
$$= q \cdot k \left(\Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},1}(\mathcal{C}) \to 1\right] - \Pr\left[\mathbf{Exp}_{\mathsf{BS}}^{\text{HS-Blind},0}(\mathcal{C}) \to 1\right]\right) = q \cdot k \cdot \mathsf{Adv}_{\mathsf{BS}}^{\text{HS-Blind}}(\mathcal{C}).$$

The theorem 1 is proven. ■

4.2. Honest-but-curious unforgeability / Security against external adversary

Here we define the particular class of blind signature schemes based on ElGamal signature equation that provides the honest-but-curious unforgeability. Namely, for such schemes we construct the security reduction to the unforgeability of the base ElGamal signature scheme. Note that all known ElGamal blind signature schemes do not provide strong unforgeability [11].

At first, let us introduce the required notations. We denote the group of points of the elliptic curve over the prime field as $\mathbb{G}$, the order of the prime subgroup of $\mathbb{G}$ as q, an elliptic curve point of order q as P and zero point as $\mathcal{O}$. We denote by H the hash function that maps binary strings to elements from $\mathbb{Z}_q$ and assume that all field operations are performed modulo q.

ElGamal blind signature scheme

The generalized ElGamal signature scheme was introduced in [12] and further extended in [13], we denote it by GenEG scheme. A key generation algorithm in this scheme involves picking random d uniformly from $\mathbb{Z}_q^*$ (secret signing key) and defining $Q = dP$ (public verifying key). A signature for message m is a pair (r, s), where $r = (kP).x \bmod q$ for some k picked uniformly at random from $\mathbb{Z}_q^*$ and s is computed from the ElGamal signature equation EG:

$$EG(d, k, r, e, s) = 0,$$

where $e = H(m)$. All possible EG equations are listed in [12]. To ensure functionality and security, certain r, e, s values need to be excluded.

ElGamal blind signature scheme, denoted by GenEG-BS, was introduced in [11]. A key generation and verification algorithms in GenEG-BS scheme are the same as in the base GenEG scheme. An interactive signing protocol assumes that the Signer performs ElGamal signature generating algorithm for the e value received from the User, the User algorithm is not determined and can be arbitrary. The parameters of the signing protocol are the base point P, public key Q, and the message m, we denote them by par.

We impose the additional requirements on the algorithm performed by the User:

— all blinding factors (we denote them by rnd) used by the User are selected according to some distribution $\mathcal{D}$ that is independent on the values received from the Signer;

— the first component of the signature r' is the x-coordinate of the R' point, which is computed as a result of applying the function parameterized by the par value (we denote it by $\mathcal{L}_1^{par}$) that takes as arguments the R point received from the Signer and rnd values. This function is linear by R for all rnd values generated according to the protocol;
— the second component of the signature s' is computed as a result of applying the function parameterized by the par value (we denote it by $\mathcal{L}_2^{par}$) that takes as arguments the s value received from the Signer, rnd values, and point R. This function is linear by s for all rnd and R values generated according to the protocol.

We denote such a scheme by $\mathsf{GenEG\text{-}BS}_{\mathcal{L}}$ scheme. The corresponding signing protocol is illustrated in Fig. 1.

The signing protocol

Signer(d)		**User**(Q, m)
$k \xleftarrow{\mathcal{U}} \mathbb{Z}_q^*$		
$R \leftarrow kP$	$\xrightarrow{\quad R \quad}$	$rnd \xleftarrow{\mathcal{D}}$
		$R' \leftarrow \mathcal{L}_1^{par}(R, rnd)$
		$r' \leftarrow R'.x \bmod q$
$r \leftarrow R.x \bmod q$	$\xleftarrow{\quad e \quad}$	compute e
if $\exists!\ s\colon EG(d, k, r, e, s) = 0$ find s		
else : **return** 0	$\xrightarrow{\quad s \quad}$	
		$s' \leftarrow \mathcal{L}_2^{par}(s, rnd, R)$
return 1		**return** (r', s')

Fig. 1. The signing protocol in $\mathsf{GenEG\text{-}BS}_{\mathcal{L}}$ scheme

Let us show that the $\mathsf{GenEG\text{-}BS}_{\mathcal{L}}$ scheme is indeed the blind version of the GenEG scheme, i.e., provides the same distribution on the signature values. The distribution on GenEG signatures is defined by the uniform distribution on k values. The distribution on $\mathsf{GenEG\text{-}BS}_{\mathcal{L}}$ signatures is defined by the distribution on k' values, where k' is such that $(k'P).x \bmod q = r'$. The k' value is linear by k since R' value is linear by R and rnd values are chosen independently on R. Thus, the distribution on k' values is also uniform.

Note that the User view in the $\mathsf{GenEG\text{-}BS}_{\mathcal{L}}$ scheme consists of the incoming messages R, s and the blinding factors rnd sampled by the User.

Now we are ready to construct the security reduction to the unforgeability of the conventional ElGamal signature scheme.

Theorem 2. For any adversary $\mathcal{A}$ for $\mathsf{GenEG\text{-}BS}_{\mathcal{L}}$ scheme in the HBC-UF model with time complexity at most t making at most q queries to the signing oracle, there exist an adversary $\mathcal{B}$ for the conventional GenEG scheme in the sUF-CMA model with the same time complexity at most t making at most q queries to the signing oracle such that

$$\mathsf{Adv}^{\text{HBC-UF}}_{\mathsf{GenEG\text{-}BS}_{\mathcal{L}}}(\mathcal{A}) \leqslant \mathsf{Adv}^{\text{sUF-CMA}}_{\mathsf{GenEG}}(\mathcal{B}).$$

Proof. Let construct the adversary $\mathcal{B}$ for the conventional GenEG scheme. The adversary $\mathcal{B}$ uses the adversary $\mathcal{A}$ as a black box. It intercepts the queries of the adversary $\mathcal{A}$ to the signing oracle and process them by itself using its own signing oracle in the following way.

Receiving the query m, adversary $\mathcal{B}$ forwards m to its own oracle and receives the signature (r', s'). Then it reconstructs R' point from the verification algorithm and selects rnd value according to the distribution $\mathcal{D}$. After that, it calculates the R value using $\mathcal{L}_1^{-1}$ function and s value using $\mathcal{L}_2^{-1}$ function. It returns as an answer the signature (r', s') and the $view = (R, s, rnd)$.

Note that $\mathcal{B}$ generates exactly the same distribution on signature values since GenEG-BS$_{\mathcal{L}}$ scheme is the blind version of the GenEG scheme. The rnd value is chosen as in the honest execution of blind signature protocol, R and s values are also computed as in the honest execution, since $\mathcal{L}_1$ and $\mathcal{L}_2$ functions are unambiguously invertible.

When $\mathcal{A}$ returns a forgery, $\mathcal{B}$ translates it to its own challenger and stops. Obviously, if $\mathcal{A}$ wins, then $\mathcal{B}$ wins, whence follows the statement of the theorem. ■

Remark 6. The same result may be formulated for the Schnorr signature scheme and its blind version defined in [5]. The proof of the theorem is conducted in the same way.

5. GOST-based blind signature scheme

We propose to use the concrete blind signature scheme in case of building the protection for GOST signature scheme [4]. This scheme was proposed in [6] in 1994 and is commonly referred to as the Camenisch scheme. We provide the definition of this scheme in terms of elliptic group notation.

The key generation algorithm is the same as in the general ElGamal signature scheme and assumes picking secret key d uniformly from $\mathbb{Z}_q^*$ and defining public key Q as dP. The signing protocol is defined in Fig. 2. The verification procedure for the message m and the signature (r', s') assumes checking $r' \neq 0$ and checking the equality $r' = R'.x \bmod q$, where $R' = (e')^{-1}(s'P - r'Q)$, e' is equal to $H(m)$, if $H(m) \neq 0$, and to 1 otherwise. Note that the signing protocol in Fig. 2 is defined for the case of using the elliptic curves of the prime order. Nevertheless, it can be slightly modified by adding the additional checks for use with non-prime order curves, e.g. with Edwards curves.

This scheme provides perfect blindness [6, Theorem 2], but does not provide unforgeability in the strong sense. In [11] it was shown that it is vulnerable to the ROS-style attack, which is possible if the adversary acting as a User is given the opportunity to open $\ell \geqslant \lceil \log q \rceil$ parallel sessions of signing protocol. However, providing such strong unforgeability is not required for our application, our purpose is the honest-but-curious unforgeability.

Camenisch scheme is the particular case of the GenEG-BS$_{\mathcal{L}}$ scheme defined in Section 4.2. Indeed, the distribution $\mathcal{D}$ in this scheme is a uniform distribution on $\mathbb{Z}_q^* \times \mathbb{Z}_q^*$ that is independent on R; $\mathcal{L}_1^{(P,Q,m)}$ and $\mathcal{L}_2^{(P,Q,m)}$ are defined as follows:

$$\mathcal{L}_1^{(P,Q,m)}(R, (\alpha, \beta)) = \alpha R + \beta P, \quad \mathcal{L}_2^{(P,Q,m)}(s, (\alpha, \beta), R) = sr'r^{-1} + \beta e',$$

where $e' = H(m)$, $r = R.x \bmod q$, $r' = (\alpha R + \beta P).x \bmod q$. These functions are linear by R and s values respectively for all possible rnd values. Moreover, zero r and e values are excluded by the corresponding checks on the Signer side as in the GOST signature scheme. Therefore, the result of Theorem 2 is applied to the Camenisch scheme, which means that it provides honest-but-curious unforgeability under the assumption that GOST

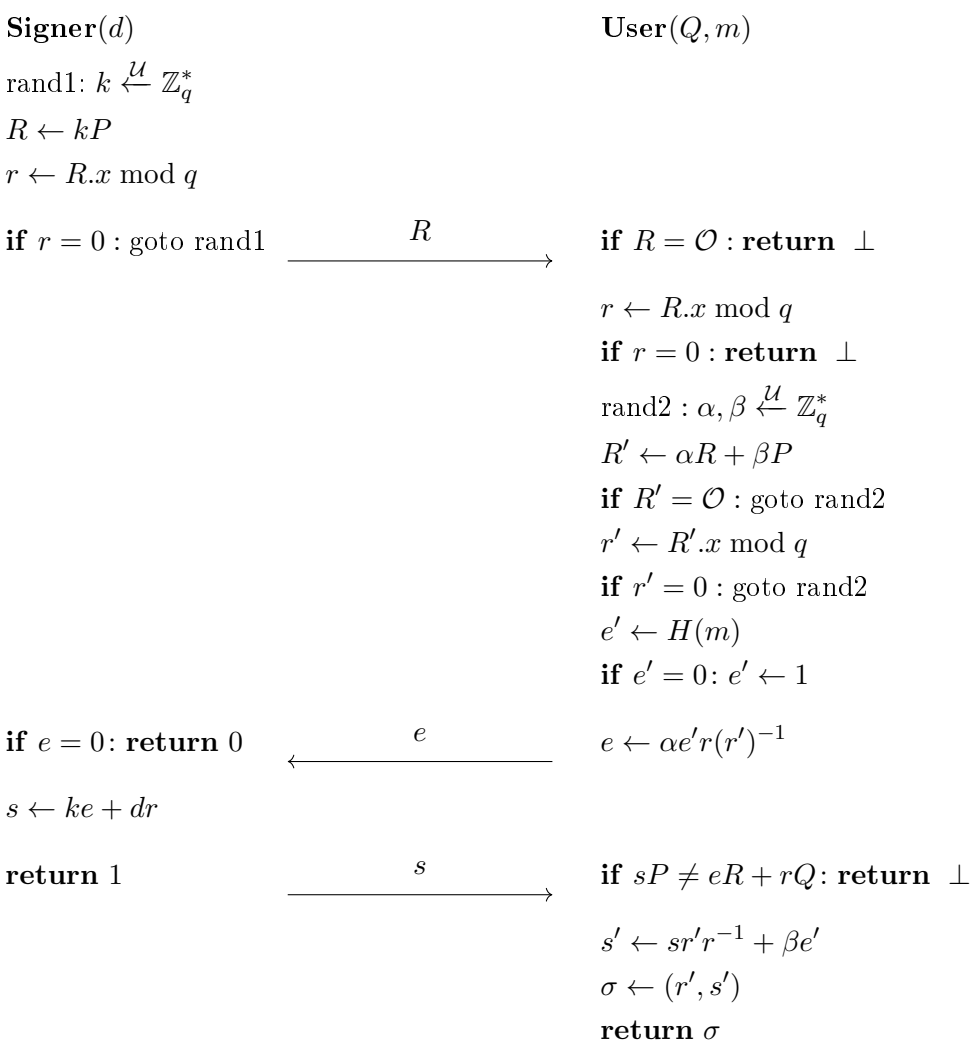

Fig. 2. The signing protocol in Camenisch scheme

scheme provides unforgeability. The security of the Camenisch scheme in the sUF-CMA model, in its turn, directly follows from the honest-but-curious unforgeability.

Thus, the Camenisch scheme is a blind version of the GOST scheme and can be applied in the systems realizing the GOST signature as the protection against backdoors in smart cards. It provides the security against external adversary and adversary with agent only by the security of the GOST signature scheme. Note that such solution, in contrast to the solution from [2], does not need any additional assumptions about the smart card such as correct implementation of low-level arithmetic operations and the absence of failures. Moreover, it requires less computations on the smart card side.

6. Conclusion

The paper addressed the security issues that arise in signing systems when the smart card used for key storage and signing is believed to contain backdoors. A novel approach based on blind signature schemes to protect against backdoors has been proposed. It has been proven that weak versions of standard security properties (honest-signer blindness and honest-but-curious unforgeability) of blind signature scheme imply security against backdoors in smart cards.

Moreover, the concrete solution in case of using the GOST signature scheme has been proposed. This solution is the well known Camenisch blind signature scheme that provides perfect blindness. It was shown that the target security is held under the sole assumption that the GOST signature scheme provides standard security, i.e., is unforgeable under chosen message attack.

One of the most interesting directions for future research is the security analysis of our solution with regard to a stronger external adversary — an active adversary that has an access to a smart card signing API (e.g. in a case when the smart card is not protected with a password or is connected to a malicious terminal).

This case corresponds to the standard unforgeability notion of the blind signatures, where the user side is treated as a fully active adversary. There are two types of unforgeability notion differing on whether the adversary can open sessions in parallel or not. In our application scenario, where the signer side is executed by low resource device, it is fairly enough to consider the adversary's capability to open sessions sequentially only (this refers to the SEQ-OMUF notion [14]).

Note that the SEQ-OMUF-security of the Camenish scheme is still an open question (as well as for the most ElGamal blind signature schemes), although there have been some positive results [14] for the Schnorr blind signature scheme.

REFERENCES

1. *Alekseev E. K., Akhmetzyanova L. R., Oshkin I. B., and Smyshlyaev S. V.* Obzor uyazvimostey nekotorykh protokolov vyrabotki obshchego klyucha s autentifikatsiey na osnove parolya i printsipy postroeniya protokola SESPAKE [A review of the password authenticated key exchange protocols vulnerabilities and principles of the SESPAKE protocol construction]. Matematicheskie Voprosy Kriptografii, 2016, vol. 7, iss. 4, pp. 7–28. (in Russian)
2. *Alekseev E. K., Akhmetzyanova L. R., Bozhko A. A., and Smyshlyaev S. V.* Bezopasnaya realizatsiya elektronnoy podpisi s ispol'zovaniem slabodoverennogo vychislitelya [Secure implementation of digital signature using semi-trusted computational core]. Matematicheskie Voprosy Kriptografii, 2021, vol. 12, iss. 4, pp. 5–23. (in Russian)
3. *Wang Y.* Password protected smart card and memory stick authentication against off-line dictionary attacks. D. Critzalis, S. Furnell, and M. Theoharidou (eds.), Information Security and Privacy Research, Berlin, Heidelberg, Springer, 2012, pp. 489–500.
4. GOST R 34.10-2012. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Protsessy formirovaniya i proverki elektronnoy tsifrovoy podpisi. [GOST R 34.10-2012. Information Technology. Cryptographic Data Security. Signature and Verification Processes of Electronic Digital Signature]. Moscow, Standartinform Publ., 2012. (in Russian)
5. *Chaum D.* Blind signatures for untraceable payments. D. Chaum, R. L. Rivest, and A. T. Sherman (eds.) Advances in Cryptology. Boston, MA, Springer, 1983. pp. 199–203.
6. *Camenisch J. L., Piveteau J. M., and Stadler M. A.* Blind signatures based on the discrete logarithm problem. LNCS, 1995, vol. 950, pp. 428–432.
7. *Bellare M. and Rogaway P.* The security of triple encryption and a framework for code-based game-playing proofs. LNCS, 2006, vol. 4004, pp. 409–426.
8. *Tessaro S. and Zhu C.* Short pairing-free blind signatures with exponential security. LNCS, 2022, vol. 13276, pp. 782–811.
9. *Juels A., Luby M., and Ostrovsky R.* Security of blind digital signatures. LNCS, 1997, vol. 1294, pp. 150–164.
10. *Fischlin M. and Mittelbach A.* An Overview of the Hybrid Argument. Cryptology ePrint Archive, paper 2021/088, `https://eprint.iacr.org/2021/088`, 2021.

11. *Akhmetzyanova L., Alekseev E., Babueva A., and Smyshlyaev S.* On the (im)possibility of ElGamal blind signatures. Cryptology ePrint Archive, paper 2022/1128, `https://eprint.iacr.org/2022/1128`, 2022.
12. *Harn L. and Xu Y.* Design of generalised ElGamal type digital signature schemes based on discrete logarithm. Electronics Letters, 1994, vol. 30, pp. 2025–2026.
13. *Fersch M.* The provable security of Elgamal-type signature schemes. Doctoral Thesis, Ruhr-Universität Bochum, 2018.
14. *Kastner J., Loss J., and Xu J.* On pairing-free blind signature schemes in the algebraic group model. LNCS, 2022, vol. 13178, pp. 468–497.

ПРИКЛАДНАЯ ТЕОРИЯ КОДИРОВАНИЯ

УДК 519.725 DOI 10.17223/20710410/63/4

ВЫЧИСЛЕНИЕ ПАР, ИСПРАВЛЯЮЩИХ ОШИБКИ ДЛЯ АЛГЕБРОГЕОМЕТРИЧЕСКОГО КОДА[1]

А. А. Кунинец*, Е. С. Малыгина**

*Балтийский федеральный университет им. И.Канта, г. Калининград, Россия
**МИЭМ НИУ ВШЭ, г. Москва, Россия

E-mail: artkuninets@yandex.ru, emalygina@hse.ru

Для произвольного алгеброгеометрического кода и дуального к нему явно вычислены пары, исправляющие ошибки. Такая пара состоит из кодов, которые необходимы для эффективного алгоритма декодирования заданного кода. Вид пар зависит от степеней дивизоров, с помощью которых строится как исходный код, так и один из кодов, входящих в пару. Для алгеброгеометрического кода $\mathcal{C}_{\mathcal{L}}(D,G)$ длины n, ассоциированного с функциональным полем $\mathcal{F}/\mathbb{F}_q$ рода g, парами, исправляющими $t=\lfloor(n-\deg(G)-g-1)/2\rfloor$ ошибок, при определённых ограничениях на степени дивизоров, участвующих в их построении, являются пары кодов $(\mathcal{C}_{\mathcal{L}}(D,F),\mathcal{C}_{\mathcal{L}}(D,G+F)^{\perp})$ или $(\mathcal{C}_{\mathcal{L}}(D,F)^{\perp},\mathcal{C}_{\mathcal{L}}(D,F-G))$. Выведены ограничения на степени дивизоров кодов $(\mathcal{C}_{\mathcal{L}}(D,F),\mathcal{C}_{\mathcal{L}}(D,G-F))$, составляющих пару, исправляющую $t=\lfloor(\deg(G)-3g+1)/2\rfloor$ ошибок для дуального кода $\mathcal{C}_{\mathcal{L}}(D,G)^{\perp}$. Рассмотрены случаи принадлежности одного из кодов, участвующих в построении пары, к классу MDS-кодов и выведены параметры, при которых данная ситуация возможна. Кроме того, вычислены возможные границы для дивизоров, участвующих в построении пар, исправляющих ошибки для подполевых подкодов $\mathcal{C}_{\mathcal{L}}(D,G)|_{\mathbb{F}_p}$ и $\mathcal{C}_{\mathcal{L}}(D,G)^{\perp}|_{\mathbb{F}_p}$ исходного алгеброгеометрического кода и дуального к нему, при степени расширения $m=2$ ($\mathbb{F}_q=\mathbb{F}_{p^2}$).

Ключевые слова: *функциональное поле, алгеброгеометрический код, исправляющая ошибки пара, подполевой подкод.*

CALCULATION OF ERROR-CORRECTING PAIRS FOR AN ALGEBRAIC-GEOMETRIC CODE

A. A. Kuninets*, E. S. Malygina**

*Immanuel Kant Baltic Federal University, Kaliningrad, Russia
**HSE, Moscow, Russia

Error-correcting pairs are calculated explicitly for an arbitrary algebraic-geometric code and its dual code. Such a pair consists of codes that are necessary for an effective decoding algorithm for a given code. The type of pairs depends on the

[1]Работа первого автора поддержана грантом Российского научного фонда № 22-41-0441, работа второго автора выполнена в рамках Программы фундаментальных исследований НИУ ВШЭ.

degrees of divisors with which both the original code and one of the codes from error-correcting pair are constructed. So for the algebraic-geometric code $\mathcal{C}_{\mathcal{L}}(D,G)$ of the length n associated with a functional field $F/\mathbb{F}_q$ of genus g the error-correcting pair with number of errors $t=\lfloor(n-\deg(G)-g-1)/2\rfloor$ is $(\mathcal{C}_{\mathcal{L}}(D,F),\mathcal{C}_{\mathcal{L}}(D,G+F)^{\perp})$ or $(\mathcal{C}_{\mathcal{L}}(D,F)^{\perp},\mathcal{C}_{\mathcal{L}}(D,F-G))$. For the dual code $\mathcal{C}_{\mathcal{L}}(D,G)^{\perp}$ the error-correcting pair with number of errors $t=\lfloor(\deg(G)-3g+1)/2\rfloor$ is $(\mathcal{C}_{\mathcal{L}}(D,F),\mathcal{C}_{\mathcal{L}}(D,G-F))$. Considering each component of pair as MDS-code, we obtain additional conditions on the degrees of the divisors G and F. In addition, error-correcting pairs are calculated for subfield subcodes $\mathcal{C}_{\mathcal{L}}(D,G)|_{\mathbb{F}_p}$ and $\mathcal{C}_{\mathcal{L}}(D,G)^{\perp}|_{\mathbb{F}_p}$, where $\mathbb{F}_p$ is a subfield of $\mathbb{F}_q$. The form of a first component in the pair depends on the degrees of the divisors G and F and, in some cases, on the genus g.

Keywords: *functional field, algebraic-geometric code, error-correcting pair, subfield subcode.*

Введение

Исследование задачи декодирования кодов, построенных на алгебраических кривых, явилось очень востребованным за последние тридцать лет. Изначально Т. Хёхольдт и др. предложили синдромный алгоритм декодирования для кодов, ассоциированных с плоской кривой [1]. Затем А. Скоробогатов и С. Влэдуц обобщили этот алгоритм на произвольные кривые [2]. Далее Р. Пелликаан и Р. Кёттер независимо друг от друга предложили алгоритм декодирования, исключающий абстрактные понятия алгебраической геометрии и использующий пары, исправляющие ошибки [3, 4]. Парой, исправляющей ошибки для кода $\mathcal{C}$, является пара кодов $\mathcal{A}$ и $\mathcal{B}$, удовлетворяющая некоторым ограничениям на размерность и минимальное расстояние, а также условию, что покомпонентное произведение кодовых слов $\mathcal{A}$ и $\mathcal{B}$ содержится в дуальном коде $\mathcal{C}^{\perp}$. Существование такой пары обеспечивает эффективный алгоритм декодирования для алгеброгеометрических кодов (АГ-кодов), который использует лишь методы линейной алгебры. Особый интерес представляет построение таких пар, поскольку сама пара является входным параметром для алгоритма декодирования. Стоит также отметить, что пары, исправляющие ошибки, заслуживают внимания и с криптографической точки зрения, поскольку лежат в основе атаки на АГ-коды [5].

Структура работы следующая: в п. 1 мы даём предварительные сведения, касающиеся базовых объектов теории функциональных полей и алгебраических кривых, необходимых для задания АГ-кода с помощью пространства Римана — Роха, а также для задания дуального АГ-кода с помощью пространства дифференциалов. В п. 2 представлен основной результат работы, заключающийся в ряде теорем. Первоначально мы задаём пары, исправляющие ошибки для АГ-кода и дуального к нему, накладывая ограничения на степени их дивизоров. Затем мы исследуем, при каких значениях код из пары или исходный код, для которого находится пара, является MDS-кодом (т. е. минимальное расстояние кода достигает максимального значения границы Синглтона). Далее мы даём полную классификацию пар, исправляющих ошибки для подполевых подкодов исходного АГ-кода и дуального к исходному АГ-коду при условии, что эти коды определены над квадратичным расширением конечного поля. Классификация включает в себя явный вид кодов из пары, значение рода кривой, длину кода, а также условия, налагаемые на степени дивизоров, ассоциированных с исходным кодом и его подполевым подкодом.

Данная работа является продолжением работы, представленной на конференции SIBECRYPT'23 [6].

1. Предварительные сведения

1.1. Алгебраические кривые и функциональные поля

Будем обозначать через $\mathbb{F}_q$ конечное поле, состоящее из q элементов, где q — степень простого числа.

Под *проективной кривой* над конечным полем $\mathbb{F}_q$ понимается проективное многообразие над $\mathbb{F}_q$ размерности один, где проективное многообразие представляет собой неприводимое замкнутое подмножество в проективном пространстве $\mathbb{P}^n$ [7]. Далее будем обозначать проективную кривую через $\mathcal{X}$.

В большинстве случаев в теории кодирования используются кривые, определённые над конечным полем. Под *проективной кривой $\mathcal{X}$, определённой над* $\mathbb{F}_q$, будем понимать кривую $\mathcal{X} \subseteq \mathbb{P}^n(\overline{\mathbb{F}}_q)$, где $\overline{\mathbb{F}}_q$ — алгебраическое замыкание поля $\mathbb{F}_q$, причём однородный многочлен, определяющий кривую, имеет коэффициенты в $\mathbb{F}_q$.

Определим *поле функций кривой* $\mathcal{X}$:

$$\mathbb{F}_q(\mathcal{X}) = \left\{ \frac{g}{h} : g, h \in \mathbb{F}_q[x_1, \ldots, x_{n-1}],\ h \neq 0 \right\}.$$

Здесь сама кривая определена однородным многочленом из кольца $\mathbb{F}_q[X_1, \ldots, X_n]$ и $x_1 = \dfrac{X_1}{X_n}$, $\ldots$, $x_{n-1} = \dfrac{X_{n-1}}{X_n}$. Говорят, что $\mathbb{F}_q(\mathcal{X})$ является *функциональным полем кривой* $\mathcal{X}/\mathbb{F}_q$.

Пусть P — точка кривой $\mathcal{X}$. Функция $f \in \mathbb{F}_q(\mathcal{X})$ называется *регулярной в точке* P, если её можно записать в виде $f = \dfrac{g}{h}$ и $g(P) \neq 0$. Множество регулярных функций в точке P образует кольцо, называемое *локальным кольцом* $\mathcal{O}_P$. Отметим, что точка кривой может иметь степень. Точки кривой, имеющие координаты в $\mathbb{F}_q$, называются *рациональными точками* или *точками степени один*. Если координаты точки кривой лежат в расширении базового конечного поля, то точка имеет степень, равную степени этого расширения.

Приведём базовые определения и свойства функциональных полей, чтобы посмотреть, как они связаны с алгебраическими кривыми.

Алгебраическим функциональным полем $\mathcal{F}/\mathbb{F}_q$ от одной переменной называется расширение $\mathbb{F}_q(x)$ поля $\mathbb{F}_q$, являющееся конечным алгебраическим расширением для некоторого трансцендентного над $\mathbb{F}_q$ элемента $x \in \mathbb{F}_q(x)$. Соответственно функциональным полем от n переменных является конечное алгебраическое расширение $\mathbb{F}_q(x_1, \ldots, x_n)$, где $x_1, \ldots, x_n$ трансцендентны над $\mathbb{F}_q$.

Для функциональных полей аналогом локального кольца в случае алгебраических кривых является кольцо нормирования. *Кольцом нормирования* функционального поля $\mathcal{F}/\mathbb{F}_q$ называется кольцо $\mathcal{O}$, такое, что:

— $\mathbb{F}_q \subsetneq \mathcal{O} \subsetneq F$;
— для любого элемента $x \in \mathcal{F}$ выполняется: $x \in \mathcal{O}$ или $x^{-1} \in \mathcal{O}$.

Следует отметить, что если мы работаем над алгебраически замкнутым полем, то существует взаимно однозначное соответствие между точками кривой и точками её функционального поля, хотя точка функционального поля имеет совсем иную специфику. *Точкой* P функционального поля $\mathcal{F}/\mathbb{F}_q$ называется максимальный идеал некоторого кольца нормирования $\mathcal{O}$.

По свойствам кольца нормирования оно является локальным кольцом, а значит, $\mathcal{O}$ можно ассоциировать с его единственным максимальным идеалом P:

$$\mathcal{O}_P = \{x \in F : x^{-1} \notin P\}.$$

Кроме того, по свойствам кольца нормирования P является главным идеалом, следовательно, $P = t_P\mathcal{O}_P$, при этом элемент t_P называется *локальным параметром точки* P. Теперь определим *степень точки* P как степень расширения поля $\mathcal{O}_P/P$ над $\mathbb{F}_q$, а именно:

$$\deg(P) = [\mathcal{O}_P/P : \mathbb{F}_p].$$

Далее будем отождествлять кривую с её функциональным полем и перейдём к рассмотрению основополагающих объектов для определения АГ-кода — дивизорам кривой (или её функционального поля).

Группой дивизоров $\mathrm{Div}(\mathcal{X})$ проективной кривой $\mathcal{X}$ называется свободная абелева группа, порождённая точками $\mathcal{X}$. Элементы группы $D \in \mathrm{Div}(\mathcal{X})$ называются *дивизорами* и представляют собой формальную сумму точек:

$$D = \sum_{P\in\mathcal{X}} n_P P,$$

причём только конечное число $n_P \in \mathbb{Z}$ отлично от нуля.

Определим *степень дивизора* как

$$\deg(D) = \sum_{P\in\mathcal{X}} n_P \cdot \deg(P).$$

В группе $\mathrm{Div}(\mathcal{X})$ определено частичное упорядочивание:

$$\sum_{P\in\mathcal{X}} n_P P \geqslant \sum_{P\in\mathcal{X}} m_P P \;\Leftrightarrow\; n_P \geqslant m_P \text{ для любой точки } P \in \mathcal{X}.$$

Теперь определим *дивизор функции*. Пусть $f \in \mathbb{F}_q(\mathcal{X})^*$. Обозначим через Z (через N) множество нулей (полюсов) функции f, определяемых с помощью точек $P \in \mathcal{X}$. Тогда для функции f определим:

— её дивизор нулей:

$$(f)_0 = \sum_{P\in Z} n_P\, P, \text{ где } n_P \text{ — кратность, соответствующая точке } P;$$

— дивизор полюсов:

$$(f)_\infty = \sum_{P\in N} (-n_P)\, P, \text{ где } n_P \text{ — кратность, соответствующая точке } P;$$

— главный дивизор:

$$(f) = (f)_0 - (f)_\infty.$$

Чтобы определить дуальный код, потребуется ряд понятий, связанных с дифференцированием и дифференциалами.

Определим *дифференцирование* над $\mathbb{F}_q(\mathcal{X})$ как $\mathbb{F}_q$-линейное отображение

$$\triangle : \mathbb{F}_q(\mathcal{X}) \to \mathbb{F}_q(\mathcal{X}),$$

удовлетворяющее правилу Лейбница $\triangle(fg) = f\triangle(g) + g\triangle(f)$ для $f, g \in \mathbb{F}_q(\mathcal{X})$. Множество таких дифференцирований $\mathrm{Der}(\mathbb{F}_q(\mathcal{X}))$ образует векторное пространство над $\mathbb{F}_q(\mathcal{X})$.

Дифференциальной формой или *дифференциалом* на кривой $\mathcal{X}$ называется $\mathbb{F}_q(\mathcal{X})$-линейное отображение $\mathrm{Der}(\mathbb{F}_q(\mathcal{X})) \to \mathbb{F}_q(\mathcal{X})$. Множество всех дифференциалов кривой $\mathcal{X}$ будем обозначать $\Omega(\mathcal{X})$.

Рассмотрим отображение

$$\delta : \begin{cases} \mathbb{F}_q(\mathcal{X}) \to \Omega(\mathcal{X}), \\ f \mapsto \delta f, \end{cases}$$

сопоставляющее всякой функции f дифференциал $\delta f : \mathrm{Der}(\mathbb{F}_q(\mathcal{X})) \to \mathbb{F}_q(\mathcal{X})$ по правилу $\delta f(\triangle) = \triangle(f)$ для любого $\triangle \in \mathrm{Der}(\mathbb{F}_q(\mathcal{X}))$.

Отметим, что любой дифференциал $\omega \in \Omega(\mathcal{X})$ можно уникально представить как $\omega = f\delta t_P$ для точки $P \in \mathcal{X}$ и локального параметра t_P, где $f \in \mathbb{F}_q(\mathcal{X})$. Будем говорить, что P является нулём ω, если P — нуль функции f, аналогично P — полюс ω, если P является полюсом функции f. Тогда по аналогии с главным дивизором для функции $f \in \mathbb{F}_q(\mathcal{X})^*$ можно определить дивизор для дифференциала $\omega \in \Omega(\mathcal{X})^*$:

$$(\omega) = \sum_{P \in N} n_P P,$$

однако специфика вычисления значения n_P достаточно сложная, поэтому за деталями можно обратиться к [8]. При этом дивизор (ω) называется *каноническим*. Обозначим $W = (\omega)$, тогда, согласно [8], $\deg(W) = 2g - 2$.

Одним из важных понятий является понятие рода кривой (её функционального поля). Если $\mathcal{X}$ является гладкой проективной плоской кривой (как правило, именно такие кривые рассматриваются для приложений в теории кодирования) степени r, то $g(\mathcal{X}) = (r-1)(r-2)/2$.

1.2. Алгеброгеометрические коды и дуальные к ним

Рассмотрим две конструкции АГ-кодов, а именно: конструкцию АГ-кода с привлечением пространства Римана — Роха и конструкцию дуального к нему АГ-кода с привлечением пространства дифференциалов.

Построение $\mathcal{C}_{\mathcal{L}}(D, G)$

Пусть G — дивизор кривой $\mathcal{X}$. Определим множество

$$\mathcal{L}(G) = \{f \in \mathbb{F}_q(\mathcal{X}) : (f) \geqslant -G\}.$$

Оно является векторным пространством над $\mathbb{F}_q$ и называется *пространством Римана — Роха*. Обозначим $\dim_{\mathbb{F}_q}(\mathcal{L}(G)) = \ell(G)$. Благодаря теореме Римана — Роха, можно получить значение $\ell(G)$.

Теорема 1 [8, Theorem 1.5.15]. Пусть $\mathcal{X}$ — гладкая проективная кривая, W — её канонический дивизор. Тогда для любого дивизора $G \in \mathrm{Div}(\mathcal{X})$ справедливо равенство

$$\ell(G) = \deg(G) + 1 - g(\mathcal{X}) + \ell(W - G).$$

Кроме того, если $\deg(G) > 2g - 2$, то $\ell(G) = \deg(G) + 1 - g(\mathcal{X})$.

Пусть $P_1, P_2, \ldots, P_n$ — попарно различные рациональные точки кривой $\mathcal{X}$ или точки функционального поля $\mathbb{F}_q(\mathcal{X})$ степени один. Обозначим $D = P_1 + \ldots + P_n$ и G — дивизоры кривой $\mathcal{X}$, причём в записи дивизора G не участвуют точки дивизора D и $\deg(G) < n$. Рассмотрим отображение

$$ev_D : \begin{cases} \mathcal{L}(G) \to \mathbb{F}_q^n, \\ f \mapsto (f(P_1), \ldots, f(P_n)). \end{cases}$$

Определение 1. АГ-кодом $\mathcal{C}_{\mathcal{L}}(D,G)$, ассоциированным с кривой $\mathcal{X}$ и дивизорами D и G, называется подпространство в $\mathbb{F}_q^n$ вида

$$\mathcal{C}_{\mathcal{L}}(D,G) = \{ev_D(f) : f \in \mathcal{L}(G)\}.$$

Отметим, что всякий код $\mathcal{C}_{\mathcal{L}}(D,G)$ можно задать параметрами $[n,k,d]$, где n — длина кода (число точек в записи дивизора D); $k = k(\mathcal{C})$ — размерность кода (размерность пространства Римана — Роха $\mathcal{L}(G)$); $d = d(\mathcal{C})$ — минимальное расстояние кода.

Согласно [8, Theorem 2.2.2], код $\mathcal{C}_{\mathcal{L}}(D,G)$ является $[n,k,d]$-кодом, причём

$$k \geqslant \deg(G) + 1 - g,\ d \geqslant n - \deg(G),$$

и если $2g - 2 < \deg(G) < n$, то $k = \deg(G) + 1 - g$.

Если $\{f_1, \ldots, f_k\}$ — базис $\mathcal{L}(G)$, то порождающая матрица кода $\mathcal{C}_{\mathcal{L}}(D,G)$ имеет следующий вид:

$$\begin{pmatrix} f_1(P_1) & f_1(P_2) & \ldots & f_1(P_n) \\ f_2(P_1) & f_2(P_2) & \ldots & f_2(P_n) \\ \vdots & \vdots & \ldots & \vdots \\ f_k(P_1) & f_k(P_2) & \ldots & f_k(P_n) \end{pmatrix}.$$

Построение $\mathcal{C}_{\Omega}(D,G)$

Для определения дуального кода к коду $\mathcal{C}_{\mathcal{L}}(D,G)$ определим множество

$$\Omega(G) = \{\omega \in \Omega(\mathcal{X}) : (\omega) \geqslant G\}.$$

Оно является векторным пространством над $\mathbb{F}_q$ и называется *пространством дифференциалов*. Размерность $\dim_{\mathbb{F}_q}(\Omega(G)) = i(G)$ называется *индексом специальности* дивизора G и равна

$$i(G) = \ell(G) - \deg(G) + g(\mathcal{X}) - 1.$$

Чтобы задать дуальный код непосредственно, нужно ввести понятия вычета дифференциала $\omega = f\delta t_P$ в точке P, где $f \in \mathbb{F}_q(\mathcal{X})$ и t_P — локальный параметр. Для этого разложим функцию f в ряд Лорана по степеням t_P:

$$f = \sum_{i=\alpha}^{\infty} \alpha_i t_P^i,$$

где $\alpha \in \mathbb{Z}$. *Вычетом дифференциала* ω *в точке* P называется коэффициент α_{-1} в представленном разложении, он обозначается $\mathrm{Res}_{\omega}(P)$.

Как и ранее, пусть $P_1, P_2, \ldots, P_n$ — попарно различные рациональные точки кривой $\mathcal{X}$, $D = P_1 + \ldots + P_n$ и G — дивизоры кривой $\mathcal{X}$, такие, что в записи дивизора G не участвуют точки дивизора D и $\deg(G) > 2g - 2$. Рассмотрим отображение

$$res_D : \begin{cases} \Omega(G) \to \mathbb{F}_q^n, \\ \omega \mapsto (\mathrm{Res}_{\omega}(P_1), \ldots, \mathrm{Res}_{\omega}(P_n)). \end{cases}$$

Определение 2. АГ-кодом $\mathcal{C}_{\Omega}(D,G)$, ассоциированным с кривой $\mathcal{X}$ и дивизорами D и G и являющимся дуальным к $\mathcal{C}_{\mathcal{L}}(D,G)$, называется

$$\mathcal{C}_{\Omega}(D,G) = \{res_D(\omega) : \omega \in \Omega(G - D)\}.$$

Если параметры кода $\mathcal{C}_\Omega(D,G)$ обозначить через $[n,k',d']$, то, согласно [8, Theorem 2.2.7],

$$k' = n + g - 1 - \deg G, \quad d' \geqslant \deg G - (2g-2),$$

если $2g-2 < \deg G < n$.

С учётом построения имеем

$$\mathcal{C}_\mathcal{L}(D,G)^\perp = \mathcal{C}_\Omega(D,G) \quad \text{и} \quad \mathcal{C}_\Omega(D,G) = \mathcal{C}_\mathcal{L}(D, D-G+W),$$

где $W = (\omega)$ — канонический дивизор кривой $\mathcal{X}$.

В общем случае рассматриваемые коды могут исправить до $\lfloor (d(d')-1)/2 \rfloor$ ошибок, где d и d' — минимальные расстояния кодов $\mathcal{C}_\mathcal{L}(D,G)$ и $\mathcal{C}_\Omega(D,G)$ соответственно. Будем называть код MDS-кодом, если его минимальное расстояние достигает границы Синглтона, т. е. $d(\mathcal{C}) = n + 1 - k(\mathcal{C})$.

1.3. Пары, исправляющие ошибки

Произведение Шура двух векторов $a, b \in \mathbb{F}_q^n$ определяется как произведение их соответствующих координат:

$$(a_1,\ldots,a_n) * (b_1,\ldots,b_n) = (a_1b_1,\ldots,a_nb_n),$$
$$(a_1,\ldots,a_n)^i = (a_1^i,\ldots,a_n^i).$$

Для кодов $\mathcal{A},\mathcal{B} \subset \mathbb{F}_q^n$ произведение Шура $\mathcal{A} * \mathcal{B}$ определяется следующим образом:

$$\mathcal{A} * \mathcal{B} = \mathrm{Span}_{\mathbb{F}_q}\{a * b \mid a \in \mathcal{A}, b \in \mathcal{B}\}.$$

Определение 3. Пусть $\mathcal{C} \in \mathbb{F}_q^n$ — линейный код. Тогда пара линейных кодов $(\mathcal{A},\mathcal{B})$, где $\mathcal{A},\mathcal{B} \subset \mathbb{F}_q^n$, называется парой, исправляющей t ошибок для кода $\mathcal{C}$, если выполняются следующие условия:

1) $\mathcal{A} * \mathcal{B} \subseteq \mathcal{C}^\perp$;
2) $\dim(\mathcal{A}) > t$;
3) $d(\mathcal{B}^\perp) > t$;
4) $d(\mathcal{A}) + d(\mathcal{C}) > n$.

В обозначениях определения считаем, что $d(\mathcal{C}) \geqslant 2t+1$.

В [9, 10] описаны условия существования пары $\mathcal{A}$ и $\mathcal{B}$, исправляющей t ошибок.

2. Основной результат

Несмотря на наличие ряда работ, посвящённых вопросу существования пар, исправляющих ошибки для линейных кодов, ни в одной из них не представлено нахождение такой пары для произвольного АГ-кода. Исключением является работа [5, Theorem 14], посвящённая криптоанализу криптосистемы Мак-Элиса, в которой рассмотрен общий вид пары, исправляющей ошибки для дуального кода. В следующих теоремах мы не только описываем вид кодов в паре, исправляющей ошибки для $\mathcal{C}_\mathcal{L}(D,G)$ и $\mathcal{C}_\mathcal{L}(D,G)^\perp$, но также задаём классификацию относительно рода функционального поля и степеней дивизоров, ассоциированных с кодами из пары. Отметим, что теорему 14 из [5] мы специализируем на случай принадлежности одного из кодов пары, исправляющей ошибки, к MDS-кодам.

Теорема 2 [11, Theorem 6]. Пусть $\mathcal{F}/\mathbb{F}_q$ — некоторое функциональное поле рода g; $D = P_1 + \ldots + P_n$ — дивизор, носитель которого состоит из точек степени один поля $\mathcal{F}$; G и H — дивизоры, такие, что $\mathrm{supp}(D) \cap (\mathrm{supp}(G) \cup \mathrm{supp}(H)) = \varnothing$. Тогда

$$\mathcal{C}_\mathcal{L}(D,G) * \mathcal{C}_\mathcal{L}(D,H) \subseteq \mathcal{C}_\mathcal{L}(D,G+H).$$

Если $\deg(G) \geqslant 2g$, $\deg(H) \geqslant 2g+1$, то выполняется равенство

$$\mathcal{C}_{\mathcal{L}}(D,G) * \mathcal{C}_{\mathcal{L}}(D,H) = \mathcal{C}_{\mathcal{L}}(D,G+H).$$

На основе теоремы 2 построим пару, исправляющую t ошибок для кода $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D,G)$.

Теорема 3. Пусть $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D,G)$ — АГ-код, ассоциированный с функциональным полем $\mathcal{F}/\mathbb{F}_q$ рода g. Тогда если $\operatorname{supp}(G) \cap \operatorname{supp}(F) = \varnothing$, то парами, исправляющими $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$ ошибок для кода $\mathcal{C}$, являются следующие коды:

1) $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D,G+F)^{\perp}$, если $t+g \leqslant \deg(F) < n - \deg(G) - t$ и $2g-2 < \deg(G) < n-g-1$;
2) $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)^{\perp}$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D,F-G)$, если $\deg(G)+t+2g-2 < \deg(F) \leqslant n+g-t-2$ и $2g-2 < \deg(G) < n-g-1$.

Доказательство. Выведем вид пары $(\mathcal{A},\mathcal{B})$, исправляющей ошибки для кода $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D,G)$ и покажем, при каких параметрах и ограничениях на степени дивизоров будут выполняться все условия определения 3.

Обоснуем равносильность условий $\mathcal{A}*\mathcal{B} \subseteq \mathcal{C}^{\perp}$ и $\mathcal{B} \subseteq (\mathcal{A}*\mathcal{C})^{\perp}$. Действительно, пусть $a \in \mathcal{A}$, $b \in \mathcal{B}$ и $c \in \mathcal{C}$, тогда если имеет место включение $\mathcal{A} * \mathcal{B} \subseteq \mathcal{C}^{\perp}$, то $\langle a*b, c\rangle = 0$, а по свойствам скалярного произведения $\langle a*b, c\rangle = \langle b, a*c\rangle$, откуда $\langle b, a*c\rangle = 0$, а значит, $\mathcal{B} \subseteq (\mathcal{A}*\mathcal{C})^{\perp}$. Обратное рассуждение, а именно: если выполняется условие $\mathcal{B} \subseteq (\mathcal{A}*\mathcal{C})^{\perp}$, то верно и $\mathcal{A}*\mathcal{B} \subseteq \mathcal{C}^{\perp}$, — аналогично.

1) Обозначим $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)$ для некоторого дивизора F. Далее оценим $\deg(F)$, но прежде построим код $\mathcal{B}$ так, чтобы выполнялось условие 1 определения 3.

Поскольку условие $\mathcal{A} * \mathcal{B} \subseteq \mathcal{C}^{\perp}$ равносильно условию $\mathcal{B} \subseteq (\mathcal{A}*\mathcal{C})^{\perp}$, покажем, что код $\mathcal{B}$ имеет вид $\mathcal{C}_{\mathcal{L}}(D,G+F)^{\perp}$:

$$\mathcal{C}_{\mathcal{L}}(D,F) * \mathcal{C}_{\mathcal{L}}(D,G) \subseteq \mathcal{C}_{\mathcal{L}}(D,G+F) \;\Leftrightarrow\; \mathcal{B} = \mathcal{C}_{\mathcal{L}}(D,G+F)^{\perp} \subseteq (\mathcal{C}_{\mathcal{L}}(D,F) * \mathcal{C}_{\mathcal{L}}(D,G))^{\perp}.$$

Далее, исходя из трёх оставшихся условий определения 3, выведем границы для $\deg(F)$:

— Если степень дивизора F лежит в границах $t+g \leqslant \deg(F) < n$, то по теореме Римана — Роха

$$k(\mathcal{A}) \geqslant \deg(F) + 1 - g \geqslant t+g+1-g = t+1 > t.$$

Учитывая верхнюю границу $\deg(G+F) < n$ на степень дивизора $G+F$, получаем ограничение $\deg(G) < n-g-t$. Раскрывая t, получаем неравенство

$$\deg(G) < n - g - \lfloor (n - \deg(G) - g - 1)/2 \rfloor,$$

откуда, накладывая ограничение $t \geqslant 0$, получаем верхнюю границу

$$\deg(G) \leqslant n - g - 1$$

на степень дивизора G.

— Если $\deg(F) \leqslant n - \deg(G) - t - 1$, то

$$d(\mathcal{B}^{\perp}) \geqslant n - \deg(F+G) = n - \deg(G) - \deg(F) \geqslant n - \deg(G) - n + \deg(G) + t + 1 = t+1 > t.$$

— Если $\deg(F+G) < n$ (что справедливо, учитывая предыдущие ограничения), то выполняется условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant n - \deg(F) + n - \deg(G) = 2n - \deg(F+G) > n.$$

2. Обозначим $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)^{\perp}$ для некоторого дивизора F. Далее проверим выполнимость условия 1 определения 3.

По аналогии с предыдущим случаем покажем, что $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, F-G)$:

$$\mathcal{C}_{\mathcal{L}}(D,F)^{\perp} * \mathcal{C}_{\mathcal{L}}(D,G) = \mathcal{C}_{\mathcal{L}}(D, D-F+(\omega)) * \mathcal{C}_{\mathcal{L}}(D,G)) \subseteq \mathcal{C}_{\mathcal{L}}(D, D+G-F+(\omega)) =$$
$$= \mathcal{C}_{\mathcal{L}}(D, D-(D+G-F+(\omega))+(\omega))^{\perp} = \mathcal{C}_{\mathcal{L}}(D, F-G)^{\perp} \;\Rightarrow\; \mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, F-G) \subseteq (\mathcal{A} * \mathcal{C})^{\perp}.$$

Рассмотрим три оставшихся условия определения 3 с целью уточнения границ для степени дивизора F:

— Если $\deg(F) > 2g-2$, то по теореме Римана — Роха выполняется

$$k(\mathcal{A}) \geqslant n + g - 1 - \deg(F).$$

Следовательно, условие 2 выполняется при $\deg(F) \leqslant n+g-t-2$. Учитывая нижнюю границу $\deg(F-G) > 2g-2$ и то, что $t \geqslant 0$, как и в прошлом случае, получаем ограничение $\deg(G) \leqslant n-g-1$ на степень дивизора G.

— Если $\deg(F) \geqslant \deg(G) + t + 2g - 1$, то выполняется следующее:

$$d(\mathcal{B}^{\perp}) \geqslant \deg(F-G) - 2g + 2 \geqslant \deg(G) + t + 2g - 1 - \deg(G) - 2g + 2 = t + 1 > t.$$

— Если $\deg(F-G) > 2g-2$ (всегда верно, учитывая предыдущие ограничения), то выполняется условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant \deg(F) - 2g + 2 + n - \deg(G) = n + \deg(F-G) - 2g + 2 > n.$$

Теорема 3 доказана. ■

Следующая теорема даёт некоторую классификацию относительно параметров как кодов из пары, исправляющей ошибки, так и самого кода, для которого эта пара представлена.

Теорема 4. Пусть $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D,G)$ — АГ-код, ассоциированный с функциональным полем $\mathcal{F}/\mathbb{F}_q$ рода g. Если пара кодов $(\mathcal{A},\mathcal{B})$ является парой, исправляющей

$$t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$$

ошибок для кода $\mathcal{C}$, то:

1. В случае $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, G+F)^{\perp}$:
   - 1.1) если $\mathcal{A} = [n, t+1, n-t]$, то g — произвольный, $2g-2 < \deg(G) < n-3g+3$ и $\deg(F) = t+g$;
   - 1.2) если $\mathcal{B} = [n, t, n-t+1]$, то $g=0$, $0 < \deg(G) < n$ и $\deg(F) = n - \deg(G) - t - 1$;
   - 1.3) если $\mathcal{C} = [n, n-2t, 2t+1]$, то $g=0$, $\deg(G) = n-2t-1$ и $\deg(F) = t$.
2. В случае $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)^{\perp}$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, F-G)$:
   - 2.1) если $\mathcal{A} = [n, t+1, n-t]$, то g — произвольный, $2g-2 < \deg(G) < n-3g+3$ и $\deg(F) = n+g-t-2$;

2.2) если $\mathcal{B} = [n, t, n-t+1]$, то $g = 0$, $0 < \deg(G) < n$ и $\deg(F) = \deg(G)+t-1$;
2.3) если $\mathcal{C} = [n, n-2t, 2t+1]$, то $g = 0$, $\deg(G) = n-2t-1$ и $\deg(F) = n-t-2$.

Доказательство.

1. Вид кодов $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, G+F)^{\perp}$ получен в теореме 3.

1.1. Очевидно, если код $\mathcal{A}$ имеет параметры $[n, t+1, n-t]$, то $\mathcal{A}$ является MDS-кодом. Таким образом, если $2g-2 < \deg(F) < n$, то

$$k(\mathcal{A}) = \deg(F) + 1 - g = t + 1,$$

откуда $\deg(F) = t+g$. Учитывая, что $\deg(F) > 2g-2$, получаем ограничение $\deg(G) < n - 3g + 3$ на степень дивизора G.

Теперь проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{A} = [n, t+1, n-t]$:

— Поскольку $\deg(F) = t + g$, имеем

$$k(\mathcal{A}) = \deg(F) + 1 - g = t + 1 > t.$$

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) = d(\mathcal{C}_{\mathcal{L}}(D, G+F)) \geqslant n - \deg(G+F) = n - \deg(G) - t - g.$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant n - \deg(G) - t - g > t$. Данное неравенство имеет место при любых значениях $\deg(G)$, поскольку $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$.

— Применяя аналогичные рассуждения, проверяем условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant n - \deg(F) + n - \deg(G) = 2n - t - g - \deg(G).$$

Соответственно $d(\mathcal{A}) + d(\mathcal{C}) > n$, если $t < n - g - \deg(G)$. Ввиду того, что $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$, неравенство $(n - \deg(G) - g - 1)/2 \leqslant n - g - \deg(G)$ выполняется всегда.

1.2. Если код $\mathcal{B}$ имеет параметры $[n, t, n-t+1]$, то он является MDS-кодом. Таким образом, если $2g-2 < \deg(G+F) < n$, то

$$k(\mathcal{B}) = n + g - 1 - \deg(G+F) = t,$$

откуда $\deg(F) = n + g - 1 - t - \deg(G)$.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{B} = [n, t, n-t+1]$:

— Поскольку $\deg(F) = n + g - 1 - t - \deg(G)$, имеем

$$k(\mathcal{A}) = \deg(F) + 1 - g = n - t - \deg(G) > t,$$

следовательно, $t < (n - \deg(G))/2$, что выполняется при любых значениях $\deg(G)$ при $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$.

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) \geqslant n - \deg(G+F) = n - \deg(G) - \deg(F) = t + 1 - g.$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant t + 1 - g > t$, что возможно лишь при $g = 0$.

— Применяя аналогичные рассуждения, проверяем условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant n - \deg(F) + n - \deg(G) = n + t + 1.$$

Соответственно $d(\mathcal{A}) + d(\mathcal{C}) > n$, если $t > -1$. Поскольку $t = \lfloor (n - \deg(G) - 1)/2 \rfloor$, неравенство $(n - \deg(G) - 1)/2 > -1$ выполняется при $\deg(G) < n + 1$, что всегда верно в силу первоначального выбора дивизора G.

1.3. Если код $\mathcal{C}$ имеет параметры $[n, n-2t, 2t+1]$, то он является MDS-кодом. Тогда если $2g - 2 < \deg(G) < n$, то

$$k(\mathcal{C}) = \deg(G) + 1 - g = n - 2t,$$

откуда $\deg(G) = n + g - 2t - 1$.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{C} = [n, n - 2t, 2t + 1]$:

— Условие $k(\mathcal{A}) = \deg(F) + 1 - g > t$ имеет место, если $\deg(F) > t + g - 1$.

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) \geqslant n - \deg(G + F) = -g + 2t + 1 - \deg(F).$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant -g + 2t + 1 - \deg(F) > t$, т. е. $\deg(F) < t + 1 - g$. Окончательно имеем

$$t + g - 1 < \deg(F) < t + 1 - g,$$

что возможно лишь при $g = 0$ и, как следствие, $\deg(F) = t$.

— Проверим условие 4 определения 3 при $g = 0$:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant n - \deg(F) + n - \deg(G) = n + t + 1.$$

Очевидно, $d(\mathcal{A}) + d(\mathcal{C}) > n$.

2. Вид кодов $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)^{\perp}$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, F - G)$ получен в теореме 3.

2.1. Если код $\mathcal{A}$ имеет параметры $[n, t+1, n-t]$, то он является MDS-кодом. Таким образом, если $2g - 2 < \deg(F) < n$, то

$$k(\mathcal{A}) = n + g - 1 - \deg(F) = t + 1,$$

откуда $\deg(F) = n + g - t - 2$. Учитывая, что $\deg(F) > 2g - 2$, получаем ограничение $\deg(G) < n - 3g + 3$ на степень дивизора G.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{A} = [n, t + 1, n - t]$:

— Поскольку $\deg(F) = n + g - t - 2$, имеем

$$k(\mathcal{A}) = n + g - 1 - \deg(F) = t + 1 > t.$$

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) = d(\mathcal{C}_{\mathcal{L}}(D, F - G)) \geqslant \deg(F - G) - 2g + 2 = n - g - t - \deg(G).$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant n - g - t - \deg(G) > t$. Данное неравенство имеет место при любых значениях $\deg(G)$ при условии, что $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$.

— Применяя аналогичные рассуждения, проверяем условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant \deg(F) - 2g + 2 + n - \deg(G) = 2n - t - g - \deg(G).$$

Соответственно $d(\mathcal{A}) + d(\mathcal{C}) > n$, если $t < n - g - \deg(G)$. Ввиду того, что $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$, неравенство $(n - \deg(G) - g - 1)/2 \leqslant n - g - \deg(G)$ выполняется всегда.

2.2. Если код $\mathcal{B}$ имеет параметры $[n, t, n-t+1]$, то он является MDS-кодом. Таким образом, если $2g - 2 < \deg(F - G) < n$, то

$$k(\mathcal{B}) = \deg(F - G) + 1 - g = \deg(F) - \deg(G) + 1 - g = t,$$

откуда $\deg(F) = \deg(G) + t + g - 1$.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{B} = [n, t, n - t + 1]$:

— Поскольку $\deg(F) = \deg(G) + t + g - 1$, имеем

$$k(\mathcal{A}) = n + g - 1 - \deg(F) = n - t - \deg(G) > t,$$

следовательно, $t < (n - \deg(G))/2$, что справедливо при любых значениях $\deg(G)$ с учётом того, что $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$.

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) \geqslant \deg(F - G) - 2g + 2 = \deg(F) - \deg(G) - 2g + 2 = t + 1 - g.$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant t + 1 - g > t$, что возможно лишь при $g = 0$.

— Применяя аналогичные рассуждения, проверяем условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant \deg(F) + 2 + n - \deg(G) = n + t + 1.$$

Соответственно $d(\mathcal{A}) + d(\mathcal{C}) > n$, если $t > -1$. Поскольку $t = \lfloor (n - \deg(G) - 1)/2 \rfloor$, неравенство $(n - \deg(G) - 1)/2 > -1$ выполняется при $\deg(G) < n + 1$, что всегда верно в силу первоначального выбора дивизора G.

2.3. Если код $\mathcal{C}$ имеет параметры $[n, n-2t, 2t+1]$, то он является MDS-кодом. Тогда, если $2g - 2 < \deg(G) < n$, то

$$k(\mathcal{C}) = \deg(G) + 1 - g = n - 2t,$$

откуда $\deg(G) = n + g - 2t - 1$.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{C} = [n, n - 2t, 2t + 1]$:

— Условие $k(\mathcal{A}) = n + g - 1 - \deg(F) > t$ имеет место, если $\deg(F) < n + g - t - 1$.

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу и $\deg(G) = n + g - 2t - 1$:

$$d(\mathcal{B}^{\perp}) \geqslant \deg(F - G) - 2g + 2 = \deg(F) - \deg(G) - 2g + 2 = \deg(F) - n - 3g + 2t + 3.$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant \deg(F) - n - 3g + 2t + 3 > t$, т. е. $\deg(F) > n + 3g - t - 3$. Окончательно имеем

$$n + 3g - t - 3 < \deg(F) < n + g - t - 1,$$

что возможно лишь при $g = 0$ и, как следствие, $\deg(F) = n - t - 2$.

— Проверим условие 4 определения 3 при $g = 0$:

$$d(\mathcal{A}) + d(\mathcal{C}) \geqslant \deg(F) + 2 + n - \deg(G) = \deg(F) + 2t + 3.$$

Соответственно $d(\mathcal{A})+d(\mathcal{C}) > n$, если $\deg(F) > n-2t-3$, что справедливо, поскольку $\deg(F) = n - t - 2$.

Теорема 4 доказана. ■

Построим пару, исправляющую t ошибок для кода $\mathcal{C}^{\perp} = \mathcal{C}_{\mathcal{L}}(D, G)^{\perp}$.

Теорема 5. Пусть $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$ — АГ-код, ассоциированный с функциональным полем $\mathcal{F}/\mathbb{F}_q$ рода g, и $\mathcal{C}^{\perp} = \mathcal{C}_{\mathcal{L}}(D, G)^{\perp}$ — код, дуальный к $\mathcal{C}$. Если пара кодов $(\mathcal{A}, \mathcal{B})$ является парой, исправляющей $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$ ошибок для кода $\mathcal{C}^{\perp}$, то в случае $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)$ и $\mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, G - F)$:

1) если $\mathcal{A} = [n, t+1, n-t]$, то g — произвольный, $5g-5 < \deg(G) < n$ и $\deg(F) = t+g$;
2) если $\mathcal{B} = [n, t, n - t + 1]$, то $g = 0$, $0 < \deg(G) < n$ и $\deg(F) = \deg(G) - t + 1$;
3) если $\mathcal{C} = [n, t - 2t, 2t + 1]$, то $g = 0$, $\deg(G) = 2t - 1$ и $\deg(F) = t$.

Доказательство. Обозначим $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)$ для некоторого дивизора F. Далее мы оценим $\deg(F)$, но прежде построим код $\mathcal{B}$ так, чтобы выполнялось условие 1 определения 3.

Поскольку $\mathcal{A} * \mathcal{B} \subseteq \mathcal{C}$, то $\mathcal{B} \subseteq (\mathcal{A} * \mathcal{C}^{\perp})^{\perp}$. Положим $\mathcal{B} = (\mathcal{A} * \mathcal{C}^{\perp})^{\perp}$. В обозначениях АГ-кодов получаем

$$\mathcal{C}_{\mathcal{L}}(D, F) * \mathcal{C}_{\mathcal{L}}(D, G^{\perp}) \subseteq \mathcal{C}_{\mathcal{L}}(D, D-G+F+(\omega)) = \mathcal{C}_{\mathcal{L}}(D, G-F)^{\perp} \Rightarrow \mathcal{B} = \mathcal{C}_{\mathcal{L}}(D, G-F) \subseteq (\mathcal{A} * \mathcal{C}^{\perp})^{\perp}.$$

1. Если код $\mathcal{A}$ имеет параметры $[n, t + 1, n - t]$, то он является MDS-кодом. Таким образом, если $2g - 2 < \deg(F) < n$, то

$$k(\mathcal{A}) = \deg(F) + 1 - g = t + 1,$$

откуда $\deg(F) = t+g$. Учитывая, что $\deg(F) > 2g-2$, получаем ограничение $\deg(G) > > 5g - 5$ на степень дивизора G.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{A} = [n, t + 1, n - t]$:

— Поскольку $\deg(F) = t + g$, имеем

$$k(\mathcal{A}) = \deg(F) + 1 - g = t + 1 > t.$$

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) = d(\mathcal{C}_{\mathcal{L}}(D, G - F)^{\perp}) \geqslant \deg(G - F) - 2g - 2 = \deg(G) - t - 3g + 2.$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant \deg(G) - t - 3g + 2 > t$. Данное неравенство имеет место при любых значениях $\deg(G)$ с учётом того, что $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$.

— Применяя аналогичные рассуждения, проверяем условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}^{\perp}) \geqslant n - \deg(F) + \deg(G) - 2g + 2 = n + \deg(G) + 2 - t - 3g.$$

Соответственно $d(\mathcal{A}) + d(\mathcal{C}^{\perp}) > n$, если $t < \deg(G) + 2 - 3g$. Ввиду того, что $t = = \lfloor(\deg(G)+1-3g)/2\rfloor$, неравенство $(\deg(G)+1-3g)/2 \leqslant \deg(G)+2-3g$ выполняется всегда.

2. Если код $\mathcal{B}$ имеет параметры $[n, t, n-t+1]$, то он является MDS-кодом. Таким образом, если $2g-2 < \deg(G-F) < n$, то

$$k(\mathcal{B}) = \deg(G-F) + 1 - g = \deg(G) - \deg(F) + 1 - g = t,$$

откуда $\deg(F) = \deg(G) + 1 - g - t$.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{B} = [n, t, n-t+1]$:

— Поскольку $\deg(F) = \deg(G) + 1 - g - t$, имеем

$$k(\mathcal{A}) = \deg(F) + 1 - g = \deg(G) + 2 - t - 2g > t,$$

следовательно, $t < (\deg(G)+2-2g)/2$. Данное неравенство выполняется при любых значениях $\deg(G)$, поскольку $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$.

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу и $\deg(F) = \deg(G) + 1 - g - t$:

$$d(\mathcal{B}^{\perp}) \geqslant \deg(G-F) - 2g + 2 = \deg(G) - \deg(F) - 2g + 2 = t + 1 - g.$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant t + 1 - g > t$, что возможно лишь при $g = 0$.

— Применяя аналогичные рассуждения, проверяем условие 4 определения 3:

$$d(\mathcal{A}) + d(\mathcal{C}^{\perp}) \geqslant n - \deg(F) + \deg(G) - 2g + 2 = n + t + 1.$$

Очевидно, что $d(\mathcal{A}) + d(\mathcal{C}^{\perp}) > n$.

3. Если код $\mathcal{C}$ имеет параметры $[n, n-2t, 2t+1]$, то он является MDS-кодом. Тогда если $2g-2 < \deg(G) < n$, то $k(\mathcal{C}) = n+g-1-\deg(G) = n-2t$, откуда $\deg(G) = 2t+g-1$.

Проверим, при каких ограничениях выполняются три оставшихся условия определения 3, если $\mathcal{C} = [n, n-2t, 2t+1]$:

— Условие $k(\mathcal{A}) = \deg(F) + 1 - g > t$ имеет место, если $\deg(F) > t + g - 1$.

— Оценим минимальное расстояние кода $\mathcal{B}^{\perp}$, учитывая его нижнюю границу:

$$d(\mathcal{B}^{\perp}) \geqslant \deg(G-F) - 2g + 2 = \deg(G) - \deg(F) - 2g + 2 = 2t - g + 1 - \deg(F).$$

По условию определения необходимо, чтобы $d(\mathcal{B}^{\perp}) \geqslant 2t - g + 1 - \deg(F) > t$, т. е. $\deg(F) < t + 1 - g$. Окончательно имеем

$$t + g - 1 < \deg(F) < t - g + 1,$$

что возможно лишь при $g = 0$ и, как следствие, $\deg(F) = t$.

— Проверим условие 4 определения 3 при $g = 0$:

$$d(\mathcal{A}) + d(\mathcal{C}^{\perp}) \geqslant n - \deg(F) + \deg(G) - 2g + 2 = n - \deg(F) + 2t + 1.$$

Соответственно $d(\mathcal{A}) + d(\mathcal{C}^{\perp}) > n$, если $\deg(F) < 2t+1$, что справедливо, поскольку $\deg(F) = t$.

Теорема 5 доказана. ■

Интересным объектом исследования относительно кодовых криптосистем являются подполевые подкоды, поскольку существует гипотеза, что именно такие коды являются стойкими к атаке на основе пар, исправляющих ошибки (по аналогии с классическими кодами Гоппы, являющимися некоторой модификацией подполевых подкодов обобщённых кодов Рида— Соломона). Дадим определение подполевого подкода.

Определение 4. Пусть код $\mathcal{C}$ определён над полем $\mathbb{F}_q$ ($\mathcal{C} \subseteq \mathbb{F}_q^n$) и $\mathbb{F}_p \subseteq \mathbb{F}_q$. *Подполевым подкодом* линейного кода $\mathcal{C}$ называется код $\mathcal{C}|\mathbb{F}_p = \mathcal{C} \cap \mathbb{F}_p^n$.

В действительности если $\mathcal{C}|\mathbb{F}_p$ — подполевой подкод кода $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$, определённого над $\mathbb{F}_q$, и $\mathbb{F}_p \subseteq \mathbb{F}_q$, то, согласно [9], пара, исправляющая t ошибок для кода $\mathcal{C}$, является парой, исправляющей такое же количество ошибок и для подполевого подкода $\mathcal{C}|\mathbb{F}_p$. При этом алгоритм декодирования работает над расширением $\mathbb{F}_q$ конечного поля $\mathbb{F}_p$ за время $\mathcal{O}((mn)^3)$, где $q = p^m$. Соответственно вопрос редукции сложности задачи декодирования подполевых подкодов сводится к нахождению пары, исправляющей ошибки для подполевого подкода над $\mathbb{F}_p$.

Теорема 6. Пусть $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$ — АГ-код, ассоциированный с функциональным полем $\mathcal{F}/\mathbb{F}_q$ рода g, где $q = p^2$. Тогда если $\operatorname{supp}(G) \cap \operatorname{supp}(F) = \varnothing$, то парой, исправляющей $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$ ошибок для кода $\mathcal{C}|\mathbb{F}_p$, является пара кодов $(\tilde{\mathcal{A}}, \tilde{\mathcal{B}})$ при условии их существования:

1) $\tilde{\mathcal{A}} = (\mathcal{C}_{\mathcal{L}}(D, F)^{\perp})|_{\mathbb{F}_p}$, если

$$g = 0 \quad \text{и} \quad \left[\begin{array}{l} n = 6,\ \deg(G) = 1,\ \deg(F) = 1 \\ \text{или} \\ n = 5,\ \deg(G) \leqslant 2,\ \deg(F) = 1 \end{array}\right.$$

или

$$g = 1,\ n = 5,\ \deg(G) = 1,\ \deg(F) = 2.$$

2) $\tilde{\mathcal{A}} = (\mathcal{C}_{\mathcal{L}}(D, F)|_{\mathbb{F}_p})^{\perp}$, если

$$g = 0, \quad n = 4, \quad \deg(G) = 1, \quad \deg(F) \leqslant 2;$$

или

$$\begin{array}{l} 0 \leqslant g \leqslant 1, \\ n \text{ — чётное} \end{array} \quad \text{и} \quad \left[\begin{array}{l} n \geqslant 6,\ 2g-2 < \deg(G) < n-g-3,\ \deg(F) \leqslant (n - \deg(G) + g + 1)/2 \\ \text{или} \\ n \geqslant 10, \quad \deg(G) = n - g - 3, \quad \deg(F) \leqslant n - \deg(G) - 1; \end{array}\right.$$

или

$g = 2$, $n \geqslant 10$, n — чётное и $2 < \deg(G) < n - 5$, $\deg(F) \leqslant (n - \deg(G) + 3)/2$;

или

$$g \geqslant 3, \quad n > 5g - 4, \quad n \text{ — чётное}, \quad 2g - 2 < \deg(G) < n - 3g + 3,$$
$$\deg(F) \leqslant (n - \deg(G) + g + 1)/2;$$

или

$$g \leqslant (n-1)/3, \quad n = 4, 6, 8, \quad \deg(G) = n - g - 3, \quad \deg(F) \leqslant n - \deg(G) - 1,$$

и в каждом случае $\deg(F) > 2g - 2$.

3) $\tilde{\mathcal{A}} = \mathcal{C}_{\mathcal{L}}(D, F)|_{\mathbb{F}_p}$, если

$$g = 0, \quad n = 3, \quad \deg(G) = 1, \quad \deg(F) = 1.$$

4) $\tilde{\mathcal{A}} = ((\mathcal{C}_{\mathcal{L}}(D, F)^{\perp})|_{\mathbb{F}_p})^{\perp}$, если

$$g = 0 \quad \text{и} \quad \left[\begin{array}{l} n = 4, \quad \deg(G) = 1, \quad \deg(F) \geqslant \deg(G) \\ \text{или} \\ n = 6, 8, 10, \quad \deg(G) \leqslant n - 5, \quad \deg(F) \geqslant (n + \deg(G) - 5)/2; \end{array}\right.$$

или

$$1 \leqslant g \leqslant 2 \quad \text{и} \quad \left[\begin{array}{l} n \geqslant 3g+1,\ \deg(G) = n-g-3,\ \deg(F) \geqslant \deg(G)+2g-1 \\ \text{или} \\ n \geqslant 4g+2,\ 2g-1 \leqslant \deg(G) \leqslant n-g-4 \quad \text{и} \\ \quad \deg(F) \geqslant (n+\deg(G)+3g-5)/2; \end{array}\right.$$
n — чётное

или

$$g=3, \quad n \geqslant 12 \text{ и } n \text{ — чётное}, \quad 4 < \deg(G) \leqslant n-6, \quad \deg(F) \geqslant (n+\deg(G)+4)/2;$$

или

$$g \geqslant 5,\ n \geqslant 5g-5 \text{ и } n \text{ — чётное},\ 2g-2 < \deg(G) \leqslant n-3g+3,$$
$$\deg(F) \geqslant (n+\deg(G)+3g-5)/2,$$

и в каждом случае $\deg(F) < n$.

Во всех четырёх случаях $\tilde{\mathcal{B}} = (\tilde{\mathcal{A}} * \mathcal{C}|_{\mathbb{F}_p})^{\perp}$.

Доказательство. Обозначим $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D,F)$ и отметим, что изначально код $\mathcal{C}$ определён над квадратичным расширением поля $\mathbb{F}_p$, т. е. $\mathbb{F}_q = \mathbb{F}_{p^2}$.

1. Проверим, является ли пара кодов $\tilde{\mathcal{A}} = \mathcal{A}^{\perp}|_{\mathbb{F}_p}$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}|_{\mathbb{F}_p})^{\perp}$ парой, исправляющей ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}}) = k(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) \geqslant 2k(\mathcal{A}^{\perp}) - n = n + 2g - 2 - 2\deg(F)$.
Так как $t = (n - \deg(G) - g - 1)/2$, условие $k(\tilde{\mathcal{A}}) > t$ имеет место, если $\deg(F) \leqslant$ $\leqslant (\deg(G) + n + 5g - 3)/4$.

— $d(\tilde{\mathcal{B}}^{\perp}) = d(\mathcal{A}^{\perp}|_{\mathbb{F}_p} * \mathcal{C}|_{\mathbb{F}_p}) \geqslant d((\mathcal{A}^{\perp} * \mathcal{C})|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp} * \mathcal{C})$.
Поскольку $\mathcal{A}^{\perp} = \mathcal{C}_{\mathcal{L}}(D, D - F + (\omega))$ и $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D,G)$, то

$$\mathcal{A}^{\perp} * \mathcal{C} \subseteq \mathcal{C}_{\mathcal{L}}(D, D + G - F + (\omega)) = \mathcal{C}_{\mathcal{L}}(D, F-G)^{\perp},$$

тогда

$$d(\mathcal{A}^{\perp} * \mathcal{C}) \geqslant d(\mathcal{C}_{\mathcal{L}}(D, F-G)^{\perp})) \geqslant \deg(F-G) - 2g + 2 = \deg(F) - \deg(G) - 2g + 2.$$

Так как $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$, условие $d(\tilde{\mathcal{B}}^{\perp}) > t$ выполняется при $\deg(F) \geqslant$ $\geqslant (\deg(G) + n + 3g - 5)/2$.

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) = d(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) + d(\mathcal{C}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp}) + d(\mathcal{C}) \geqslant \deg(F) - 2g + 2 + n - \deg(G)$.
Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) > n$, если $\deg(F) \geqslant \deg(G) + 2g - 1$.

Таким образом, $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$ — пара, исправляющая ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$, если имеет место следующая система:

$$\begin{cases} \deg(F) \leqslant (\deg(G) + n + 5g - 3)/4, \\ \deg(F) \geqslant (\deg(G) + n + 3g - 5)/2, \\ \deg(F) \geqslant \deg(G) + 2g - 1. \end{cases}$$

Здесь следует рассмотреть два случая:

— Условие $(\deg(G)+n+3g-5)/2 \leqslant \deg(F) \leqslant (\deg(G)+n+5g-3)/4$ выполняется, если $\deg(G) \leqslant n-g-3$ и $\deg(G) \leqslant -n-g+7$. Соответственно получаем следующие значения для рода g, длины n и степени $\deg(G)$:

$$\begin{array}{lll} g=0 & \text{и} & \begin{array}{lll} n=6, & \deg(G)=1, & \text{откуда } \deg(F)=1, \\ n=5, & \deg(G)\leqslant 2, & \text{откуда } \deg(F)=1; \end{array} \\ g=1 & \text{и} & n=5, \quad \deg(G)=1, \quad \text{откуда } \deg(F)=2. \end{array}$$

— Условие $\deg(G)+2g-1 \leqslant \deg(F) \leqslant (\deg(G)+n+5g-3)/4$ выполняется, если $n-g-3 \leqslant deg(G) \leqslant (n-3g+1)/3$. Соответственно получаем следующие значения для рода g, длины n и степени $\deg(G)$:

$$\begin{array}{llllll} g=0 & \text{и} & n=5, & \deg(G)=2, & \text{откуда} & \deg(F)=1; \\ g=1 & \text{и} & n=5, & \deg(G)=1, & \text{откуда} & \deg(F)=2. \end{array}$$

2. Проверим, может ли пара кодов $\tilde{\mathcal{A}} = (\mathcal{A}|_{\mathbb{F}_p})^{\perp}$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}|_{\mathbb{F}_p})^{\perp}$ являться парой, исправляющей ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}}) = k((\mathcal{A}|_{\mathbb{F}_p})^{\perp}) = n - k(\mathcal{A}|_{\mathbb{F}_p})$.
Так как $t = \lfloor (n-\deg(G)-g-1)/2 \rfloor$, условие $k(\tilde{\mathcal{A}}) > t$ имеет место, если $\deg(F) \leqslant$ $\leqslant (\deg(G)+3n+5g-3)/4$.

— $d(\tilde{\mathcal{B}}^{\perp}) = d((\mathcal{A}|_{\mathbb{F}_p})^{\perp} * \mathcal{C}|_{\mathbb{F}_p})$.
Рассмотрим случай, когда код $\tilde{\mathcal{A}}$ является самодуальным, т. е. $(\mathcal{A}|_{\mathbb{F}_p})^{\perp} = \mathcal{A}|_{\mathbb{F}_p}$. Следовательно, $k(\mathcal{A}|_{\mathbb{F}_p}) = n/2$, что возможно при $\deg(F) \leqslant (3n+4g-4)/4$. Тогда получаем

$$d(\tilde{\mathcal{B}}^{\perp}) = d(\mathcal{A}|_{\mathbb{F}_p} * \mathcal{C}|_{\mathbb{F}_p}) \geqslant d((\mathcal{A}*\mathcal{C})|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}*\mathcal{C}) \geqslant d(\mathcal{C}_{\mathcal{L}}(D, F+G)) \geqslant n-\deg(F)-\deg(G).$$

Поскольку $t = \lfloor (n-\deg(G)-g-1).2 \rfloor$, условие $d(\tilde{\mathcal{B}}^{\perp}) > t$ выполняется при $\deg(F) \leqslant$ $\leqslant (n+g+1-\deg(G)).2$.
Имеем два случая для определения $\deg(F)$:

— Случай $\deg(F) \leqslant (3n+4g-4)/4$ имеет место при $\deg(G) \leqslant (6-n-2g)/2$, что возможно, если

$$g=0, \quad n=2,4, \quad \deg(G) \leqslant 2$$

или

$$g=1, \quad n=2, \quad \deg(G)=1.$$

— Случай $\deg(F) \leqslant (n+g+1-\deg(G))/2$ имеет место при $\deg(G) > (6-n-2g)/2$.

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) = d((\mathcal{A}|_{\mathbb{F}_p})^{\perp}) + d(\mathcal{C}|_{\mathbb{F}_p}) = d(\mathcal{A}|_{\mathbb{F}_p}) + d(\mathcal{C}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}) + d(\mathcal{C}) \geqslant 2n - \deg(F) -$ $- \deg(G)$. Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) > n$, если $\deg(F) < n - \deg(G)$.

Таким образом, построение пары $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, исправляющей ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$, возможно, если справедлива одна из систем:

$$\begin{cases} \deg(F) \leqslant (\deg(G)+3n+5g-3)/4, \\ \deg(F) \leqslant (3n+4g-4)/4, \\ \deg(G) \leqslant (6-n-2g)/2, \\ \deg(F) \leqslant n-\deg(G)-1 \end{cases} \quad \text{или} \quad \begin{cases} \deg(F) \leqslant (\deg(G)+3n+5g-3)/4, \\ \deg(F) \leqslant (n+g+1-\deg(G))/2, \\ \deg(G) > (6-n-2g)/2, \\ \deg(F) \leqslant n-\deg(G)-1. \end{cases}$$

Уточняя обе системы, окончательно получаем следующие результаты:

$$g=0,\quad n=4,\quad \deg(G)=1,\quad \deg(F)\leqslant 2,$$
$$0\leqslant g\leqslant 2,\ n\geqslant 9,\ 2g{-}2<\deg(G)<n{-}g{-}3,\ 2g{-}2<\deg(F)\leqslant (n{-}\deg(G){+}g{+}1)/2,$$
$$g\geqslant 3,\ n>5g{-}4,\ 2g{-}2<\deg(G)<n{-}3g{+}3,\ 2g{-}2<\deg(F)\leqslant (n{-}\deg(G){+}g{+}1)/2,$$
$$0\leqslant g\leqslant 1,\quad n\geqslant 9,\quad \deg(G)=n-g-3,\quad 2g-2<\deg(F)\leqslant n-\deg(G)-1,$$
$$g=0,1,\ n=6,8,\ 2g{-}2<\deg(G)<n{-}g{-}3,\ 2g{-}2<\deg(F)\leqslant (n{-}\deg(G){+}g{+}1)/2,$$
$$g\leqslant (n-1)/3,\quad n=4,6,8,\quad \deg(G)=n-g-3,\quad 2g-2<\deg(F)\leqslant n-\deg(G)-1.$$

Во всех случаях n — чётное.

3. Проверим, является ли пара кодов $\tilde{\mathcal{A}}=\mathcal{A}|_{\mathbb{F}_p}$ и $\tilde{\mathcal{B}}=(\tilde{A}*\mathcal{C}|_{\mathbb{F}_p})^{\perp}$ парой, исправляющей ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}}*\tilde{\mathcal{B}}\subseteq(\mathcal{C}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}})=k(\mathcal{A}|_{\mathbb{F}_p})\geqslant 2k(\mathcal{A})-n=2\deg(F)+2-2g-n$.
Так как $t=\lfloor(n-\deg(G)-g-1)/2\rfloor$, условие $k(\tilde{\mathcal{A}})>t$ имеет место, если $\deg(F)\geqslant$ $\geqslant(3n+3g-\deg(G)-5)/4$.

— $d(\tilde{\mathcal{B}}^{\perp})=d(\mathcal{A}|_{\mathbb{F}_p}*\mathcal{C}|_{\mathbb{F}_p})\geqslant d((\mathcal{A}*\mathcal{C})|_{\mathbb{F}_p})\geqslant d(\mathcal{A}*\mathcal{C})$.
Поскольку $\mathcal{A}=\mathcal{C}_{\mathcal{L}}(D,F)$ и $\mathcal{C}=\mathcal{C}_{\mathcal{L}}(D,G)$, то $\mathcal{A}*\mathcal{C}\subseteq\mathcal{C}_{\mathcal{L}}(D,G+F)$ и

$$d(\tilde{\mathcal{B}}^{\perp})\geqslant d(\mathcal{C}_{\mathcal{L}}(D,F+G)))\geqslant n-\deg(F+G)=n-\deg(F)-\deg(G).$$

Так как $t=\lfloor(n-\deg(G)-g-1)/2\rfloor$, условие $d(\tilde{\mathcal{B}}^{\perp})>t$ выполняется при $\deg(F)\leqslant$ $\leqslant(n+g+1-\deg(G))/2$.

— $d(\tilde{\mathcal{A}})+d(\mathcal{C}|_{\mathbb{F}_p})=d(\mathcal{A}|_{\mathbb{F}_p})+d(\mathcal{C}|_{\mathbb{F}_p})\geqslant d(\mathcal{A})+d(\mathcal{C})\geqslant 2n-\deg(F)-\deg(G)$.
Очевидно, что $d(\tilde{\mathcal{A}})+d(\mathcal{C}|_{\mathbb{F}_p})>n$, если $\deg(F)\leqslant n-\deg(G)-1$.

Таким образом, $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$ — пара, исправляющая ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$, если справедлива следующая система:

$$\begin{cases}\deg(F)\geqslant(3n+3g-\deg(G)-5)/4,\\ \deg(F)\leqslant(n+g+1-\deg(G))/2,\\ \deg(F)\leqslant n-\deg(G)-1.\end{cases}$$

Здесь следует рассмотреть два случая:

— Условие $(3n+3g-\deg(G)-5)/4\leqslant\deg(F)\leqslant n-\deg(G)-1$ выполняется, если $\deg(G)\geqslant n-g-2$ и $\deg(G)\leqslant(n+1-3g)/3$. Соответственно получаем следующие значения рода g, длины n, степени $\deg(G)$, а также степени $\deg(F)$:

$$g=0,\quad n=3,\quad \deg(G)=1,\quad \deg(F)=1.$$

— Условие $(3n+3g-\deg(G)-5)/4\leqslant\deg(F)\leqslant(n+g+1-\deg(G))/2$ не выполняется никогда, поскольку в результате накладывания ограничений на степени дивизоров получаем следующие несовместные системы:

$$\begin{cases}\deg(F)\geqslant(3n+3g-\deg(G)-5)/4,\\ \deg(F)\leqslant(n+g+1-\deg(G))/2,\\ n\leqslant 3;\quad 0<\deg(G)\leqslant n-g-3\end{cases}\qquad\text{или}\qquad\begin{cases}\deg(F)\geqslant(3n+3g-\deg(G)-5)/4,\\ \deg(F)\leqslant(n+g+1-\deg(G))/2,\\ n\geqslant 4;\quad 0<\deg(G)\leqslant 4-n-g.\end{cases}$$

4. Проверим, является ли пара кодов $\tilde{\mathcal{A}} = (\mathcal{A}^{\perp}|_{\mathbb{F}_p})^{\perp}$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}|_{\mathbb{F}_p})^{\perp}$ парой, исправляющей ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}}) = k((\mathcal{A}^{\perp}|_{\mathbb{F}_p})^{\perp}) = n - k(\mathcal{A}^{\perp}|_{\mathbb{F}_p})$.
Для выполнения условия 2 определения 3 необходимо, чтобы $k(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) < n - t$. С другой стороны, $k(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) \geqslant 2k(\mathcal{A}^{\perp}) - n = n + 2g - 2\deg(F) - 2$. Принимая во внимание, что $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$, и уточняя, при каком ограничении на $\deg(F)$ выполняются неравенства

$$n + 2g - 2\deg(F) - 2 \leqslant k(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) < n - t,$$

получаем $\deg(F) \geqslant (n + 3g - \deg(G) - 5)/4$.

— $d(\tilde{\mathcal{B}}^{\perp}) = d((\mathcal{A}^{\perp}|_{\mathbb{F}_p})^{\perp} * \mathcal{C}|_{\mathbb{F}_p})$.
Здесь снова будем рассматривать случай, когда код $\tilde{\mathcal{A}}$ является самодуальным, т. е. $(\mathcal{A}^{\perp}|_{\mathbb{F}_p})^{\perp} = \mathcal{A}^{\perp}|_{\mathbb{F}_p}$. Следовательно, $k(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) = n/2$, что возможно при $\deg(F) \geqslant (n+4g-4)/4$. Тогда $d(\tilde{\mathcal{B}}^{\perp}) = d(\mathcal{A}^{\perp}|_{\mathbb{F}_p} * \mathcal{C}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp} * \mathcal{C})$. Поскольку $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)$ и $\mathcal{C} = \mathcal{C}_{\mathcal{L}}(D, G)$, то

$$d(\tilde{\mathcal{B}}^{\perp}) \geqslant d(\mathcal{C}_{\mathcal{L}}(D, D-F+(\omega)) * \mathcal{C}_{\mathcal{L}}(D, G)) \geqslant d(\mathcal{C}_{\mathcal{L}}(D, F-G)^{\perp}) \geqslant \deg(F) - \deg(G) - 2g + 2.$$

Так как $t = \lfloor (n - \deg(G) - g - 1)/2 \rfloor$, условие $d(\tilde{\mathcal{B}}^{\perp}) > t$ выполняется при $\deg(F) \geqslant (n + \deg(G) + 3g - 5)/2$.
Следует рассмотреть два случая, чтобы определить $\deg(F)$:
— Случай $\deg(F) \geqslant (n + \deg(G) + 3g - 5)/2$ имеет место при $\deg(G) > (6 - n - 2g)/2$.
— Случай $\deg(F) \geqslant (n + 4g - 4)/4$ имеет место при $\deg(G) \leqslant (6 - n - 2g)/2$, что возможно, если

$$g = 0, \quad n = 2, 4, \quad \deg(G) \leqslant 2$$

или

$$g = 1, \quad n = 2, \quad \deg(G) = 1.$$

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) = d((\mathcal{A}^{\perp}|_{\mathbb{F}_p})^{\perp}) + d(\mathcal{C}|_{\mathbb{F}_p}) = d(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) + d(\mathcal{C}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp}) + d(\mathcal{C}) \geqslant \deg(F) - 2g + 2 + n - \deg(G)$. Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) > n$, если $\deg(F) > \deg(G) + 2g - 2$.

Таким образом, построение пары $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, исправляющей ошибки для кода $\mathcal{C}|_{\mathbb{F}_p}$, возможно, если справедлива одна из систем:

$$\begin{cases} \deg(F) \geqslant (n + 3g - \deg(G) - 5)/4, \\ \deg(F) \geqslant (n + \deg(G) + 3g - 5)/2, \\ \deg(G) > (6 - n - 2g)/2, \\ \deg(F) > \deg(G) + 2g - 2 \end{cases} \quad \text{или} \quad \begin{cases} \deg(F) \geqslant (n + 3g - \deg(G) - 5)/4, \\ \deg(F) \geqslant (n + 4g - 4)/2, \\ \deg(G) \leqslant (6 - n - 2g)/2, \\ \deg(F) > \deg(G) + 2g - 2. \end{cases}$$

Уточняя обе системы, окончательно получаем следующие результаты:

$$g = 0, \quad n = 4, \quad \deg(G) = 1, \quad \deg(G) \leqslant \deg(F) \leqslant n - 1,$$
$$g = 0,\ n = 6, 8, 10,\ (6 - n)/2 < \deg(G) \leqslant n - 5,\ (n + \deg(G) - 5)/2 \leqslant \deg(F) \leqslant n - 1,$$
$$g = 0,\ n \geqslant 5 \text{ и } n \text{ — чётное},\ n - 2 \leqslant \deg(G) \leqslant n - 1,\ \deg(G) - 1 \leqslant \deg(F) \leqslant n - 1,$$
$$g = 1,\ n \geqslant 4 \text{ и } n \text{ — чётное},\ n - 4 \leqslant \deg(G) \leqslant n - 2,\ \deg(G) + 1 \leqslant \deg(F) \leqslant n - 1,$$
$$g = 1,\ n \geqslant 6 \text{ и } n \text{ — чётное},\ 1 \leqslant \deg(G) \leqslant n - 5,\ (n + \deg(G) - 2)/2 \leqslant \deg(F) \leqslant n - 1,$$

$$g=2,\ n\geqslant 8 \text{ и } n \text{ — чётное},\ n-5\leqslant \deg(G)\leqslant n-4,\ \deg(G)+3\leqslant \deg(F)\leqslant n-1,$$
$$g=2,\ n\geqslant 9 \text{ и } n \text{ — чётное},\ 3\leqslant \deg(G)\leqslant n-6,\ (n+\deg(G)+1)/2\leqslant \deg(F)\leqslant n-1,$$
$$g=3,\ n\geqslant 12 \text{ и } n \text{ — чётное},\ 4<\deg(G)\leqslant n-6,\ \deg(F)\geqslant (n+\deg(G)+4)/2,$$
$$g\geqslant 5, n\geqslant 5g-5 \text{ и } n \text{ — чётное}, 2g-2<\deg(G)\leqslant n-3g+3, \deg(F)\geqslant (n+\deg(G)+3g-5)/2.$$

Теорема 6 доказана. ■

Теорема 7. Пусть $\mathcal{C}=\mathcal{C}_{\mathcal{L}}(D,G)$ — АГ-код, ассоциированный с функциональным полем $\mathcal{F}/\mathbb{F}_q$ рода g, где $q=p^2$, и $\mathcal{C}^{\perp}$ — дуальный к $\mathcal{C}$. Тогда если $\operatorname{supp}(G)\cap\operatorname{supp}(F)=\varnothing$, то парой, исправляющей $t=\lfloor(\deg(G)-3g+1)/2\rfloor$ ошибок для кода $(\mathcal{C}^{\perp})|_{\mathbb{F}_p}$, является пара кодов $(\tilde{\mathcal{A}},\tilde{\mathcal{B}})$ при условии их существования:

1) $\tilde{\mathcal{A}}=\mathcal{C}_{\mathcal{L}}(D,F)|_{\mathbb{F}_p}$, если

$$g=0 \quad \text{и} \quad \left[\begin{array}{llll} n=7, & \deg(G)=5, & \deg(F)=4,\\ \text{или} \quad n=6, & \deg(G)=3, & \deg(F)=3,\\ \text{или} \quad n=5, & \deg(G)=1, & \deg(F)=3,\\ \text{или} \quad n=3, & \deg(G)=1, & \deg(F)\leqslant 2. \end{array}\right.$$

2) $\tilde{\mathcal{A}}=(\mathcal{C}_{\mathcal{L}}(D,F)^{\perp})|_{\mathbb{F}_p}$, если

$$g=0 \quad \text{и} \quad n=5, \quad \deg(G)=1, \quad \deg(F)=1$$
$$\text{или} \quad g=1 \quad \text{и} \quad n=5, \quad \deg(G)=4, \quad \deg(F)=2.$$

3) $\tilde{\mathcal{A}}=(\mathcal{C}_{\mathcal{L}}(D,F)|_{\mathbb{F}_p})^{\perp}$, если

$$g=0 \text{ и } \left[\begin{array}{l} n=4,6,\ (3n-10)/2\leqslant \deg(G)\leqslant n,\ \deg(F)\leqslant (4n-\deg(G)-5)/4,\\ \text{или}\\ n=10,\ 1\leqslant \deg(G)\leqslant 8,\ \deg(F)\leqslant (35-\deg(G))/4,\\ \text{или}\\ n\geqslant 12 \text{ и } n \text{ — чётное},\ 1\leqslant \deg(G)\leqslant n-2,\ \deg(F)\leqslant (\deg(G)+3)/2, \end{array}\right.$$

или

$$g=1, \quad n\geqslant 4 \text{ и } n \text{ — чётное}, \quad 2\leqslant \deg(G)\leqslant n-1, \quad \deg(F)\leqslant (\deg(G)+2)/2,$$

или $\quad g\geqslant 2, \quad n\geqslant 6 \text{ и } n \text{ — чётное}, \quad \deg(G)=4, \quad \deg(F)=1,$

или

$$g\geqslant 2,\ n\geqslant 5g-5 \text{ и } n \text{ — чётное},\ 5g-6\leqslant \deg(G)\leqslant n-1,\ \deg(F)\leqslant (\deg(G)+3-g)/2.$$

4) $\tilde{\mathcal{A}}=((\mathcal{C}_{\mathcal{L}}(D,F)^{\perp})|_{\mathbb{F}_p})^{\perp}$, если

$$g=0 \quad \text{и} \quad \left[\begin{array}{l} n=4, \quad \deg(G)=2, \quad 1\leqslant \deg(F)\leqslant 3\\ \text{или}\\ n=6,8,\ 2\leqslant \deg(G)\leqslant (3n-11)/2,\ \deg(F)\geqslant (2n-\deg(G)-7)/2 \end{array}\right.$$

или $\quad 1\leqslant g\leqslant 3, \quad n\geqslant 3g+3 \text{ и } n \text{ — чётное}, \quad 3g+2<\deg(G)\leqslant n-1,$

$$\deg(F)\geqslant (2n-\deg(G)+5g-7)/2.$$

Во всех четырёх случаях $\mathcal{B}=(\mathcal{A}*(\mathcal{C}^{\perp})|_{\mathbb{F}_p})^{\perp}$.

Доказательство. Обозначим $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)$ и отметим, что изначально код $\mathcal{C}^{\perp}$ определён над квадратичным расширением поля $\mathbb{F}_p$, т. е. над $\mathbb{F}_q = \mathbb{F}_{p^2}$.

1. Проверим, является ли пара кодов $\tilde{\mathcal{A}} = \mathcal{A}|_{\mathbb{F}_p}$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}^{\perp}|_{\mathbb{F}_p})^{\perp}$ парой, исправляющей ошибки для кода $\mathcal{C}^{\perp}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}^{\perp}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}}) = 2k(\mathcal{A}|_{\mathbb{F}_p}) \geqslant k(\mathcal{A}) - n = 2\deg(F) - 2g - n + 2$.
Так как $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$, условие $k(\tilde{\mathcal{A}}) > t$ имеет место, если $\deg(F) \geqslant$ $\geqslant (2n + \deg(G) + g - 3)/4$.

— $d(\tilde{\mathcal{B}}^{\perp}) = d(\mathcal{A}|_{\mathbb{F}_p} * \mathcal{C}^{\perp}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A} * \mathcal{C}^{\perp})$.
Поскольку $\mathcal{A} = \mathcal{C}_{\mathcal{L}}(D, F)$ и $\mathcal{C}^{\perp} = \mathcal{C}_{\mathcal{L}}(D, D - G + (\omega))$, то

$$\mathcal{A} * \mathcal{C}^{\perp} \subseteq \mathcal{C}_{\mathcal{L}}(D, D + F - G + (\omega)) = \mathcal{C}_{\mathcal{L}}(D, G - F)^{\perp},$$

тогда

$$d(\mathcal{A} * \mathcal{C}^{\perp}) \geqslant d(\mathcal{C}_{\mathcal{L}}(D, G - F)^{\perp}) \geqslant \deg(G) + 2 - \deg(F) - 2g.$$

Так как $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$, условие $d(\tilde{\mathcal{B}}^{\perp}) > t$ выполняется при $\deg(F) \leqslant$ $\leqslant (\deg(G) + 3 - g)/2$.

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}) + d(\mathcal{C}^{\perp}) \geqslant n - \deg(F) + \deg(G) - 2g + 2$.
Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) > n$, если $\deg(F) < \deg(G) - 2g + 2$.

Таким образом, $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$ — пара, исправляющая ошибки для кода $\mathcal{C}^{\perp}|_{\mathbb{F}_p}$, если имеет место система

$$\begin{cases} \deg(F) \geqslant (2n + \deg(G) + g - 3)/4, \\ \deg(F) \leqslant (\deg(G) + 3 - g)/2, \\ \deg(F) < \deg(G) + 2 - 2g. \end{cases}$$

Здесь следует рассмотреть два случая:

— Условие $(\deg(G) + g + 2n - 3)/4 \leqslant \deg(F) \leqslant (\deg(G) + 3 - g)/2$ выполняется, если

$$g = 0 \quad \text{и} \quad \left[\begin{array}{llll} & n = 7, & \deg(G) = 5, & \deg(F) = 4, \\ \text{или} & n = 6, & \deg(G) = 3, & \deg(F) = 3, \\ \text{или} & n = 5, & \deg(G) = 1, & \deg(F) = 3, \\ \text{или} & n = 3, & \deg(G) = 1, & \deg(F) \leqslant 2. \end{array}\right.$$

— Условие $(\deg(G) + g + 2n - 3)/4 \leqslant \deg(F) < \deg(G) - 2g + 2$ не выполняется никогда.

2. Проверим, является ли пара кодов $\tilde{\mathcal{A}} = (\mathcal{A}^{\perp})|_{\mathbb{F}_p}$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}^{\perp}|_{\mathbb{F}_p})^{\perp}$ парой, исправляющей ошибки для кода $\mathcal{C}^{\perp}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}^{\perp}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}}) = k((\mathcal{A}^{\perp})|_{\mathbb{F}_p}) \geqslant 2k(\mathcal{A}^{\perp}) - n = n + 2g - 2 - 2\deg(F)$.
Так как $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$, условие $k(\tilde{\mathcal{A}}) > t$ имеет место, если $\deg(F) \leqslant$ $\leqslant (2n + 7g - \deg(G) - 5)/4$.

— $d(\tilde{\mathcal{B}}^{\perp}) = d((\mathcal{A}^{\perp})|_{\mathbb{F}_p} * \mathcal{C}^{\perp}|_{\mathbb{F}_p}) \geqslant d((\mathcal{A}^{\perp} * \mathcal{C}^{\perp})|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp} * \mathcal{C}^{\perp})$.
Поскольку $\mathcal{A}^{\perp} = \mathcal{C}_{\mathcal{L}}(D, D - F + (\omega))$ и $\mathcal{C}^{\perp} = \mathcal{C}_{\mathcal{L}}(D, D - G + (\omega))$, то $\mathcal{A}^{\perp} * \mathcal{C}^{\perp} \subseteq$ $\subseteq \mathcal{C}_{\mathcal{L}}(D, 2D - G - F + 2(\omega))$, тогда

$$d(\mathcal{A}^{\perp} * \mathcal{C}^{\perp}) \geqslant d(\mathcal{C}_{\mathcal{L}}(D, 2D - G - F + 2(\omega))) \geqslant \deg(G) + \deg(F) - n - 4g + 4.$$

Поскольку $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$, условие $d(\tilde{\mathcal{B}}^{\perp}) > t$ выполняется при $\deg(F) \geqslant$ $\geqslant (2n + 5g - \deg(G) - 7)/2$.

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp}) + d(\mathcal{C}^{\perp}) \geqslant \deg(F) + \deg(G) - 4g + 4$.
Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) > n$, если $\deg(F) > n + 4g - \deg(G) - 4$.

Таким образом, $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$ — пара, исправляющая ошибки для кода $\mathcal{C}^{\perp}|_{\mathbb{F}_p}$, если имеет место следующая система:

$$\begin{cases} \deg(F) \leqslant (2n + 7g - \deg(G) - 5)/4, \\ \deg(F) \geqslant (2n + 5g - \deg(G) - 7)/2, \\ \deg(F) > n + 4g - \deg(G) - 4. \end{cases}$$

Здесь следует рассмотреть два случая:

— Условие $(2n + 5g - \deg(G) - 7)/2 \leqslant \deg(F) \leqslant (2n + 7g - \deg(G) - 5)/4$ выполняется, если

$$g = 1 \quad \text{и} \quad n = 5, \quad \deg(G) = 4, \quad \deg(F) = 2$$

или

$$g = 0 \quad \text{и} \quad n = 5, \quad \deg(G) = 1, \quad \deg(F) = 1.$$

— Условие $n + 4g - \deg(G) - 4 \leqslant \deg(F) \leqslant (2n + 7g - \deg(G) - 5)/4$ выполняется, если

$$1 \leqslant n \leqslant 3 \quad \text{и} \quad (2n + 9g - 11)/3 < \deg(G) < 3g - 1.$$

3. Проверим, является ли пара кодов $\tilde{\mathcal{A}} = (\mathcal{A}|_{\mathbb{F}_p})^{\perp}$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}^{\perp}|_{\mathbb{F}_p})^{\perp}$ парой, исправляющей ошибки для кода $\mathcal{C}^{\perp}|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}^{\perp}|_{\mathbb{F}_p})^{\perp}$.

— $k(\tilde{\mathcal{A}}) = k((\mathcal{A}|_{\mathbb{F}_p})^{\perp}) = n - k(\mathcal{A}|_{\mathbb{F}_p})$.
Отметим, что $k(\mathcal{A}|_{\mathbb{F}_p}) \geqslant 2k(\mathcal{A}) - n = 2\deg(F) + 2 - 2g - n$. С другой стороны, необходимо, чтобы выполнялось $k(\tilde{\mathcal{A}}) > t$. Поскольку $t = \lfloor(\deg(G) - 3g + 1)/2\rfloor$, окончательно имеем $\deg(F) \leqslant (4n + 7g - \deg(G) - 5)/4$.

— $d(\tilde{\mathcal{B}}^{\perp}) = d((\mathcal{A}|_{\mathbb{F}_p})^{\perp} * \mathcal{C}|_{\mathbb{F}_p})$.
Рассмотрим случай, когда код $\tilde{\mathcal{A}}$ является самодуальным, т. е. $(\mathcal{A}|_{\mathbb{F}_p})^{\perp} = \mathcal{A}|_{\mathbb{F}_p}$. Следовательно, $k(\mathcal{A}|_{\mathbb{F}_p}) = n/2$, что возможно при $\deg(F) \leqslant (3n + 4g - 4)/4$. Тогда получаем

$$d(\tilde{\mathcal{B}}^{\perp}) = d(\mathcal{A}|_{\mathbb{F}_p} * \mathcal{C}^{\perp}|_{\mathbb{F}_p}) \geqslant d((\mathcal{A} * \mathcal{C}^{\perp})|_{\mathbb{F}_p}) \geqslant d(\mathcal{A} * \mathcal{C}^{\perp}) \geqslant$$
$$\geqslant d(\mathcal{C}_{\mathcal{L}}(D, G - F)^{\perp}) \geqslant \deg(G) - \deg(F) - 2g + 2.$$

Так как $t = (\deg(G) + 1 - 3g)/2$, условие $d(\tilde{\mathcal{B}}^{\perp}) > t$ выполняется при $\deg(F) \leqslant$ $\leqslant (\deg(G) + 3 - g)/2$.
Следует рассмотреть два случая, чтобы определить $\deg(F)$:

— Случай $\deg(F) \leqslant (3n + 4g - 4)/4$ имеет место при $\deg(G) > (3n + 6g - 10)/2$, что возможно, если

$$g = 0, \quad n = 4, 6, 8, \quad \deg(G) \geqslant (3n - 10)/2, \quad \deg(F) \leqslant (3n - 4)/4$$

или

$$g = 1, \quad n = 2, \quad \deg(G) = 1, \quad \deg(F) = 1.$$

— Случай $\deg(F) \leqslant (\deg(G) + 3 - g)/2$ имеет место при $\deg(G) < (3n + 6g - 10)/2$.

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) = d((\mathcal{A}|_{\mathbb{F}_p})^\perp) + d(\mathcal{C}^\perp|_{\mathbb{F}_p}) = d(\mathcal{A}|_{\mathbb{F}_p}) + d(\mathcal{C}^\perp|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}) + d(\mathcal{C}) \geqslant n - \deg(F) + \deg(G) - 2g + 2$.
Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}|_{\mathbb{F}_p}) > n$, если $\deg(F) < \deg(G) + 2 - 2g$.

Таким образом, построение пары $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, исправляющей ошибки для кода $\mathcal{C}^\perp|_{\mathbb{F}_p}$, возможно, если справедлива одна из следующих систем:

$$\begin{cases} g = 0, n = 4, 6, 8, \\ \deg(G) > (3n - 10)/2, \\ \deg(F) \leqslant (3n - 4)/4, \\ \deg(F) \leqslant (4n - \deg(G) - 5)/4, \\ \deg(F) < \deg(G) + 2, \end{cases} \quad \text{или} \quad \begin{cases} g = 1, n = 2, \\ \deg(G) = 1, \\ \deg(F) = 1, \\ \deg(F) \leqslant (4n + 2 - \deg(G))/4, \\ \deg(F) < \deg(G), \end{cases}$$

$$\text{или} \quad \begin{cases} \deg(G) < (3n + 6g - 10)/2, \\ \deg(F) \leqslant (4n + 7g - \deg(G) - 5)/4, \\ \deg(F) \leqslant (\deg(G) + 3 - g)/2, \\ \deg(F) < \deg(G) + 2 - 2g. \end{cases}$$

Уточняя все три системы, окончательно получаем следующие результаты:

$$g = 0, \quad n = 4, 6, \quad (3n - 10)/2 \leqslant \deg(G) \leqslant n - 2, \quad \deg(F) \leqslant (4n - \deg(G) - 5)/4,$$
$$g = 0, \quad n = 10, \quad \deg(G) \leqslant 8, \quad \deg(F) \leqslant (35 - \deg(G))/4,$$
$$g = 0, \quad n > 10 \text{ и } n \text{ — чётное}, \quad 1 \leqslant \deg(G) \leqslant n - 2, \quad \deg(F) \leqslant (\deg(G) + 3)/2,$$
$$g = 1, \quad n \geqslant 4 \text{ и } n \text{ — чётное}, \quad 2 \leqslant \deg(G) \leqslant n - 1, \quad \deg(F) \leqslant (\deg(G) + 2)/2,$$
$$g \geqslant 2, \quad n \geqslant 6 \text{ и } n \text{ — чётное}, \quad \deg(G) = 4, \quad \deg(F) = 1,$$
$$g \geqslant 2, \ n \geqslant 5g - 5 \text{ и } n \text{ – чётное}, \ 5g - 6 \leqslant \deg(G) \leqslant n - 1, \ \deg(F) \leqslant (\deg(G) + 3 - g)/2.$$

4. Проверим, является ли пара кодов $\tilde{\mathcal{A}} = (\mathcal{A}^\perp|_{\mathbb{F}_p})^\perp$ и $\tilde{\mathcal{B}} = (\tilde{A} * \mathcal{C}^\perp|_{\mathbb{F}_p})^\perp$ парой, исправляющей ошибки для кода $\mathcal{C}^\perp|_{\mathbb{F}_p}$:

— Учитывая вид $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, получаем $\tilde{\mathcal{A}} * \tilde{\mathcal{B}} \subseteq (\mathcal{C}^\perp|_{\mathbb{F}_p})^\perp$.

— $k(\tilde{\mathcal{A}}) = k((\mathcal{A}^\perp|_{\mathbb{F}_p})^\perp) = n - k(\mathcal{A}^\perp|_{\mathbb{F}_p})$.
Для выполнения условия 2 определения 3 необходимо, чтобы $k(\mathcal{A}^\perp|_{\mathbb{F}_p}) < n - t$. С другой стороны, $k(\mathcal{A}^\perp|_{\mathbb{F}_p}) \geqslant 2k(\mathcal{A}^\perp) - n = n + 2g - 2\deg(F) - 2$. Принимая во внимание, что $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$, и уточняя, при каком ограничении на $\deg(F)$ выполняются неравенства

$$n + 2g - 2\deg(F) - 2 \leqslant k(\mathcal{A}^\perp|_{\mathbb{F}_p}) < n - t,$$

получаем $\deg(F) \geqslant (\deg(G) + g - 3)/4$.

— $d(\tilde{\mathcal{B}}^\perp) = d((\mathcal{A}^\perp|_{\mathbb{F}_p})^\perp * \mathcal{C}^\perp|_{\mathbb{F}_p})$.
Рассмотрим случай, когда код $\tilde{\mathcal{A}}$ является самодуальным, т. е. $(\mathcal{A}|_{\mathbb{F}_p})^\perp = \mathcal{A}|_{\mathbb{F}_p}$. Следовательно, $k(\mathcal{A}|_{\mathbb{F}_p}) = n/2$, что возможно при $\deg(F) \geqslant (n + 4g - 4)/4$. Тогда $d(\tilde{\mathcal{B}}^\perp) = d(\mathcal{A}^\perp|_{\mathbb{F}_p} * \mathcal{C}^\perp|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^\perp * \mathcal{C}^\perp)$. Поскольку $\mathcal{A} = \mathcal{C}_\mathcal{L}(D, F)$ и $\mathcal{C} = \mathcal{C}_\mathcal{L}(D, G)$, то

$$d(\tilde{\mathcal{B}}^\perp) \geqslant d(\mathcal{C}_\mathcal{L}(D, D - F + (\omega)) * \mathcal{C}_\mathcal{L}(D, D - G + (\omega))) \geqslant$$
$$\geqslant d(\mathcal{C}_\mathcal{L}(D, 2D - G - F + 2(\omega))) \geqslant n + 2g - 2 - 2\deg(F).$$

Так как $t = \lfloor(\deg(G) + 1 - 3g)/2\rfloor$, условие $d(\tilde{\mathcal{B}}^\perp) > t$ выполняется при $\deg(F) \geqslant (2n + 5g - \deg(G) - 7)/2$.
Следует рассмотреть два случая, чтобы определить $\deg(F)$:

— Случай $\deg(F) \geqslant (2n+5g-\deg(G)-7)/2$ имеет место при $\deg(G) \leqslant (3n+6g-10)/2$.
— Случай $\deg(F) \geqslant (n+4g-4)/4$ имеет место при $\deg(G) > (3n+6g-10)/2$, что возможно, если

$$g=0, \quad n=4,6,8, \quad \deg(G) > (3n-10)/2, \quad \deg(F) \geqslant (n-4)/4.$$

— $d(\tilde{\mathcal{A}}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) = d((\mathcal{A}^{\perp}|_{\mathbb{F}_p})^{\perp}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) = d(\mathcal{A}^{\perp}|_{\mathbb{F}_p}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) \geqslant d(\mathcal{A}^{\perp}) + d(\mathcal{C}^{\perp}) \geqslant \deg(F) + \deg(G) - 4g + 4$.
Очевидно, что $d(\tilde{\mathcal{A}}) + d(\mathcal{C}^{\perp}|_{\mathbb{F}_p}) > n$, если $\deg(F) > n - \deg(G) + 4g - 4$.

Таким образом, построение пары $\tilde{\mathcal{A}}$ и $\tilde{\mathcal{B}}$, исправляющей ошибки для кода $\mathcal{C}^{\perp}|_{\mathbb{F}_p}$, возможно, если справедлива одна из систем:

$$\begin{cases} g=0, \\ n=4,6,8, \\ \deg(G) > (3n-10)/2, \\ \deg(F) \geqslant (n-4)/4, \\ \deg(F) \geqslant n - \deg(G) - 3, \end{cases} \quad \text{или} \quad \begin{cases} g=1, \\ n=2, \\ \deg(G)=1, \\ \deg(F)=1, \\ \deg(F) \geqslant n+1-\deg(G), \end{cases}$$

$$\text{или} \quad \begin{cases} \deg(G) \leqslant (3n+6g-10)/2, \\ \deg(F) \geqslant (2n+5g-\deg(G)-7)/2, \\ \deg(F) > n+4g-\deg(G)-3. \end{cases}$$

Уточняя системы, окончательно получаем следующие результаты:

$$g=0, \quad n=4, \quad \deg(G)=2, \quad 1 \leqslant \deg(F) \leqslant 3;$$

$g=0$, $6 \leqslant n \leqslant 8$ и n — чётное, $2 \leqslant \deg(G) \leqslant (3n-11)/2$, $\deg(F) \geqslant (2n-\deg(G)-7)/2$;

$1 \leqslant g \leqslant 3$, $n \geqslant 3g+3$ и n — чётное, $3g+2 < \deg(G) \leqslant n-1$, $\deg(F) \geqslant (2n-\deg(G)+5g-7)/2$.

Теорема 7 доказана. ■

Замечание 1. Стоит отметить, что в условиях теорем 6 и 7 вовсе не гарантируется, что пара, исправляющая ошибки, существует для любого кода C с заданными параметрами; получены границы, при которых существование пары в принципе возможно. В п. 2 и 4 теоремы 6 и в п. 3 и 4 теоремы 7 рассматривается случай самодуальности кода $\tilde{\mathcal{A}}$, что на практике труднодостижимо. В дополнение, ввиду грубости границы для оценки размерности подполевого подкода, в общем случае коды, составляющие пару, могут вырождаться. Необходимы дополнительные вычислительные эксперименты для уточнения полученных границ для параметров пар, исправляющих ошибки для подполевого подкода.

Заключение

Для обеспечения условия 2 в определении пары, исправляющей ошибки, в теоремах 4 и 5 мы ограничиваемся рассмотрением случаев, когда $\deg(F) = t+g$ и $\deg(F) = n+g-t-2$, хотя данные значения являются нижней и верхней границами соответственно для $\deg(F)$ в зависимости от вида кода $\mathcal{A}$.

Отметим, что теоремы 6 и 7 доказаны для случая, когда исходный АГ-код определён над квадратичным расширением $\mathbb{F}_{p^2}$, чтобы получить более компактные соотношения. Построение пар, исправляющих ошибки для произвольного АГ-кода, определённого над расширениями бóльших степеней, — всё ещё открытый вопрос. Кроме того,

следует отметить, что в подслучаях, где мы рассматриваем самодуальный код, наличие пары, исправляющей ошибки, возможно, но необязательно выполнимо. Для более сильного утверждения необходимо провести ряд вычислительных экспериментов.

Весьма интересным представляется также вычисление пар, исправляющих ошибки для трэйс-кодов (такие коды получены с помощью применения к кодовым словам кода $\mathcal{C}$, определённым над $\mathbb{F}_q$, функции следа $\mathrm{tr} : \mathbb{F}_{p^m} \to \mathbb{F}_p$), поскольку такие коды связаны с дуальными соотношением $(\mathcal{C}|_{\mathbb{F}_p})^\perp = \mathrm{tr}(\mathcal{C}^\perp)$.

ЛИТЕРАТУРА

1. *Justesen J., Larsen K., Jensen H., et al.* Construction and decoding of a class of algebraic geometry codes // IEEE Trans. Inform. Theory. 1989. No. 35(4). P. 811–821.
2. *Skorobogatov A. N. and Vlădut S. G.* On the decoding of algebraic-geometric codes // IEEE Trans. Inform. Theory. 1990. No. 36(5). P. 1051–1060.
3. *Pellikaan R.* On decoding by error location and dependent sets of error positions // Discrete Math. 1992. No. 106–107. P. 369–381.
4. *Kötter R.* A unified description of an error locating procedure for linear codes // Proc. Algebraic Combinatorial Coding Theory III. Hermes, 1992. P. 113–117.
5. *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. No. 63. P. 5404–5418.
6. *Малыгина Е. С., Кунинец А. А.* Вычисление пар, исправляющих ошибки, для алгеброгеометрического кода // Прикладная дискретная математика. Приложение. 2023. № 16. С. 136–140.
7. *Milne J. S.* Algebraic Geometry. `https://www.jmilne.org/math/CourseNotes/AG510.pdf`.
8. *Stichtenoth H.* Algebraic Function Fields and Codes. Springer Verlag, 1991.
9. *Pellikaan R.* On the existence of error-correcting pairs // Statistical Planning and Inference. 1996. No. 51. P. 229–242.
10. *Marquez-Corbella I. and Pellikaan R.* Error-correcting pairs: a new approach to code-based cryptography // 20th Conf. ACA 2014, Jul 2014, New York, USA. `https://hal.science/hal-01088433`.
11. *Mumford D.* Varieties defined by quadratic equations // Questions on Algebraic Varieties. Berlin; Heidelberg: Springer, 2011. P. 29–100.

REFERENCES

1. *Justesen J., Larsen K., Jensen H., et al.* Construction and decoding of a class of algebraic geometry codes. IEEE Trans. Inform. Theory, 1989, no. 35(4), pp. 811–821.
2. *Skorobogatov A. N. and Vlădut S. G.* On the decoding of algebraic-geometric codes. IEEE Trans. Inform. Theory, 1990, no. 36(5), pp. 1051–1060.
3. *Pellikaan R.* On decoding by error location and dependent sets of error positions. // Discrete Math., 1992, no. 106–107, pp. 369–381.
4. *Kötter R.* A unified description of an error locating procedure for linear codes. Proc. Algebraic Combinatorial Coding Theory III, Hermes, 1992, pp. 113–117.
5. *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, no. 63, pp. 5404–5418.
6. *Malygina E. S. and Kuninets A. A.* Vychislenie par, ispravlyayushchikh oshibki, dlya algebro-geometricheskogo koda [Calculation of error-correcting pairs for an algebraic-geometric code]. Prikladnaya Diskretnaya Matematika. Prilozhenie, 2023, no. 16, pp. 136–140. (in Russian)

7. *Milne J. S.* Algebraic Geometry. https://www.jmilne.org/math/CourseNotes/AG510.pdf.
8. *Stichtenoth H.* Algebraic Function Fields and Codes. Springer Verlag, 1991.
9. *Pellikaan R.* On the existence of error-correcting pairs. Statistical Planning and Inference, 1996, no. 51, pp. 229–242.
10. *Marquez-Corbella I. and Pellikaan R.* Error-correcting pairs: a new approach to code-based cryptography. 20th Conf. ACA 2014, Jul 2014, New York, USA. https://hal.science/hal-01088433.
11. *Mumford D.* Varieties defined by quadratic equations. Questions on Algebraic Varieties. Berlin, Heidelberg, Springer, 2011, pp. 29–100.

ПРИКЛАДНАЯ ТЕОРИЯ ГРАФОВ

УДК 519.1, 004.05 DOI 10.17223/20710410/63/5

КОЛИЧЕСТВО АТТРАКТОРОВ И ЦИКЛИЧЕСКИХ СОСТОЯНИЙ В КОНЕЧНЫХ ДИНАМИЧЕСКИХ СИСТЕМАХ ОРИЕНТАЦИЙ ПОЛНЫХ ГРАФОВ

А. В. Жаркова

Саратовский национальный исследовательский государственный университет имени Н. Г. Чернышевского, г. Саратов, Россия

E-mail: ZharkovaAV3@gmail.com

Графовые модели занимают важное место в задачах, связанных с защитой информации и информационной безопасностью, в том числе при построении моделей и методов управления непрерывным функционированием и восстановлением систем, противодействия отказам в обслуживании. Рассматривается конечная динамическая система (Γ_{K_n}, α), $n \geqslant 1$, состояниями которой являются все возможные ориентации полного графа K_n, а эволюционная функция задаётся следующим образом: динамическим образом орграфа является орграф, полученный из исходного путём переориентации всех дуг, входящих в стоки, других отличий между исходным орграфом и его образом нет. Получены формулы для подсчёта количества циклических (принадлежащих аттракторам) состояний системы; состояний, не являющихся циклическими; аттракторов системы, в том числе различных типов. Приведены соответствующие таблицы для n от 1 до 20 включительно.

Ключевые слова: *аттрактор, граф, кибербезопасность, конечная динамическая система, отказоустойчивость, полный граф, циклическое состояние, эволюционная функция.*

NUMBER OF ATTRACTORS AND CYCLIC STATES IN FINITE DYNAMIC SYSTEMS OF COMPLETE GRAPHS ORIENTATIONS

A. V. Zharkova

Saratov State University, Saratov, Russia

Graph models occupy an important place in information security tasks, including the construction of models and methods for managing the continuous operation of systems and system recovery, countering denials of service. Finite dynamic systems of complete graphs orientations are considered. States of a dynamic system (Γ_{K_n}, α), $n \geqslant 1$, are all possible orientations of the complete graph K_n, and evolutionary function transforms the graph orientation by reversing all the arcs that enter into sinks, and there are no other differences between the given and the next digraphs. Formulas are obtained for counting the number of cyclic (belonging to attractors) system states and the number of states that are not cyclic (not belonging to attractors), namely, the number of states belonging to attractors is 1, if $n = 1$; $2^{(n-1)(n-2)/2}(2^{n-1} - n) + n!$, if $n > 1$,

the number of states not belonging to attractors is 0, if $n = 1$; $n \cdot 2^{(n-1)(n-2)/2} - n!$, if $n > 1$. Formulas are obtained for counting the number of attractors of the system, including various types, namely, the number of attractors of length 1 is 1, if $n = 1$; $2^{(n-1)(n-2)/2}(2^{n-1} - n)$, if $n > 1$, the number of attractors of length n is $(n-1)!$, the number of attractors (basins) is 1, if $n = 1$; $2^{(n-1)(n-2)/2}(2^{n-1} - n) + (n-1)!$, if $n > 1$. The corresponding tables are given for $n = 1, \ldots, 20$.

Keywords: *attractor, complete graph, cybersecurity, cyclic state, evolutionary function, fault-tolerance, finite dynamic system, graph.*

Введение

Графовые модели занимают важное место в задачах, связанных с информационной безопасностью. В вопросах кибербезопасности с помощью графовых моделей можно, например, выявлять связи между сущностями системы, группировать их, оценивать поведение, выявлять различные аномалии. В задачах, связанных с отказоустойчивостью компьютерных сетей, отказы процессоров интерпретируются как удаление соответствующих вершин, а отказы сетевых каналов — как удаление дуг.

При изучении модельных графов можно применять идеи и методы теории конечных динамических систем. В работе [1] представлены нетрадиционные приложения автоматов в алгебре, теории динамических систем, теории графов и спектральной теории. В модели [2] в качестве механизма восстановления работоспособности сети предлагается так называемая SER-динамика бесконтурных связных ориентированных графов. В [3] на множестве всех двоичных векторов заданной размерности вводится структура динамической системы, исследуются её свойства и устанавливается связь с динамикой из предыдущей модели. В [4] представлены конструктивные методы символической динамики и их приложения к изучению непрерывных и дискретных динамических систем. В работе [5] рассмотрены методологические аспекты динамического программирования, в том числе анализируются основные графовые интерпретации динамического программирования и представление структуры задачи динамического программирования с помощью графа взаимосвязей. В [6] рассматривается задача оптимального сопоставления для взвешенных графов и развивается новая аппроксимация этой проблемы путём построения динамических систем на многообразии ортогональных матриц. В [7] характеризуется циклическая эквивалентность класса конечных графовых динамических систем, при этом две конечные графовые динамические системы циклически эквивалентны, если их аттракторы изоморфны как ориентированные графы.

Модель InterSim [8] представляет собой гибкую среду общего назначения для моделирования графовых динамических систем и их обобщений. В [9] изучаются динамические системы, связанные с конечными двудольными разделёнными графами, алгебрами графов и парадоксальными разбиениями. В работе [10] описывается вебприложение GDSCalc для вычисления и характеристики динамики дискретных графовых динамических систем. В [11] характеризуется широкое обобщение динамических систем над графами, состояния которых могут принимать значения в произвольной булевой алгебре с 2^p элементами, $p \in \mathbb{N}$. В работе [12] излагаются концептуальные основы общей теории дискретных динамических, релейных и логико-динамических систем на основе использования общих фундаментальных свойств рассматриваемых классов — дискретности структур и физической декомпозиции. В [13] предлагаются формализация графовых моделей структур многокомпонентных динамических систем с примене-

нием маркированных графов и матричный способ описания процесса функционирования ориентированных и неориентированных маркированных графов. В [14] решается проблема сосуществования аттракторов в однородных булевых графовых динамических системах, которые индуцируются булевыми функциями минтерма и макстерма, с направленным базовым графом зависимостей. В модели [15] изучается влияние графа взаимодействия на конечную динамическую систему.

В настоящей работе полные графы изучаются с точки зрения динамического подхода к кибербезопасности и отказоустойчивости графовых систем. Подсчитываются количества циклических и не являющихся циклическими состояний, количество аттракторов в конечных динамических системах ориентаций полных графов. Предварительные результаты частично были анонсированы на научных конференциях [16, 17]. Данная работа является полной и завершающей эти исследования.

1. Основные определения и постановка задачи

Основные понятия теории дискретных систем, в частности графов, используются согласно [18].

Под *конечной динамической системой* понимается пара $(S,\ \delta)$, где S — конечное непустое множество состояний системы; $\delta : S \to S$ — отображение множества состояний в себя, называемое *эволюционной функцией системы*. Каждой конечной динамической системе сопоставляется карта, представляющая собой функциональный орграф с множеством вершин S и дугами, проведёнными из каждой вершины $s \in S$ в вершину $\delta(s)$. Компоненты связности орграфа, задающего динамическую систему, называются её *бассейнами*. Каждый бассейн представляет собой контур с входящими в него деревьями. Контуры, в свою очередь, называются *предельными циклами*, или *аттракторами*. Под *длиной* аттрактора будем понимать количество различных состояний в соответствующем контуре. Состояние, принадлежащее аттрактору, называется *циклическим*.

Основными проблемами теории конечных динамических систем являются задачи отыскания эволюционных параметров системы без построения карты и проведения динамических исследований на её основе. К числу таких характеристик относятся принадлежность состояния аттрактору, количество таких состояний, описание аттракторов системы, их количество.

В [19] описаны свойства принадлежности состояний аттракторам, сами аттракторы, подсчитано их количество в конечных динамических системах ориентаций некоторых типов графов. В данной работе подсчитываются количества циклических и не являющихся циклическими состояний, количество аттракторов в конечных динамических системах ориентаций полных графов.

2. Описание конечной динамической системы (Γ_{K_n}, α)

Пусть дан полный граф $G = K_n$, $n \geqslant 1$, $m = n(n-1)/2$ — число рёбер. Пометим его вершины и придадим его рёбрам произвольную ориентацию, тем самым получив направленный граф $\overrightarrow{G} = (V, \beta)$, где отношение смежности β антирефлексивно и антисимметрично. Применим к полученному орграфу эволюционную функцию α, которая у данного орграфа одновременно переориентирует все дуги, входящие в стоки, а остальные дуги оставляет без изменения, в результате получим орграф $\alpha(\overrightarrow{G})$. Если проделать указанные действия со всеми возможными ориентациями данного графа, то получим карту конечной динамической системы, состоящую из одного или нескольких бассейнов.

Таким образом, рассмотрим конечную динамическую систему (Γ_{K_n}, α), $n \geqslant 1$, где через Γ_{K_n} обозначено множество всех возможных ориентаций полного графа K_n, $|\Gamma_{K_n}| = 2^m$, а эволюционная функция α задана следующим образом: если дан некоторый орграф $\overrightarrow{G} \in \Gamma_{K_n}$, то его динамическим образом $\alpha(\overrightarrow{G})$ является орграф, полученный из $\overrightarrow{G}$ одновременной переориентацией всех дуг, входящих в стоки, других отличий между $\overrightarrow{G}$ и $\alpha(\overrightarrow{G})$ нет.

На рис. 1 изображён граф K_3 и карта конечной динамической системы (Γ_{K_3}, α).

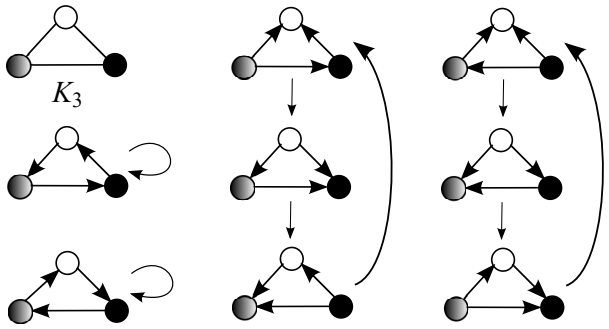

Рис. 1. Граф K_3 и карта конечной динамической системы (Γ_{K_3}, α)

В [2] рассматривается конечная динамическая система (Ω, α), где Ω — множество всех бесконтурных ориентаций данного связного графа, и отмечается, что для полного графа существует $n!$ бесконтурных ориентаций, где $n!$ — количество перестановок его вершин, при этом система имеет $(n-1)!$ бассейнов, каждый из которых состоит исключительно из аттрактора длины n, то есть все состояния данной системы являются циклическими.

Под *вектором степеней захода* орграфа будем понимать вектор, компонентами которого являются расположенные в убывающем порядке степени захода всех его вершин. Например, на рис. 1 расположенный сверху справа орграф имеет вектор степеней захода $(2, 1, 0)$.

3. Количество циклических состояний в конечной динамической системе (Γ_{K_n}, α)

Теорема 1 [20]. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, состояние $\overrightarrow{G} \in \Gamma_{K_n}$ принадлежит аттрактору (является циклическим) тогда и только тогда, когда орграф $\overrightarrow{G}$:

1) не имеет стока или
2) имеет вектор степеней захода $(n-1, n-2, \ldots, 0)$.

Теорема 2. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, количество принадлежащих аттракторам (циклических) состояний равно

$$\begin{cases} 1, & \text{если } n = 1, \\ 2^{(n-1)(n-2)/2}(2^{n-1} - n) + n!, & \text{если } n > 1. \end{cases}$$

Доказательство. Количество состояний системы (Γ_{K_n}, α) равно $2^{n(n-1)/2}$. В теореме 1 приведён критерий принадлежности состояния системы аттрактору (циклического состояния).

Пусть $n = 1$.

В системе существует единственное состояние $\overrightarrow{G} \in \Gamma_{K_1}$, которое является циклическим. Таким образом, в конечной динамической системе (Γ_{K_1}, α) количество принадлежащих аттракторам состояний равно 1.

Пусть $n > 1$.

Заметим, что множества состояний, подходящих под п. 1 и 2 теоремы 1, не пересекаются, так как компонента $n-1$ в векторе степеней захода указывает на наличие стока в соответствующем орграфе. Таким образом, общее число циклических состояний равно сумме количеств состояний, соответствующих п. 1 и 2 теоремы 1, подсчитаем их.

1) Найдём количество состояний системы, у которых нет стока.

Очевидно, что в ориентации полного графа может быть не более одного стока.

Пусть дано состояние $\overrightarrow{G} \in \Gamma_{K_n}$, у которого есть сток. Удалим у орграфа $\overrightarrow{G}$ сток и обозначим полученный новый орграф через $\overrightarrow{G}'$. Орграф $\overrightarrow{G}'$ имеет $n-1$ вершину и его симметризация G' также является полным графом. Количество всех возможных ориентаций полного графа G' с $n-1$ вершиной равно $2^{(n-1)(n-2)/2}$. Удалённый сток мог быть на месте любой из n вершин орграфа $\overrightarrow{G}$. Таким образом, количество состояний $\overrightarrow{G} \in \Gamma_{K_n}$, у которых есть сток, равно $n \cdot 2^{(n-1)(n-2)/2}$. Получаем, что количество состояний $\overrightarrow{G} \in \Gamma_{K_n}$, у которых нет стока, равно

$$2^{n(n-1)/2} - n \cdot 2^{(n-1)(n-2)/2} = 2^{(n-1)(n-2)/2} \left(2^{n-1} - n\right).$$

2) Найдём количество состояний системы, которые имеют вектор степеней захода $(n-1, n-2, \ldots, 0)$. Докажем, что оно равно числу перестановок n-элементного множества $\{n-1, n-2, \ldots, 0\}$, то есть $n!$.

Предположим, что это не так, а именно: есть перестановка n-элементного множества $\{n-1, n-2, \ldots, 0\}$, не соответствующая ни одному из состояний системы. Попробуем последовательно построить соответствующую ориентацию графа G с n вершинами. При построении будем нумеровать вершины согласно их степени захода.

Начинаем с вершины v_0', степень захода которой равна 0: $d^-(v_0') = 0$, то есть она является источником, все рёбра ориентируем из неё.

Находим вершину v_1', степень захода которой равна 1: $d^-(v_1') = 1$, то есть она достижима только из одной вершины, а именно из вершины v_0', все остальные рёбра ориентируем из неё.

Переходим к вершине v_2', степень захода которой равна 2: $d^-(v_2') = 2$, то есть она достижима только из двух вершин, а именно из v_0' и v_1', все остальные рёбра ориентируем из неё.

Продолжая аналогично, доходим до вершины v_{n-1}', у которой степень захода равна $n-1$: $d^-(v_{n-1}') = n-1$, то есть она является стоком, и на данном шаге все рёбра уже ориентированы в данную вершину.

Таким образом, получили ориентацию полного графа G, причём единственную, у которой вектор, компонентами которого являются расположенные в заданном порядке степени захода вершин, совпадает с данной перестановкой — противоречие.

Таким образом, в конечной динамической системе (Γ_{K_n}, α), $n > 1$, количество принадлежащих аттракторам (циклических) состояний равно $2^{(n-1)(n-2)/2} \left(2^{n-1} - n\right) + n!$.

Теорема 2 доказана. ■

Например, в конечной динамической системе (Γ_{K_3}, α) все восемь состояний являются циклическими (см. рис. 1), при этом по теореме 2 имеем $2^1 \left(2^2 - 3\right) + 3! = 8$.

В табл. 1 приведены данные по количеству принадлежащих аттракторам состояний в конечных динамических системах (Γ_{K_n}, α) для $1 \leqslant n \leqslant 20$. Можно заметить, что абсолютное большинство составляют циклические состояния.

Таблица 1

Количество циклических состояний в (Γ_{K_n}, α)

n	$\lvert\Gamma_{K_n}\rvert$	Количество циклических состояний	%
1	2^0	1	100
2	2^1	2	100
3	2^3	8	100
4	2^6	56	87,5
5	2^{10}	824	≈ 80
6	2^{15}	27344	≈ 83
7	2^{21}	1872816	≈ 89
8	2^{28}	251698560	≈ 94
9	2^{36}	66303920512	≈ 96
10	2^{45}	34497180950272	≈ 98
11	2^{55}	35641768965903616	$\approx 98{,}9$
12	2^{66}	73354630731089640448	$\approx 99{,}4$
13	2^{78}	301272224211830624013312	$\approx 99{,}7$
14	2^{91}	2471648838202109434865068032	$\approx 99{,}8$
15	2^{105}	40527681006124779440955203213312	$\approx 99{,}9$
16	2^{120}	1328578958677599019450261671029080064	$\approx 99{,}95$
17	2^{136}	870896890558319030767845351381953243709 44	$\approx 99{,}97$
18	2^{153}	1141641352050090736402664852541131787684931 1744	$\approx 99{,}986$
19	2^{171}	2992938411604397870579225677935591422639720079360000	$\approx 99{,}993$
20	2^{190}	1569215570739605117175417732871168545075536656127224971264	$\approx 99{,}996$

Следствие 1. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, количество не принадлежащих аттракторам (не являющихся циклическими) состояний равно

$$\begin{cases} 0, & \text{если } n = 1, \\ n \cdot 2^{(n-1)(n-2)/2} - n!, & \text{если } n > 1. \end{cases}$$

4. Количество аттракторов в конечной динамической системе (Γ_{K_n}, α)

Теорема 3 [20]. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, существуют следующие аттракторы:

1) длины 1, каждый из которых образован состоянием $\overrightarrow{G} \in \Gamma_{K_n}$, у которого нет стока;
2) длины n, каждый из которых состоит из состояний $\overrightarrow{G} \in \Gamma_{K_n}$, у которых вектор степеней захода есть $(n-1, n-2, \ldots, 0)$, при этом аттрактор представляет собой контур, в котором каждое следующее состояние получается из предыдущего таким образом: если $(d^-(v_1), d^-(v_2), \ldots, d^-(v_n))$ — вектор, составленный из степеней захода вершин в порядке их нумерации для $\overrightarrow{G}$, то для $\alpha(\overrightarrow{G}) \in \Gamma_{K_n}$ соответствующий вектор равен $(d^-(v_1)+1, d^-(v_2)+1, \ldots, d^-(v_n)+1)$, где сложение осуществляется по модулю n,

и только они.

Теорема 4. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, количество аттракторов длины 1 равно

$$\begin{cases} 1, & \text{если } n = 1, \\ 2^{(n-1)(n-2)/2}\left(2^{n-1} - n\right), & \text{если } n > 1. \end{cases} \tag{1}$$

Доказательство. Согласно теореме 3, количество аттракторов длины 1 совпадает с количеством состояний $\overrightarrow{G} \in \Gamma_{K_n}$, у которых нет стока, подсчитано в доказательстве теоремы 2 и совпадает с (1). ■

Например, в конечной динамической системе (Γ_{K_3}, α) количество аттракторов длины 1 равно 2 (см. рис. 1), при этом по теореме 4 имеем $2^{(3-1)(3-2)/2}\left(2^{3-1} - 3\right) = 2$.

В табл. 2 приведены данные по количеству аттракторов длины 1 в конечных динамических системах (Γ_{K_n}, α) для $1 \leqslant n \leqslant 20$. Можно заметить, что с ростом n аттракторы длины 1 начинают составлять абсолютное большинство по сравнению с аттракторами длины n.

Таблица 2

Количество аттракторов длины 1 **в** (Γ_{K_n}, α)

n	Количество аттракторов длины 1	%
1	1	100
2	0	0
3	2	50
4	32	≈ 84
5	704	≈ 97
6	26624	$\approx 99{,}6$
7	1867776	$\approx 99{,}96$
8	251658240	$\approx 99{,}998$
9	66303557632	≈ 100
10	34497177321472	≈ 100
11	35641768925986816	≈ 100
12	73354630730610638848	≈ 100
13	30127222421182439699 2512	≈ 100
14	24716488382021093476867768 32	≈ 100
15	40527681006124779439647528845312	≈ 100
16	13285789586775990194502407482391 92064	≈ 100
17	87089689055831903076784534782507896274944	≈ 100
18	11416413520500907364026648525404915503143583744	≈ 100
19	29929384116043978705792256779355913009946196705280 00	≈ 100
20	15692155707396051171754177328711685450731037541190483312 64	≈ 100

Теорема 5. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, количество аттракторов длины n равно $(n-1)!$

Доказательство. При $n = 1$ имеем $1 = 0!$ аттрактор длины 1 (по теореме 4).

Пусть $n > 1$. В теореме 3 описаны аттракторы длины n в системе; согласно доказательству теоремы 2, п. 2, количество соответствующих циклических состояний равно $n!$. Таким образом, в системе (Γ_{K_n}, α), $n > 1$, количество аттракторов длины n равно $n!/n = (n-1)!$. ■

Например, в конечной динамической системе (Γ_{K_3}, α) количество аттракторов длины 3 равно 2 (см. рис. 1), при этом по теореме 5 имеем $(3-1)! = 2$.

В табл. 3 приведены данные по количеству аттракторов длины n в конечных динамических системах (Γ_{K_n}, α) для $1 \leqslant n \leqslant 20$.

Таблица 3
Количество аттракторов длины n в (Γ_{K_n}, α)

n	Количество аттракторов длины n	%
1	1	100
2	1	100
3	2	50
4	6	≈ 16
5	24	≈ 3
6	120	$\approx 0{,}4$
7	720	$\approx 0{,}04$
8	5040	$\approx 0{,}002$
9	40320	$\approx 6 \cdot 10^{-5}$
10	362880	$\approx 1 \cdot 10^{-6}$
11	3628800	$\approx 1 \cdot 10^{-8}$
12	39916800	$\approx 5 \cdot 10^{-11}$
13	479001600	$\approx 2 \cdot 10^{-13}$
14	6227020800	$\approx 3 \cdot 10^{-16}$
15	87178291200	$\approx 2 \cdot 10^{-19}$
16	1307674368000	$\approx 1 \cdot 10^{-22}$
17	20922789888000	$\approx 2 \cdot 10^{-26}$
18	355687428096000	$\approx 3 \cdot 10^{-30}$
19	6402373705728000	$\approx 2 \cdot 10^{-34}$
20	121645100408832000	$\approx 8 \cdot 10^{-39}$

Теорема 6. В конечной динамической системе (Γ_{K_n}, α), $n \geqslant 1$, количество аттракторов (бассейнов) равно

$$\begin{cases} 1, & \text{если } n = 1, \\ 2^{(n-1)(n-2)/2}\,(2^{n-1} - n) + (n-1)!, & \text{если } n > 1. \end{cases}$$

Доказательство. При $n = 1$, очевидно, количество аттракторов равно 1.

Пусть $n > 1$. Согласно доказательству теоремы 2 и теореме 3, общее число аттракторов в системе (Γ_{K_n}, α) равно сумме количеств аттракторов длины 1 и n, которые подсчитаны в теоремах 4 и 5. ■

Например, в конечной динамической системе (Γ_{K_3}, α) четыре аттрактора (см. рис. 1), при этом по теореме 6 имеем $2^{(3-1)(3-2)/2}\,(2^{3-1} - 3) + (3-1)! = 4$.

В табл. 4 приведены данные по количеству аттракторов в конечных динамических системах (Γ_{K_n}, α) для $1 \leqslant n \leqslant 20$.

Например, карта системы (Γ_{K_7}, α), $|\Gamma_{K_7}| = 2097152$, состоит из 1868496 бассейнов, при этом 224336 состояний не являются циклическими (что составляет $\approx 11\,\%$ от общего числа состояний), 1872816 состояний являются циклическими, которые образуют 1867776 аттракторов длины 1 (что составляет $\approx 99{,}96\,\%$ от общего числа аттракторов) и 720 аттракторов длины 7.

Таблица 4

Количество аттракторов в (Γ_{K_n}, α)

n	Количество аттракторов (бассейнов)
1	1
2	1
3	4
4	38
5	728
6	26744
7	1868496
8	251663280
9	66303597952
10	34497177684352
11	35641768929615616
12	73354630730650555648
13	301272224211824875994112
14	2471648838202109353913797632
15	40527681006124779439734707136512
16	1328578958677599019450242055913560064
17	870896890558319030767845348034306861629 44
18	114164135205009073640266485254052711905716797 44
19	2992938411604397870579225677935591307396993376256000
20	1569215570739605117175417732871168545073225399219457163264

Заключение

В работе получены формулы для подсчёта количества циклических (принадлежащих аттракторам) и не являющихся циклическими состояний конечной динамической системы (Γ_{K_n}, α), $n \geqslant 1$, всех возможных ориентаций полного графа K_n; получены формулы для подсчёта количества аттракторов системы, в том числе различных типов, что является полезным для задач, связанных с информационной безопасностью, например для построения отказоустойчивых графовых систем с непрерывным функционированием и восстановлением.

ЛИТЕРАТУРА

1. *Григорчук Р. И., Некрашевич В. В., Сущанский В. И.* Автоматы, динамические системы и группы // Труды МИАН. 2000. Т. 231. С. 134–214.
2. *Barbosa V. C.* An Atlas of Edge-Reversal Dynamics. London: Chapman&Hall/CRC, 2001. 372 p.
3. *Салий В. Н.* Об одном классе конечных динамических систем // Вестник Томского госуниверситета. Приложение. 2005. № 14. С. 23–26.
4. *Osipenko G.* Dynamical Systems, Graphs, and Algorithms. Berlin; Heidelberg: Springer Verlag, 2007. 300 p.
5. *Щербина О. А.* Методологические аспекты динамического программирования // Динамические системы. 2007. Вып. 22. С. 21–36.
6. *Zavlanos M. M. and Pappas G. J.* A dynamical systems approach to weighted graph matching // Automatica. 2008. V. 44. No. 11. P. 2817–2824.
7. *Macauley M. and Mortveit H. S.* Cycle equivalence of graph dynamical systems // Nonlinearity. 2009. V. 22. No. 2. P. 421–436.
8. *Kuhlman C. J., Kumar V. S. A., Marathe M. V., et al.* A general-purpose graph dynamical system modeling framework // Proc. 2011 Winter Simulation Conf. Phoenix, USA, 2011. P. 296–308.

9. *Ara P. and Exel R.* Dynamical systems associated to separated graphs, graph algebras, and paradoxical decompositions // Adv. Math. 2014. V. 252. P. 748–804.
10. *Abdelhamid S. H. E., Kuhlman C. J., Marathe M. V., et al.* GDSCalc: a web-based application for evaluating discrete graph dynamical systems // PLoS ONE. 2015. No. 10 (8). 24 p. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0133660.
11. *Aledo J. A., Martinez S., and Valverde J. C.* Graph dynamical systems with general Boolean states // Appl. Math. & Inform. Sci. 2015. V. 9. No. 4. P. 1803–1808.
12. *Кадыров А. А., Кадыров А. А.* Концептуальные основы общей теории дискретных динамических, релейных и логико-динамических систем на базе физической декомпозиции и графовых моделей // Вестник Волгогр. гос. ун-та. Сер. 10. Иннов. деят. 2015. № 2 (17). С. 80–89.
13. *Волгина М. А.* Формализация информационных потоков графовых моделей динамических систем // Альманах современной науки и образования. 2015. № 3 (93). С. 23–26.
14. *Aledo J. A., Diaz L. G., Martinez S., and Valverde J. C.* Coexistence of periods in parallel and sequential boolean graph dynamical systems over directed graphs // Math. 2020. No. 8 (10). P. 1812–1825.
15. *Gadouleau M.* On the influence of the interaction graph on a finite dynamical system // Natural Computing. 2020. No. 19. P. 15–28.
16. *Жаркова А. В.* О количестве циклических состояний в конечных динамических системах ориентаций полных графов // Компьютерные науки и информационные технологии: Материалы Междунар. науч. конф. Саратов, 2018. С. 149–151.
17. *Жаркова А. В.* О количестве аттракторов в конечных динамических системах ориентаций полных графов // Прикладная дискретная математика. Приложение. 2018. № 11. С. 106–109.
18. *Богомолов А. М., Салий В. Н.* Алгебраические основы теории дискретных систем. М.: Наука, Физматлит, 1997. 368 с.
19. *Власова А. В.* Аттракторы в динамических системах двоичных векторов. Деп. в ВИНИТИ 23.06.2010. № 392–В2010. 19 с.
20. *Жаркова А. В.* Аттракторы и циклические состояния в конечных динамических системах ориентаций полных графов // Прикладная дискретная математика. 2023. № 59. С. 80–87.

REFERENCES

1. *Grigorchuk R. I., Nekrashevych V. V., Sushchansky V. I.* Avtomaty, dinamicheskie sistemy i gruppy [Automata, Dynamic Systems and Groups]. Trudy MIAN, 2000, vol. 231, pp. 134–214. (in Russian)
2. *Barbosa V. C.* An Atlas of Edge-Reversal Dynamics. London, Chapman&Hall/CRC, 2001. 372 p.
3. *Salii V. N.* Ob odnom klasse konechnykh dinamicheskikh sistem [On a class of finite dynamic systems]. Vestnik Tomskogo Gosuniversiteta. Prilozhenie, 2005, no. 14, pp. 23–26 (in Russian).
4. *Osipenko G.* Dynamical Systems, Graphs, and Algorithms. Berlin, Heidelberg, Springer Verlag, 2007. 300 p.
5. *Shcherbina O. A.* Metodologicheskie aspekty dinamicheskogo programmirovaniya [Methodological aspects of dynamic programming]. Dinamicheskie Sistemy, 2007, no. 22, pp. 21–36. (in Russian)
6. *Zavlanos M. M. and Pappas G. J.* A dynamical systems approach to weighted graph matching. Automatica, 2008, vol. 44, no. 11, pp. 2817–2824.
7. *Macauley M. and Mortveit H. S.* Cycle equivalence of graph dynamical systems. Nonlinearity, 2009, vol. 22, no. 2, pp. 421–436.

8. *Kuhlman C. J., Kumar V. S. A., Marathe M. V., et al.* A general-purpose graph dynamical system modeling framework. Proc. 2011 Winter Simulation Conf., Phoenix, USA, 2011, pp. 296–308.
9. *Ara P. and Exel R.* Dynamical systems associated to separated graphs, graph algebras, and paradoxical decompositions. Adv. Math., 2014, vol. 252, pp. 748–804.
10. *Abdelhamid S. H. E., Kuhlman C. J., Marathe M. V., et al.* GDSCalc: a web-based application for evaluating discrete graph dynamical systems. PLoS ONE, 2015, no. 10 (8), 24 p. `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0133660`.
11. *Aledo J. A., Martinez S., and Valverde J. C.* Graph dynamical systems with general boolean states. Appl. Math. & Inform. Scie., 2015, vol. 9, no. 4, pp. 1803–1808.
12. *Kadyrov A. A. and Kadyrov A. A.* Kontseptual'nye osnovy obshchey teorii diskretnykh dinamicheskikh, releynykh i logiko-dinamicheskikh sistem na baze fizicheskoy dekompozitsii i grafovykh modeley [Conceptual foundations of general theory of discrete dynamic, relay and logical-dynamic systems based on physical decomposition and graph models]. Vestnik VolSU, Ser. 10, Innov. Deyat., 2015, no. 2 (17), pp. 80–89. (in Russian)
13. *Volgina M. A.* Formalizatsiya informatsionnykh potokov grafovykh modeley dinamicheskikh sistem [Formalization of information flows of graph models of dynamical systems]. Al'manakh Sovremennoy Nauki i Obrazovaniya, 2015, no. 3 (93), pp. 23–26 (in Russian).
14. *Aledo J. A., Diaz L. G., Martinez S., and Valverde J. C.* Coexistence of periods in parallel and sequential Boolean graph dynamical systems over directed graphs. Math., 2020, no. 8 (10), pp. 1812–1825.
15. *Gadouleau M.* On the influence of the interaction graph on a finite dynamical system. Natural Computing, 2020, no. 19, pp. 15–28.
16. *Zharkova A. V.* O kolichestve tsiklicheskikh sostoyaniy v konechnykh dinamicheskikh sistemakh orientatsiy polnykh grafov [On the number of cyclic states in finite dynamic systems of complete graphs orientations]. Komp'iuternye Nauki i Informatsionnye Tekhnologii, Saratov, 2018, pp. 149–151. (in Russian)
17. *Zharkova A. V.* O kolichestve attraktorov v konechnykh dinamicheskikh sistemakh orientatsiy polnykh grafov [On the number of attractors in finite dynamic systems of complete graphs orientations]. Prikladnaya Diskretnaya Matematika. Prilozhenie, 2018, no. 11, pp. 106–109. (in Russian)
18. *Bogomolov A. M. and Salii V. N.* Algebraicheskie osnovy teorii diskretnykh sistem [Algebraic Foundations of the Theory of Discrete Systems]. Moscow, Nauka Publ., 1997. 368 p. (in Russian)
19. *Vlasova A. V.* Attraktory v dinamicheskikh sistemakh dvoichnykh vektorov [Attractors in Dynamic Systems of Binary Vectors]. Dep. in VINITI 23.06.2010, no. 392-B2010, 19 p. (in Russian)
20. *Zharkova A. V.* Attraktory i tsiklicheskie sostoyaniya v konechnykh dinamicheskikh sistemakh orientatsiy polnykh grafov [Attractors and cyclic states in finite dynamic systems of complete graphs orientations]. Prikladnaya Diskretnaya Matematika, 2023, no. 59, pp. 80–87. (in Russian)

УДК 519.178 DOI 10.17223/20710410/63/6

КОНТЕКСТНЫЙ АНАЛИЗ СВЯЗНОСТИ ДВУХПОЛЮСНЫХ СТРУКТУР[1]

А. С. Лосев

ИПМ ДВО РАН, г. Владивосток, Россия

E-mail: A.S.Losev@yandex.ru

Разрабатывается способ повышения вероятности связности двухполюсника, состоящего из низконадёжных рёбер. Методами контекстного анализа выделяется множество доминант, к которому относятся рёбра с наибольшим влиянием на связность всего двухполюсника. Разработаны два метода воздействия на множество доминант, приводящие к желаемому эффекту. В результате их сравнительного анализа получены соответствующие условия, позволяющие выбрать подходящий метод в зависимости от структуры двухполюсника.

Ключевые слова: *связность, двухполюсник, множество доминант, сетевые структуры.*

CONTEXTUAL ANALYSIS OF THE BIPOLAR STRUCTURES CONNECTIVITY

A. S. Losev

IAM FEB RAS, Vladivostok, Russia

The paper discusses an original approach to the analysis of network structures of various natures, which are represented in the form of graphs. It is assumed that the probability of connectivity of individual edges functionally depends on their physical characteristics (edge length) and tends to zero. The issue of increasing the connectivity probability of the entire graph is being addressed, which is defined as the probability of the existence of a sequential set of edges connecting the selected start and end vertices. A distinctive feature of the proposed method is the move away from the traditional reservation of obviously weak connection points towards a functional impact on individual connections of the structure, changing their physical characteristics. Based on the theory of systems functioning and the theory of dominants, contextual approach is being developed aimed at identifying the dominant connections of the graph, reflecting the functional and meaningful meaning of the connections between the vertices of the original modeled object. As a result, a set of dominants $\mathcal{S}$ is identified, consisting of edges whose parameters meet the given criteria. The selection criterion is the length of the edge included in the asymptotic relation, which characterizes the connectivity probability of the entire graph. It has been proven that changing the length of edges from a given set by $\varepsilon > 0$ increases the probability of connectedness of the original graph in the maximum case in $h^{-\varepsilon}$ or $h^{-\varepsilon\beta}$ times, where $h \to 0$ and β is a parameter depending on the number of graph paths passing through this edge. Various methods have been proposed to increase the connectivity probability of the graph under consideration: from the point of view of the point effect, by reducing the length of a

[1]Работа выполнена в рамках госзадания ИПМ ДВО РАН № 075-01290-23-00.

single edge from the set $\mathcal{S}$; from the position of influencing the maximum set of edges from $\mathcal{S}$, allowing to obtain the overall maximum effect. A comparative analysis of the proposed ways to increase the probability of graph connectivity has been carried out, and the appropriate conditions have been identified to achieve the maximum effect from the selected methods of influence.

Keywords: *connectivity, bipolar network, set of dominants, network structure.*

Введение

Объёмы передаваемых данных в различных сферах растут с невероятной скоростью и требуют новых математических подходов и методов по их обработке. Существующие мощности обработки не всегда могут охватить весь объём информации и требуют всё большего технического ресурса. В задаче обработки большого количества данных на помощь приходят современные разработки в информационных системах, связанные с распараллеливанием, самообучением и участием искусственного интеллекта [1–4]. Например, в работе [1] на основе методов распараллеливания и самообучения нейросети предлагаются решения нестандартных или нечётко сформулированных задач. В работах [5, 6] исследуются интеллектуальные системы слежения и контроля в непрерывном потоке информации с элементами распознавания объектов и субъектов.

Первоначально задача алгоритмической обработки данных, представленных в виде сетевых структур или графов, связана с исследованиями У. Мак-Каллока в области биологических процессов головного мозга и У. Питтса, работавшего над созданием искусственного интеллекта на основе нейронных сетей [7]. Она нашла своё продолжение во многих исследованиях, в том числе у С. Гроссберга, Т. Кохонена, Д. Хопфилда [8, 9]. Отдельное место данная задача занимает в исследованиях нейропроцессов, моделируемых в нейросетях.

Среди возможных подходов, направленных на решение задачи алгоритмической обработки данных, представленных в виде различных систем, особый интерес представляет теория функционирования систем П. К. Анохина [10]. Он предлагает отбросить догмат о саморегулировании систем и оценивать функционирование сложных систем как совокупность конкретных и частных вопросов, которые решаются с меньшими затруднениями и затратами ресурса. Действительно, с одной стороны, метод декомпозиции существенно уменьшает размерность модели и входных параметров исследуемого объекта. Но, с другой стороны, обладает существенным недостатком: отражая характеристики отдельных частей системы, плохо характеризует объект в целом, упуская из виду возможные синергетические эффекты, возникающие при композиции обработанных частей системы в единое целое. Однако в сочетании с теорией доминант А. А. Ухтомского [11] он позволяет оценить объект в целом. В своих исследованиях А. А. Ухтомский [12] показал, что в любой системе присутствует множество доминант, которое формирует фокус рассмотрения системы в целом, отражая её основные характеристики вплоть до того, что отсутствие доминанты в системе влечёт ликвидацию системы полностью. Таким образом, наблюдается взаимно однозначное соответствие доминанты системы и контекста рассмотрения системы, где изменение одного влечёт изменение другого.

В данной работе методами контекстного анализа исследуется влияние отдельных рёбер двухполюсника на вероятность его связности. Выделяется множество доминант, состоящее из рёбер, оказывающих наибольшее влияние на вероятность связности двухполюсника по сравнению с прочими элементами. Рассматриваются различные спосо-

бы повышения вероятности связности двухполюсника через изменение длины рёбер из множества доминант. Проводится сравнительный анализ предложенных подходов.

1. Основные обозначения и постановка задачи

Введём в рассмотрение двухполюсник $\Gamma = \{U, W\}$ в виде неориентированного графа с конечным множеством вершин U, множеством рёбер W и выделенными начальной u_* и конечной v_* вершинами. Обозначим через $\mathcal{R}$ множество всех ациклических путей R двухполюсника Γ из вершины u_* в v_*. Положим, что каждое ребро $w \in W$ характеризуется длиной $d(w)$ и работает независимо с вероятностью p_w, $0 < p_w < 1$. Обозначим P_Γ вероятность связности двухполюсника, которая характеризуется наличием хотя бы одного работающего пути между начальной и конечной вершинами. Длину пути R определим как $D(R) = \sum\limits_{w \in R} d(w)$, а длину минимального пути — $D(\mathcal{R}) = \min\limits_{R \in \mathcal{R}} D(R)$.

В работах [13, 14] получены асимптотические соотношения, характеризующие вероятность связности двухполюсника, состоящего из низконадёжных рёбер — таких, что $p_w = p_w(h)$, где $p_w(h)$ — функция от некоторого параметра h, характеризующая вероятность работы ребра длиной $d(w)$. Здесь и далее $h > 0$.

Теорема 1. Пусть $p_w(h) \sim h^{d(w)}$ при $h \to 0$, где $d(w) > 0$, $w \in W$, тогда

$$\mathsf{P}_\Gamma \sim \mathcal{N}(\mathcal{R}) h^{D(\mathcal{R})}, \tag{1}$$

где $\mathcal{N}(\mathcal{R})$ — количество путей минимальной длины в двухполюснике Γ.

Соотношение (1) позволяет построить асимптотическую оценку вероятности связности двухполюсника через характеристики отдельных рёбер. Обозначим множество таких рёбер через $\mathcal{S} = \{w \in R : D(R) = D(\mathcal{R})\}$. С позиции контекстного анализа, множество $\mathcal{S}$ является множеством доминант и представляет отдельный интерес, так как его элементы считаются основополагающими в вопросах функционирования двухполюсника в целом и его связности в частности.

В данном представлении вероятность связности отдельно взятого ребра двухполюсника выражается через функциональную зависимость от его длины. Соответственно изменение длины любого ребра из множества доминант приводит к значимому изменению связности двухполюсника в целом. При этом нужно отметить, что в силу определения множества доминант изменение длины отдельно взятого ребра из $\mathcal{S}$ может привести к аналогичному эффекту, сравнимому с изменением некоторого набора рёбер.

Рассмотрим различные способы повышения вероятности связности всего двухполюсника через изменение длины отдельно взятого ребра из множества доминант и соответствующего набора. Проведём сравнительный анализ этих подходов с целью выявления условий, позволяющих получить наилучший эффект в зависимости от выбранного подхода и особенности строения двухполюсника.

2. Повышение связности структуры через контекстное воздействие

2.1. Точечный подход

Выделим из множества $\mathcal{R}$ подмножество всех кратчайших путей $\widetilde{\mathcal{R}} = \{R \in \mathcal{R} : D(R) = D(\mathcal{R})\}$. Обозначим $N(A)$ число элементов множества A.

Утверждение 1. Если заменить $d(w')$ на $d(w') - \varepsilon$ для одного любого $w' \in \mathcal{S}$, то

$$\mathsf{P}_\Gamma \sim N(\widetilde{\mathcal{R}}_1) h^{D(\mathcal{R})-\varepsilon}, \quad h \to 0,$$

где $1 < \varepsilon < d(w')$, $0 < N(\widetilde{\mathcal{R}}_1) \leqslant \mathcal{N}(\mathcal{R})$.

Доказательство. В силу теоремы 1 имеем

$$\mathsf{P}_\Gamma \sim \mathcal{N}(\mathcal{R})h^{D(\mathcal{R})} = \sum_{R\in\widetilde{\mathcal{R}}}\prod_{w\in R} h^{d(w)}.$$

Представим $\widetilde{\mathcal{R}} = \widetilde{\mathcal{R}}_0 \cup \widetilde{\mathcal{R}}_1$, где $\widetilde{\mathcal{R}}_0 = \{R \in \widetilde{\mathcal{R}} : w' \notin R\}$, $\widetilde{\mathcal{R}}_1 = \{R \in \widetilde{\mathcal{R}} : w' \in R\}$, тогда

$$\sum_{R\in\widetilde{\mathcal{R}}}\prod_{w\in R} h^{d(w)} = \sum_{R\in\widetilde{\mathcal{R}}_0}\prod_{w\in R} h^{d(w)} + \sum_{R\in\widetilde{\mathcal{R}}_1}\prod_{w\in R} h^{d(w)}.$$

Заменим $d(w')$ на $d(w') - \varepsilon$ и получим

$$P_\Gamma \sim \sum_{R\in\widetilde{\mathcal{R}}_0}\prod_{w\in R} h^{d(w)} + \sum_{R\in\widetilde{\mathcal{R}}_1}\left(h^{d(w')-\varepsilon}\prod_{w\in R\setminus w'} h^{d(w)}\right) = \sum_{R\in\widetilde{\mathcal{R}}_0}\prod_{w\in R} h^{d(w)} +$$
$$+ \sum_{R\in\widetilde{\mathcal{R}}_1}\left(h^{-\varepsilon}\prod_{w\in R} h^{d(w)}\right) = \sum_{R\in\widetilde{\mathcal{R}}_0} h^{D(\mathcal{R})} + \sum_{R\in\widetilde{\mathcal{R}}_1} h^{D(\mathcal{R})-\varepsilon} =$$
$$= \sum_{R\in\widetilde{\mathcal{R}}_1} h^{D(\mathcal{R})-\varepsilon}\left(1 + \frac{\sum\limits_{R\in\widetilde{\mathcal{R}}_0} h^{D(\mathcal{R})}}{\sum\limits_{R\in\widetilde{\mathcal{R}}_1} h^{D(\mathcal{R})-\varepsilon}}\right) = N(\widetilde{\mathcal{R}}_1)h^{D(\mathcal{R})-\varepsilon}\left(1 + \frac{N(\widetilde{\mathcal{R}}_0)}{N(\widetilde{\mathcal{R}}_1)}h^{\varepsilon}\right).$$

При условии $h \to 0$ и $1 < \varepsilon < d(w')$ получаем

$$\mathsf{P}_\Gamma \sim N(\widetilde{\mathcal{R}}_1)h^{D(\mathcal{R})-\varepsilon}(1 + o(1)) \sim N(\widetilde{\mathcal{R}}_1)h^{D(\mathcal{R})-\varepsilon}.$$

Утверждение 1 доказано. ■

2.2. Множественный подход

Выделим из множества $\widetilde{\mathcal{R}}$ подмножество $\widetilde{\mathcal{R}}' = \{R \in \widetilde{\mathcal{R}} : N(R) = \widetilde{N}(\widetilde{\mathcal{R}})\}$, где $\widetilde{N}(\widetilde{\mathcal{R}}) = \max\limits_{R\in\widetilde{\mathcal{R}}} N(R)$, а из множества доминант $\mathcal{S}$ — подмножество $\mathcal{S}' = \{w \in R : R \in \widetilde{\mathcal{R}}'\}$.

Утверждение 2. Если заменить $d(w)$ на $d(w) - \varepsilon$ для всех $w \in \mathcal{S}'$, то

$$\mathsf{P}_\Gamma \sim N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})}, \quad h \to 0,$$

где $1 < \varepsilon < \min\limits_{w\in\mathcal{S}'} d(w)$, $0 < N(\widetilde{\mathcal{R}}') \leqslant \mathcal{N}(\mathcal{R})$.

Доказательство. В силу теоремы 1 имеем

$$\mathsf{P}_\Gamma \sim \mathcal{N}(\mathcal{R})h^{D(\mathcal{R})} = \sum_{R\in\widetilde{\mathcal{R}}}\prod_{w\in R} h^{d(w)} = \sum_{R\in\widetilde{\mathcal{R}}\setminus(\widetilde{\mathcal{R}}'\cup\widetilde{\mathcal{R}}'_0)}\prod_{w\in R} h^{d(w)} + \sum_{R\in\widetilde{\mathcal{R}}'_0}\prod_{w\in R} h^{d(w)} + \sum_{R\in\widetilde{\mathcal{R}}'}\prod_{w\in R} h^{d(w)},$$

где $\widetilde{\mathcal{R}}'_0 = \{R \in \widetilde{\mathcal{R}} : N(R) \neq \widetilde{N}(\widetilde{\mathcal{R}}), R \cap \mathcal{S}' \neq \varnothing\}$. Заменим $d(w)$ на $d(w) - \varepsilon$ для всех $w \in \mathcal{S}'$, тогда

$$\mathsf{P}_\Gamma \sim \sum_{R\in\widetilde{\mathcal{R}}\setminus(\widetilde{\mathcal{R}}'\cup\widetilde{\mathcal{R}}'_0)}\prod_{w\in R} h^{d(w)} + \sum_{R\in\widetilde{\mathcal{R}}'_0}\left(\prod_{w\in R\setminus\mathcal{S}'} h^{d(w)}\prod_{w\in R\cap\mathcal{S}'} h^{d(w)-\varepsilon}\right) + \sum_{R\in\widetilde{\mathcal{R}}'}\prod_{w\in R} h^{d(w)-\varepsilon} =$$
$$= \sum_{R\in\widetilde{\mathcal{R}}\setminus(\widetilde{\mathcal{R}}'\cup\widetilde{\mathcal{R}}'_0)} h^{D(\mathcal{R})} + \sum_{R\in\widetilde{\mathcal{R}}'_0} h^{D(\mathcal{R})-\varepsilon N(R\cap\mathcal{S}')} + \sum_{R\in\widetilde{\mathcal{R}}'} h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})} =$$

$$= \sum_{R\in\widetilde{\mathcal{R}}'} h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})} \left(\frac{\sum\limits_{R\in\widetilde{\mathcal{R}}\setminus(\widetilde{\mathcal{R}}'\cup\widetilde{\mathcal{R}}'_0)} h^{D(\mathcal{R})}}{\sum\limits_{R\in\widetilde{\mathcal{R}}'} h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})}} + \frac{\sum\limits_{R\in\widetilde{\mathcal{R}}'_0} h^{D(\mathcal{R})-\varepsilon N(R\cap\mathcal{S}')}}{\sum\limits_{R\in\widetilde{\mathcal{R}}'} h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})}} + 1 \right) =$$

$$= N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})} \left(1 + k_0 h^{\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})} + \frac{1}{N(\widetilde{\mathcal{R}}')} \sum_{R\in\widetilde{\mathcal{R}}'_0} h^{\varepsilon(\widetilde{N}(\widetilde{\mathcal{R}})-N(R\cap\mathcal{S}'))} \right),\ k_0 = \text{const}\,.$$

Отдельно рассмотрим разность $\widetilde{N}(\widetilde{\mathcal{R}}) - N(R\cap\mathcal{S}')$. Из определения $\widetilde{N}(\widetilde{\mathcal{R}})$ следует, что $\widetilde{N}(\widetilde{\mathcal{R}}) - N(R\cap\mathcal{S}') > 0$. Действительно, если существует $R^*\in\widetilde{\mathcal{R}}'_0$, для которого $\widetilde{N}(\widetilde{\mathcal{R}}) - N(R^*\cap\mathcal{S}') = 0$, то $R^*\in\widetilde{\mathcal{R}}'$ и, следовательно, $R^*\notin\widetilde{\mathcal{R}}'_0$. С другой стороны, если существует $R^*\in\widetilde{\mathcal{R}}'_0$, для которого $\widetilde{N}(\widetilde{\mathcal{R}}) - N(R^*\cap\mathcal{S}') < 0$, то $N(R^*\cap\mathcal{S}') > \widetilde{N}(\widetilde{\mathcal{R}})$, что противоречит определению $\widetilde{N}(\widetilde{\mathcal{R}}) = \max\limits_{R\in\widetilde{\mathcal{R}}} N(R)$. Значит,

$$\mathsf{P}_\Gamma \sim N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})} \left(1 + k_0 h^{\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})} + \frac{1}{N(\widetilde{\mathcal{R}}')} \sum_{R\in\widetilde{\mathcal{R}}'_0} h^{\varepsilon k_1(R)} \right),\ k_0, k_1(R) > 0.$$

При условии $1 < \varepsilon < \min\limits_{w\in\mathcal{S}'} d(w)$ и $h\to 0$ получаем

$$\mathsf{P}_\Gamma \sim N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})}(1+o(1)) \sim N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon\widetilde{N}(\widetilde{\mathcal{R}})}.$$

Утверждение 2 доказано. ■

3. Сравнительный анализ подходов

Эффект, связанный с увеличением вероятности связности двухполюсника, зависит не только от выбранного подхода, но и от особенностей строения рассматриваемого графа. И хотя очевидно, что $N(\mathcal{S}) \geqslant N(\mathcal{S}')$ и один из подходов предполагает изменение длины одного ребра, а другой — некоторого количества рёбер, заранее неизвестно, какой из них приведет к наилучшему результату.

Обозначим P_{Γ_1} и P_{Γ_2} — вероятности связности фиксированного двухполюсника Γ после изменения длины фиксированного ребра $w'\in\mathcal{S}$ и множества рёбер $\mathcal{S}'$ соответственно. Тогда

$$\mathsf{P}_{\Gamma_1} \sim N(\widetilde{\mathcal{R}}_1)h^{D(\mathcal{R})-\varepsilon_1},\quad \mathsf{P}_{\Gamma_2} \sim N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})},\quad h\to 0,$$

где $1 < \varepsilon_1 < d(w')$, $1 < \varepsilon_2 < \min\limits_{w\in\mathcal{S}'} d(w)$. Сравним полученные асимптотические соотношения:

$$\frac{\mathsf{P}_{\Gamma_1}}{\mathsf{P}_{\Gamma_2}} \sim \frac{N(\widetilde{\mathcal{R}}_1)h^{D(\mathcal{R})-\varepsilon_1}}{N(\widetilde{\mathcal{R}}')h^{D(\mathcal{R})-\varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})}} = \frac{N(\widetilde{\mathcal{R}}_1)}{N(\widetilde{\mathcal{R}}')}h^{-\varepsilon_1+\varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})},\quad h\to 0.$$

Рассмотрим частный случай, когда $\varepsilon_1 = \varepsilon_2$. Тогда

$$\frac{\mathsf{P}_{\Gamma_1}}{\mathsf{P}_{\Gamma_2}} \sim \frac{N(\widetilde{\mathcal{R}}_1)}{N(\widetilde{\mathcal{R}}')}h^{\varepsilon_1(\widetilde{N}(\widetilde{\mathcal{R}})-1)},\quad h\to 0.$$

Если $\widetilde{N}(\widetilde{\mathcal{R}}) - 1 = 0$, то минимальный путь состоит из одного ребра и $N(\widetilde{\mathcal{R}}_1) = 1$. Если в графе нет кратных рёбер одинаковой длины, то $N(\widetilde{\mathcal{R}}_1) = N(\widetilde{\mathcal{R}}')$ и $\mathsf{P}_{\Gamma_1} \sim \mathsf{P}_{\Gamma_2}$, в противном случае $N(\widetilde{\mathcal{R}}_1) < N(\widetilde{\mathcal{R}}')$ и $\mathsf{P}_{\Gamma_1} \lesssim \mathsf{P}_{\Gamma_2}$.

Если $\widetilde{N}(\widetilde{\mathcal{R}}) - 1 > 0$, то $0 < h^{\varepsilon_1(\widetilde{N}(\widetilde{\mathcal{R}})-1)} < 1$, $h \to 0$. Следовательно, при условии $N(\widetilde{\mathcal{R}}_1) \leqslant N(\widetilde{\mathcal{R}}')$ получаем, что $\mathsf{P}_{\Gamma_1} \lesssim \mathsf{P}_{\Gamma_2}$. С другой стороны, при $N(\widetilde{\mathcal{R}}_1) > N(\widetilde{\mathcal{R}}')$ возможна ситуация, в которой как $\mathsf{P}_{\Gamma_1} \lesssim \mathsf{P}_{\Gamma_2}$, так и $\mathsf{P}_{\Gamma_1} \gtrsim \mathsf{P}_{\Gamma_2}$ в зависимости от структурных особенностей двухполюсника.

Аналогичным образом проведён сравнительный анализ полученных асимптотических соотношений в общем виде, его результаты представлены в таблице.

Сравнение точечного и множественного подходов

$\mathsf{P}_{\Gamma_1} \gtrsim \mathsf{P}_{\Gamma_2}$	$\mathsf{P}_{\Gamma_1} \lesssim \mathsf{P}_{\Gamma_2}$
$N(\widetilde{\mathcal{R}}_1) > N(\widetilde{\mathcal{R}}')$ и $\varepsilon_1 > \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$	$N(\widetilde{\mathcal{R}}_1) < N(\widetilde{\mathcal{R}}')$ и $\varepsilon_1 < \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$
$N(\widetilde{\mathcal{R}}_1) = N(\widetilde{\mathcal{R}}')$ и $\varepsilon_1 > \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$	$N(\widetilde{\mathcal{R}}_1) = N(\widetilde{\mathcal{R}}')$ и $\varepsilon_1 < \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$
$N(\widetilde{\mathcal{R}}_1) > N(\widetilde{\mathcal{R}}')$ и $\varepsilon_1 = \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$	$N(\widetilde{\mathcal{R}}_1) < N(\widetilde{\mathcal{R}}')$ и $\varepsilon_1 = \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$

Неопределённой остается ситуация, когда $N(\widetilde{\mathcal{R}}_1) < N(\widetilde{\mathcal{R}}')$ при $\varepsilon_1 > \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$ и $N(\widetilde{\mathcal{R}}_1) > N(\widetilde{\mathcal{R}}')$ при $\varepsilon_1 < \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$. Действительно, если положить $k = \varepsilon_1 - \varepsilon_2\widetilde{N}(\widetilde{\mathcal{R}})$ и $\Delta = N(\widetilde{\mathcal{R}}_1)/N(\widetilde{\mathcal{R}}')$, то в первом случае $\Delta < 1$ и $k > 0$, значит, $h^{-k} > 1$ при $h \to 0$. Во втором случае $\Delta > 1$ и $k < 0$, т. е. $0 < h^{-k} < 1$ при $h \to 0$. В обоих случаях $\Delta h^{-k} > 0$, но остаётся неопределённым значение полученного выражения относительно единицы, что не позволяет однозначно ответить на вопрос эффективности того или иного подхода. Полученная неопределённость при наличии обобщённого результата является отличительной особенностью контекстного подхода, направленного на выделение доминант. В данном случае только определение множества доминант в отдельно взятом двухполюснике позволяет однозначно разрешить неопределённость и выбрать наиболее эффективный метод повышения его связности.

Заключение

Полученные асимптотические соотношения, характеризующие вероятность связности двухполюсника, в отличие от традиционных способов повышения связности графов позволяют существенно влиять на связность всего соединения через изменение длины отдельных рёбер, входящих в множество доминант. В итоге результатом применения контекстного анализа является решение поставленной прикладной задачи, обладающей практической значимостью, а не поиск места модели в классификации имеющихся математических моделей.

В целом, применение контекстного подхода для оценки вероятности связности сетевой структуры произвольного вида приводит к более детализированному результату, который отражает особенности её строения. Последнее особенно важно, если предполагается дальнейшая работа со структурой, направленная на изменение её связности через воздействие на её отдельные части.

ЛИТЕРАТУРА

1. *Аксенов С. В.* Организация и использование нейронных сетей (методы и технологии). Томск: Изд-во НТЛ, 2006. 128 с.
2. *Богачев К. Ю.* Основы параллельного программирования. М.: БИНОМ. Лаборатория знаний, 2003.
3. *Воеводин В. В.* Суперкомпьютеры: вчера, сегодня, завтра // Наука и жизнь. 2000. № 5. С. 76–83.
4. *Воеводин В. В.* Параллельные вычисления. СПб.: БХВ-Петербург, 2002.
5. *Круг П. Г.* Нейронные сети и нейрокомпьютеры. М.: МЭИ, 2002. 176 с.
6. *Collinger J. L.* High-performance neuroprosthetic control by an individual with tetraplegia // The Lancet. 2012. V. 6736(12). P. 61816–61819.

7. *Culloch W. C. and Pitts W. H.* Logical calculus of ideas immanent in nervous activity // Bull. Math. Biophysics. 1943. V. 5. P. 115–119.
8. *Hopfield J. J.* Neural networks and physical systems with emergent collective computational abilities // Proc. National Academy of Sci. 1982. V. 79. P. 2554–2558.
9. *Hopfield J. J.* Neural computation of decision in optimization problems // Biol. Cybernet. 1985. V. 52. P. 141–152.
10. *Анохин П. К.* Избранные труды: Кибернетика функциональных систем. М.: Медицина, 1998. 400 с.
11. *Ухтомский А. А.* Доминанта. СПб.: Питер, 2002. 448 с.
12. *Ухтомский А. А.* Доминанта: физиология поведения. М.: АСТ, 2020. 117 с.
13. *Лосев А. С.* Асимптотический анализ надежности стохастических сетей // Информатика и системы управления. 2008. № 4(18). С. 101–105.
14. *Цицашвили Г. Ш., Лосев А. С., Осипова М. А.* Асимптотические формулы для вероятностей связности случайных графов // Автоматика и вычислительная техника. 2013. № 2. С. 22–28.

REFERENCES

1. *Aksenov S. V.* Organizatsiya i ispol'zovanie neyronnykh setey (metody i tekhnologii) [Organization and Use of Neural Networks (Methods and Technologies)]. Tomsk, NTL Publ., 2006. 128 p. (in Russian)
2. *Bogachev K. Yu.* Osnovy parallel'nogo programmirovaniya [Basics of Parallel Programming]. Moscow, BINOM Publ., 2003. (in Russian)
3. *Voevodin V. V.* Superkomp'yutery: vchera, segodnya, zavtra [Supercomputers: yesterday, today, tomorrow]. Nauka i Zhizn', 2000, no. 5, pp. 76–83. (in Russian)
4. *Voevodin V. V.* Parallel'nye vychisleniya [Parallel Computing]. Saint Petersburg, BKhV Publ., 2002. (in Russian)
5. *Krug P. G.* Neyronnye seti i neyrokomp'yutery [Neural Networks and Neurocomputers]. Moscow, MPEI Publ., 2002. 176 p. (in Russian)
6. *Collinger J. L.* High-performance neuroprosthetic control by an individual with tetraplegia. The Lancet, 2012, vol. 6736(12), pp. 61816–61819.
7. *Culloch W. C. and Pitts W. H.* Logical calculus of ideas immanent in nervous activity. Bull. Math. Biophysics, 1943, vol. 5, pp. 115–119.
8. *Hopfield J. J.* Neural networks and physical systems with emergent collective computational abilities. Proc. National Academy of Sciences, 1982, vol. 79, pp. 2554–2558.
9. *Hopfield J. J.* Neural computation of decision in optimization problems. Biol. Cybernet., 1985, vol. 52, pp. 141–152.
10. *Anokhin P. K.* Izbrannye trudy: Kibernetika funktsional'nykh sistem [Selected Works: Cybernetics of Functional Systems]. Moscow, Meditsina, 1998. 400 p. (in Russian)
11. *Ukhtomskiy A. A.* Dominanta [Dominant]. Saint Petersburg, Piter, 2002. 448 p. (in Russian)
12. *Ukhtomskiy A. A.* Dominanta: fiziologiya povedeniya [Dominant: Physiology of Behavior]. Moscow, AST Publ., 2020. 117 p. (in Russian)
13. *Losev A. S.* Asimptoticheskiy analiz nadezhnosti stokhasticheskikh setey [Asymptotic analysis of the reliability of stochastic networks]. Informatika i Sistemy Upravleniya, 2008, no. 4(18), pp. 101–105. (in Russian)
14. *Tsitsiashvili G. Sh., Losev A. S., and Osipova M. A.* Asimptoticheskie formuly dlya veroyatnostey svyaznosti sluchaynykh grafov [Asymptotic formulas for the connectedness probabilities of random graphs]. Avtomatika i Vychislitel'naya Tekhnika, 2013, no. 2, pp. 22–28. (in Russian)

МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

УДК 510.52 DOI 10.17223/20710410/63/7

ГЕНЕРИЧЕСКИ НЕРАЗРЕШИМЫЕ И ТРУДНОРАЗРЕШИМЫЕ ПРОБЛЕМЫ[1]

А. Н. Рыбалов

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

E-mail: alexander.rybalov@gmail.com

Памяти Виталия Анатольевича Романькова

В рамках генерического подхода изучается поведение алгоритмов на типичных (почти всех) входах, а остальные входы игнорируются. А. Мясниковым и автором ранее был предложен метод генерической амплификации для построения генерически неразрешимых алгоритмических проблем. Основной идеей этого метода является объединение эквивалентных входов в достаточно большие множества. Эквивалентность входов означает, что рассматриваемая проблема на них решается одинаково. Предлагается обобщение этого метода, строится пример разрешимой в классическом смысле проблемы, не являющейся генерически разрешимой за полиномиальное время. Для этого используются другие методы, так как, скорее всего, метод генерической амплификации здесь не работает.

Ключевые слова: *генерическая сложность, амплификация, алгоритмическая проблема.*

GENERICALLY UNDECIDABLE AND HARD PROBLEMS

A. N. Rybalov

Sobolev Institute of Mathematics, Omsk, Russia

The generic-case approach to algorithmic problems examines the behavior of an algorithm on typical (almost all) inputs and ignores the rest of the inputs. The method of generic amplification was proposed by A. Myasnikov and author for constructing of generically undecidable problems. The main ingredient of this method is the cloning technique, which combines the input data of a problem into sufficiently large sets of equivalent input data. Equivalence is understood in the sense that the problem is solved in the same way for them. We present a generalization of this method. We also construct a problem that is decidable in the classical sense, but which is not generically decidable in polynomial time. We use a different method to generic amplification, because generic amplification is unlikely to be applicable here.

Keywords: *generic complexity, amplyfication, algorithmic problems.*

[1]Работа выполнена в рамках госзадания ИМ СО РАН, проект FWNF-2022-0003.

Введение

Генерический подход [1] — это один из подходов к изучению алгоритмических проблем для «почти всех» входов. Исследования вычислительной сложности для «почти всех» входов начались в 1970–80-х гг., после того как был выделен огромный пласт трудноразрешимых алгоритмических проблем — NP-полных проблем, для которых не удалось найти эффективных алгоритмов, работающих за полиномиальное время для всех входов. Оказалось, что если немного ослабить требование эффективности — рассматривать не все входы, а «почти все» или случайные входы, то иногда можно быстро решать задачу для таких типичных входов. Это имеет практический смысл, когда алгоритм должен решать задачу для случайных входных данных: если вероятность «наткнуться» на «плохой» вход пренебрежимо мала, то алгоритм будет быстро работать практически всегда. В рамках генерического подхода изучается поведение алгоритмов на множестве «почти всех» входов (это множество называется генерическим) и игнорируется его поведение на остальных входах, на которых алгоритм может работать медленно или вообще не останавливаться. Понятие «почти все» формализуется введением асимптотической плотности на множестве входных данных.

В исследованиях по генерической вычислимости и сложности вычислений можно выделить два основных направления. Первое связано с построением генерических (полиномиальных) алгоритмов для алгоритмических проблем, которые являются неразрешимыми или трудноразрешимыми в классическом смысле. Второе направление концентрируется на поиске алгоритмических проблем, которые остаются неразрешимыми или трудноразрешимыми и в генерическом смысле. Данная работа относится ко второму направлению исследований.

Первые генерически неразрешимые алгоритмические проблемы были найдены А. Г. Мясниковым и А. Н. Рыбаловым в [2]. Для доказательства генерической неразрешимости предложен метод генерической амплификации, который позволяет по проблеме, неразрешимой в классическом смысле, строить проблему, которая генерически неразрешима. Данный метод успешно применён к следующим алгоритмическим проблемам: проблема остановки для машин Тьюринга [3], проблема равенства для полугрупп [2], проблема разрешимости элементарных теорий [2, 4], десятая проблема Гильберта [5]. Однако формализация метода, предложенная в [2], оказалась не совсем удобной: напрямую её удается применить только для построения конечно определённой полугруппы с генерически неразрешимой проблемой равенства, а в остальных случаях генерическая амплификация используется неформально. В данной работе предлагается формализация более общей схемы генерической амплификации, которая работает во всех случаях.

Применение генерической амплификации для построения генерически трудноразрешимых проблем сталкивается с трудностями, которые связаны с необходимостью контролировать скорость сходимости последовательности частот множества «плохих» входов. Поэтому тут удается получить лишь результаты об отсутствии сильно генерических полиномиальных алгоритмов, которые решают проблему быстро на множестве входов, относительные частоты которых экспоненциально быстро стремятся к единице. Например, в таком виде генерическая амплификация применима для арифметики Пресбургера [6]. В данной работе с помощью других методов строится пример разрешимой в классическом смысле проблемы, для которой не существует полиномиального генерического алгоритма.

1. Предварительные сведения

Пусть I — некоторое множество входов. Для подмножества $S \subseteq I$ определим последовательность

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

где I_n — множество входов размера n, а $S_n = S \cap I_n$ — множество входов из S размера n. Здесь для конечного множества A через $|A|$ обозначено число его элементов. *Асимптотической плотностью* S назовём предел (если он существует)

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

Множество S называется *генерическим*, если $\rho(S) = 1$, и *пренебрежимым*, если $\rho(S) = 0$. Назовём множество S *сильно пренебрежимым*, если последовательность $\rho_n(S)$ экспоненциально быстро сходится к нулю, т. е. существуют константы σ, $0 < \sigma < 1$, и $C > 0$, такие, что для любого n

$$\rho_n(S) < C\sigma^n.$$

Теперь S называется *сильно генерическим*, если его дополнение $I \setminus S$ сильно пренебрежимо.

Алгоритм $\mathcal{A}$ с множеством входов I называется (*сильно*) *генерическим*, если множество $\{x \in I : \mathcal{A}(x)\downarrow\}$ (сильно) генерическое. Здесь $\mathcal{A}(x)\downarrow$ означает, что алгоритм $\mathcal{A}$ останавливается на входе x. Генерический алгоритм $\mathcal{A}$ вычисляет функцию $f : I \to J$, если

$$\forall x \in I\ \big(\mathcal{A}(x)\downarrow \Rightarrow\ f(x) = \mathcal{A}(x)\big).$$

Генерический алгоритм $\mathcal{A}$ работает за полиномиальное время, если существует полином $p(n)$, такой, что

$$\forall x \in I\ \big(\mathcal{A}(x)\downarrow \Rightarrow t_{\mathcal{A}}(x) < p(\mathrm{size}(x))\big),$$

где $\mathrm{size}(x)$ — размер входа x; а $t_{\mathcal{A}}(x)$ — время работы алгоритма $\mathcal{A}$ на входе x. Такие алгоритмы будем называть полиномиальными генерическими.

С практической точки зрения, когда требуется построить алгоритм, решающий конкретную алгоритмическую проблему для почти всех входов, удобнее рассматривать алгоритмы следующего типа [7]. Каждый такой алгоритм останавливается на всех входах, на входах из некоторого генерического множества выдает правильный ответ, а на пренебрежимом множестве остальных входов выдает специальный ответ «?» — «Не знаю».

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

1) $\forall x \in I\ \mathcal{A}(x)\downarrow$;
2) множество $\{x \in I : \mathcal{A}(x) = ?\}$ (сильно) пренебрежимо.

Эффективно генерический алгоритм $\mathcal{A}$ вычисляет функцию $f : I \to J$, если

$$\forall x \in I\ \big(\mathcal{A}(x) \neq ?\ \Rightarrow\ f(x) = \mathcal{A}(x)\big).$$

Множество $S \subseteq I$ и соответствующая проблема распознавания (S, I) (*эффективно*) (*сильно*) *генерически разрешимы* (*за полиномиальное время*), если существует (эффективно) (сильно) генерический (полиномиальный) алгоритм, вычисляющий характеристическую функцию S.

Легко видеть, что из эффективной генерической разрешимости следует генерическая разрешимость. Действительно, любой эффективный генерический алгоритм можно без труда переделать в генерический, заменив выдачу ответа «?» на бесконечное зацикливание. В обратную сторону это неверно — см., например, [8, теорема 2.22 и следствие 2.24]. Однако для полиномиальной сложности верно и обратное: из полиномиальной генерической разрешимости следует полиномиальная эффективная разрешимость. Действительно, если имеется полиномиальная оценка $p(n)$ на время работы генерического алгоритма в случае, когда он останавливается, то можно завести счётчик T числа шагов и если $T > p(n)$, обрывать вычисление и выдавать ответ «?», — в этом случае генерический алгоритм уже не остановится. Таким образом получается эффективно генерический полиномиальный алгоритм, решающий ту же проблему.

2. Генерическая амплификация

Опишем сначала схему, обобщающую конструкцию генерической амплификации, предложенную в [2].

Пусть I — множество входов. *Клонирование* множества I — это функция $C : I \to P(I)$, где $P(I)$ есть множество всех подмножеств множества I. Будем называть клонирование $C : I \to P(I)$ *эффективным*, если существует всюду определённая вычислимая функция $E : I \times \mathbb{N} \to I$, такая, что для любого $x \in I$

$$C(x) = \{E(x, 0), E(x, 1), \dots, \}.$$

Таким образом, с помощью алгоритма E можно эффективно перечислять все элементы каждого клона $C(x)$. Клонирование $C : I \to P(I)$ называется *непренебрежимым* (*не сильно пренебрежимым*), если для любого $x \in I$ множество $C(x)$ не является пренебрежимым (сильно пренебрежимым).

Пусть $S \subseteq I$. Будем говорить, что клонирование $C : I \to P(I)$ *сохраняет* множество S, если:

1) $\forall x \in S \big(C(x) \subseteq S\big)$;
2) $\forall x \notin S \big(C(x) \subseteq I \setminus S\big)$.

Теорема 1. Пусть I — множество входов, $S \subseteq I$ и $C : I \to P(I)$ — эффективное клонирование, сохраняющее S. Тогда:

1) если C непренебрежимое клонирование и проблема распознавания (S, I) генерически разрешима, то проблема распознавания (S, I) разрешима;
2) если C не сильно пренебрежимое клонирование и проблема распознавания (S, I) сильно генерически разрешима, то проблема распознавания (S, I) разрешима.

Доказательство. Докажем п. 1. Пусть $\mathcal{A}$ — генерический алгоритм, распознающий S, такой, что множество

$$G(\mathcal{A}) = \{x \in I : \mathcal{A}(x) \downarrow\}$$

генерическое. Построим алгоритм $\mathcal{B}$, который решает проблему распознавания S для всех входов. На входе $x \in I$ алгоритм $\mathcal{B}$ работает следующим образом:

1) установить $i = 0$;
2) сделать $i + 1$ шагов вычисления алгоритма $\mathcal{A}$ на $E(x, 0)$, $E(x, 1)$, …, $E(x, i)$;
3) если алгоритм $\mathcal{A}$ остановился на каком-то $E(x, k)$, $k \leqslant i$, и выдал ответ, остановиться и выдать этот ответ;
4) иначе увеличить i на 1 и вернуться на шаг 2.

Так как $C(x)$ непренебрежимо, то $C(x)$ имеет непустое пересечение с множеством $G(\mathcal{A})$. Поэтому, параллельно запуская алгоритм $\mathcal{A}$ на элементах $E(x,0)$, $E(x,1)$, ..., мы найдём зависящее от x число i_x, такое, что $x' = E(x,i_x) \in G(\mathcal{A})$. Очевидно, $x \in S$ тогда и только тогда, когда $x' \in S$, и тогда и только тогда, когда $\mathcal{A}$ выдаёт ответ «ДА» для x'. Поэтому мы можем эффективно решать, выполнено ли $x \in S$, и п. 1 доказан.

Доказательство п. 2 аналогично. ■

Теорему 1 можно переформулировать в терминах генерической неразрешимости.

Теорема 2. Пусть I — множество входов, $S \subseteq I$ и $C : I \to P(I)$ — эффективное клонирование, сохраняющее S. Если проблема распознавания (S, I) неразрешима, то имеет место следующее:

1) если C — непренебрежимое клонирование, то проблема распознавания (S, I) не является генерически разрешимой;
2) если C — не сильно пренебрежимое клонирование, то проблема распознавания (S, I) не является сильно генерически разрешимой.

Перейдём к описанию метода генерической амплификации [2]. Назовём клонирование $C : I \to P(I)$ *разделяющим*, если

$$\forall x, y \in I \ (x \neq y) \Rightarrow C(x) \cap C(y) = \varnothing.$$

Для множества $S \subseteq I$ определим *клон* $C(S)$ как объединение всех клонов элементов из S:

$$C(S) = \bigcup_{x \in S} C(x).$$

Легко видеть, что для любого $S \subseteq I$ разделяющее клонирование $C : I \to P(I)$ сохраняет множество $C(S)$. Поэтому непосредственным следствием из теоремы 2 является следующее утверждение:

Теорема 3 [2]. Пусть I — множество входов, $S \subseteq I$ и $C : I \to P(I)$ — эффективное разделяющее клонирование. Тогда если проблема распознавания (S, I) неразрешима, то имеет место следующее:

1) если C — непренебрежимое клонирование, то проблема распознавания $(C(S), I)$ не является генерически разрешимой;
2) если C — не сильно пренебрежимое клонирование, то проблема распознавания $(C(S), I)$ не является сильно генерически разрешимой.

Отметим, что в конкретных ситуациях клонирование редко получается разделяющим. Например, в доказательствах генерической неразрешимости проблемы остановки для нормализованных машин Тьюринга [3] и теорий первого порядка для нормализованных формул [4] клоны для различных элементов пересекаются. Однако соответствующие клонирования сохраняют рассматриваемые множества, а потому, по теореме 2, эти проблемы не являются генерически разрешимыми.

3. Генерически трудноразрешимые проблемы

Рассмотрим машины Тьюринга, которые распознают подмножества двоичных строк из $\{0,1\}^*$. Такие машины имеют два завершающих состояния: q_a — допускающее и q_r — отвергающее. Заканчивать работу они могут только в одном из этих состояний. Таким образом, эти машины выдают только ответы «ДА» или «НЕТ». Под размером строки w понимается её длина $|w|$, поэтому число входов размера n равно 2^n.

Под *эффективной нумерацией всех полиномиальных машин Тьюринга* будем подразумевать эффективную нумерацию всех пар $\{(M_i, p_k(n)) : i \in \mathbb{N},\ k \in \mathbb{N}\}$, где M_i — машина Тьюринга с номером i; $p_k(n) = n^k + k$. Такая пара на входе x моделирует работу некоторой полиномиальной машины Тьюринга следующим образом:

$$(M_i, p_k(n))(x) = \begin{cases} M_i(x), & \text{если } M_i(x)\downarrow \text{ за } \leqslant p_k(|x|) \text{ шагов,} \\ \text{НЕТ} & \text{иначе.} \end{cases}$$

Ясно, что любая полиномиальная машина Тьюринга встретится в этой последовательности.

Теорема 4. Существует рекурсивное множество, не являющееся генерически разрешимым за полиномиальное время.

Доказательство. Пусть есть эффективная нумерация полиномиальных машин Тьюринга $P_1, P_2, P_3, \ldots$ Образуем из них следующую последовательность:

$$\{M_i, i = 1, 2, 3, \ldots\} = \{P_1, P_1, P_2, P_1, P_2, P_3, P_1, P_2, P_3, P_4, P_1, P_2, \ldots\}.$$

В ней каждый раз, после того как были выписаны машины $P_1, \ldots, P_k$, выписываются машины $P_1, \ldots, P_{k+1}$. Таким образом, каждая полиномиальная машина P_i выписывается бесконечно много раз.

Построение нужного множества S будет проходить по шагам. Стартуя с множества всех двоичных строк $\{0,1\}^*$ на нулевом шаге, мы будем на шаге i вычеркивать или оставлять некоторые числа в зависимости от поведения машины M_i. Опишем подробно шаг $i > 0$. Запускаем машину M_i на каждом входе размера i и считаем количество ответов «ДА» и «НЕТ». Если ответов «ДА» получилось больше половины, то вычёркиваем все входы размера i, иначе все их оставляем. Предельное множество в этом процессе и есть искомое множество S.

Действительно, заметим, что для любой полиномиальной машины M множество входов, на которых M дает неправильный ответ, имеет вид

$$E(M) = \bigcup_{i=1}^{\infty} A_i,$$

где $A_i = \{w \in \{0,1\}^* : |w| = m_i\}$, $m_i > m_{i-1}$, причём $|A_i| \geqslant 2^{m_i}/2$ для любого i. Если теперь рассмотреть последовательность

$$\rho_n(E(M)) = \frac{|E(M)_n|}{2^n},\ n = 1, 2, 3, \ldots,$$

то легко видеть, что $\rho_n(E(M)) \geqslant 1/2$ для бесконечно большого числа значений n. Поэтому множество $E(M)$ непренебрежимо.

Допустим, что существует генерический полиномиальный алгоритм $\mathcal{A}$, распознающий множество S. Без ограничения общности можно считать, что $\mathcal{A}$ — полиномиальный эффективно генерический алгоритм. По нему легко получить полиномиальную машину M, которая на любом входе x работает следующим образом:

1) вычисляет $\mathcal{A}(x)$;
2) если $\mathcal{A}(x) = 1$, выдаёт 1;
3) если $\mathcal{A}(x) = 0$, выдаёт 0;
4) если $\mathcal{A}(x) = ?$, выдаёт 0.

Очевидно, что M, распознавая элементы S, ошибается на пренебрежимом множестве. Но это противоречит построению множества S.

Рекурсивность множества S следует из алгоритмической природы процедуры его построения. ■

Заключение

В работе предложена формализация схемы генерической амплификации, которая обобщает схему [2]. Новый метод работает во всех случаях, в которых напрямую не удаётся применить схему из [2]. Кроме того, с помощью других идей строится пример разрешимой в классическом смысле проблемы, для которой не существует полиномиального генерического алгоритма. В данном случае, по-видимому, метод генерической амплификации неприменим.

В качестве дальнейших направлений исследований представляет интерес связать понятие генерической разрешимости с мерой М. Громова, определяемой также с помощью асимптотической плотности. Особо интересны случаи, когда эта плотность отлична от нуля и единицы. В этом направлении Р. Гилманом, А. Г. Мясниковым и В. А. Романьковым получены очень интересные результаты о плотности разрешимых уравнений в свободных группах [9] и в нильпотентных группах [10].

Автор выражает благодарность рецензенту за полезные замечания и предложения по улучшению текста статьи.

ЛИТЕРАТУРА

1. *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.
2. *Myasnikov A. and Rybalov A.* Generic complexity of undecidable problems // J. Symbolic Logic. 2008. V. 73. No. 2. P. 656–673.
3. *Rybalov A.* On the generic undecidability of the Halting Problem for normalized Turing machines // Theory of Computing Systems. 2017. V. 60. No. 4. P. 671–676.
4. *Рыбалов А. Н.* О генерической сложности элементарных теорий // Вестник Омского университета. 2015. № 4. С. 14–17.
5. *Rybalov A.* Generic complexity of the Diophantine problem // Groups Complexity Cryptology. 2013. V. 5. No. 1. P. 25–30.
6. *Rybalov A.* Generic complexity of Presburger arithmetic // Theory of Computing Systems. 2010. V. 46. No. 1. P. 2–8.
7. *Hirschfeldt D.* Some questions in computable mathematics // LNCS. 2017. V. 10010. P. 22–55.
8. *Jockusch C. and Schupp P.* Generic computability, Turing degrees, and asymptotic density // J. London Math. Soc. 2012. V. 85. No. 2. P. 472–490.
9. *Gilman A., Myasnikov A., and Roman'kov V. A.* Random equations in free groups // Groups Complexity Cryptology. 2011. V. 3. No. 2. P. 257–284.
10. *Gilman A., Myasnikov A., and Roman'kov V. A.* Random equations in nilpotent groups // J. Algebra. 2012. V. 352. No. 1. P. 192–214.

REFERENCES

1. *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.
2. *Myasnikov A. and Rybalov A.* Generic complexity of undecidable problems. J. Symbolic Logic, 2008, vol. 73, no. 2, pp. 656–673.
3. *Rybalov A.* On the generic undecidability of the Halting Problem for normalized Turing machines. Theory of Computing Systems, 2017, vol. 60, no. 4, pp. 671–676.
4. *Rybalov A. N.* O genericheskoy slozhnosti elementarnykh teoriy [On the generic complexity of elementary theories]. Vestnik OmSU, 2015, no. 4, pp. 14–17. (in Russian)
5. *Rybalov A.* Generic complexity of the Diophantine problem. Groups Complexity Cryptology, 2013, vol. 5, no. 1, pp. 25–30.

6. *Rybalov A.* Generic complexity of Presburger arithmetic. // Theory of Computing Systems, 2010. vol. 46, no. 1, pp. 2–8.
7. *Hirschfeldt D.* Some questions in computable mathematics. LNCS, 2017, vol. 10010, pp. 22–55.
8. *Jockusch C. and Schupp P.* Generic computability, Turing degrees, and asymptotic density. J. London Math. Soc., 2012, vol. 85, no. 2, pp. 472–490.
9. *Gilman A., Myasnikov A. and Roman'kov V. A.* Random equations in free groups. Groups Complexity Cryptology, 2011, vol. 3, no. 2, pp. 257–284.
10. *Gilman A., Myasnikov A. and Roman'kov V. A.* Random equations in nilpotent groups. J. Algebra, 2012, vol. 352, no. 1, pp. 192–214.

ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ

УДК 519.16 DOI 10.17223/20710410/63/8

КОМБИНАТОРНЫЕ СВОЙСТВА ЗАДАЧИ ОБ ОГРАНИЧЕННОМ РЮКЗАКЕ

М. С. А. Волков

Московский государственный технический университет им. Н. Э. Баумана, г. Москва, Россия

E-mail: sabina-volkoff@yandex.ru

Рассматриваются комбинаторные свойства множества решений в задаче об ограниченном рюкзаке. Как и в общем случае, эта задача является NP-полной задачей комбинаторной оптимизации и её точное решение требует применения алгоритмов перебора с декомпозицией множества допустимых решений. В связи с этим актуален вопрос определения и оценки свойств множества допустимых решений. Получены формулы, позволяющие вычислять среднее значение функционала задачи на множестве её допустимых решений и мощность этого множества через число решений подзадач меньшей размерности. Базовой техникой получения результатов служит метод производящих функций. Рассмотрена задача о рюкзаке с произвольными значениями переменных, в которой совпадают коэффициенты вектора ограничений и целевой функции. Для неё предполагается «сюръективность» множества решений. Найдены оценки значений функционала в этой задаче. Результаты могут представлять интерес для конструирования вычислительных алгоритмов нахождения и оценки числа решений и значения функционала на оптимальных решениях. Найденные выражения также могут быть использованы во вспомогательных процедурах для оценки оптимальности решения в декомпозиционных или эвристических алгоритмах решения задачи о рюкзаке.

Ключевые слова: *задача о рюкзаке, производящие функции, NP-полные задачи, метод коэффициентов, вычет, методы декомпозиции.*

COMBINATORIAL PROPERTIES OF THE BOUNDED KNAPSACK PROBLEM

M. S. A. Volkov

Bauman Moscow State Technical University, Moscow, Russia

The combinatorial properties of the set of solutions to the bounded knapsack problem are considered. As in the general case, this problem is an NP-complete combinatorial optimization problem and its exact solution requires the use of search algorithms with the decomposition of a set of feasible solutions. In this regard, the question of determining and evaluating the properties of the set of acceptable solutions to the problem is relevant. In this paper, formulas are obtained which allow to calculate the

average value of the functional of a problem on the set of its feasible solutions and the power of this set through the number of solutions of subtasks of smaller dimension. The basic technique for obtaining results is the method of generating functions. We also consider the knapsack problem with arbitrary values of variables, in which the coefficients of the constraint vector and the objective function coincide. For this, the "continuity" of the set of solutions is assumed. Estimates of the values of the functional in this problem are found. The results may be of interest for the design of computational algorithms for finding and estimating the number of solutions and the value of the functional for optimal solutions. The expressions found can also be used in auxiliary procedures to evaluate the optimality of the solution in decomposition or heuristic algorithms for solving the knapsack problem.

Keywords: *knapsack problem, generating functions, NP-complete problems, coefficient method, deduction, decomposition methods.*

Введение

Задача об ограниченном рюкзаке — это вариант классической задачи о рюкзаке, в которой каждый предмет доступен в определённом ограниченном количестве. Имеется набор предметов, содержащий m копий каждого предмета, где k-й предмет $(1 \leqslant k \leqslant n)$ имеет два неотрицательных целочисленных параметра — вес a_k и ценность c_k. Определено ограничение грузоподъёмности рюкзака b. Задача состоит в том, чтобы выбрать подмножество предметов с максимальной общей ценностью, суммарный вес которого не превышает грузоподъёмности рюкзака. В виде оптимизации задача об ограниченном рюкзаке задаётся выражением [1]

$$\sum_{j=1}^{n} c_j x_j \to \max; \tag{1}$$

$$\sum_{i=1}^{n} a_i x_i \leqslant b, \tag{2}$$

где $x = (x_1, \ldots, x_n)$ — n-мерный вектор с целочисленными компонентами $x_i \in \{0, 1, \ldots, m\}$; $c_1, \ldots, c_n$, $a_1, \ldots, a_n$, b — неотрицательные целые числа.

Так как задача (1), (2) является NP-полной и для получения её точного решения используются переборные и декомпозиционные алгоритмы, то актуален вопрос о связи сложности задачи со сложностью её подзадач меньшей размерности. В эвристических подходах используются процедуры получения приближённых оценок значений функционала и распределения значений функционала в области допустимых значений переменных, поэтому формулы для вычисления таких оценок могут непосредственно применяться в подобных алгоритмах либо служить для сравнения используемых алгоритмов.

Для доказательства основных результатов в данной работе использован метод производящих функций. Базовой техникой для выражения ограничений на множество допустимых решений послужил метод коэффициентов. Данный метод определяет линейный функционал на множестве формальных степенных рядов с конечным числом членов отрицательной степени, который ставит в соответствие каждому степенному ряду коэффициент при его члене в минус первой степени. Для степенных рядов, сходящихся в окрестности нуля, этот коэффициент совпадает с вычетом в точке ноль. В ряде случаев этот метод существенно удобнее классического варианта с применением вычетов. Подробное описание метода приведено в [2].

Задача о рюкзаке и её разновидности находят применение в области математического программирования, в частности в теории кодирования и криптографии [3, 4]. Например, задача о рюкзаке стала основой для нескольких криптографических систем, безопасность которых зависит от сложности получения её решения [5]. Поскольку различные вариации задачи о рюкзаке часто возникают при ослаблении задач целочисленного программирования, она интенсивно изучалась в последние десятилетия. Как следствие, литература по ней обширна и охватывает как вопросы, связанные с разработкой алгоритмов, так и теоретические аспекты, связанные со свойствами задачи. Алгоритмическая сторона рассматривается, например, в работах [1, 6].

В последнее десятилетие широкое распространение получило применение эвристических и метаэвристических подходов к решению данной задачи. В [7] предложен алгоритм амебоидного организма для решения задачи о 0-1 рюкзаке. В работе [8] авторы использовали алгоритм когортного интеллекта — метод оптимизации, навеянный естественной склонностью людей учиться друг у друга. Модифицированный генетический алгоритм для решения задачи о многомерном рюкзаке, основанный на предварительном анализе данных, предложен в [9]. В [10] для решения многомерной задачи о рюкзаке представлена эвристика, основанная на методе поиска гармонии. В этом алгоритме внимание уделяется распределению вероятностей значений переменных вместо поиска их точного значения. Гибридный алгоритм решения задачи о 0-1 рюкзаке, основанный на разделении предметов по регионам по степени «жадности», построен в работе [11]. В [12] разработана эвристика, сочетающая методы уменьшения размерности задачи, основанные на правилах фиксации переменных, с решением результирующей целочисленной линейной задачи. В [13] авторы представили переформулировку многомерной задачи о рюкзаке с множественным выбором как задачи разделения множества, позволяющую уменьшить размерность задачи при сохранении общего числа переменных и ограничений.

Распространение получили также работы, связанные с исследованием свойств области допустимых решений в задачах о рюкзаке. В [14] реализован распределённый итерационный метод с фиксированной точкой для решения задачи выполнимости рюкзачных ограничений. В работе [15] предложены полиномиальные по времени алгоритмы оценки числа решений отдельных ограничений. В [16] изучается структура многогранников задач о рюкзаке специального вида. Исчерпывающий обзор последних исследований рюкзачных многогранников приведён в [17].

В данной работе получены комбинаторные формулы, позволяющие вычислять и оценивать значения функционала в зависимости от набора заданных параметров задачи. В п. 1 приведены вспомогательные утверждения, выражающие производящие функции в виде полиномов для множества допустимых решений и значений функционала задачи на этом множестве. В п. 2 найдены выражения для числа решений и математического ожидания значения функционала задачи через число решений подзадач меньшей размерности. В п. 3 рассмотрен случай совпадения коэффициентов вектора ограничений и целевой функции, найдены оценки функционала при его сюръективности на всём множестве решений. Ряд результатов с применением подхода на основе метода производящих функций для задачи о рюкзаке с булевыми переменными приведён в работах [18, 19]. Этот подход использован и в настоящей работе для получения формул для случая ограниченного рюкзака с возможностью повторения предметов.

1. Вспомогательные утверждения

Выразим производящие функции в виде полиномов для множества допустимых решений и значений функционала задачи на этом множестве. Множество допустимых решений задачи V_b — это множество n-мерных векторов x, $x_i \in \{0, 1, \ldots, m\}$, $i = 1, \ldots, n$, удовлетворяющих неравенству (2). По аналогии с непрерывным случаем будем называть множество V_b многогранником допустимых решений задачи. Объёмом V_b назовём число $|V_b|$ допустимых решений неравенства (2).

Для анализа распределения точек в многограннике V_b используется полином

$$P_b(z_1, z_2, \ldots, z_n) = \sum_{x \in V_b} z_1^{a_1 x_1} z_2^{a_2 x_2} \ldots z_n^{x_n a_n}. \tag{3}$$

Для исследования свойств значений функционала задачи (1), (2) в допустимых точках многогранника решений будем рассматривать полином

$$F_b(z_1, z_2, \ldots, z_n) = \sum_{x \in V_b} z_1^{c_1 x_1} z_2^{c_2 x_2} \ldots z_n^{x_n c_n}. \tag{4}$$

Примеры использования этих полиномов для получения оценок в различных типах задачи о рюкзаке приведены в работах [20, 21].

Лемма 1. Для задачи об ограниченном рюкзаке (1), (2) справедлива формула

$$\sum_{b=0}^{\infty} P_b(z_1, \ldots, z_n) u^b = \frac{(1+(z_1u)^{a_1}+\ldots+(z_1u)^{ma_1}) \ldots (1+(z_nu)^{a_n}+\ldots+(z_nu)^{ma_n})}{1-u}. \tag{5}$$

Доказательство. Преобразуем сумму (3), используя метод коэффициентов. Внутреннее суммирование проводится по всему множеству векторов $(x_1, x_2, \ldots, x_n)$ с координатами из $\{0, 1, \ldots, m\}$. Использование метода коэффициентов позволяет отбирать из этого множества только векторы, удовлетворяющие ограничениям (2):

$$P_b(z_1, \ldots, z_n) = \sum_{t=0}^{b} \sum_{\{x_1, \ldots x_n\}} z_1^{a_1 x_1} z_2^{a_2 x_2} \ldots z_n^{a_n x_n} \frac{1}{2\pi i} \oint_{|u|=\rho} \frac{u^{\sum_{i=1}^{n} a_i x_i}}{u^{t+1}} du =$$

$$= \frac{1}{2\pi i} \oint_{|u|=\rho} \sum_{t=0}^{b} \frac{1}{u^{t+1}} \sum_{x_1=0}^{m} (z_1u)^{a_1 x_1} \ldots \sum_{x_n=0}^{m} (z_nu)^{a_n x_n} du =$$

$$= \frac{1}{2\pi i} \oint_{|u|=\rho} \frac{u^{-(b+2)} - u^{-1}}{u^{-1} - 1} \prod_{k=1}^{n} (1 + (z_ku)^{a_k} + \ldots + (z_ku)^{ma_k}) du.$$

Раскладывая полученное выражение по числителю дроби, имеем

$$P_b(z_1, \ldots, z_n) = \frac{1}{2\pi i} \oint_{|u|=\rho} \frac{1}{u^{b+1}(1-u)} \prod_{k=1}^{n} (1 + (z_ku)^{a_k} + \ldots + (z_ku)^{ma_k}) du +$$

$$+ \frac{1}{2\pi i} \oint_{|u|=\rho} \frac{1}{(1-u)} \prod_{k=1}^{n} (1 + (z_ku)^{a_k} + \ldots + (z_ku)^{ma_k}) du.$$

Ввиду теоремы о вычетах, последнее слагаемое равно нулю; окончательно получим

$$P_b(z_1, \ldots, z_n) = \frac{1}{2\pi i} \oint_{|u|=\rho} \frac{1}{u^{b+1}(1-u)} \prod_{k=1}^{n} (1 + (z_ku)^{a_k} + \ldots + (z_ku)^{ma_k}) du. \tag{6}$$

Воспользуемся правилом снятия коэффициента [2]:

$$\frac{1}{2\pi i}\oint\limits_{|u|=\rho} A(u)du = \underset{u}{\mathrm{coef}}\{A(u)\} = a_{-1},$$

где a_{-1} — коэффициент при минус первой степени многочлена $A(u)$.

Подставляя выражение (6) в левую часть формулы (5), получим

$$\sum_{b=0}^{\infty} P_b(z_1,\ldots,z_n)u^b = \sum_{b=0}^{\infty} u^b \frac{1}{2\pi i}\oint\limits_{|u|=\rho} \frac{1}{u^{b+1}(1-u)}\prod_{k=1}^{n}(1+(z_k u)^{a_k}+\ldots+(z_k u)^{ma_k})du =$$

$$= \sum_{b=0}^{\infty} u^b \underset{u}{\mathrm{coef}}\Big\{\frac{1}{u^{b+1}(1-u)}\prod_{k=1}^{n}(1+(z_k u)^{a_k}+\ldots+(z_k u)^{ma_k})\Big\}.$$

Теперь, воспользовавшись правилом замены переменной [2]

$$\sum_{k=0}^{\infty} z^k \underset{u}{\mathrm{coef}}\{A(u)u^{-k-1}\} = A(z)$$

для u, получим искомое соотношение. ■

Следствие 1. Для объёма области допустимых решений задачи (1), (2) с $m \in \mathbb{N}$ имеет место равенство

$$|V_b| = \frac{1}{2\pi i}\oint\limits_{|u|=\rho} \frac{(1+u^{a_1}+\ldots+u^{ma_1})\ldots(1+u^{a_n}+\ldots+u^{ma_n})}{(1-u)u^{b+1}}du. \tag{7}$$

Здесь и далее параметр ρ удовлетворяет условиям $0 < \rho < 1$.

Доказательство. Для нахождения числа допустимых решений задачи необходимо подставить $z = 1$ в ряд (3) и провести рассуждения, аналогичные доказательству леммы 1. ■

Лемма 2. Имеет место равенство:

$$F_b(z_1,\ldots,z_n) = \frac{1}{2\pi i}\oint\limits_{|u|=\rho} \frac{(1+z_1^{c_1}u^{a_1}+\ldots+z_1^{mc_1}u^{ma_1})\ldots(1+z_n^{c_n}u^{a_n}+\ldots+z_n^{mc_n}u^{ma_n})}{(1-u)u^{b+1}}du. \tag{8}$$

Доказательство. Преобразуем сумму (4), используя метод коэффициентов. Аналогично лемме 1, введение интеграла позволяет получить ограничение области допустимых решений задачи:

$$F_b(z_1,\ldots,z_n) = \sum_{t=0}^{b}\sum_{\{x_1,\ldots x_n\}} z_1^{c_1x_1}z_2^{c_2x_2}\ldots z_n^{c_nx_n}\frac{1}{2\pi i}\oint\limits_{|u|=\rho}\frac{u^{\sum\limits_{i=1}^{n}a_ix_i}}{u^{t+1}}du =$$

$$= \frac{1}{2\pi i}\oint\limits_{|u|=\rho}\sum_{t=0}^{b}\frac{1}{u^{t+1}}\sum_{x_1=0}^{m}(z_1^{c_1}u^{a_1})^{x_1}\ldots\sum_{x_n=0}^{m}(z_n^{c_n}u^{a_n})^{x_n}du =$$

$$= \frac{1}{2\pi i}\oint\limits_{|u|=\rho}\frac{1}{u^{b+1}(1-u)}\prod_{k=1}^{n}(1+z_k^{c_k}u^{a_k}+\ldots+z_k^{mc_k}u^{ma_k})du. \tag{9}$$

Напомним, что метод коэффициентов — это выражение коэффициента при минус первой степени переменной через сумму вычетов, что имеет тот же смысл, что и интеграл Коши в формуле (8). Таким образом, выражение (9) эквивалентно интегралу (8). ■

2. Свойства функционала в задаче об ограниченном рюкзаке

Для эффективного решения задачи о рюкзаке при помощи алгоритмов декомпозиции и перебора необходимо иметь способы оценки значений функционала решений задачи. В этом контексте может быть полезна формула, которая выражает среднее значение функционала на множестве допустимых решений.

Рассмотрим производящую функцию (3), которая характеризует распределение значений функционала $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} c_j x_j$ задачи (1), (2). Для целого неотрицательного k обозначим через A_k число допустимых решений задачи, для которых $f(x_1, \ldots, x_n) = k$. Также введём следующее обозначение:

$$\Phi_b(z) = F_b(z, \ldots, z) = \sum_{x \in V_b} z^{c_1 x_1} z^{c_2 x_2} \ldots z^{c_n x_n} = \sum_{k=0}^{\infty} A_k z^k. \tag{10}$$

Из введённых определений, обозначений и формулы (10) следует соотношение

$$|V_b| = \Phi_b(1) = F_b(1, \ldots, 1) = \sum_{x \in V_b} 1^{c_1 x_1} 1^{c_2 x_2} \ldots 1^{c_n x_n} = \sum_{k=0}^{\infty} A_k.$$

В частности, заметим, что

$$\max_{x \in V_b} f(x_1, \ldots, x_n) = \max_{x \in V_b} \sum_{j=1}^{n} c_j x_j = \max_{k : A_k \geqslant 1} k.$$

Далее из леммы 2 получим формулу

$$\Phi_b(z) = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{(1 + u^{a_1} z^{c_1} + \ldots + u^{m a_1} z^{m c_1}) \ldots (1 + u^{a_n} z^{c_n} + \ldots + u^{m a_n} z^{m c_n})}{(1-u) u^{b+1}} du. \tag{11}$$

Будем считать, что все точки многогранника V_b равновероятны. Тогда значения функционала $f(x_1, \ldots, x_n)$ — это случайная величина $\xi = \xi(a_1, \ldots, a_n, c_1, \ldots, c_n, b)$ с производящей функцией вероятностей $P(z) = \dfrac{\Phi_b(z)}{\Phi_b(1)}$. Обозначим её математическое ожидание $\mu(\xi)$. Математическое ожидание (первый момент) случайной величины определяется первой производной её производящей функции вероятностей в точке $z = 1$.

Для определения первой производной функции $P(z)$ введём обозначение

$$\phi(z, u) = \prod_{k=1}^{n} (1 + z^{c_k} u^{a_k} + \ldots + z^{m c_k} u^{m a_k}).$$

Тогда производная функции $\phi(z, u)$ имеет следующий вид:

$$\phi'(z, u) =$$

$$= \sum_{k=1}^{n} \Big((c_k z^{c_k - 1} u^{a_k} + 2 c_k z^{2c_k - 1} u^{2 a_k} + \ldots + m c_k z^{m c_k - 1} u^{m a_k}) \prod_{\substack{i=1, \\ i \neq k}}^{n} (1 + z^{c_i} u^{a_i} + \ldots + z^{m c_i} u^{m a_i}) \Big).$$

Выражая её через $\phi(z, u)$, получим

$$\phi'(z, u) = \sum_{k=1}^{n} \frac{c_k z^{c_k - 1} u^{a_k} + 2 c_k z^{2c_k - 1} u^{2 a_k} + \ldots + m c_k z^{m c_k - 1} u^{m a_k}}{(1 + z^{c_k} u^{a_k} + \ldots + z^{m c_k} u^{m a_k})} \phi(z, u).$$

Отсюда найдём значение первой производной функции $\Phi_b(z)$:

$$\Phi^{'}(z) = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \sum_{k=1}^{n} \frac{c_k z^{c_k-1} u^{a_k} + 2c_k z^{2c_k-1} u^{2a_k} + \ldots + mc_k z^{mc_k-1} u^{ma_k}}{(1 + z^{c_k} u^{a_k} + \ldots + z^{mc_k} u^{ma_k})} \frac{\phi(z,u)}{(1-u)u^{b+1}} du.$$

Подставив в это выражение $z = 1$, получим

$$\Phi^{'}(1) = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \sum_{k=1}^{n} \frac{c_k u^{a_k} + 2c_k u^{2a_k} + \ldots + mc_k u^{ma_k}}{(1 + u^{a_k} + \ldots + u^{ma_k})} \frac{\prod\limits_{i=1}^{n}(1 + u^{a_i} + \ldots + u^{ma_i})}{(1-u)u^{b+1}} du. \quad (12)$$

Для каждой из n переменных введём $m+1$ «сечений» множества допустимых решений задачи V_b следующим образом. Для переменной x_k $(1 \leqslant k \leqslant n)$ «сечение» с номером d $(0 \leqslant d \leqslant m)$ содержит все решения, удовлетворяющие условию

$$\sum_{\substack{i=1,\\ i\neq k}}^{n} a_i x_i \leqslant b - da_k, \quad x_i \in \{0, 1, \ldots, m\}.$$

Эти решения соответствуют подмножеству решений задачи (1), (2) с $x_k = d$. Обозначим это множество через V_b^{dk}. Из следствия 1 получаем

$$|V_b^{dk}| = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1, i\neq k}^{n}(1 + u^{a_i} + \ldots + u^{ma_i})}{(1-u)u^{b+1-da_k}} du. \quad (13)$$

Теорема 1. Справедливо соотношение

$$\mu(\xi) = \frac{1}{|V_b|} \sum_{k=1}^{n} c_k(|V_b^{1k}| + 2|V_b^{2k}| + \ldots + m|V_b^{mk}|). \quad (14)$$

Доказательство. Вспомним, что $P(z) = \Phi_b(z)/\Phi_b(1)$ и математическое ожидание случайной величины выражается как первая производная её производящей функции $P(z)$ в точке $z = 1$:

$$\mu(\xi) = P^{'}(1) = \frac{\Phi_b^{'}(1)}{\Phi_b(1)}. \quad (15)$$

По формуле (11) имеем

$$\Phi^{'}(1) = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \sum_{k=1}^{n} \frac{c_k u^{a_k} + 2c_k u^{2a_k} + \ldots + mc_k u^{ma_k}}{1 + u^{a_k} + \ldots + u^{ma_k}} \frac{\prod\limits_{i=1}^{n}(1 + u^{a_i} + \ldots + u^{ma_i})}{(1-u)u^{b+1}} du.$$

Разложим это выражение по первому множителю на m слагаемых:

$$\Phi^{'}(1) = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \sum_{k=1}^{n} \left(\frac{c_k u^{a_k}}{(1-u)u^{b+1}} \frac{\prod\limits_{i=1}^{n}(1 + u^{a_i} + \ldots + u^{ma_i})}{1 + u^{a_k} + \ldots + u^{ma_k}} + \right.$$

$$\left. + \frac{2c_k u^{2a_k}}{(1-u)u^{b+1}} \frac{\prod\limits_{i=1}^{n}(1 + u^{a_i} + \ldots + u^{ma_i})}{1 + u^{a_k} + \ldots + u^{ma_k}} + \ldots + \frac{mc_k u^{ma_k}}{(1-u)u^{b+1}} \frac{\prod\limits_{i=1}^{n}(1 + u^{a_i} + \ldots + u^{ma_i})}{1 + u^{a_k} + \ldots + u^{ma_k}} \right) du.$$

Теперь вынесем c_k за знак интеграла и заметим, что слагаемые данного выражения содержат правые части выражений (13) для $d = 1, \ldots, m$:

$$\Phi'(1) = \sum_{k=1}^{n} \left(c_k \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1, i\neq k}^{n} (1 + u^{a_i} + \ldots + u^{ma_i})}{(1-u)u^{b+1-a_k}} du + \right.$$

$$\left. + 2c_k \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1, i\neq k}^{n} (1 + u^{a_i} + \ldots + u^{ma_i})}{(1-u)u^{b+1-2a_k}} du + \ldots + mc_k \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1, i\neq k}^{n} (1 + u^{a_i} + \ldots + u^{ma_i})}{(1-u)u^{b+1-ma_k}} du \right).$$

С учетом равенств (13) произведём замены интегралов их обозначениями и получим соотношение

$$\Phi_b'(1) = \sum_{k=1}^{n} c_k (|V_b^{1k}| + 2|V_b^{2k}| + \ldots + m|V_b^{mk}|).$$

Из выражений (11) и (7) следует

$$\Phi_b(1) = |V_b| = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{(1 + u^{a_1} + \ldots + u^{ma_1}) \ldots (1 + u^{a_n} + \ldots + u^{ma_n})}{(1-u)u^{b+1}} du.$$

Подставляя найденные значения в (15), получим искомое соотношение (14). ■

Выражение (14) может быть полезным при оценке эффективности алгоритмов, применяемых для решения задач о рюкзаке. В частности, среднее значение оптимизируемого функционала задачи может служить показателем качества решения при сравнении с результатами, полученными с применением эвристических или аппроксимационных алгоритмов. Если значения, полученные таким алгоритмом, существенно превышают среднее значение функционала, это говорит о том, что алгоритм обеспечивает решения, близкие к оптимальным. Кроме того, данная формула может быть применена для нахождения нижней оценки оптимального значения функционала задачи на подобласти допустимых значений переменной при использовании алгоритмов декомпозиции, например в методе ветвей и границ.

Выражение $|V_b|$ также можно представить через сумму $|V_b^{dk}|$, раскладывая по скобке, соответствующей переменной x_k:

$$|V_b| = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{(1 + u^{a_1} + \ldots + u^{ma_1}) \ldots (1 + u_n^a + \ldots + u^{ma_n})}{(1-u)u^{b+1}} du.$$

Домножим и разделим на $(1 + u^{a_k} + \ldots + u^{ma_k})$:

$$|V_b| = \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{(1 + u^{a_k} + \ldots + u^{ma_k})}{(1 + u^{a_k} + \ldots + u^{ma_k})} \frac{(1 + u^{a_1} + \ldots + u^{ma_1}) \ldots (1 + u^{a_n} + \ldots + u^{ma_n})}{(1-u)u^{b+1}} du.$$

Разложим теперь по числителю $(1+u^{a_k}+\ldots+u^{ma_k})$ на $(m+1)$ слагаемых и заметим, что они содержат выражения (13) для $d=0,\ldots,m$:

$$|V_b| = \left(\frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1,i\neq k}^{n} 1+u^{a_i}+\ldots+u^{ma_i}}{(1-u)u^{b+1}} du + \right.$$

$$\left. + \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1,i\neq k}^{n} 1+u^{a_i}+\ldots+u^{ma_i}}{(1-u)u^{b+1-a_k}} du + \ldots + \frac{1}{2\pi i} \oint\limits_{|u|=\rho} \frac{\prod\limits_{i=1,i\neq k}^{n} 1+u^{a_i}+\ldots+u^{ma_i}}{(1-u)u^{b+1-ma_k}} du \right).$$

Заменяя интегралы их обозначениями из (13), получим

$$|V_b| = |V_b^{0k}| + |V_b^{1k}| + \ldots + |V_b^{mk}|. \tag{16}$$

Такой метод вычисления числа решений задачи может быть эффективнее прямого подсчёта, поскольку данная формула декомпозирует решение задачи на подзадачи меньшей размерности. При этом области допустимых значений для этих подзадач вложены друг в друга, т.е. $V_b^{di} \subset V_b^{ci}$ при $d \leqslant c$ для всех $i=1,\ldots,n$, поэтому при последовательном вычислении $|V_b^{di}|$ для $d=1,\ldots,m$ можно использовать $|V_b^{di}|$ при нахождении $|V_b^{ci}|$, $c=d,\ldots,m$, что сокращает объём вычислений.

Формула (16) позволяет также сократить количество рассчитываемых значений в (14). Для этого достаточно подставить в (14) значение $|V_b|$, определённое по формуле (16) для какой-нибудь одной переменной x_j, например наименьшего значения a_j. Тогда

$$\mu(\xi) = \frac{1}{|V_b|} \sum_{k=1}^{n} c_k(|V_b^{1k}| + 2|V_b^{2k}| + \ldots + m|V_b^{mk}|) =$$

$$= \frac{1}{|V_b^{0j}| + |V_b^{1j}| + \ldots + |V_b^{mj}|} \sum_{k=1}^{n} c_k(|V_b^{1k}| + 2|V_b^{2k}| + \ldots + m|V_b^{mk}|).$$

Вычисление среднего значения функционала по данной формуле также удобно, поскольку оно может выполняться параллельно. Ввиду того, что «сечения» V_b^{di} по каждой переменной i представляют собой непересекающиеся подмножества, значения их объёмов $|V_b^{di}|$ могут быть вычислены независимо, а затем просуммированы для получения среднего значения. Такой подход может значительно ускорить процесс вычислений, особенно при работе с большими экземплярами задач.

Формулы (14) и (16) могут быть уточнены для учёта уникальных особенностей начальных условий при изучении задач, возникающих в конкретных областях применения. Например, в криптографии интерес представляет решение задачи о рюкзаке в несколько изменённом виде: известно, что уравнение $\sum\limits_{i=1}^{n} a_i x_i = b$ имеет решение в числах $\{0,1\}$, и требуется найти это решение. В этом случае выражение (14) имеет вид

$$\mu(\xi) = \frac{1}{|V_b|} \sum_{k=1}^{n} a_k |V_b^{1k}|.$$

Данная формула выражает среднее значение допустимых решений задачи $\sum\limits_{i=1}^{n} a_i x_i \leqslant b$ и может применяться для определения вероятности успешной атаки на рюкзачные криптосистемы методом перебора.

Проиллюстрируем выполнение найденных формул на простых примерах.

Пример 1. Рассмотрим следующую задачу:

$$\begin{cases} x_1 + 2x_2 + 3x_3 \to \max, \\ x_1 + 2x_2 + x_3 \leqslant 3, \\ x_1, x_2, x_3 \in \{0, 1, 2\}. \end{cases}$$

У этой задачи 11 допустимых решений: $V_b = \{(000), (001), (010), (011), (100), (101), (110), (002), (200), (102), (201)\}$.

Для нахождения V_b вычислим $|V_b^{0k}|$ для наименьшего $a_k = 1$: $V_b^{01} = \{(00), (01), (10), (11), (02)\}$ — решения неравенства $2x_2 + x_3 \leqslant 3$, $|V_b^{01}| = 5$.

Далее для нахождения $\mu(\xi)$ вычислим $|V_b^{ik}|$ для $k = 1, 2, 3$, $i = 1, 2$ по формуле (13):
$V_b^{11} = \{(00), (10), (01), (02)\}$ — решения неравенства $2x_2 + x_3 \leqslant 2$, $|V_b^{11}| = 4$;
$V_b^{21} = \{(00), (01)\}$ — решения неравенства $2x_2 + x_3 \leqslant 1$, $|V_b^{21}| = 2$;
$V_b^{12} = \{(00), (01), (10)\}$ — решения неравенства $x_1 + x_3 \leqslant 1$, $|V_b^{12}| = 3$;
$V_b^{22} = \varnothing$ — решения неравенства $x_1 + x_3 \leqslant -1$, $|V_b^{22}| = 0$;
$V_b^{13} = \{(00), (01), (10), (20)\}$ — решения неравенства $x_1 + 2x_2 \leqslant 2$, $|V_b^{13}| = 4$;
$V_b^{23} = \{(00), (10)\}$ — решения неравенства $x_1 + 2x_2 \leqslant 1$, $|V_b^{23}| = 2$.

Подставляя эти значения в (16) и (14), получаем

$$|V_b| = |V_b^{01}| + |V_b^{11}| + |V_b^{21}| = 5 + 4 + 2 = 11,$$

$$\mu(\xi) = \sum_{k=1}^{n} c_k \left(\frac{|V_b^{1k}|}{|V_b|} + 2\frac{|V_b^{2k}|}{|V_b|} \right) = 1\Big(\frac{4}{11} + \frac{2 \cdot 2}{11}\Big) + 2\Big(\frac{3}{11} + \frac{2 \cdot 0}{11}\Big) + 3\Big(\frac{4}{11} + \frac{2 \cdot 2}{11}\Big) = \frac{38}{11}.$$

Это соответствует значению математического ожидания функционала задачи при прямом подсчёте:

$$\mu(\xi) = \frac{1}{11}\Big(0 + 1 + 2 + 3 + 5 + 4 + 3 + 6 + 2 + 7 + 5\Big) = \frac{38}{11}.$$

Как видно из примера, приведённые в теореме 1 формулы позволяют декомпозировать задачу на подзадачи меньшей размерности, в которых значение ограничения b меньше исходного.

Пример 2. Рассмотрим следующую задачу:

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 \to \max, \\ 2x_1 + 3x_2 + 4x_3 + 6x_4 \leqslant 6, \\ x_1, x_2, x_3, x_4 \in \{0, 1, 2, 3\}. \end{cases}$$

У этой задачи 10 допустимых решений: $\{(0000), (0001), (0010), (0100), (0200), (1000), (1010), (1100), (2000), (3000)\}$.

По формуле (13) получим

$$|V_b^{01}| = 5, \quad |V_b^{11}| = 3, \quad |V_b^{21}| = 1, \quad |V_b^{31}| = 1, \quad |V_b^{12}| = 2, \quad |V_b^{22}| = 1, \quad |V_b^{32}| = 0,$$
$$|V_b^{13}| = 2, \quad |V_b^{23}| = 0, \quad |V_b^{33}| = 0, \quad |V_b^{14}| = 2, \quad |V_b^{24}| = 0, \quad |V_b^{34}| = 0.$$

Подставляя эти значения в (16) для $k = 1$, находим число решений исходной задачи:

$$|V_b| = |V_b^{01}| + |V_b^{11}| + |V_b^{21}| + |V_b^{31}| = 5 + 3 + 1 + 1 = 10.$$

Подставляя все найденные значения в (14), находим среднее значение функционала:

$$\mu(\xi) = \sum_{k=1}^{n} c_k \left(\frac{|V_b^{1k}|}{|V_b|} + 2\frac{|V_b^{2k}|}{|V_b|} + 3\frac{|V_b^{3k}|}{|V_b|} \right) =$$
$$= 2\,\frac{3+2\cdot 1+3\cdot 1}{10} + 3\,\frac{2+2\cdot 1+3\cdot 0}{10} + 4\,\frac{2+2\cdot 0+3\cdot 0}{10} + 6\,\frac{2+2\cdot 0+3\cdot 0}{10} = \frac{48}{13}.$$

Это соответствует решению, полученному прямой подстановкой:

$$\mu(\xi) = \frac{1}{13}\Big(0+4+2+6+4+1+5+3+5+2+6+4+6\Big) = \frac{48}{13}.$$

Как показывают приведённые примеры, большинство членов в формуле (14) равны нулю, что уменьшает количество необходимых для вычислений значений.

Пример 3. Пусть $a_i = 2^{i-1}$ для всех $i = 1, 2, \ldots, n$, а $b = 2^{n+1}$, тогда решения неравенства $\sum\limits_{i=1}^{n} a_i x_i \leqslant b$, $x_i \in \{0, 1, 2\}$ — это все элементы пространства n-мерных векторов с координатами из $\{0, 1, 2\}$. Получаем $|V_b| = 3^n$.

Для любого k находим решения неравенства $\sum\limits_{i=1, i\neq k}^{n} a_i x_i \leqslant b - a_k$, $x_i \in \{0, 1, 2\}$. Это все элементы пространства $(n-1)$-мерных векторов с координатами из $\{0, 1, 2\}$, поэтому $|V_b^{1k}| = 3^{n-1}$. Рассуждая аналогично, получаем $|V_b^{2k}| = 3^{n-1}$. Отсюда среднее значение $\mu(\xi) = \sum\limits_{k=1}^{n} c_k(1/3 + 2/3) = \sum\limits_{k=1}^{n} c_k$, а максимальное значение функционала равно $\sum\limits_{k=1}^{n} 2c_k$.

3. Сюръективность в задаче о рюкзаке

Рассмотрим случай, при котором коэффициенты целевой функции и векторы ограничений совпадают, т. е. $a_i = c_i$, $i = 1, \ldots, n$.

Определение 1. Пусть $M = \{0, 1, \ldots, m\}$; M^n — множество всех n-мерных векторов с координатами из M. Назовём функцию $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} c_j x_j$ сюръективной на множестве $N \subset M^n$, если она принимает на N все значения из интервала $[f_{\min}, \ldots, f_{\max}]$, где $f_{\min}, f_{\max}$ — минимальное и максимальное значение данной функции на множестве N.

Ранее определение линейной функции с булевыми переменными $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} c_j x_j$, принимающей все значения из интервала $[0, \sum\limits_{j=1}^{n} c_j]$, было дано, например, в работе [22]. Подробный анализ свойств таких функций приведён в [23].

Примерами сюръективных функций на всём M^n при любом m являются: $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} 2^{j-1} x_j$; $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} j x_j$. Функция $2x_1 + 3x_2 + 4x_3$ при любом m не является сюръективной, поскольку она не принимает значение 1.

Теорема 2. Если $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} c_j x_j$ — сюръективная функция на V_b, где $x_i \in \{0, 1, \ldots, m\}$, $i = 1, \ldots, n$, то

$$(m+1)^n - 1 \geqslant \max f(x_1, \ldots, x_n) \geqslant \sum_{k=1}^{n} c_k \left(\frac{|V_b^{1k}|}{|V_b|} + 2\frac{|V_b^{2k}|}{|V_b|} + \ldots + m\frac{|V_b^{mk}|}{|V_b|} \right).$$

Доказательство. Пусть $C = \{c_1, c_2, \ldots, c_n\}$ и $S(C, n)$ — число различных сумм из элементов C. Очевидно, что $S(C, n) \leqslant (m+1)^n - 1$. Наименьшее значение $f(x_1, \ldots, x_n)$ равно нулю. Поскольку функция $f(x_1, \ldots, x_n)$ сюръективная, она должна принимать все значения, начиная с нуля, поэтому её максимальное значение не может превосходить числа $S(C, n)$:

$$(m+1)^n - 1 \leqslant \max f(x_1, \ldots, x_n).$$

Нижняя оценка следует из теоремы 1. ■

Пример 4. Верхняя оценка из теоремы 2 достигается для произвольного m при $c_i = (m+1)^{i-1}$, $i = 1, \ldots, n$, и $b \leqslant \sum\limits_{j=1}^{n} c_j$. В этом случае $S(C, n) = (m+1)^n - 1$ и $f(x_1, \ldots, x_n) = \sum\limits_{j=1}^{n} c_j x_j$ — сюръективная функция с максимальным значением $(m+1)^n - 1$.

Заключение

Рассмотрены вопросы, связанные с вычислением и оценкой значений функционала задачи об ограниченном рюкзаке. Приведены формулы и оценки числа допустимых решений задачи в зависимости от числа решений подзадач меньшей размерности. Метод производящих функций может быть успешно применён для анализа подобных задач, имеющих комбинаторную природу. Полученные формулы могут быть уточнены при рассмотрении задач специального вида, которые возникают в конкретных прикладных областях, в частности в математических моделях информационной безопасности, учитывающих реальные особенности исходных постановок. Изложенные результаты могут послужить базой для дальнейших исследований свойств структуры многогранников задач о рюкзаке. Найденные выражения могут быть также использованы непосредственно в вычислительных алгоритмах в качестве вспомогательных процедур.

ЛИТЕРАТУРА

1. *Kellerer H., Pferschy U., and Pisinger D.* Knapsack Problems. Berlin: Springer, 2004. 548 p.
2. *Егорычев Г. П.* Интегральное представление и вычисление комбинаторных сумм. Новосибирск: Наука, 1977. 285 с.
3. *Zhang H., Han W., Lai X., et al.* Survey on cyberspace security // Sci. China Inform. Sci. 2015. V. 58. No. 11. P. 1–43.
4. *Lyubashevsky V., Palacio A., and Segev G.* Public-key cryptographic primitives provably as secure as subset sum // LNCS. 2010. V. 5978. P. 382–400.
5. *Ranjith J. and Mahantesh K.* Blockchain-based knapsack system for security and privacy preserving to medical data // SN Comput. Sci. 2021. V. 2. `https://www.researcher-app.com/paper/7553542`.
6. *Martello S. and Toth P.* Knapsack Problems: Algorithms and Computer Implementations. N.Y.: John Wiley & Sons, 1990. 308 p.
7. *Zhang X., Huang S., Hu Y., et al.* Solving 0-1 knapsack problems based on amoeboid organism algorithm // Appl. Math. Comput. 2013. V. 219. No. 19. P. 9959–9970.
8. *Kulkarni A. J. and Shabir H.* Solving 0-1 knapsack problem using Cohort Intelligence Algorithm // Int. J. Mach. Learn. & Cyber. 2014. V. 7. P. 427–441.
9. *Rezoug A., Bader-El-Den M., and Boughaci D.* Guided genetic algorithm for the multi-dimensional knapsack problem // Memetic Computing. 2018. V. 10. P. 29–42.

10. *Kong X., Gao L., Ouyang H., and Li S.* Solving large-scale multidimensional knapsack problems with a new binary harmony search algorithm // Computers & Operations Res. 2015. V. 63. P. 7–22.
11. *Lv J., Wang X., Huang M., et al.* Solving 0-1 knapsack problem by greedy degree and expectation efficiency // Appl. Soft Comput. 2016. V. 41. P. 94–103.
12. *Chen Y. and Hao J.-K.* A "reduce and solve" approach for the multiple-choice multidimensional knapsack problem // Europ. J. Operational Res. 2014. V. 239. No. 2. P. 312–322.
13. *Voß S. and Lalla-Ruiz E.* A set partitioning reformulation for the multiple-choice multidimensional knapsack problem // Engin. Optimization. 2016. V. 48. No. 5. P. 831–850.
14. *Dang C. and Ye Y.* A fixed point iterative approach to integer programming and its distributed computation // Fixed Point Theory Appl. 2015. `https://fixedpointtheoryandalgorithms.springeropen.com/articles/10.1186/s13663-015-0429-8`.
15. *Pesant G.* Counting solutions of CSPs: A structural spproach // Proc. IJCAI'05. Edinburgh, Scotland, 2005. P. 260–265.
16. *Louveaux Q. and Weismantel R.* Polyhedral properties for the intersection of two knapsacks // Math. Program. 2008. V. 113. P. 15–37.
17. *Hojny C., Gally T., Habeck O., et al.* Knapsack polytopes: a survey // Ann. Oper. Res. 2020. V. 292. P. 469–517.
18. *Леонтьев В. К., Гордеев Э. Н.* Производящие функции в задаче о ранце // Доклады Академии наук. 2018. Т. 481. № 5. С. 478–480.
19. *Гордеев Э. Н., Леонтьев В. К.* О некоторых комбинаторных свойствах задачи о рюкзаке // Ж. вычисл. матем. и матем. физ. 2019. Т. 59. № 8. С. 1439–1447.
20. *Леонтьев В. К., Гордеев Э. Н.* О числе решений системы булевых уравнений // Автоматика и телемеханика. 2021. № 9. С. 150–168.
21. *Леонтьев В. К., Гордеев Э. Н.* О некоторых особенностях задачи разрешимости систем булевых уравнений // Вопросы кибербезопасности. 2021. № 1(41). С. 18–28.
22. *Леонтьев В. К.* О псевдобулевых полиномах // Ж. вычисл. матем. и матем. физ. 2015. Т. 55. № 11. С. 1952–1958.
23. *Леонтьев В. К., Гордеев Э. Н., Волков М. С. А.* Классическая непрерывность и ее дискретный вариант // Прикладная физика и математика. 2022. № 1. С. 31–37.

REFERENCES

1. *Kellerer H., Pferschy U., and Pisinger D.* Knapsack Problems. Berlin, Springer, 2004. 548 p.
2. *Egorychev G. P.* Integral'noe predstavlenie i vychislenie kombinatornykh summ [Integral Representation and the Computation of Combinatorial Sums]. Novosibirsk, Nauka, 1977. 285 p. (in Russian)
3. *Zhang H., Han W., Lai X., et al.* Survey on cyberspace security. Sci. China Inform. Sci., 2015. vol. 58, no. 11, pp. 1–43.
4. *Lyubashevsky V., Palacio A., and Segev G.* Public-key cryptographic primitives provably as secure as subset sum. LNCS, 2010, vol. 5978, pp. 382–400.
5. *Ranjith J. and Mahantesh K.* Blockchain-based knapsack system for security and privacy preserving to medical data. // SN Comput. Sci., 2021, vol. 2, `https://www.researcher-app.com/paper/7553542`.
6. *Martello S. and Toth P.* Knapsack Problems: Algorithms and Computer Implementations. N.Y., John Wiley & Sons, 1990. 308 p.
7. *Zhang X., Huang S., Hu Y., et al.* Solving 0-1 knapsack problems based on amoeboid organism algorithm. Appl. Math. Comput., 2013, vol. 219, no. 19, pp. 9959–9970.

8. *Kulkarni A. J. and Shabir H.* Solving 0-1 knapsack problem using Cohort Intelligence Algorithm. Int. J. Mach. Learn. & Cyber., 2014, vol. 7, pp. 427–441.
9. *Rezoug A., Bader-El-Den M., and Boughaci D.* Guided genetic algorithm for the multi-dimensional knapsack problem. Memetic Computing, 2018, vol. 10, pp. 29–42.
10. *Kong X., Gao L., Ouyang H., and Li S.* Solving large-scale multidimensional knapsack problems with a new binary harmony search algorithm. Computers & Operations Res., 2015, vol. 63, pp. 7–22.
11. *Lv J., Wang X., Huang M., et al.* Solving 0-1 knapsack problem by greedy degree and expectation efficiency. Appl. Soft Comput., 2016, vol. 41, pp. 94–103.
12. *Chen Y. and Hao J.-K.* A "reduce and solve" approach for the multiple-choice multidimensional knapsack problem. Europ. J. Operational Res., 2014, vol. 239, no. 2, pp. 312–322.
13. *Voß S. and Lalla-Ruiz E.* A set partitioning reformulation for the multiple-choice multidimensional knapsack problem. Engin. Optimization, 2016, vol. 48, no. 5, pp. 831–850.
14. *Dang C. and Ye Y.* A fixed point iterative approach to integer programming and its distributed computation. Fixed Point Theory Appl., 2015, `https://fixedpointtheoryand algorithms.springeropen.com/articles/10.1186/s13663-015-0429-8`.
15. *Pesant G.* Counting solutions of CSPs: A structural spproach, Proc. IJCAI'05, Edinburgh, Scotland, 2005, pp. 260–265.
16. *Louveaux Q. and Weismantel R.* Polyhedral properties for the intersection of two knapsacks. Math. Program., 2008, vol. 113, pp. 15–37.
17. *Hojny C., Gally T., Habeck O., et al.* Knapsack polytopes: a survey. Ann. Oper. Res., 2020, vol. 292, pp. 469–517.
18. *Leont'ev V. K. and Gordeev E. N.* Proizvodyashchie funktsii v zadache o rantse [The generating functions in the knapsack problem]. Doklady Akademii Nauk, 2018, vol. 481, no. 5, pp. 478–480. (in Russian)
19. *Gordeev E. N. and Leont'ev V. K.* On combinatorial properties of the knapsack problem. Comput. Math. Math. Phys., 2019, vol. 59, no. 8, pp. 1380–1388.
20. *Leontiev V. K. and Gordeev E. N.* On the number of solutions to a system of Boolean equations. Autom. Remote Control, 2021, vol. 82, no. 9, pp. 1581–1596.
21. *Leont'ev V. K. and Gordeev E. N.* O nekotorykh osobennostyakh zadachi razreshimosti sistem bulevykh uravneniy [On some features of the problem of solvability of Boolean equations systems]. Voprosy Kiberbezopasnosti, 2021, no. 1(41), pp. 18–28. (in Russian)
22. *Leontiev V. K.* On pseudo-Boolean polynomials. // Comput. Math. Math. Phys., 2015, vol. 55, no. 11, pp. 1926–1932.
23. *Leont'ev V. K., Gordeev E. N., and Volkov M. S. A.* Klassicheskaya nepreryvnost' i ee diskretnyy variant [Classical continuity and its discrete variant]. Prikladnaya Fizika i Matematika, 2022, no. 1, pp. 31–37. (in Russian)

СВЕДЕНИЯ ОБ АВТОРАХ

АХМЕТЗЯНОВА Лилия Руслановна — кандидат физико-математических наук, заместитель начальника отдела криптографических исследований ООО «КРИПТО-ПРО», г. Москва. E-mail: **lah@cryptopro.ru**

БАБУЕВА Александра Алексеевна — ведущий инженер-аналитик отдела криптографических исследований ООО «КРИПТО-ПРО», г. Москва.
E-mail: **babueva@cryptopro.ru**

БОЖКО Андрей Алексеевич — инженер-аналитик ООО «КРИПТО-ПРО», г. Москва. E-mail: **bozhko@cryptopro.ru**

ВОЛКОВ Мария Сабина Александровна — аспирантка Московского государственного технического университета им. Н. Э. Баумана, г. Москва.
E-mail: **sabina-volkoff@yandex.ru**

ЖАРКОВА Анастасия Владимировна — кандидат физико-математических наук, доцент кафедры теоретических основ компьютерной безопасности и криптографии Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского, г. Саратов. E-mail: **ZharkovaAV3@gmail.com**

КИРЮХИН Виталий Александрович — главный специалист ООО «СФБ Лаб», АО «ИнфоТеКС», г. Москва. E-mail: **vitaly.kiryukhin@sfblaboratory.ru**

КУНИНЕЦ Артем Андреевич — студент специальности «Компьютерная безопасность» ОНК «Институт высоких технологий» БФУ им. И. Канта, г. Калининград.
E-mail: **artkuninets@yandex.ru**

ЛОСЕВ Александр Сергеевич — кандидат физико-математических наук, доцент, старший научный сотрудник ИПМ ДВО РАН, г. Владивосток.
E-mail: **A.S.Losev@yandex.ru**

МАЛЫГИНА Екатерина Сергеевна — кандидат физико-математических наук, доцент департамента прикладной математики МИЭМ НИУ ВШЭ, г. Москва.
E-mail: **emalygina@hse.ru**

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**

СЕРГЕЕВ Андрей Михайлович — старший специалист ООО «СФБ Лаб», г. Москва. E-mail: **andrey.sergeev@sfblaboratory.ru**

ШАПОРЕНКО Александр Сергеевич — аспирант Новосибирского государственного университета, г. Новосибирск. E-mail: **shaporenko.alexandr@gmail.com**

Журнал «Прикладная дискретная математика» входит в перечень ВАК рецензируемых научных изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание учёной степени кандидата и доктора наук по специальностям 2.3.5. «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей» (технические науки), 2.3.6. «Методы и системы защиты информации, информационная безопасность» (физико-математические и технические науки), 1.1.5. «Математическая логика, алгебра, теория чисел и дискретная математика» (физико-математические науки), 1.2.3. «Теоретическая информатика, кибернетика» (физико-математические науки), а также в перечень журналов, рекомендованных ФУМО ВО ИБ в качестве учебной литературы по специальности «Компьютерная безопасность».

Журнал индексируется в базах данных Web of Science (Emerging Sources Citation Index (ESCI) и Russian Science Citation Index (RSCI)), Scopus, MathSciNet и Zentralblatt MATH. По решению ВАК от 21.12.2023 он отнесён к первой категории (К1) научных журналов, входящих в Перечень ВАК.

Журнал «Прикладная дискретная математика» распространяется по подписке; его подписной индекс 38696 в объединённом каталоге «Пресса России». Полнотекстовые электронные версии вышедших номеров журнала доступны на его сайте journals.tsu.ru/pdm и на Общероссийском математическом портале www.mathnet.ru. На сайте журнала можно найти также правила подготовки рукописей статей для публикации в журнале.

Тематика публикаций журнала:

- *Теоретические основы прикладной дискретной математики*
- *Математические методы криптографии*
- *Математические методы стеганографии*
- *Математические основы компьютерной безопасности*
- *Математические основы надёжности вычислительных и управляющих систем*
- *Прикладная теория кодирования*
- *Прикладная теория автоматов*
- *Прикладная теория графов*
- *Логическое проектирование дискретных автоматов*
- *Математические основы информатики и программирования*
- *Вычислительные методы в дискретной математике*
- *Математические основы интеллектуальных систем*
- *Исторические очерки по дискретной математике и её приложениям*