МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА

Приложение

№ 15 **Сентябрь 2022**

Зарегистрирован в Федеральной службе по надзору
в сфере связи, информационных технологий и массовых коммуникаций
(Роскомнадзор)

Свидетельство о регистрации ПИ № ФС 77-50702 от 17 июля 2012 г.

ТРУДЫ
XXI Международной конференции
«Сибирская научная школа-семинар
“Компьютерная безопасность и криптография” — SIBECRYPT’22»
имени Г. П. Агибалова
(Красноярск, 5–10 сентября 2022 г.)

УЧРЕДИТЕЛЬ
Томский государственный университет

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

XXI Международная конференция «Сибирская научная школа-семинар "Компьютерная безопасность и криптография" — SIBECRYPT'22» имени Г. П. Агибалова проведена Сибирским государственным университетом науки и технологий имени академика М. Ф. Решетнёва, Новосибирским государственным университетом и Международным математическим центром в Академгородке в сотрудничестве с Томским государственным университетом и Академией криптографии РФ с 5 по 10 сентября 2022 г. в г. Красноярске при финансовой поддержке Международного математического центра в Академгородке (***соглашение с Министерством науки и высшего образования РФ № 075-15-2022-281***) ***и Северо-Западного центра математических исследований имени Софьи Ковалевской*** (***соглашение с Министерством науки и высшего образования РФ № 075-02-2022-872***)***.***

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

Подписано к печати 15.08.2022. Формат $60 \times 84\frac{1}{8}$. Усл. п. л. 15,24. Тираж 300 экз.
Заказ № 5121. Цена свободная. Дата выхода в свет 18.08.2022.

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

СОДЕРЖАНИЕ

Секция 1

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

Секция 2

ДИСКРЕТНЫЕ ФУНКЦИИ

Секция 3

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

Секция 4

МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ, ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

Секция 5

ПРИКЛАДНАЯ ТЕОРИЯ КОДИРОВАНИЯ И ГРАФОВ

Секция 1

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

УДК 519.7 DOI 10.17223/2226308X/15/1

ОБ ИНВАРИАНТНЫХ ПОДПРОСТРАНСТВАХ ФУНКЦИЙ, АФФИННО ЭКВИВАЛЕНТНЫХ ОБРАЩЕНИЮ ЭЛЕМЕНТОВ КОНЕЧНОГО ПОЛЯ[1]

Н. А. Коломеец, Д. А. Быков

Рассматриваются аффинные подпространства над $\mathbb{F}_p$ конечного поля $\mathbb{F}_{p^n}$, p — простое, образ которых под действием функции x^{-1}, обращающей элемент x поля (считаем, что $0^{-1} = 0$), также является аффинным подпространством. Доказано, что образ аффинного подпространства U, $|U| > 2$, является аффинным подпространством, если и только если $U = q\mathbb{F}_{p^k}$, где $q \in \mathbb{F}_{p^n}^*$ и $k|n$. Другими словами, все такие подпространства выражаются через подполя поля $\mathbb{F}_{p^n}$. В качестве следствия предложено достаточное условие, при котором функция $A(x^{-1}) + b$ не имеет инвариантных аффинных подпространств U мощности $2 < |U| < p^n$, где $A : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ — обратимое линейное преобразование, $b \in \mathbb{F}_{p^n}^*$. Приведены примеры функции, у которых, исключая само $\mathbb{F}_{p^n}$, отсутствуют инвариантные аффинные подпространства.

Ключевые слова: *конечные поля, обратный элемент, аффинные подпространства, инвариантные подпространства.*

В работе рассматриваются аффинные подпространства, образ которых под действием функции, обращающей элементы конечного поля, также является аффинным подпространством. Обозначим эту функцию через σ, т. е. $\sigma : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$, такая, что

$$\sigma(x) = \begin{cases} x^{-1}, & \text{если } x \neq 0, \\ 0, & \text{если } x = 0, \end{cases}$$

где $\mathbb{F}_{p^n}$ — конечное поле, состоящее из p^n элементов. Здесь и далее p — простое число. Под *линейным подпространством* L поля $\mathbb{F}_{p^n}$ мы подразумеваем его линейное подпространство над $\mathbb{F}_p$, другими словами, аддитивную подгруппу $\mathbb{F}_{p^n}$. Через $a+L$ обозначим *аффинное подпространство*. Отметим, что для любого множества $S \subseteq \mathbb{F}_{p^n}$, элемента $a \in \mathbb{F}_{p^n}$ и функции $f : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$

$$a + S = \{a + s : s \in S\},\ aS = \{as : s \in S\} \text{ и } f(S) = \{f(s) : s \in S\}.$$

Известно, что у $\mathbb{F}_{p^n}$ существует подполе $\mathbb{F}_{p^k}$ тогда и только тогда, когда $k|n$. В этом случае будем полагать, что $\mathbb{F}_{p^k} \subseteq \mathbb{F}_{p^n}$.

Рассмотрим аффинные подпространства $U \subseteq \mathbb{F}_{p^n}$, такие, что $\sigma(U)$ также является аффинным подпространством. Выделим *нетривиальные* подпространства, такие, что

[1]Работа выполнена в рамках госзадания ИМ СО РАН (проект № FWNF–2022–0018).

$2 < |U| < p^n$, поскольку любое множество мощности 1, 2 (если $p = 2$) или p^n является аффинным подпространством.

Интерес к этим подпространствам обусловлен их связью с инвариантными подпространствами отображений. Знание всех таких U позволяет определить все инвариантные аффинные подпространства относительно функций, аффинно эквивалентных σ. Напомним, что множество $S \subseteq \mathbb{F}_{p^n}$ называется *инвариантным* относительно f, если $f(S) \subseteq S$. Функция σ, на базе которой построен S-блок AES [1, 2], а также её инвариантные подпространства интересны в криптографическом контексте. Например, они использовались для исследования групповых свойств раундовых функций у AES-подобных шифров [3 – 5]. В [6] предложена атака, использующая инвариантное подпространство для нескольких раундов шифрования, которая к настоящему времени рассмотрена для многих схем [7]. Известна также более общая атака [8], построение которой можно начинать с S-блоков [9]. Отметим, что все линейные подпространства, являющиеся инвариантными относительно σ, описаны в работах [10, 11]. Таким образом, мы обобщаем эти результаты, во-первых, на аффинные подпространства и, во-вторых, на подпространства, являющиеся инвариантными не для σ, а для некоторой функции, аффинно эквивалентной σ.

Если $f(U)$ не является аффинным подпространством для любого аффинного подпространства $U \subseteq \mathbb{F}_{2^n}$ размерности 2, то f принадлежит к классу APN-функций [12], имеющих большую криптографическую значимость. Более того, только APN-функции обладают этим свойством. С ними связано множество открытых вопросов [13].

Сначала покажем, что для функции σ достаточно рассматривать только линейные подпространства.

Теорема 1. Пусть U и $\sigma(U)$ — аффинные подпространства $\mathbb{F}_{p^n}$, $|U| > 2$. Тогда U является линейным подпространством $\mathbb{F}_{p^n}$.

Следующая теорема описывает все искомые подпространства U.

Теорема 2. Пусть U — аффинное подпространство $\mathbb{F}_{p^n}$, $|U| > 2$. Тогда $\sigma(U)$ является аффинным подпространством $\mathbb{F}_{p^n}$, если и только если

$$U = q\mathbb{F}_{p^k}, \text{ где } k|n \text{ и } q \in \mathbb{F}_{p^n}^*.$$

Доказательство теоремы использует тождество Хуа [14] аналогично рассмотренному в [10, 11] случаю линейных подпространств, инвариантных относительно σ. Теорема 2 также показывает, что при простых $n > 2$ существуют APN-функции, такие, что образ нетривиального аффинного подпространства (любой размерности, а не только размерности 2) никогда не является аффинным подпространством.

Покажем, что теоремы 1 и 2 позволяют построить с помощью σ функции без нетривиальных инвариантных подпространств, имеющие вид

$$\sigma_{A,b}(x) = A(\sigma(x)) + b,$$

где $A : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ — обратимая линейная функция; $b \in \mathbb{F}_{p^n}^*$ и $x \in \mathbb{F}_{p^n}$. Рассматривая линейные отображения, мы, аналогично подпространствам, подразумеваем, что $\mathbb{F}_{p^n}$ является векторным пространством над полем $\mathbb{F}_p$. Определим также подмножество поля

$$\mathcal{S}(\mathbb{F}_{p^n}) = \{x \in \mathbb{F}_{p^n} : x \notin \mathbb{F}_{p^k}, \text{ где } k|n \text{ и } k < n\}.$$

Другими словами, оно содержит все элементы $\mathbb{F}_{p^n}$, не лежащие в его подполях, и всегда является непустым.

Теорема 3. Пусть $A : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ — обратима и линейна, $b \in \mathbb{F}_{p^n}^*$, причём

$$b^{-1}\sigma_{A,b}(b) \in \mathcal{S}(\mathbb{F}_{p^n}). \tag{1}$$

Тогда не существует инвариантных аффинных подпространств U относительно $\sigma_{A,b}$, таких, что $2 < |U| < p^n$.

Даже если $\sigma_{A,b}$ не удовлетворяет условию (1), можно привести её к нужному виду.

Следствие 1. Пусть $A : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ — обратима и линейна, $b \in \mathbb{F}_{p^n}^*$. Определим функцию $A' : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ следующим образом:

$$A'(x) = \alpha\beta A(x), \text{ где } \alpha \in \mathcal{S}(\mathbb{F}_{p^n}) \text{ и } \beta = (b^{-1}A(b^{-1}))^{-1}.$$

Тогда функция $\sigma_{A',b}$ удовлетворяет условию (1).

Отметим, что S-блок $S_{\texttt{AES}}$ шифра AES, имеющий вид $\sigma_{A,b}$ в поле $\mathbb{F}_{2^8}$, удовлетворяет условию (1) теоремы 3. Тем не менее существует аффинное подпространство размерности 1, инвариантное относительно $S_{\texttt{AES}}$ [1, табл. 7]:

$$S_{\texttt{AES}}(\texttt{0x73}) = \texttt{0x8F} \text{ и } S_{\texttt{AES}}(\texttt{0x8F}) = \texttt{0x73}.$$

Таким образом, утверждения выше не гарантируют отсутствия инвариантных подпространств U, где $1 \leqslant |U| \leqslant 2$. В дополнение к примеру выше, функция вида $\sigma_{A,b}$ может удовлетворять условию (1), но при этом иметь неподвижные точки (т. е. инвариантные аффинные подпространства размерности 0). Следующая теорема гарантирует существование функций, единственным инвариантным подпространством которых является $\mathbb{F}_{p^n}$.

Теорема 4. Пусть $\sigma_{\alpha,b}(x) = \alpha\sigma(x) + b$, где $\alpha, b \in \mathbb{F}_{p^n}^*$, $x \in \mathbb{F}_{p^n}$. Тогда $\sigma_{\alpha,b}$ не имеет инвариантных аффинных подпространств, кроме $\mathbb{F}_{p^n}$, если и только если

— $\alpha b^{-2} \in M_2$ при $p = 2$, где

$$M_2 = \{x \in \mathcal{S}(\mathbb{F}_{2^n}) : \mathrm{tr}(x) = 1\},\ \mathrm{tr}(x) = x^{2^0} + x^{2^1} + \ldots + x^{2^{n-1}}; \tag{2}$$

— $\alpha b^{-2} + 4^{-1} \in M_p$ при $p \neq 2$, где

$$M_p = \{x \in \mathcal{S}(\mathbb{F}_{p^n}) : x \neq y^2 \text{ для любого } y \in \mathbb{F}_{p^n}\}. \tag{3}$$

Множества M_p, определённые в (2) и (3), всегда непустые. Однако функции $\sigma_{\alpha,b}$ из теоремы 4 имеют очень простую алгебраическую структуру. Заметим также, что при непростых n всегда существует некоторое линейное подпространство $L \neq \mathbb{F}_{p^n}$, такое, что $\sigma_{\alpha,b}(u + L) = v + L$ для некоторых $u, v \in \mathbb{F}_{p^n}$.

ЛИТЕРАТУРА

1. `http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf`. FIPS Publ. 197. Advanced Encryption Standard. 2001.
2. *Daemen J. and Rijmen V.* The Design of Rijndael: AES — the Advanced Encryption Standard. Springer Verlag, 2002.
3. *Caranti A., Volta F., and Sala M.* Imprimitive permutations groups generated by the round functions of key-alternating block ciphers and truncated differential cryptanalysis. `https://arxiv.org/abs/math/0606022`. 2006.

4. *Caranti A., Volta F., and Sala M.* An application of the O'Nan-Scott theorem to the group generated by the round functions of an AES-like cipher // Des. Codes Cryptogr. 2009. V. 52. P. 293–301.
5. *Caranti A., Volta F., and Sala M.* On some block ciphers and imprimitive groups // Appl. Algebra Eng. Commun. Comput. 2009. V. 20. P. 339–350.
6. *Leander G., Abdelraheem M. A., AlKhzaimi H., and Zenner E.* A cryptanalysis of PRINTcipher: The invariant subspace attack // LNCS. 2011. V. 6841. P. 206–221.
7. *Трифонов Д. И., Фомин Д. Б.* Об инвариантных подпространствах в XSL-шифрах // Прикладная дискретная математика. 2021. № 54. С 58–76.
8. *Todo Y., Leander G., and Sasaki Y.* Nonlinear invariant attack: practical attack on full SCREAM, iSCREAM, and Midori64 // ASIACRYPT 2016. LNCS. 2016. V. 10032. P. 3–33.
9. *Буров Д. А.* О существовании нелинейных инвариантов специального вида для раундовых преобразований XSL-алгоритмов // Дискретная математика. 2021. Т. 33. № 2. С. 31–45.
10. *Mattarei S.* Inverse-closed additive subgroups of fields // Israel J. Math. 2007. V. 159. P. 343–347.
11. *Goldstein D., Guralnick R., Small L., and Zelmanov E.* Inversion-invariant additive subgroups of division rings // Pacific J. Math. 2006. V. 227. P. 287–294.
12. *Nyberg K.* Differentially uniform mappings for cryptography // LNCS. 1994. V. 765. P. 55–64.
13. *Carlet C.* Open questions on nonlinearity and on APN Functions // LNCS. 2015. V. 9061. P. 83–107.
14. *Hua L.-K.* Some properties of a sfield // Proc. NAS USA. 1949. V. 35. P. 533–537.

УДК 519.214 DOI 10.17223/2226308X/15/2

ОБ АСИМПТОТИЧЕСКОЙ НОРМАЛЬНОСТИ ЧИСЛА КРАТНЫХ СОВПАДЕНИЙ ЦЕПОЧЕК В ПОЛНЫХ q-ИЧНЫХ ДЕРЕВЬЯХ И ЛЕСАХ СО СЛУЧАЙНЫМИ МЕТКАМИ

В. Г. Михайлов, В. И. Круглов

Рассматриваются полные q-ичные корневые деревья высоты H, каждой вершине которых независимо от остальных вершин присвоена случайная метка, выбираемая из множества $\{1, 2, \ldots, N\}$. Исследуются случайные величины, равные числу наборов из $r \geqslant 2$ путей одинаковой длины s, у которых совпадают соответствующие s-цепочки меток вершин. Представлена теорема о достаточных условиях асимптотической нормальности рассматриваемых случайных величин при неограниченном увеличении высоты дерева. При исследовании повторений цепочек в лесе деревьев предполагается, что имеется r деревьев, которые могут иметь разные высоты $H_1, \ldots, H_r$ и вершинам которых аналогичным образом поставлены в соответствие независимые в совокупности случайные метки. Изучается число наборов из r путей длины s, в которые входит по одному пути с каждого дерева, для которых совпадают соответствующие цепочки меток вершин, для этой случайной величины также получены достаточные условия асимптотической нормальности.

Ключевые слова: *деревья с метками, цепочки меток на дереве, повторения цепочек, условия асимптотической нормальности.*

Введение

Повторения событий могут свидетельствовать о наличии закономерностей, при анализе которых возникают задачи о вычислении или оценке значений вероятностей повторений событий в наборах независимых случайных величин. В исследованиях по этой тематике первоначально рассматривались задачи о повторениях цепочек случайных символов [1 – 4], естественным продолжением этих исследований стали работы, связанные с повторениями паттернов в деревьях со случайно помеченными вершинами. Распределения числа вхождений заданного поддерева в случайное дерево рассматривались в [5, 6], задачи такого рода возникают в компьютерных науках [7, 8] при анализе алгоритмов или, например, в связи с древовидной структурой XML-документов, которые используются, в частности, на портале Госуслуг. Подобные задачи также могут возникать в связи с построением статистических критериев и анализом генетических последовательностей.

Предельные пуассоновские теоремы для числа совпадений меток цепочек в двоичном или q-ичном дереве, метки вершин которого независимы и имеют равновероятное распределение на конечном алфавите, получены в [9, 10], предельная пуассоновская теорема для числа совпадений паттернов в q-ичном дереве с равновероятными метками вершин доказана в [4].

В настоящей работе рассматриваются полные q-ичные корневые деревья и леса, составленные из таких корневых деревьев. Некорневым вершинам деревьев присвоены случайные метки, выбранные независимо из множества $\{1, 2, \ldots, N\}$ в соответствии с некоторым вероятностным распределением.

Изучается число наборов по $r \geqslant 2$ путей длины s на одном или нескольких деревьях, для которых совпадают соответствующие цепочки меток вершин. Получены достаточные условия асимптотической нормальности этой случайной величины при неограниченном увеличении высоты деревьев.

1. Повторения цепочек на дереве

Пусть $Tr(H)$ — полное q-ичное корневое дерево высоты H и вершинам этого дерева присвоены незавимые в совокупности случайные метки, выбираемые из конечного множества $\{1, 2, \ldots, N\}$ в соответствии с положительными вероятностями $p_1, \ldots, p_N$, где $p_1 + \ldots + p_N = 1$.

Пусть $s < H$. Через $W(H, s)$ будем обозначать множество цепочек длины s в дереве $Tr(H)$, начало которых имеет высоту, не превосходящую $H - s$. Нетрудно показать [11], что

$$|W(H, s)| = \frac{q^{H-s+1} - 1}{q - 1}\, q^s = \frac{q^{H+1} - q^s}{q - 1}.$$

Определим случайную величину $\xi_r(H, s)$, которая равна числу всех таких наборов из r различных путей длины s в дереве $Tr(H)$, для которых совпадают соответствующие s-цепочки меток вершин, составляющих эти пути. Для этого занумеруем элементы множества $W(H, s)$ числами от 1 до $|W(H, s)|$, путь с номером u, где $1 \leqslant u \leqslant |W(H, s)|$, будем обозначать ω_u, а цепочку меток вершин на этом пути — $Y(\omega_u)$. Тогда

$$\xi_r(H, s) = \sum_{1 \leqslant u_1 < \ldots < u_r \leqslant |(W(H,s)|} I\{Y(\omega_{u_1}) = \ldots = Y(\omega_{u_r})\}.$$

Теорема 1. Пусть $H \to \infty$ и параметры $s = s(H)$ и $q = q(H)$ изменяются так, что $s/H \to 0$. Пусть существуют такие числа $C > 0$ и $\varepsilon \in (0, 1]$, что при всех достаточно

больших H выполнено неравенство

$$\mathbf{D}\xi_r(H,s) \geqslant C\left(\frac{q^{H+1}-q^s}{q-1}\right)^{2(r-1)+\varepsilon}. \tag{1}$$

Тогда функции распределения и моменты случайной величины

$$\tilde{\xi}_r(H,s) = \frac{\xi_r(H,s) - \mathbf{E}\xi_r(H,s)}{\sqrt{\mathbf{D}\xi_r(H,s)}}$$

сходятся к функции распределения и моментам стандартного нормального распределения.

Доказательство теоремы 1 для случая $r = 2$ опубликовано в [11].

Можно отметить, что при $s = 1$ величина $\xi_r(H,1)$ совпадает по распределению с числом ξ_r наборов по r одинаковых исходов в последовательности из $|W(H,1)| - 1$ независимых случайных величин X_i, принимающих значения на множестве $\{1,\ldots,N\}$ с вероятностями $\mathsf{P}[X_i = k] = p_k > 0$, $k = 1,\ldots,N$, $\sum\limits_{k=1}^{N} p_k = 1$. Свойства распределения величины ξ_r известны: для неё условие (1) выполняется для любых неравновероятных распределений величин X_i и не выполняется, если $p_1 = \ldots = p_N = 1/N$.

2. Повторения цепочек в лесах

Рассмотрим набор из r полных q-ичных корневых деревьев $Tr_1(H_1),\ldots,Tr_r(H_r)$ высот $H_1,\ldots,H_r$ соответственно, и пусть вершинам этих деревьев присвоены независимые в совокупности случайные метки, выбираемые из множества $\{1,2,\ldots,N\}$ в соответствии с положительными вероятностями $p_1,p_2,\ldots,p_N$, где $p_1+p_2+\ldots+p_N = 1$.

Пусть случайная величина $\xi_{(r)}(H_1,\ldots,H_r;s)$ равна числу таких наборов из r путей длины s, что в эти наборы входит по одному пути из каждого из деревьев $Tr_1(H_1),\ldots,Tr_r(H_r)$ и для этих путей совпадают соответствующие s-цепочки меток вершин. Тогда

$$\xi_{(r)}(H_1,\ldots,H_r;s) = \sum_{\omega_{u_1}\in W(H_1,s)} \ldots \sum_{\omega_{u_r}\in W(H_r,s)} I\{Y(\omega_{u_1}) = \ldots = Y(\omega_{u_r})\}.$$

Минимальную высоту деревьев $Tr_1(H_1),\ldots,Tr_r(H_r)$ будем обозначать через $H_{\min} = \min\{H_1,\ldots,H_r\}$.

Для любого $l \in \mathbb{N}$ определим величину $P_l = \sum\limits_{k=1}^{N} p_k^l$, которая равна вероятности того, что l различных вершин, лежащих в одном или нескольких деревьях, имеют одинаковые метки.

Теорема 2. Пусть $H_1,\ldots,H_r \to \infty$ и параметры $s = s(H_1,\ldots,H_r)$ и $q = q(H_1,\ldots,H_r)$ изменяются так, что $s/H_{\min} \to 0$. Пусть существуют такие числа $C > 0$ и $\varepsilon \in (0,1]$, что при всех достаточно больших $H_{\min}$ выполнено неравенство

$$\mathbf{D}\xi_{(r)}(H_1,\ldots,H_r;s) \geqslant Cq^{2(H_1+\ldots+H_r)-(2-\varepsilon)H_{\min}}.$$

Тогда функции распределения и моменты случайной величины

$$\tilde{\xi}_{(r)}(H_1,\ldots,H_r;s) = \frac{\xi_{(r)}(H_1,\ldots,H_r;s) - P_r^s \prod\limits_{k=1}^{r} \dfrac{q^{H_k+1}-q^s}{q-1}}{\sqrt{\mathbf{D}\xi_{(r)}(H_1,\ldots,H_r;s)}}$$

сходятся к функции распределения и моментам стандартного нормального распределения.

Доказательства теорем 1 и 2 основаны на модификации метода Янсона [12], предложенной в работе В. Г. Михайлова [13].

ЛИТЕРАТУРА

1. *Guibas L. J. and Odlyzko A. M.* Long repetitive patterns in random sequences // Z. Wahrscheinlichkeitstheorie verw. Geb. 1980. No. 1. P. 241–262.
2. *Зубков А. М., Михайлов В. Г.* Предельные распределения случайных величин, связанных с длинными повторениями в последовательности независимых испытаний // Теория вероятн. и ее примен. 1974. Т. 19. № 1. С. 173–181.
3. *Михайлов В. Г.* Оценка точности сложной пуассоновской аппроксимации для распределения числа совпадающих цепочек // Теория вероятн. и ее примен. 2001. Т. 46. № 4. С. 713–723.
4. *Kruglov V. and Zubkov A.* Number of pairs of template matchings in q-ary tree with randomly marked vertices // LNCS. 2017. V. 10684. P. 336–346.
5. *Hoffmann C. M. and O'Donnell M. J.* Pattern matching in trees // J. ACM. 1982. V. 29. No. 1. P. 68–95.
6. *Steyaert J.-M. and Flajolet P.* Patterns and pattern-matching in trees: an analysis // Inf. & Control. 1983. V. 58. No. 1. P. 19–58.
7. *Singh G., Smolka S. A., and Ramakrishnan I. V.* Distributed algorithms for tree pattern matching // LNCS. 1988. V. 312. P. 92–107.
8. *Tahraoui M. A., Pinel-Sauvagnat K., Laitang C., et al.* A survey on tree matching and XML retrieval // Computer Science Rev. 2013. No. 8. P. 1–23.
9. *Зубков А. М., Круглов В. И.* Повторения цепочек на бинарном дереве со случайными метками вершин // Дискретная математика. 2015. Т. 27. № 4. С. 38–48.
10. *Kruglov V. I.* On coincidences of tuples in a q-ary tree with random labels of vertices // Discr. Math. Appl. 2018. V. 28. No. 5. P. 293–307.
11. *Михайлов В. Г., Круглов В. И.* Об асимптотической нормальности в задаче о повторениях цепочек в помеченном полном дереве // Матем. вопр. криптогр. 2021. Т. 12. № 4. С. 59–64.
12. *Janson S.* Normal convergence by higher semiinvariants with applications to sums of dependent random variables and random graphs // Ann. Probab. 1988. V. 16. No. 1. P. 306–312.
13. *Михайлов В. Г.* Об одной теореме Янсона // Теория вероятн. и ее примен. 1991. Т. 36. № 1. С. 168–170.

УДК 519.214 DOI 10.17223/2226308X/15/3

О ТОЧНОСТИ НОРМАЛЬНОЙ АППРОКСИМАЦИИ ДЛЯ РАСПРЕДЕЛЕНИЯ ЧИСЛА КРАТНЫХ ПОВТОРЕНИЙ ЗНАКОВ В СТАЦИОНАРНОЙ СЛУЧАЙНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ

В. Г. Михайлов, Н. М. Меженная

Изучается задача об асимптотической нормальности числа r-кратных повторений знаков в отрезке длины n стационарной в узком смысле случайной последовательности со значениями в конечном множестве, удовлетворяющей условию равномерно сильного перемешивания. Показано, что если существует такое число $\alpha > 0$, что коэффициент равномерно сильного перемешивания $\varphi(t)$ убывает

как $t^{-6-\alpha}$, то расстояние в равномерной метрике между функцией распределения стандартизованного числа повторений и функцией распределения стандартного нормального закона с увеличением длины отрезка последовательности n убывает со скоростью $O(n^{-\delta})$ для любого $\delta \in (0, \alpha(32+4\alpha)^{-1})$.

Ключевые слова: *кратные повторения, зависимые случайные величины, равномерно сильное перемешивание, нормальная аппроксимация, оценка скорости сходимости.*

С развитием вычислительной техники всё бóльшую роль играет метод статистического моделирования изучаемых природных явлений и теоретических построений. Этот метод предполагает генерацию последовательностей случайных или псевдослучайных чисел, что делает необходимой проверку соответствия их свойств требованиям модели. Особенно важно это для криптографических применений. Для такой проверки в криптографической литературе разработаны наборы статистических тестов [1, 2]. Обоснование этих тестов использует свойства последовательностей независимых случайных величин.

Теоретические исследования дискретных случайных последовательностей вышли за пределы этих чисто утилитарных потребностей. В частности, сформировалось отдельное научное направление, изучающее предельное поведение распределения числа повторений, в том числе кратных, в независимых и конечно зависимых последовательностях случайных величин. Наиболее успешными оказались исследования условий сходимости распределений чисел повторений к распределению Пуассона [3, 4]. В качестве примера построения достаточных условий асимптотической нормальности числа повторений следует привести работу А. М. Шойтова [5].

В последние годы активно велись исследования повторяемости знаков в дискретных цепях Маркова. Например, в работах [6 – 8] исследована возможность аппроксимации распределения чисел повторений цепочек в цепи Маркова распределением Пуассона. Аналогичная задача о доказательстве центральной предельной теоремы для таких случайных величин решена в [9]. В частности, в [9, теорема 3] установлена оценка скорости сходимости к нормальному закону для числа кратных совпадений знаков в конечной стационарной цепи Маркова. В настоящей работе показано, что вместо цепей Маркова можно рассматривать более широкий класс — стационарные случайные последовательности, удовлетворяющие условию равномерно сильного перемешивания.

Пусть $X_1, \ldots, X_n$ — отрезок стационарной (в узком смысле) случайной последовательности со значениями из множества $\{1, \ldots, N\}$, стационарным распределением $\mathsf{P}[X_1 = k] = p_k$, $k = 1, \ldots, N$, $p_1 + \ldots + p_N = 1$, удовлетворяющей условию равномерно сильного перемешивания.

Напомним, что процесс $\{X_t\}_{-\infty}^{\infty}$ называется стационарным (в узком смысле), если распределения случайных векторов вида $(X_{t_1+h}, \ldots, X_{t_s+h})$ при всех натуральных s не зависят от h. Стационарная последовательность $\{X_t\}_{-\infty}^{\infty}$ обладает свойством *равномерно сильного перемешивания*, если

$$\varphi(t) = \sup_{A \in \mathcal{F}_{-\infty}^{0}, B \in \mathcal{F}_{t}^{\infty}} |\mathsf{P}[B|A] - \mathsf{P}[B]| \downarrow 0, \quad t \to \infty,$$

где $\mathcal{F}_a^b$ — σ-алгебра событий, порождённая величинами $X_a, \ldots, X_b$. Величину $\varphi(t)$ называют коэффициентом равномерно сильного перемешивания.

Рассмотрим случайную величину

$$\xi_r = \sum_{1 \leqslant j_1 < \ldots < j_r \leqslant n} I\{X_{j_1} = \ldots = X_{j_r}\} \tag{1}$$

— число r-кратных повторений знаков в отрезке $X_1, \ldots, X_n$.

Для исследования распределения случайной величины (1) определим случайные величины

$$\zeta_k = \sum_{t=1}^{n} I\{X_t = k\}, \quad k = 1, \ldots, N,$$

— числа одинаковых знаков в $X_1, \ldots, X_n$, и пусть $\eta_k = \zeta_k - np_k$. Обозначим

$$U_n = \frac{1}{(r-1)!} \sum_{k=1}^{N} p_k^{r-1} \eta_k,$$

$$\xi_r^* = \frac{\xi_r}{n^{r-1}\sqrt{\mathbf{D}U_n}} - \frac{1}{n^{r-1} r! \sqrt{\mathbf{D}U_n}} \sum_{k=1}^{N} \left(\prod_{j=0}^{r-1} (np_k - j) \right).$$

Теорема 1. Пусть $X_1, \ldots, X_n$ — отрезок стационарной (в узком смысле) случайной последовательности со значениями из множества $\{1, \ldots, N\}$, стационарным распределением $\mathsf{P}[X_1 = k] = p_k > 0$, $k = 1, \ldots, N$, $p_1 + \ldots + p_N = 1$ (причём среди вероятностей $p_1, \ldots, p_N$ есть различные), удовлетворяющей условию равномерно сильного перемешивания с коэффициентом φ, таким, что при некотором $\alpha > 0$ выполнено условие

$$\varphi(t) \leqslant t^{-(6+\alpha)}.$$

Тогда для любого $0 < \delta < \dfrac{\alpha}{4(8+\alpha)}$

$$\sup_{-\infty < x < \infty} |\mathsf{P}[\xi_r^* \leqslant x] - \Phi(x)| = O\left(n^{-\delta}\right), \quad n \to \infty.$$

ЛИТЕРАТУРА

1. *Иванов М. А., Чугунков И. В.* Теория, применение и оценка качества генераторов псевдослучайных последовательностей. М.: КУДИЦ-ОБРАЗ, 2003. 240 с.
2. *Rukhin A., Soto J., Nechvatal J., et al.* A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST, Apr. 2010. `https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf`.
3. *Михайлов В. Г.* Предельная теорема пуассоновского типа для числа пар почти полностью совпавших цепочек // Теория вероятностей и ее применения. 2008. Т. 53. Вып. 1. С. 59–71.
4. *Михайлов В. Г., Шойтов А. М.* О числах множеств эквивалентных цепочек в последовательности независимых случайных величин // Математические вопросы криптографии. 2013. Т. 4. Вып. 1. С. 77–86.
5. *Шойтов А. М.* Нормальное приближение в задаче об эквивалентных цепочках // Труды по дискретной математике. 2007. Т. 10. С. 326–349.
6. *Михайлов В. Г.* Оценки точности пуассоновской аппроксимации для распределения числа серии повторений длинных цепочек в цепи Маркова // Дискретная математика. 2015. Т. 27. Вып. 4. С. 67–78.
7. *Михайлов В. Г., Шойтов А. М.* О длинных повторениях цепочек в цепи Маркова // Дискретная математика. 2014. Т. 26. Вып. 3. С. 79–89.
8. *Михайлов В. Г., Шойтов А. М.* Многократные повторения длинных цепочек в цепи Маркова // Математические вопросы криптографии. 2015. Т. 6. Вып. 3. С. 117–134.
9. *Михайлов В. Г., Меженная Н. М., Волгин А. В.* Об условиях асимптотической нормальности числа повторений в стационарной случайной последовательности // Дискретная математика. 2021. Т. 33. Вып. 3. С. 64–78.

УДК 519.7 DOI 10.17223/2226308X/15/4

О РАССЕИВАЮЩИХ СВОЙСТВАХ ОБОБЩЁННЫХ КВАЗИАДАМАРОВЫХ ПРЕОБРАЗОВАНИЙ НА КОНЕЧНЫХ АБЕЛЕВЫХ ГРУППАХ

Б. А. Погорелов, М. А. Пудовкина

Для произвольной конечной группы X предлагаются обобщения квазиадамаровых преобразований. При $X = \mathbb{Z}_{2^m}$ они включают в себя псевдоадамаровы преобразования алгоритмов блочного шифрования Safer, Safer+, Safer++, Twofish, а также квазиадамаровы преобразования, предложенные Х. Липмаа. Описаны свойства рассеивания биективными обобщёнными квазиадамаровыми преобразованиями систем импримитивности регулярных подстановочных представлений аддитивных групп $\mathbb{Z}_{2^m}^2$ и $\mathbb{Z}_{2^{2m}}$. Получены условия, при которых обобщённые квазиадамаровы преобразования максимально рассеивают все нетривиальные системы импримитивности этих двух групп.

Ключевые слова: *алгоритмы шифрования семейства Safer, алгоритм шифрования Twofish, псевдоадамарово преобразование, квазиадамарово преобразование, система импримитивности, примитивная группа, регулярное подстановочное представление.*

В ряде алгоритмов блочного шифрования (Safer [1], Hight [2], CS-Cipher [3], Speed [4] и др.) большинство преобразований, составляющих раундовую функцию, реализует побайтовую обработку блоков текста без взаимного влияния байтов с использованием абелевых групп. Наибольшее распространение среди таких «межбайтовых» преобразований получил класс псевдоадамаровых преобразований на двух байтах в алгоритмах семейства Safer. В [5] он обобщён до класса квазиадамаровых преобразований на аддитивных группах $\mathbb{Z}_{2^m}^2$, $\mathbb{Z}_2^{2m}$, для которых получены формулы нахождения элементов разностной матрицы. В [6] найден показатель рассеивания (branch number) тензорного произведения псевдоадамаровых преобразований алгоритма Safer.

Пусть $m \geqslant 2$; $(X, *)$ — конечная группа с бинарной операцией $*$; e — тождественная подстановка; $S(X)$ — симметрическая группа на X; $P(X) = \{b \mid b\colon X \to X\}$ — симметрическая полугруппа; $\alpha^g = \alpha g = g(\alpha)$ — образ элемента $\alpha \in X$ при действии на него преобразованием $g \in P(X)$;

$$\mathrm{Hom}(X) = \left\{v\colon X \to X \mid \forall x, y \in X \left(v(x * y) = v(x) * v(y)\right)\right\}.$$

Для алгоритмов блочного шифрования рассмотрим обобщение $h_{\bar\vartheta,\bar\psi}\colon X^2 \to X^2$ квазиадамаровых и псевдоадамаровых преобразований на группу $(X, *)$, которое задаётся набором отображений $\bar\vartheta = (\vartheta_1, \vartheta_2)$, $\bar\psi = (\psi_1, \psi_2)$, $\bar\vartheta, \bar\psi \in P(X)^2$б и условием

$$h_{\bar\vartheta,\bar\psi} : (\alpha_1, \alpha_2) \mapsto \left(\alpha_1^{\vartheta_1} * \alpha_2^{\vartheta_2}, \alpha_1^{\psi_1} * \alpha_2^{\psi_2}\right). \tag{1}$$

При

$$(X, *) \in \{(\mathbb{Z}_{2^m}, +), (\mathbb{Z}_2^m, \oplus)\},\ r_1, r_2 \in \{0, \ldots, m-1\}^2,$$
$$r_1 = (r_{11}, r_{12}),\ r_2 = (r_{21}, r_{22}),\ r_1 \neq r_2,\ \upsilon \in \{0, 1\}$$

квазиадамарово преобразование $u_{r_1,r_2}^{(\upsilon)}$ на X, введённое в [5], является частным случаем обобщения (1). Так, при $X = \mathbb{Z}_{2^m}$ оно задаётся условием

$$u_{r_1,r_2}^{(\upsilon)}\colon (\alpha_1, \alpha_2) \mapsto (2^{r_{11}}\alpha_1 + (-1)^{\upsilon}2^{r_{12}}\alpha_2, 2^{r_{21}}\alpha_1 + (-1)^{\upsilon}2^{r_{22}}\alpha_2)$$

и совпадает с $h_{\bar{\vartheta},\bar{\psi}}$ при следующих отображениях:

$$\vartheta_1\colon \alpha \mapsto 2^{r_{11}}\alpha \mod 2^m,\ \vartheta_2\colon \alpha \mapsto (-1)^v 2^{r_{12}}\alpha \mod 2^m,$$
$$\psi_1\colon \alpha \mapsto 2^{r_{21}}\alpha \mod 2^m,\ \psi_2\colon \alpha \mapsto (-1)^v 2^{r_{22}}\alpha \mod 2^m.$$

Отметим, что при чётном $r_{11} > 0$, $r_2 = (0,0)$ преобразование

$$u^{(0)}_{(r_{11},0),(0,0)}\colon (\alpha_1,\alpha_2) \mapsto (2^{r_{11}}\alpha_1 + \alpha_2, \alpha_1 + \alpha_2)$$

применено в хеш-функции FFT [7]. Кроме того, при $r_{11} = 1$ преобразование $u^{(0)}_{(r_{11},0),(0,0)}$, известное как псевдоадамарово, используется для улучшения рассеивающих свойств в алгоритмах блочного шифрования Safer, Safer+ [8], Safer++ [9] и Twofish [10]. В данной работе получены условия биективности преобразования $h_{\bar{\vartheta},\bar{\psi}}$. Так, для биективности $h_{\bar{\vartheta},\bar{\psi}}$ необходимо, чтобы одно из преобразований ϑ_1, ϑ_2, ψ_1, ψ_2 было биективным. Отсюда вытекает, что при $X = \mathbb{Z}_{2^m}$ биективное преобразование $h_{\bar{\vartheta},\bar{\psi}}$ принимает вид

$$h_{\bar{\vartheta},\bar{\psi}}\colon (\alpha_1,\alpha_2) \mapsto \left(\alpha_{v_1}^{\vartheta_{v_1}} + b_1\alpha_{v_2},\ a\cdot\alpha_{v_1}^{\vartheta_{v_1}} + b_2\alpha_{v_2}\right) \text{ для всех } (\alpha_1,\alpha_2) \in \mathbb{Z}_{2^m}^2, \tag{2}$$

где $\{v_1,v_2\} = \{1,2\}$, $v_2 = v_1(1,2)$, $\vartheta_{v_1} \in S(\mathbb{Z}_{2^m})$, $a, b_1, b_2 \in \mathbb{Z}_{2^m}$, причём

$$b_2 - a\cdot b_1 \equiv 1 \,(\mathrm{mod}\ 2). \tag{3}$$

В работе показывается, что условию (2) удовлетворяет биективное квазиадамарово преобразование.

Для импримитивной группы $G \leqslant S(X)$ с системой импримитивности $\overline{W}$ (т. е. $\overline{W}$ — нетривиальное разбиение множества X на равномощные блоки, сохраняемое группой G) рассеивание подстановкой $g \in S(X)$ системы $\overline{W}$, а также расстояние от g до G будем характеризовать посредством матрицы $c^{(\overline{W})}(g)$, введённой в [11]. Разбиению $\overline{W} = \{W_0,\dots,W_{p-1}\}$ множества X и подстановке $g \in G$ ставится в соответствие матрица $c^{(\overline{W})}(g) = \left[c^{(\overline{W})}_{i,j}(g)\right]$, где

$$c^{(\overline{W})}_{i,j}(g) = |W_i^g \cap W_j|,\ W_i^g = \{\beta^g \mid \beta \in W_i\},\ i,j \in \{0,\dots,p-1\}.$$

Для группы $(G,*)$ рассмотрим её правое подстановочное представление $\varphi_G\colon G \to S(G)$, заданное условием

$$\varphi_G(k)\colon x \mapsto x * k \ \text{ для всех } \ x,k \in G.$$

Пусть $\bar{G} = \varphi_G(G) = \{\varphi_G(k) \mid k \in G\}$. Для $d \in \{0,\dots,2m\}$, $t \in \{0,\dots,d\}$ положим $\overline{W}^{(d,t)} = \left\{W_0^{(d,t)},\dots,W_{2^t-1}^{(d,t)}\right\}$, где

$$W_i^{(d,t)} = \left\{j \equiv i \,(\mathrm{mod}\ 2^t) \mid j \in \mathbb{Z}_{2^d}\right\},\ i = 0,\dots,2^t-1.$$

Легко видеть, что $\overline{W}^{(d,1)},\dots,\overline{W}^{(d,d-1)}$ — нетривиальные системы импримитивности группы $\bar{\mathbb{Z}}_{2^d}$.

Для $t_1,t_2 \in \{0,\dots,m\}$ положим

$$W_{i,j}^{(m,t_1,t_2)} = W_i^{(m,t_1)} \times W_j^{(m,t_2)},\ i = 0,\dots,2^{t_1}-1,\ j = 0,\dots,2^{t_2}-1,$$
$$\overline{W}^{(m,t_1,t_2)} = \left\{W_{i,j}^{(m,t_1,t_2)} \mid i \in \{0,\dots,2^{t_1}-1\}, j \in \{0,\dots,2^{t_2}-1\}\right\}.$$

В работе описываются свойства рассеивания преобразованием $h_{\bar{\vartheta},\bar{\psi}}$ систем импримитивности регулярных подстановочных представлений $\bar{\mathbb{Z}}_{2^m}^2 = \bar{\mathbb{Z}}_{2^m} \times \bar{\mathbb{Z}}_{2^m}$ и $\bar{\mathbb{Z}}_{2^{2m}}$ соответственно двух групп наложения ключа $\mathbb{Z}_{2^m}^2$ и $\mathbb{Z}_{2^{2m}}$.

Легко видеть, что группа $\bar{\mathbb{Z}}_{2^m}^2$ импримитивна, а $\left\{\overline{W}^{(m,t_1,t_2)} \mid t_1, t_2 \in \{0,\ldots,m\}\right\}$ — множество всех её систем импримитивности. Найдены элементы матрицы $c^{(\overline{W}^{(m,t,t)})}(h_{\bar{\vartheta},\bar{\psi}})$ для преобразования $h_{\bar{\vartheta},\bar{\psi}}$ и системы импримитивности $\overline{W}^{(m,t,t)}$ для каждого $t \in \in \{0,\ldots,m\}$.

Утверждение 1. Пусть преобразование $h_{\bar{\vartheta},\bar{\psi}}\colon \mathbb{Z}_{2^m}^2 \to \mathbb{Z}_{2^m}^2$ удовлетворяет условиям (2) и (3), $v_1 = 1$, $\vartheta_1 \in S(\mathbb{Z}_{2^m})$. Группа $\left\langle h_{\bar{\vartheta},\bar{\psi}}, \bar{\mathbb{Z}}_{2^m}^2 \right\rangle \leqslant S(\mathbb{Z}_{2^m}^2)$ является примитивной тогда и только тогда, когда примитивна группа $\left\langle \vartheta_1, \bar{\mathbb{Z}}_{2^m} \right\rangle \leqslant S(\mathbb{Z}_{2^m})$.

Нетривиальной системой импримитивности группы $\bar{\mathbb{Z}}_{2^{2m}}$ является $\overline{W}^{(2m,t)}$ для каждого $t \in \{1,\ldots,2m-1\}$. Преобразование $h_{\bar{\vartheta},\bar{\psi}}$, заданное условием (1), зависит от элемента $v_1 \in \{1,2\}$.

Утверждение 2. Пусть $t \in \{1,\ldots,2m-1\}$, преобразование $h_{\bar{\vartheta},\bar{\psi}}\colon \mathbb{Z}_{2^m}^2 \to \mathbb{Z}_{2^m}^2$ удовлетворяет условиям (2) и (3), $v_1 = 1$, $a \equiv 1 \pmod 2$, $\vartheta_1 \in S(\mathbb{Z}_{2^m})$. Тогда для каждых $j_1, j_2 \in \{0,\ldots,2^t-1\}$ элементы матрицы $c^{\left(\overline{W}^{(2m,t)}\right)}\left(h_{\bar{\vartheta},\bar{\psi}}\right)$ удовлетворяют следующим свойствам:

1) $c_{j_1,j_2}^{\left(\overline{W}^{(2m,t)}\right)}\left(h_{\bar{\vartheta},\bar{\psi}}\right) = 2^{2m-2t}$, если $t \in \{1,\ldots,m\}$;
2) $c_{j_1,j_2}^{\left(\overline{W}^{(2m,t)}\right)}\left(h_{\bar{\vartheta},\bar{\psi}}\right) \in \{0,1\}$, если $t \in \{m+1,\ldots,2m-1\}$.

Следовательно, матрица $c^{\left(\overline{W}^{(2m,t)}\right)}\left(h_{\bar{\vartheta},\bar{\psi}}\right)$ является «максимально равномерной» для каждого $t \in \{1,\ldots,2m-1\}$, а преобразование $h_{\bar{\vartheta},\bar{\psi}}$ максимально удалено от группы $\mathrm{IG}_{\overline{W}^{(2m,t)}}$, которая состоит из всех подстановок, сохраняющих систему импримитивности $\overline{W}^{(2m,t)}$. Поэтому преобразование $h_{\bar{\vartheta},\bar{\psi}}$ максимально рассеивает все нетривиальные системы импримитивности группы $\bar{\mathbb{Z}}_{2^{2m}}$. Показано также, что при чётном a матрица $c^{\left(\overline{W}^{(2m,t)}\right)}\left(h_{\bar{\vartheta},\bar{\psi}}\right)$ не является «максимально равномерной». Схожие результаты получены для $v_1 = 2$. Таким образом, введённый класс обобщённых квазиадамаровых преобразований существенно шире классов псевдоадамаровых и квазиадамаровых преобразований, причём он содержит преобразования, которые отличны от квазиадамаровых, но также обладают хорошими рассеивающими свойствами.

ЛИТЕРАТУРА

1. *Massey J. L.* SAFER K-64: a byte-oriented block-ciphering algorithm // FSE 1994. LNCS. 1994. V. 1267. P. 1–17.
2. *Hong D., Sung J., Hong S., et al.* A new block cipher suitable for low-resource device // CHES 2006. LNCS. 2006. V. 4249. P. 46–59.
3. *Stern J. and Vaudenay S.* CS-Cipher // FSE 1998. LNCS. 1998. V. 1372. P. 189–204.
4. *Zheng Y.* The SPEED cipher // Financial Cryptography. LNCS. 1997. V. 1318. P. 71–89.
5. *Lipmaa H.* On differential properties of pseudo-Hadamard transform and related mappings // INDOCRYPT 2002. LNCS. 2002. V. 2551. P. 48–61.
6. *St Denis T.* Fast Pseudo-Hadamard Transforms. Cryptology ePrint Archive, Report 2004/010. 2004. `https://eprint.iacr.org/2004/010.pdf`.
7. *Schnorr C.-P.* FFT-Hash II, efficient cryptographic hashing // EUROCRYPT'92. LNCS. 1992. V. 658. P. 45–54.

8. *Massey J., Khachatrian G., and Kuregian M.* Nomination of SAFER+ as Candidate Algorithm for the Advanced Encryption Standard (AES). NIST AES Proposal, 1998. `http://www.princeton.edu/~rblee/safer+/`.
9. *Massey J., Khachatrian G., and Kuregian M.* Nomination of SAFER++ as Candidate Algorithm for NESSIE. 2003. `https://www.cosic.esat.kuleuven.be/nessie/workshop/submissions/safer++.zip`.
10. *Schneier B., Kelsey J., Whiting D., et al.* The Twofish Encryption Algorithm: A 128-Bit Block Cipher. N.Y.: John Wiley & Sons, 1999.
11. *Погорелов Б. А., Пудовкина М. А.* О расстояниях от подстановок до импримитивных групп при фиксированной системе импримитивности // Дискретная математика. 2013. Т. 25. № 3. С. 78–95.

Секция 2

ДИСКРЕТНЫЕ ФУНКЦИИ

УДК 519.7 DOI 10.17223/2226308X/15/5

ПРИМЕНЕНИЕ ЭВРИСТИЧЕСКИХ МЕТОДОВ ДЛЯ ПОИСКА БУЛЕВЫХ ФУНКЦИЙ С КРИПТОГРАФИЧЕСКИМИ ХАРАКТЕРИСТИКАМИ[1]

Н. Д. Атутова

Для успешного противостояния шифров линейному и алгебраическому криптоанализу в их структуре необходимо использовать функции с высокой нелинейностью и алгебраической иммунностью. Работа является продолжением исследования, в котором используется комбинированный подход к поиску криптографических булевых функций на основе эвристических методов, в частности генетического алгоритма и поиска восхождением к вершине (алгоритм Hill Climbing). Проведён сравнительный анализ вариаций мутации и скрещивания для генетического алгоритма и сравнительный анализ ранее полученных результатов со случайным поиском. На основе полученных булевых функций построены векторные булевы функции и среди них подсчитано количество функций, обаладающих высокой компонентной алгебраической иммунностью и нелинейностью.

Ключевые слова: *генетический алгоритм, алгоритм Hill Climbing, алгебраическая иммунность, нелинейность, эвристики.*

Успех криптоанализа в основном зависит от нахождения уязвимостей в математических свойствах булевых функций, являющихся компонентами шифра. Для противостояния разным видам криптоанализа существуют различные математические требования к функциям.

Функции, которые удовлетворяют этим требованиям, называются криптографическими. Для повышения стойкости необходимо искать и применять в шифрах булевы функции с оптимальными криптографическими свойствами. Для ослабления статистической зависимости между входом и выходом функции по умолчанию предполагается, что функция обладает сбалансированностью, т. е. принимает значения 0 и 1 одинаково часто.

Работа является продолжением исследования, начальные результаты которого представлены в [1]. Рассматриваются булевы функции от n переменных с такими криптографическими характеристиками: сбалансированность, нелинейность и алгебраическая иммунность. Известны теоретические оценки максимума этих характеристик: для нелинейности — $N_f \leqslant 2^{n-1} - 2^{n/2-1}$, для алгебраической иммунности — $\mathrm{AI}(f) \leqslant \lceil n/2 \rceil$.

Цель данной работы — построение сбалансированных булевых функций с хорошими криптографическими характеристиками, а также построение на их основе векторных булевых функций с компонентной алгебраической иммунностью и нелинейностью.

[1]Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования РФ № 075-15-2022-281.

Функция вида $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ называется *векторной булевой функцией*. Векторную булеву функцию можно представить в виде $F = (f^1(x) \dots f^m(x))$, где $f^j(x)$, $j = 1, \dots, m$, — координатные (булевы) функции от n переменных.

Компонентной функцией называется любая ненулевая линейная комбинация координатных функций, т. е. булева функция $vF = v_1 f_1 \oplus \dots \oplus v_m f_m$, где $v = (v_1 \dots v_m) \in \in \mathbb{F}_2^m \setminus 0^m$ и 0^m — нулевой вектор длины m.

Нелинейностью N_f и *компонентной алгебраической иммунностью* $\mathrm{AI}_{\mathrm{comp}}(F)$ векторной булевой функции $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$ называются минимальная нелинейность и алгебраическая иммунность её компонентных функций соответственно:

$$N_F = \min_{v \in \mathbb{F}_2^m \setminus 0^m} N_{vF}, \qquad \mathrm{AI}_{\mathrm{comp}}(F) = \min_{v \in \mathbb{F}_2^m \setminus 0^m} \mathrm{AI}(vF).$$

Можно выделить три способа нахождения таких функций: полный перебор, алгебраическое конструирование и эвристики. Для булевой функции $f : \mathbb{F}_2^n \to \mathbb{F}_2$ длина вектора значений — 2^n. При росте числа переменных количество булевых функций растёт дважды экспоненциально, что делает невозможным полный перебор. Алгебраическое построение заведомо сужает множество решений и повышает вероятность успеха криптоанализа из-за предсказуемости построения. Перспективным является подход, использующий эвристические методы, в основе которых лежит итерационный перебор с параметрами для достижения желаемого результата.

Для поиска криптографических булевых функций и построения на их основе векторных булевых функций предложен гибридный алгоритм на основе генетического алгоритма и поиска восхождением к вершине (алгоритм Hill Climbing) и проведены вычислительные эксперименты, показывающие его эффективность.

Генетический алгоритм (ГА) — это эвристический подход к комбинаторной оптимизации в сложном пространстве пригодности. Идея, лежащая в основе генетического алгоритма, состоит в том, чтобы начать с набора потенциальных решений (генофонда или родительского множества) и каким-то образом объединить их пары (схема размножения) для получения новых решений (дочерних).

Схемы размножения могут включать родительский отбор и мутацию полученного в результате ребенка. Новый набор выбирается из числа родителей и детей (отбор) на основе функции пригодности, и процесс повторяется до тех пор, пока не будет найдено оптимальное решение или не будет достигнуто максимальное число итераций.

В работе особями являются векторы значений булевых функций от n переменных, начальная популяция — случайное множество сбалансированных особей.

Поиск восхождением к вершине (Hill Climbing) — эвристический алгоритм, который начинает с произвольного решения, а затем итерационно пытается найти лучшее решение путём пошагового изменения одного из его элементов. На вход поступает вектор значений булевой функции. Алгоритм итеративно пытается его улучшить, изменяя одну из координат.

Более подробное описание алгоритма и теоретическое обоснование его эффективности представлено в [2]. В работе поиск восхождением к вершине применяется после оператора мутации потомков на каждой итерации ГА для поддержания нелинейности.

Рассмотрим два варианта реализации оператора скрещивания.

Первый — однородное скрещивание. На вход поступают булевы функции f и g от n переменных, заданные векторами значений $(f_0, f_1, \dots, f_{2^n-1})$ и $(g_0, g_1, \dots, g_{2^n-1})$ соответственно. На выходе булева функция h от n переменных, вектор значений $(h_0, h_1, \dots, h_{2^n-1})$ которой определяется следующим образом: если $f_i = g_i$, то $h_i = f_i$;

если $f_i \neq g_i$, то h_i принимается равным значению f_i или g_i с одинаковой вероятностью, где $i = 0, 1, \ldots, 2^{n-1}$.

Второй вариант предложен в [3], он похож на предыдущий, но, в отличие от первого, позволяет поддерживать сбалансированность потомков.

Введём некоторые ограничения на скрещивание. *Расстоянием Хэмминга* ($\mathrm{dist}(f, g)$) между булевыми функциями f и g от n переменных называется число координат, в которых различаются их векторы значений.

Если $\mathrm{dist}(f, g) > 2^{n-1}$, то вместо вектора значений функции g рассматривается вектор, полученный инверсией всех битов вектора значений функции g.

Рассмотрено применение двух видов мутации.

Первый — к двум различным случайным битам входного вектора значений применялась инверсия.

Второй предложен в [4]. Выбираются две случайные позиции в векторе значений. Значение из большей позиции переставляется в меньшую и происходит сдвиг битов вправо.

Для генетического алгоритма с целевой функцией $\mathrm{cost} = \mathrm{AI}(f)$ проведён сравнительный анализ эффективности различных вариантов операций скрещивания и мутации. Полученные результаты приведены в табл. 1, где n — число переменных; P — размер популяции; T — число итераций.

Таблица 1

Результаты применения ГА с различными операторами

n	P	T	max, min, среднее значение $\mathrm{AI}(f)$ в исходной популяции	max, min, среднее значение $\mathrm{AI}(f)$ после применения ГА	Количество функций с $\max \mathrm{AI}(f)$	Количество функций с $\max \mathrm{AI}(f)$ с новыми операторами
6	10	20	(3, 1, 1,9)	(3, 3, 3)	659	682
7	10	20	(4, 2, 2,8)	(4, 4, 4)	14	34
8	10	20	(4, 1, 2,7)	(4, 4, 4)	671	831

Как видно из табл. 1, все потомки имеют ненулевую алгебраическую иммунность вне зависимости от вариации операторов, но неклассические мутации находят большее число функций с максимальной алгебраической иммунностью.

Для оценки эффективности гибридного подхода, основанного на сочетании генетического алгоритма с новыми операторами и поиска восхождением к вершине проведён сравнительный анализ по времени и количеству найденных функций со случайным перебором из одного миллиона случайно сгенерированных сбалансированных функций. Результаты представлены в табл. 2.

Таблица 2

Сравнение гибридного подхода со случайным поиском

n	P	T	Найдено функций ГА+НС	Время, с	Случайный поиск из 1000000 функций	Время, с
4	20	5000	314222	432	91002	972
6	20	5000	564111	765	123611	1187
7	20	5000	457211	1056	198715	1623
8	20	5000	732888	1434	261238	2111

Как видно из табл. 2, гибридный подход показывает высокую эффективность в задаче поиска булевых функций с алгебраической иммунностью ($\mathrm{cost} = \mathrm{AI}(f)$).

Полученные булевы функции используются для построения векторных булевых функций. Изменим целевую функцию на сочетание криптографических характеристик и применим гибридный подход для поиска булевых функций и на основе получившихся построим векторные булевы функции для $m = 4$ и $n = 4, 5, 6, 8$. Целевая функция — нелинейность и алгебраическая иммунность: $\text{cost} = \dfrac{N_f}{\max N_g} + \dfrac{\text{AI}(f)}{\max \text{AI}(g)}$, где g пробегает множество особей популяции. Результаты представлениы в табл. 3

Таблица 3

Результаты для векторных булевых функций

n	P	T	Найдено булевых функций	Найдено векторных булевых функций	Значение N_F и $\text{AI}_{\text{comp}}(F)$	Теор. оценка N_F	Теор. оценка $\text{AI}_{\text{comp}}(F)$	Время, с
4	20	10	1245	415661	6; 2	$\leqslant 6$	$\leqslant 2$	162,202
5	20	10	76	1797608	12; 3	< 13	$\leqslant 3$	5323,85
6	20	10	1346	310295	24; 3	$\leqslant 28$	$\leqslant 3$	363,098
8	20	10	1426	321203	108; 4	$\leqslant 120$	$\leqslant 4$	3061,11

Таким образом, в работе приведены результаты сравнительного анализа гибридного подхода и случайного перебора. Найденные булевы функции использованы для поиска векторных булевых функций. Представлены результаты поиска векторных булевых функций с нелинейностью и компонентной алгебраической иммунностью для $m = 4$ и $n \leqslant 8$. Полученные векторные булевы функции могут быть использованы для построения S-блоков. Наличие компонентной алгебраической иммунности и нелинейности S-блоков способствует противостоянию алгебраическому и линейному криптоанализу шифров соответственно.

ЛИТЕРАТУРА

1. *Амутова Н. Д.* Гибридный подход к поиску булевых функций с высокой алгебраической иммунностью на основе эвристических методов // Прикладная дискретная математика. Приложение. 2021. № 14. С. 37–40.
2. *Millan W., Clark A., and Dawson E.* An effective genetic algorithm for finding highly nonlinear Boolean functions // LNCS. 1997. V. 1334. P. 149–158.
3. *Kang M. and Wang M.* New genetic operators for developing S-boxes with low boomerang uniformity // IEEE Access. 2022. V. 10. P. 10898–10906.
4. *Behera P. and Gangopadhyay S.* Evolving bijective S-Boxes using hybrid adaptive genetic algorithm with optimal cryptographic properties // J. Ambient Intell. Human. Comput. 2021. P. 2640–2658.

УДК 519.7 DOI 10.17223/2226308X/15/6

О НИЖНЕЙ ОЦЕНКЕ ЧИСЛА БЕНТ-ФУНКЦИЙ НА МИНИМАЛЬНОМ РАССТОЯНИИ ОТ БЕНТ-ФУНКЦИЙ ИЗ КЛАССА МЭЙОРАНА — МАКФАРЛАНДА[1]

Д. А. Быков

Исследуется построение бент-функций на некотором расстоянии от заданной бент-функции. Для функции f из класса Мэйорана — МакФарланда $\mathcal{M}_{2n}$ доказан критерий того, что функция, полученная из f прибавлением индикатора аффинного подпространства размерности n, является бент-функцией. Показано, что для простых $n \geqslant 5$ достигается нижняя оценка $2^{2n+1} - 2^n$ числа бент-функций на минимальном расстоянии от бент-функций из класса $\mathcal{M}_{2n}$. Найдены бент-функции, для которых оценка точна. Показано, что эта нижняя оценка не достигается для бент-функций из класса $\mathcal{M}_{2n}$, где перестановка, по которой построена бент-функция, не является APN-функцией. Для некоторых расстояний, в частности 2^{2n-1}, получены нижние оценки числа бент-функций из класса $\mathcal{M}_{2n}$ на этих расстояниях от бент-функций из класса $\mathcal{C}$.

Ключевые слова: *бент-функции, булевы функции, минимальное расстояние, класс Мэйорана — МакФарланда, нижние оценки.*

Введение

Пусть $\mathbb{F}_2^n$ — линейное пространство над полем $\mathbb{F}_2$, состоящее из двоичных векторов размерности n. Множество всех *булевых функций* от n переменных $f : \mathbb{F}_2^n \to \mathbb{F}_2$ обозначим через $\mathcal{F}_n$. *Расстоянием* $\operatorname{dist}(f, g)$ между двумя булевыми функциями $f, g \in \mathcal{F}_n$ называется число позиций, в которых векторы значений этих функций различаются. Булева функция от чётного числа переменных $2n$, расстояние от которой до множества всех аффинных функций максимально и равно $2^{2n-1} - 2^{n-1}$, называется *бент-функцией*. Везде далее будем рассматривать бент-функции от $2n$ переменных. Обозначим множество всех бент-функций от $2n$ переменных через $\mathcal{B}_{2n}$. Класс бент-функций Мэйорана — МакФарланда $\mathcal{M}_{2n}$ состоит из функций вида

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

где $\varphi \in \mathcal{F}_n$ и π — перестановка на $\mathbb{F}_2^n$. Напомним, что функции $f, g \in \mathcal{F}_n$ называются *EA-эквивалентными*, если существуют аффинная функция h, невырожденная матрица A размера $n \times n$ с элементами из $\mathbb{F}_2$ и $b \in \mathbb{F}_2^n$, такие, что $f(x) = g(xA \oplus b) \oplus h(x)$ для всех $x \in \mathbb{F}_2^n$. Через *индикатор* Ind_S обозначим характеристическую функцию множества $S \subseteq \mathbb{F}_2^n$.

Бент-функции введены О. Ротхаусом в [1]. Они интересны как своими экстремальными значениями нелинейности, так и приложениями в криптографии, теории кодирования, теории символьных последовательностей. Однако есть много открытых вопросов об устройстве множества всех бент-функций, о соотнесении различных известных классов бент-функций. Например, классы $\mathcal{C}$ и $\mathcal{D}$, введённые в [2], лежат вне замыкания класса $\mathcal{M}_{2n}$ относительно EA-эквивалентности. Но это известно благодаря построенным примерам [2, 3] и непонятно, какая часть бент-функций из $\mathcal{C}$ и $\mathcal{D}$ лежит вне замыкания $\mathcal{M}_{2n}$.

Для построения бент-функции на некотором расстоянии от исходной бент-функции $f \in \mathcal{B}_{2n}$ можно воспользоваться универсальной конструкцией $f \mapsto f \oplus \operatorname{Ind}_S$. При этом

[1]Работа выполнена в рамках госзадания ИМ СО РАН (проект № FWNF-2022-0018).

часто удобно рассматривать не произвольное множество $S \subseteq \mathbb{F}_2^{2n}$, а, например, некоторое подпространство U. Так, в работе [2] для случая, когда U — аффинное подпространство $\mathbb{F}_2^{2n}$ размерности n, доказано, что $f \oplus \mathrm{Ind}_U$ — бент-функция.

Для двух различных $f, g \in \mathcal{B}_{2n}$ известно, что $\mathrm{dist}(f, g) \geqslant 2^n$. В [4] показано, что все бент-функции на минимально возможном расстоянии 2^n от заданной бент-функции f могут быть выражены как $f \oplus \mathrm{Ind}_U$, где U — аффинное подпространство $\mathbb{F}_2^{2n}$, $\dim U = n$. Иными словами, в поиске всех бент-функций на расстоянии 2^n достаточно ограничиться прибавлением индикатора аффинного подпространства размерности n.

В [5] приведён вид всех бент-функций из $\mathcal{M}_{2n}$, лежащих на расстоянии 2^n от исходной бент-функции из $\mathcal{M}_{2n}$. Число таких бент-функций может быть использовано в качестве следующей оценки.

Утверждение 1 (Н. Коломеец, 2017). Число всех бент-функций на минимальном расстоянии от бент-функций из $\mathcal{M}_{2n}$ можно оценить снизу как $2^{2n+1} - 2^n$. При этом все бент-функции, учитывающиеся в этой оценке, также лежат в классе $\mathcal{M}_{2n}$.

1. Число бент-функций на минимально возможном расстоянии 2^n от бент-функций из $\mathcal{M}_{2n}$

Рассмотрим $f \in \mathcal{M}_{2n}$ и конструкцию $f \mapsto f \oplus \mathrm{Ind}_U$, где U — аффинное подпространство $\mathbb{F}_2^{2n}$, $\dim U = n$. Сначала построим критерий того, что $f \oplus \mathrm{Ind}_U$ является бент-функцией.

Пусть E — линейное подпространство $\mathbb{F}_2^n$ размерности k. Тогда для E существует единственная GJB-матрица M — приведённая ступенчатая матрица полного ранга размера $k \times n$, строки которой составляют базис E. Через $\langle M \rangle$ обозначим линейную оболочку строк матрицы M.

В следующей теореме приведён критерий для функций $f(x, y) = \langle y, \pi(x) \rangle \oplus \varphi(x)$, другими словами, $f \notin \mathcal{M}_{2n}$, а $h(x, y) = f(y, x) \in \mathcal{M}_{2n}$. Такой переход нужен для удобного представления базисов подпространств с помощью GJB-матриц. Критерий для бент-функций из $\mathcal{M}_{2n}$ является следствием этой теоремы.

Теорема 1. Пусть $f(x, y) = \langle y, \pi(x) \rangle \oplus \varphi(x)$, такая, что $h(x, y) = f(y, x) \in \mathcal{M}_{2n}$; $U = (a, b) \oplus E$ — аффинное подпространство $\mathbb{F}_2^{2n}$, $\dim U = n$, где $a, b \in \mathbb{F}_2^n$; линейное подпространство E имеет GJB-матрицу вида

$$M = \left(\begin{array}{c|c} L & T \\ \hline 0 & R \end{array} \right),$$

где L является GJB-матрицей размера $(n - k) \times n$. Тогда

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

является бент-функцией, если и только если выполнены следующие условия:

1) $\pi(a \oplus \langle L \rangle) = \pi(a) \oplus \langle R \rangle^{\perp}$;
2) $\langle uT \oplus b, \pi(uL \oplus a) \rangle \oplus \varphi(uL \oplus a)$ — аффинная функция от переменных $u = (u_1, \ldots, u_{n-k}) \in \mathbb{F}_2^{n-k}$.

Отметим, что в частном случае $U = U_1 \times U_2$, где $\dim U = n$, U_1, U_2 — аффинные подпространства $\mathbb{F}_2^n$, теорему 1 можно переписать в более простом виде. В определении класса $\mathcal{D}$ используются бент-функции $f(x, y) = \langle x, \pi(y) \rangle \oplus \varphi(y) \in \mathcal{M}_{2n}$ с $\varphi \equiv 0$, к которым прибавляется Ind_E, где $E = E_1 \times E_2$; $\dim E = n$; E_1, E_2 — линейные подпространства $\mathbb{F}_2^n$. Поэтому теорема 1 обобщает определение класса $\mathcal{D}$, так как даёт критерий на бент-функции при произвольной φ и аффинных подпространствах U_1, U_2.

Функция $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ называется *APN-функцией*, если для любых $a \neq 0, b \in \mathbb{F}_2^n$ уравнение $F(x) \oplus F(x \oplus a) = b$ имеет не более двух решений. Рассмотрим случай, когда нижняя оценка гарантированно не достигается.

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

Таким образом, для функций $f(x, y) = \langle x, \pi(y) \rangle \oplus \varphi(y) \in \mathcal{M}_{2n}$, где π не является APN-перестановкой, существует подпространство U размерности n, такое, что бент-функция $f \oplus \mathrm{Ind}_U \notin \mathcal{M}_{2n}$.

Далее будем представлять булевы функции как функции из $\mathbb{F}_{2^n}$ в $\mathbb{F}_2$, зафиксировав в поле $\mathbb{F}_{2^n}$ некоторый базис над $\mathbb{F}_2$. Функция $\mathrm{tr} : \mathbb{F}_{2^n} \to \mathbb{F}_2$, определённая как $\mathrm{tr}(x) = x^{2^0} + x^{2^1} + \ldots + x^{2^{n-1}}$, называется *следом*. Любая $f \in \mathcal{F}_n$ представима в виде $f = \mathrm{tr}\left(\sum\limits_{k=0}^{2^n-1} c_k x^k\right)$. Такое представление не единственно, но его можно сделать таковым [6]. Функции класса $\mathcal{M}_{2n}$ в этом представлении записываются как

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

где $\varphi : \mathbb{F}_{2^n} \mapsto \mathbb{F}_2$; $\pi : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ — взаимно однозначная функция; $x, y \in \mathbb{F}_{2^n}$. В поле $\mathbb{F}_{2^n}$ функция обращения элемента $F(x) = x^{2^n-2}$ является взаимно однозначной. Отметим, что при нечётных n это APN-перестановка.

Теорема 3. Пусть $n \geqslant 5$ — простое; функция $f(x, y) = \mathrm{tr}(xy^{2^n-2}) + \varphi(y) \in \mathcal{M}_{2n}$, $x, y \in \mathbb{F}_{2^n}$, такова, что φ не является функцией вида

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

где $y \in \mathbb{F}_{2^n}$, $c_0, c_{2^n-1} \in \mathbb{F}_2$, $\beta_1, \beta_2, \ldots, \beta_{n-1} \in \mathbb{F}_{2^n}$. Тогда для f существует ровно $2^{2n+1} - 2^n$ бент-функций, лежащих на минимальном расстоянии от неё, т. е. рассматриваемая оценка является достижимой.

Пример 1. Условию теоремы 3 при любом простом $n \geqslant 5$ удовлетворяют, в частности, функции $f(x, y) = \mathrm{tr}(xy^{2^n-2}) + \mathrm{tr}(y^5)$ и $g(x, y) = \mathrm{tr}(xy^{2^n-2}) + \mathrm{tr}(y^{11})$.

Не составляет труда посчитать число функций f из теоремы 3.

Следствие 1. Пусть $n \geqslant 5$ — простое. Тогда число функций из $\mathcal{M}_{2n}$, для которых достигается нижняя оценка $2^{2n+1} - 2^n$, не меньше $2^{2^n} - 2^{n^2-n+2}$.

Известны некоторые бент-функции, для которых точно посчитано количество бент-функций на минимально возможном расстоянии 2^n от них:

— это количество равно нулю для неслабонормальных бент-функций. Такие бент-функции рассматривались в работах [7, 8];

— это количество равно $2^n(2^1+1)\ldots(2^n+1)$ для квадратичных бент-функций [9]. Это значение является также верхней оценкой числа бент-функций на расстоянии 2^n от исходной бент-функции, при этом достигается она только для квадратичных бент-функций [10].

Теорема 3 даёт точное число бент-функций, лежащих на минимальном расстоянии от ещё ряда бент-функций из $\mathcal{M}_{2n}$.

2. Число бент-функций из $\mathcal{M}_{2n}$ на некотором расстоянии от бент-функций из $\mathcal{C}$

Класс $\mathcal{C}$ введён в работе [2], он состоит из функций вида

$$f(x,y) = \langle x, \pi(y)\rangle \oplus \mathrm{Ind}_{L^\perp}(x),$$

где L — линейное подпространство $\mathbb{F}_2^n$ и π — перестановка на $\mathbb{F}_2^n$, такая, что для любого $a \in \mathbb{F}_2^n$ множество $\pi^{-1}(a \oplus L)$ — аффинное подпространство.

Рассмотрим конструкцию $f \mapsto f \oplus \mathrm{Ind}_S$ для бент-функций $f(x,y) = \langle x, \pi(y)\rangle \oplus \oplus \mathrm{Ind}_{L^\perp}(x) \in \mathcal{C}$ и множеств S, не обязательно являющихся подпространствами. В следующей теореме для оценки использованы бент-функции $g(x,y) = \langle x, \tau(y)\rangle \oplus \varphi(y) \in \mathcal{M}_{2n}$, такие, что $\tau(y) = \pi(y) \oplus c$, $c \in \mathbb{F}_2^n$. Отметим, что число таких перестановок τ равно 2^n и мало по сравнению с числом всех перестановок $2^n!$.

Теорема 4. Количество бент-функций из класса $\mathcal{M}_{2n}$, лежащих на расстоянии m от бент-функций $f(x,y) = \langle x, \pi(y)\rangle \oplus \mathrm{Ind}_{L^\perp}(x) \in \mathcal{C}$ от $2n$ переменных, таких, что $\dim L^\perp = k$, не менее

1) $\binom{2^n}{w}$, если $m \in \{2^k(w(2^{n-k}-2)+2^n) : w = 0,1,\ldots,2^n\}$;
2) $\binom{2^n}{w}(2^{n-k}-1)$, если $m \in \{2^{2n-1} + 2^k(2^n - 2w) : w = 0,1,\ldots,2^n\}$, где $0 \leqslant k < n$.

Если рассмотреть расстояние 2^{2n-1} отдельно, то для него можно получить гораздо бóльшую нижнюю оценку, чем даёт предыдущая теорема.

Теорема 5. На расстоянии 2^{2n-1} от бент-функций из класса $\mathcal{C}$ от $2n$ переменных лежит не менее $2^{2^n}(2^{n-1})!$ бент-функций из класса $\mathcal{M}_{2n}$.

ЛИТЕРАТУРА

1. *Rothaus O.* On bent functions // J. Comb. Theory. Ser. A. 1976. V. 20. No. 3. P. 300–305.
2. *Carlet C.* Two new classes of bent functions // LNCS. 1993. V. 765. P. 77–101.
3. *Zhang F., Cepak N., Pasalic E., and Wei Y.* Further analysis of bent functions from $\mathcal{C}$ and $\mathcal{D}$ which are provably outside or inside $\mathcal{M}^\#$ // Discr. Appl. Math. 2020. V. 285. P. 458–472.
4. *Коломеец Н. А., Павлов А. В.* Свойства бент-функций, находящихся на минимальном расстоянии друг от друга // Прикладная дискретная математика. 2009. № 4(6). С. 5–20.
5. *Kolomeec N.* The graph of minimal distances of bent functions and its properties // Des. Codes Cryptogr. 2017. V. 85. P. 395–410.
6. *Логачев О. А., Сальников А. А., Ященко В. В.* Булевы функции в теории кодирования и криптологии. М.: МЦНМО, 2004.
7. *Canteaut A., Daum M., Dobbertin H., and Leander G.* Finding nonnormal bent functions // Discr. Appl. Math. 2006. V. 154. Iss. 2. P. 202–218.
8. *Leander G. and McGuire G.* Construction of bent functions from near-bent functions // J. Comb. Theory. Ser. A. 2009. V. 116. No. 4. P. 960–970.
9. *Коломеец Н. А.* Перечисление бент-функций на минимальном расстоянии от квадратичной бент-функции // Дискретн. анализ и исслед. опер. 2012. Т. 19. Вып. 1. С. 41–58.
10. *Коломеец Н. А.* Верхняя оценка числа бент-функций на расстоянии 2^k от произвольной бент-функции от $2k$ переменных // Прикладная дискретная математика. 2014. № 3(25). С. 28–39.

УДК 519.7 DOI 10.17223/2226308X/15/7

СВОЙСТВА ПОДФУНКЦИЙ САМОДУАЛЬНЫХ БЕНТ-ФУНКЦИЙ[1]

А. В. Куценко

Бент-функция называется самодуальной, если она совпадает со своей дуальной бент-функцией. Исследованы подфункции самодуальных бент-функций, полученные фиксацией первой переменной, а также первых двух переменных. Для описания подфункций от $n-1$ переменной введено понятие самодуальности почти бент-функции от нечётного числа переменных. Доказано, что между множествами самодуальных бент-функций от n переменных и почти бент-функций от $n-1$ переменной существует взаимно однозначное соответствие. Получено достаточное условие того, что подфункции от $n-2$ переменных самодуальной бент-функции являются бент-функциями. Предложен ряд новых итеративных конструкций бент-функций. Получена новая итеративная нижняя оценка числа самодуальных бент-функций.

Ключевые слова: *самодуальная бент-функция, подфункция, почти бент-функция, отношение Рэлея.*

Через $\mathbb{F}_2^n$ обозначим линейное пространство всех двоичных векторов длины n над полем $\mathbb{F}_2$. *Булевой функцией* от n переменных называется отображение вида $\mathbb{F}_2^n \to \mathbb{F}_2$. Множество всех булевых функций от n переменных обозначается через $\mathcal{F}_n$. *Характеристическим вектором* (характеристической последовательностью) булевой функции $f \in \mathcal{F}_n$ называется вектор

$$F \equiv (-1)^f = \left((-1)^{f(0)}, (-1)^{f(1)}, \ldots, (-1)^{f(2^n-1)}\right) \in \{\pm 1\}^{2^n},$$

где $(f(0), f(1), \ldots, f(2^n-1)) \in \mathbb{F}_2^{2^n}$ — вектор значений функции f. Для каждой пары $x, y \in \mathbb{F}_2^n$ через $\langle x, y\rangle$ обозначим значение $\bigoplus\limits_{i=1}^{n} x_i y_i$. *Преобразованием Уолша — Адамара* булевой функции f от n переменных называется целочисленная функция $W_f : \mathbb{F}_2^n \to \mathbb{Z}$, заданная равенством

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

Булева функция g от нечётного числа переменных m называется *почти бент-функцией*, если $W_f(y) \in \left\{0, \pm 2^{(m+1)/2}\right\}$ для каждого $y \in \mathbb{F}_2^n$. Булева функция f от чётного числа переменных n называется *почти бент-функцией*, если $W_f(y) \in \left\{0, \pm 2^{(n+2)/2}\right\}$ для каждого $y \in \mathbb{F}_2^n$.

Булева функция f от чётного числа переменных n называется *бент-функцией*, если $|W_f(y)| = 2^{n/2}$ для каждого $y \in \mathbb{F}_2^n$ [1]. Для множества бент-функций от n переменных используется обозначение $\mathcal{B}_n$. Для каждой $f \in \mathcal{B}_n$ из соотношения $W_f(y) = (-1)^{\widetilde{f}(y)} 2^{n/2}$ однозначным образом определяется *дуальная* к ней бент-функция $\widetilde{f} \in \mathcal{B}_n$, значения которой находятся из соответствия для каждого $y \in \mathbb{F}_2^n$. Бент-функция f называется *самодуальной* (*антисамодуальной*), если $f = \widetilde{f}$ (соответственно $f = \widetilde{f} \oplus 1$). Множество самодуальных бент-функций от n переменных обозначаются через $\mathcal{SB}_n^+$.

Открытой проблемой является полная характеризация и описание класса самодуальных бент-функций. Данные вопросы исследовались в ряде работ. В частности, в [2]

[1]Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования РФ № 075-15-2022-281.

приведена аффинная классификация самодуальных бент-функций от $2, 4, 6$ переменных и всех квадратичных самодуальных бент-функций от 8 переменных относительно преобразования, сохраняющего самодуальность. В работе [3] приведена классификация всех квадратичных самодуальных бент-функций. Аффинную классификацию квадратичных и кубических самодуальных бент-функций от 8 переменных относительно преобразования, сохраняющего самодуальность, можно найти в [4]. В работах [5, 6] представлены конструкции самодуальных бент-функций.

В настоящей работе исследованы подфункции самодуальных бент-функций, полученные фиксацией первой переменной, а также первых двух переменных. Другими словами, если $\mathbf{f}$ — вектор значений булевой функции f, то упомянутые подфункции есть в точности булевы функции с векторами значений f_i в представлении $\mathbf{f} = (f_0, f_1, \ldots, f_{2^k-1})$ для $k = 1, 2$ соответственно. Предложены новые конструкции, а также новая итеративная нижняя оценка числа самодуальных бент-функций.

1. Подфункции от $n-1$ переменной

Известно, что подфункции от $n-1$ переменной каждой бент-функции являются почти бент-функциями, при этом носители их спектров Уолша — Адамара не пересекаются [7].

Отношением Рэлея булевой функции f от n переменных называется число

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

Применительно к бент-функциям данная характеристика изучалась в работе [8]. Она представляет интерес в силу того, что число S_f полностью характеризует расстояние Хэмминга между бент-функцией и дуальной к ней. Заметим, что задача поиска максимального (минимального) значения отношения Рэлея решена лишь для случая чётного числа переменных. Для нечётного случая она является открытой проблемой.

Известно [2], что для каждой булевой функции f от чётного числа n переменных справедливо $|S_f| \leqslant 2^{3n/2}$, при этом равенство достигается в том и только в том случае, когда f — самодуальная $\left(+2^{3n/2}\right)$ или антисамодуальная $\left(-2^{3n/2}\right)$ бент-функция. Нетрудно видеть, что в случае чётного числа переменных экстремальные значения отношения Рэлея достигаются лишь на тех булевых функциях, для которых $(-1)^{f(y)} W_f(y) = 2^{n/2}$ или $(-1)^{f(y)} W_f(y) = -2^{n/2}$ при каждом $y \in \mathbb{F}_2^n$.

Введём следующее понятие самодуальности почти бент-функции от нечётного числа переменных. Пусть m — нечётное положительное число. Почти бент-функцию g от m переменных будем называть *самодуальной*, если

$$(-1)^{g(y)} W_g(y) \geqslant 0 \text{ для любого } y \in \mathbb{F}_2^m.$$

В свою очередь, функция g называется *антисамодуальной* почти бент-функцией, если

$$(-1)^{g(y)} W_g(y) \leqslant 0 \text{ для любого } y \in \mathbb{F}_2^m.$$

Содержательно оба определения описывают ситуации, когда знаки коэффициентов Уолша — Адамара и значения характеристического вектора почти бент-функции согласованы. Можно показать, что

Утверждение 1. Пусть g — почти бент-функция от m переменных, тогда

$$|S_g| \leqslant 2^{(3m-1)/2},$$

при этом равенство достигается, если и только если g — самодуальная $\left(+2^{(3m-1)/2}\right)$ или антисамодуальная $\left(-2^{(3m-1)/2}\right)$ почти бент-функция.

То есть (анти-)самодуальные почти бент-функции от нечётного числа переменных на множестве почти бент-функций, так же как и самодуальные бент-функции на множестве булевых функций от чётного числа переменных, являются экстремальными объектами в спектральном смысле.

Понятия самодуальности для чётного и нечётного числа переменных тесно связаны, что показывает следущая

Теорема 1. Между множествами самодуальных бент-функций от $n \geqslant 4$ переменных и множеством (анти-)самодуальных почти бент-функций от $n-1$ переменных существует взаимно однозначное соответствие.

Упомянутая связь устанавливается на основе отображения, которое каждой самодуальной бент-функции ставит в соответствие её подфункцию, получаемую фиксацией первой координаты.

Таким образом, подфункциями от $n-1$ переменной, получаемыми фиксацией первой координаты, являются самодуальные почти бент-функции, и только они.

2. Свойства подфункций от $n-2$ переменных

Известно [9], что для каждой бент-функции от n переменных все её подфункции от $n-2$ переменных имеют одинаковые спектры Уолша — Адамара. Более того, все подфункции являются бент-функциями, либо все являются почти бент-функциями, либо их спектры Уолша — Адамара состоят из чисел 0, $\pm 2^{(n-2)/2}$, $\pm 2^{n/2}$.

Случай, когда все подфункции являются бент-функциями, ведёт к итеративной конструкции бент функции от $n+2$ переменных на основе четырёх бент-функций от n переменных. В работе [10] найдены необходимые и достаточные условия того, что конкатенация векторов значений четырёх бент-функций от n переменных даёт вектор значений бент-функции от $n+2$ переменных.

Известны две итеративные конструкции самодуальных бент-функций от $n+2$ переменных, в основе которых лежит конкатенация четырёх векторов значений бент-функций от n переменных. Они представлены ниже:

— конструкция **C1**: $\left(h, \widetilde{h}, \widetilde{h}, h \oplus 1\right)$, где h — бент-функция от n переменных [2];

— конструкция **C2**: $(f, g \oplus 1, g, f)$, где f — самодуальная, а g — антисамодуальная бент-функции от n переменных [11].

Сумма мощностей данных непересекающихся конструкций **C1** и **C2** даёт нижнюю оценку $|\mathcal{B}_{n-2}| + \left|\mathcal{SB}^{+}_{n-2}\right|^2$ числа самодуальных бент-функций. Прямые вычисления показывают, что данная оценка превосходит другие известные оценки.

Очевидно, что для обоих конструкций характеристические векторы подфункций образуют линейно зависимые множества. В настоящей работе доказано утверждение, обобщающее данный факт:

Теорема 2. Если характеристические векторы подфункций f_0, f_1, f_2, f_3 самодуальной бент-функции f линейно зависимы, то данные подфункции являются бент-функциями.

Теорема 2 даёт достаточное условие того, что все подфункции, полученные фиксацией первых двух переменных, являются бент-функциями.

Отметим, что для случая $n = 4$ данное условие также является достаточным.

3. Новые конструкции и оценка числа самодуальных бент-функций

В настоящей работе мы предлагаем три новых конструкции **C3**, **C4** и **C5** самодуальных бент-функций. В данных конструкциях используются бент-функции от $n-4$ переменных. Пусть h — бент-функция от $n-4$ переменных, f — самодуальная и g — антисамодуальная бент-функции от $n-4$ переменных. Опишем конструкции:

— **C3**: вектор значений функции имеет вид

$$\big(h, g, g\oplus 1, h, \widetilde{h}, f, f\oplus 1, \widetilde{h}, \widetilde{h}, f\oplus 1, f, \widetilde{h}, h\oplus 1, g, g\oplus 1, h\oplus 1\big);$$

все подфункции от $n-2$ переменных являются бент-функциями;

— **C4**: вектор значений функции имеет вид

$$\big(h, g, \widetilde{h}, f, g\oplus 1, h, f\oplus 1, \widetilde{h}, \widetilde{h}, f\oplus 1, h\oplus 1, g, f, \widetilde{h}, g\oplus 1, h\oplus 1\big);$$

подфункции от $n-2$ переменных являются бент-функциями тогда и только тогда, когда $h\oplus\widetilde{h}\oplus f\oplus g=0$. Таким образом, данная конструкция даёт класс самодуальных бент-функций, которые нельзя представить в виде конкатенации четырёх бент-функций;

— **C5**: вектор значений функции имеет вид

$$\big(h, h\oplus 1, \widetilde{h}, \widetilde{h}, h, h, \widetilde{h}\oplus 1, \widetilde{h}, \widetilde{h}, \widetilde{h}, h\oplus 1, h, \widetilde{h}\oplus 1, \widetilde{h}, h\oplus 1, h\oplus 1\big);$$

все подфункции от $n-2$ переменных являются бент-функциями.

На основе анализа данных конструкций получена

Теорема 3. Число самодуальных бент-функций от $n\geqslant 6$ переменных не меньше чем

$$|\mathcal{B}_{n-2}|+\left|\mathcal{SB}^{+}_{n-2}\right|^2+|\mathcal{B}_{n-4}|\left(2\left|\mathcal{SB}^{+}_{n-4}\right|^2+1\right)-2\left|\mathcal{SB}^{+}_{n-4}\right|.$$

Таким образом, известная итеративная нижняя оценка увеличивается на слагаемое, соответствующее самодуальным бент-функциям от $n-4$ переменных.

4. Линейная независимость характеристических векторов подфункций

Конструкции, предложенные в работе, позволяют однозначно ответить на вопрос о возможности обращения теоремы 2 для случая $n\geqslant 6$.

Теорема 4. Для каждого чётного $n\geqslant 6$ существуют самодуальные бент-функции от n переменных, подфункции которых образуют линейно независимые множества векторов.

Таким образом, обращение теоремы 2 не имеет места при $n\geqslant 6$, то есть линейная зависимость характеристических векторов не является необходимым условием, а обеспечивает лишь *достаточное* условие того, что подфункции от $n-2$ переменных являются бент-функциями.

ЛИТЕРАТУРА

1. *Rothaus O.* On bent functions // J. Combin. Theory. Ser. A. 1976. V. 20. No. 3. P. 300–305.
2. *Carlet C., Danielsen L. E., Parker M. G., and Solé P.* Self-dual bent functions // Int. J. Inform. Coding Theory. 2010. V. 1. P. 384–399.
3. *Hou X.-D.* Classification of self dual quadratic bent functions // Des. Codes Cryptogr. 2012. V. 63. No. 2. P. 183–198.

4. *Feulner T., Sok L., Solé P., and Wassermann A.* Towards the classification of self-dual bent functions in eight variables // Des. Codes Cryptogr. 2013. V. 68. No. 1. P. 395–406.
5. *Luo G., Cao X., and Mesnager S.* Several new classes of self-dual bent functions derived from involutions // Cryptogr. Commun. 2019. V. 11. No. 6. P. 1261–1273.
6. *Li Y., Kan H., Mesnager S., et al.* Generic constructions of (Boolean and vectorial) bent functions and their consequences // IEEE Trans. Inform. Theory. 2022. V. 68. No. 4. P. 2735–2751.
7. *Wolfmann A.* Special bent and near-bent functions // Adv. Math. Commun. 2014. V. 8. No. 1. P. 21–33.
8. *Danielsen L. E., Parker M. G., and Solé P.* The Rayleigh quotient of bent functions // LNCS. 2009. V. 5921. P. 418–432.
9. *Canteaut A. and Charpin P.* Decomposing bent functions // IEEE Trans. Inf. Theory. 2003. V. 49. No. 8. P. 2004–2019.
10. *Preneel B., Van Leekwijck W., Van Linden L., et al.* Propagation characteristics of Boolean functions // LNCS. 1990. V. 473. P. 161–173.
11. *Kutsenko A.* Metrical properties of self-dual bent functions // Des. Codes Cryptogr. 2020. V. 88. No. 1. P. 201–222.

УДК 519.7 DOI 10.17223/2226308X/15/8

ГЕНЕРАЦИЯ ВЕКТОРНЫХ БУЛЕВЫХ ФУНКЦИЙ С НЕВЫРОЖДЕННЫМИ КООРДИНАТНЫМИ ФУНКЦИЯМИ

И. А. Панкратова, Е. А. Рубан, С. В. Чикалова

Предложен алгоритм генерации обратимой векторной булевой функции, все координатные функции которой существенно зависят от всех переменных.

Ключевые слова: *векторные булевы функции, подстановки, существенная зависимость функции от переменной.*

Обозначим через $P_2(n)$ множество всех булевых функций от n переменных. Говорят, что переменная x_i, $1 \leqslant i \leqslant n$, *существенная для функции* $f(x_1, \ldots, x_n) \in P_2(n)$ (f *существенно зависит от переменной* x_i), если найдётся пара наборов $a, b \in \mathbb{Z}_2^n$, соседних по i-й координате, такая, что $f(a) \neq f(b)$; будем называть такую пару наборов *доказывающей существенность переменной* x_i *для функции* f (или просто *доказывающей парой*, если из контекста ясно, о каких переменной и функции идёт речь). Переменная, от которой функция не зависит существенно, называется *фиктивной для этой функции*; функции, существенно зависящие от всех переменных, — *невырожденными*.

Векторной булевой функцией $((n, m) -)$ называется отображение $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$. Такую функцию можно рассматривать как упорядоченный набор из m булевых функций f_i, $i = 1, \ldots, m$, которые называются *координатными функциями*: $F = (f_1 \ldots f_m)$.

Оценим вероятность того, что случайно сгенерированная векторная булева функция невырожденная. По формуле включений и исключений определим D_n — количество функций в $P_2(n)$, имеющих фиктивные переменные:

$$D_n = \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} 2^{2^{n-i}}.$$

Вероятность того, что булева функция от n переменных невырожденная, равна

$$\mathsf{P}_n = 1 - D_n / 2^{2^n}. \tag{1}$$

Вычисления по формуле (1) показывают, что $\mathsf{P}_n \approx 1$ при $n \geqslant 5$. Для случайной (n, m)-функции получаем вероятность невырожденности

$$\mathsf{P}_{n,m} = \mathsf{P}_n^m = \left(1 - D_n/2^{2^n}\right)^m. \tag{2}$$

Рассмотрим случай, когда $n = m$ и F обратима, т. е. является подстановкой на $\mathbb{Z}_2^n$. Подстановки используются во многих криптосистемах, в частности в криптосистемах с функциональными ключами [1, 2]. Для обеспечения стойкости криптосистем функции в них должны обладать определёнными свойствами; одно из таких свойств — существенная зависимость всех координат от всех переменных (т. е. их невырожденность). Векторную булеву функцию назовём *невырожденной*, если все её координатные функции невырожденные.

Для подстановок не удалось найти или получить формулу, аналогичную (2), но проведённые эксперименты дают тот же результат: начиная с $n = 5$, почти все случайно сгенерированные подстановки оказываются невырожденными. Тем не менее считаем, что задача генерации невырожденной подстановки не теряет своей значимости; например, в некоторой криптосистеме могут использоваться не любые подстановки, а из какого-то класса, для которого доля невырожденных подстановок в нём может оказаться другой (не близкой к 100 %).

Ранее в [3] предложен алгоритм генерации невырожденных подстановок с помощью n независимых транспозиций из тождественной подстановки или из такой, все координатные функции которой имеют ровно одну существенную переменную; обозначим классы подстановок, доставляемых этим алгоритмом, через $\mathcal{K}_n$ и $\mathcal{K}'_n$ соответственно. В [4, 5] описаны свойства функций этих классов, некоторые из них оказались неудовлетворительными с точки зрения криптографической стойкости. Кроме того, алгоритм в [3] не обладает свойством полноты, т. е. не может получить *любую* невырожденную подстановку. В данной работе эта проблема решена.

Докажем вспомогательные утверждения для булевых функций.

Утверждение 1. Пусть $f \in P_2(n)$, $f \neq \mathrm{const}$ и x_i — фиктивная переменная для f. Выберем два произвольных набора $a, b \in \mathbb{Z}_2^n$, таких, что $f(a) \neq f(b)$, и построим функцию $g \in P_2(n)$ так: $g(a) = f(b)$, $g(b) = f(a)$, $g(c) = f(c)$ для всех $c \in \mathbb{Z}_2^n \setminus \{a, b\}$. Тогда функция g существенно зависит от переменной x_i.

Доказательство. Пусть c — набор, соседний с a по i-й координате. Заметим, что $c \neq b$, так как x_i — фиктивная переменная для f и $f(a) \neq f(b)$. Получим

$$g(c) = f(c) = f(a) \neq f(b) = g(a),$$

т. е. x_i — существенная переменная для g. ■

Замечание 1. Поскольку выбор наборов a, b в утверждении 1 не зависит от номера фиктивной переменной, все переменные, фиктивные для f, являются существенными для построенной функции g.

Обозначим через $\mathbf{e}_i$, $1 \leqslant i \leqslant n$, булев вектор длины n с единственной единицей в i-й координате; $\mathrm{w}(f)$ — вес функции f.

Утверждение 2. Пусть $f \in P_2(n)$, $n \geqslant 3$, f уравновешенная и имеет фиктивную переменную. Тогда для каждой её существенной переменной найдётся не менее трёх доказывающих пар.

Доказательство. Пусть x_i — существенная и x_j — фиктивная переменные для функции f. Тогда $f(y) \neq f(y \oplus \mathbf{e}_i)$ для некоторого $y \in \mathbb{Z}_2^n$. В силу фиктивности переменной x_j можем записать, что

$$f(y \oplus \mathbf{e}_j) = f(y) \neq f(y \oplus \mathbf{e}_i) = f(y \oplus \mathbf{e}_i \oplus \mathbf{e}_j),$$

т. е. имеем две доказывающие пары: $\langle y, y \oplus \mathbf{e}_i \rangle$ и $\langle y \oplus \mathbf{e}_j, y \oplus \mathbf{e}_i \oplus \mathbf{e}_j \rangle$.

Разобьём множество $\mathbb{Z}_2^n$ на четвёрки наборов вида $\langle x, x \oplus \mathbf{e}_i, x \oplus \mathbf{e}_j, x \oplus \mathbf{e}_i \oplus \mathbf{e}_j \rangle$; одна из них (при $x = y$) содержит две доказывающие пары, а функция f на наборах этой четвёрки дважды принимает значение 0 и дважды 1.

Предположим, что больше доказывающих пар нет, тогда ввиду $f(x \oplus \mathbf{e}_j) = f(x) = f(x \oplus \mathbf{e}_i) = f(x \oplus \mathbf{e}_i \oplus \mathbf{e}_j)$ на наборах из каждой четвёрки (кроме случая $x = y$) функция f принимает одно и то же значение; пусть это значение равно 1 на k четвёрках. Получаем

$$\mathrm{w}(f) = 2 + 4k = 2^{n-1};$$

второе равенство должно выполняться в силу уравновешенности f, но оно невозможно при $n \geqslant 3$. ■

Следствие 1. Пусть для функции f выполнены условия утверждения 2. Тогда функция g, построенная способом, описанным в утверждении 1, невырожденная.

Доказательство. Если переменная фиктивная для функции f, то она является существенной для g (см. замечание 1).

Если переменная x_i существенная для функции f, то по утверждению 2 для неё существует не менее трёх доказывающих пар. Функция g отличается от функции f на двух наборах, поэтому независимо от того, как будут выбраны эти наборы, хотя бы одна пара, доказывающая существенность переменной x_i для функции g, останется. ■

На основании следствия 1 с учётом того, что координаты любой подстановки на $\mathbb{Z}_2^n$ уравновешены, получаем следующие два утверждения.

Утверждение 3. Пусть $n \geqslant 3$; $a, b \in \mathbb{Z}_2^n$ — произвольные взаимно инверсные наборы; G — подстановка на $\mathbb{Z}_2^n$, такая, что $G(a) = b$, $G(b) = a$, $G(c) = c$ для всех $c \in \mathbb{Z}_2^n \setminus \{a, b\}$. Тогда G невырожденная.

Утверждение 4. Пусть $n \geqslant 3$; F — подстановка на $\mathbb{Z}_2^n$, каждая координата которой имеет хотя бы одну фиктивную переменную; $F(x) = a$ и $F(y) = b$, где $a, b \in \mathbb{Z}_2^n$ — произвольные взаимно инверсные наборы. Построим подстановку G на $\mathbb{Z}_2^n$ так: $G(x) = b$, $G(y) = a$, $G(c) = F(c)$ для всех $c \in \mathbb{Z}_2^n \setminus \{a, b\}$. Тогда G невырожденная.

Оба утверждения имеют недостатки — функция G в утверждении 3 близка к тождественной, а в утверждении 4 строится из «особой» подстановки F, которую ещё надо как-то получить. Оба недостатка преодолены в алгоритме 1.

Полнота алгоритма 1 обеспечивается выбором любой подстановки на шаге 1.

Корректность алгоритма следует из того, что для функций f_i, которые меняются на шаге 6, выполнены условия следствия 1, следовательно, после обмена $F(x) \leftrightarrow F(y)$ они станут невырожденными. Остальные координатные функции F невырожденные уже с шага 1.

Для выполнения шага 1 в алгоритме 2 можно адаптировать на случай векторной функции алгоритм из [6, разд. 6.2].

Алгоритм 1. Генерация невырожденной подстановки на $\mathbb{Z}_2^n$

Вход: $n \in \mathbb{N}$, $n \geqslant 3$.

Выход: подстановка $F = (f_1 \dots f_n) : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$, такая, что функции f_i существенно зависят от всех переменных, $i = 1, \dots, n$.

1: Построить случайную подстановку $F = (f_1 \dots f_n)$ на $\mathbb{Z}_2^n$ любым известным алгоритмом (например, методом тасования Фишера — Йетса [7]).
2: Построить вектор $v = (v_1 \dots v_n) \in \mathbb{Z}_2^n$, такой, что $v_i = 1 \Leftrightarrow f_i$ имеет фиктивную переменную, $i = 1, \dots, n$ (см. алгоритм 2).
3: **Если** $v = 0 \dots 0$, **то**
   выход, ответ — F.
4: Выбрать любой набор $x \in \mathbb{Z}_2^n$.
5: Найти $y \in \mathbb{Z}_2^n$, такой, что $F(y) = F(x) \oplus v$.
6: Поменять местами значения $F(x)$ и $F(y)$.
7: Выход, ответ — F.

Алгоритм 2. Анализ вырожденности векторной булевой функции

Вход: функция $F = (f_1 \dots f_m) : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$.

Выход: вектор $v = (v_1 \dots v_m) \in \mathbb{Z}_2^m$, такой, что $v_i = 1 \Leftrightarrow f_i$ имеет фиктивную переменную, $i = 1, \dots, m$.

1: Выполнить преобразование Мёбиуса $G = (g_1 \dots g_m) = \mu(F)$, где $g_i(a_1, \dots, a_n) = 1 \Leftrightarrow$ моном $x_1^{a_1} \dots x_n^{a_n}$ входит в АНФ функции f_i.
2: **Для всех** $i = 1, \dots, m$:
3: вычислить $c := \bigvee\limits_{a \in \mathbb{Z}_2^n : g_i(a) = 1} a$;
4: **Если** $c = 1, \dots, 1$, **то**
   $v_i := 0$,
5: **иначе**
6: $v_i := 1$.

ЛИТЕРАТУРА

1. *Agibalov G. P.* Substitution block ciphers with functional keys // Прикладная дискретная математика. 2017. № 38. С. 57–65.
2. *Agibalov G. P. and Pankratova I. A.* Asymmetric cryptosystems on Boolean functions // Прикладная дискретная математика. 2018. № 40. С. 23–33.
3. *Pankratova I. A.* Construction of invertible vectorial Boolean functions with coordinates depending on given number of variables // Материалы Междунар. науч. конгресса по информатике: Информационные системы и технологии. Республика Беларусь, Минск, 24–27 окт. 2016. Минск: БГУ, 2016. С. 519–521.
4. *Карпова Л. А., Панкратова И. А.* Свойства координатных функций одного класса подстановок на $\mathbb{F}_2^n$ // Прикладная дискретная математика. Приложение. 2017. № 10. С. 38–40.
5. *Панкратова И. А.* Свойства компонент некоторых классов векторных булевых функций // Прикладная дискретная математика. 2019. № 44. С. 5–11.
6. *Панкратова И. А.* Булевы функции в криптографии. СПб., М., Краснодар: Лань, 2019.
7. *Кнут Д. Э.* Искусство программирования. Т. 2. Получисленные алгоритмы. 3-е изд. М.: Вильямс, 2007.

УДК 519.7 DOI 10.17223/2226308X/15/9

О КОРРЕЛЯЦИОННО-ИММУННЫХ ФУНКЦИЯХ С МАКСИМАЛЬНОЙ АЛГЕБРАИЧЕСКОЙ ИММУННОСТЬЮ[1]

И. С. Хильчук, Д. А. Зюбина, Н. Н. Токарева

Рассматривается геометрическое представление корреляционно-иммунных булевых функций с максимальной алгебраической иммунностью. Найдено пересечение классов функций с максимальной алгебраической иммунностью и функций, обладающих корреляционной иммунностью, от малого числа переменных. Для $n = 3, 4, 5$ произведена классификация таких функций.

Ключевые слова: *булевы функции, алгебраическая иммунность, корреляционная иммуность, булев куб.*

Булевы функции являются основными компонентами симметричных шифров, их криптографические свойства обеспечивают стойкость шифра к различным видам криптоанализа. Например, в работе [1] рассмотрены связи корреляционной иммунности с другими свойствами булевых функций. В данной работе основным рассматриваемым свойством является алгебраическая иммунность, обеспечивающая стойкость шифра к алгебраическому криптоанализу, введённому Н. Куртуа в 2003 г. [2]. Корреляционная иммунность используется в качестве вспомогательного свойства для сокращения числа рассматриваемых функций и поиска закономерностей и интересных структур, так как накладывает ограничения на расположение носителя булевой функции в булевом кубе.

Так как полный перебор множества всех булевых функций затруднён, наибольший интерес представляют подходы, при которых булевы функции с нужными свойствами находятся с помощью средств машинного обучения, таких, как генетические алгоритмы [3], или строятся, как, например, в [4]. В данной работе изучается способ построения булевых функций от большего числа переменных на основе функций от меньшего числа переменных с сохранением криптографических свойств исходных функций. Используется геометрическое представление булевой функции: носитель булевой функции рассматривается как подможество вершин булева куба соответствующей размерности.

Любую булеву функцию можно единственным образом записать в *алгебраической нормальной форме* (АНФ, полином Жегалкина):

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

где при каждом k все индексы $i_1, \ldots, i_k$ различны и параметры $a_0, a_{i_1,\ldots,i_k}$ принимают значения 0 или 1.

Носитель булевой функции — множество всех векторов, на которых функция принимает значение 1:

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

Булев куб — граф $\mathbb{E}^n$, вершинами которого являются все двоичные векторы длины n, т. е. $V = \{(x_1, \ldots, x_n) : x_i \in \mathbb{Z}_2\}$, а рёбрами соединяются только те векторы, расстояние Хэмминга между которыми равно единице. Число n называется *размерностью* булева куба.

[1]Работа выполнена в рамках госзадания ИМ СО РАН (проект № FWNF-2022-0018).

Весом Хэмминга $\mathrm{wt}(f)$ булевой функции f от n переменных называется число ненулевых координат её вектора значений. *Гранью размерности* k в булевом кубе $\mathbb{E}^n$ называется множество

$$\Gamma^{a_1,\ldots,a_{n-k}}_{i_1,\ldots,i_{n-k}} = \{x_{i_1} = a_1, \ldots, x_{i_{n-k}} = a_{n-k}\}.$$

Булева функция $f(x)$ от n переменных называется *сбалансированной*, если принимает каждое из значений 0 и 1 ровно 2^{n-1} раз [5].

Алгебраической иммунностью $\mathrm{AI}(f)$ булевой функции f от n переменных называется минимальное число d, такое, что существует не тождественно равная нулю булева функция g от n переменных степени d, для которой выполняется $f \cdot g = 0$ или $(f+1)g = 0$ [5]. Высокая алгебраическая иммунность обеспечивает стойкость шифра к алгебраическому криптоанализу. Известна оценка сверху алгебраической иммунности булевой функции от n переменных: $\mathrm{AI}(f) \leqslant \lceil n/2 \rceil$ [2].

Булева функция f от n переменных называется *корреляционно-иммунной порядка* r, $1 \leqslant r \leqslant n$, если для любой её подфункции $f^{a_1,\ldots,a_r}_{i_1,\ldots,i_r}$, полученной фиксацией r переменных, выполняется равенство

$$\mathrm{wt}(f^{a_1,\ldots,a_r}_{i_1,\ldots,i_r}) = \frac{\mathrm{wt}(f)}{2^r}.$$

Эквивалентное определение: булева функция f от n переменных является корреляционно-иммунной порядка $n-k$, $1 \leqslant k \leqslant n$, если любой грани $\Gamma^{a_1,\ldots,a_{n-k}}_{i_1,\ldots,i_{n-k}}$ булева куба $\mathbb{E}^n$ размерности k принадлежит одинаковое число точек носителя функции f, а именно $2^{-(n-k)}\mathrm{wt}(f)$ точек.

Так как корреляционно-иммунная порядка m булева функция является также корреляционно-иммунной порядка ℓ для всех $\ell < m$ [6], можем ввести обозначение для максимального порядка корреляционной иммунности:

$$\mathrm{CI}(f) = \max\{m \in \mathbb{N} : f \text{ — корреляционно-иммунная порядка } m\}.$$

Пусть f — булева функция от n переменных. *Геометрическим представлением* булевой функции f назовём подграф булева куба $\mathbb{E}^n$, индуцированный носителем функции f. Напомним, что два графа F и G называются *изоморфными*, если существует биекция ψ на множествах их вершин, такая, что образы $\psi(u)$ и $\psi(v)$ в графе G смежны тогда и только тогда, когда смежны вершины u и v в графе F.

Два подграфа G_1 и G_2 булева куба $\mathbb{E}^n$, индуцированные носителями функций f_1 и f_2 соответственно, назовём *изоморфными по метрическому вложению*, если найдётся автоморфизм $\varphi : \mathbb{E}^n \to \mathbb{E}^n$ булева куба $\mathbb{E}^n$, под действием которого G_1 переходит в G_2. Напомним, что любой автоморфизм булева куба $\mathbb{E}^n$ как графа является изометрией $\mathbb{E}^n$ относительно расстояния Хэмминга.

1. Булевы функции от трёх переменных

Для $n = 3$ получена полная классификация булевых функций с корреляционной иммунностью порядков 3, 2, 1. Всего существуют:

- 2 функции порядка 3 — функции-константы;
- 4 функции порядка 2 — функции-константы и функции-счётчики чётности;
- 18 функций порядка 1.

Из всех этих функций ни одна не имеет максимально возможного значения алгебраической иммунности (т. е. $\mathrm{AI}(f) < 2$).

2. Булевы функции от четырёх переменных

Исследуем пересечение множеств булевых функций от четырёх переменных с максимальной алгебраической иммунностью (равной 2) и с максимальным порядком корреляционной иммунности 1. Заметим, что среди функций от четырёх переменных с максимальной алгебраической иммунностью не существует функций с более высоким максимальным порядком корреляционной иммунности.

В этом пересечении лежат 392 функции, среди которых функции веса 6 (96 функций), 8 (200 функций) и 10 (96 функций), при этом функции веса 6 совпадают с инверсиями функций веса 10. Ограничения на вес функций накладывают рассматриваемые характеристики: корреляционная иммунность требует чётного веса, а для того, чтобы алгебраическая иммунность имела максимальный показатель $\mathrm{AI}(f) = 2$, необходимо, чтобы выполнялись следующие ограничения: $\mathrm{wt}(f) \geqslant \sum\limits_{i=0}^{d} \binom{n}{i}$ и $\mathrm{wt}(f \oplus 1) \geqslant \sum\limits_{i=0}^{d} \binom{n}{i}$, где $d = \mathrm{AI}(f) - 1 = 1$, $n = 4$.

Будем рассматривать только функции, которые принимают значение 1 на нулевом векторе: 36 функций веса 6, 100 функций веса 8, 60 функций веса 10. Мы можем это сделать, так как инвертирование функции не изменяет показатели её алгебраической и корреляционной иммунности.

Заметим, что все функции разбиваются на классы сообразно своему геометрическому представлению (или виду АНФ). Переход от одной булевой функции к другой внутри класса осуществляется переобозначением переменных.

Произведена классификация данных булевых функций и написана программа, с помощью которой на основе рассматриваемых 196 функций от четырёх переменных построены функции от шести переменных и проверена их алгебраическая и корреляционная иммунность.

2.1. Булевы функции веса 6 с $\mathrm{AI}(f) = 2$, $\mathrm{CI}(f) = 1$

Рассмотрим функции веса 6 с $\mathrm{AI}(f) = 2$, $\mathrm{CI}(f) = 1$, принимающие значение 1 на нулевом векторе, и индуцированные их носителями подграфы булева куба $\mathbb{E}^4$ (рис. 1). Здесь и далее метками у тонких линий обозначены расстояния между несмежными вершинами графа. Подграфы данного вида симметричны относительно вертикальной оси, мы не различаем симметричные вершины.

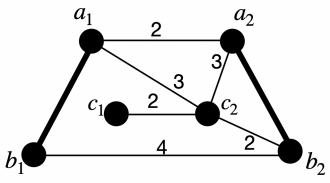


Рис. 1. Подграф из G_6

Все подграфы, индуцированные носителями таких функций, изоморфны по вложению между собой. Обозначим данное множество индуцированных подграфов как G_6.

Утверждение 1. Булева функция f от четырёх переменных веса 6, принимающая значение 1 на нулевом векторе, имеет характеристики $\mathrm{AI}(f) = 2$ и $\mathrm{CI}(f) = 1$, если и только если индуцированный её носителем подграф булева куба $\mathbb{E}^4$ изоморфен по вложению графу на рис. 1. От того, какой вершиной индуцированного носителем подграфа является нулевой вектор булева куба, зависит вид алгебраической нормальной формы функции f:

1) нулевой вектор — изолированная вершина (12 функций), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_1x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus x_4 \oplus x_3 \oplus x_2 \oplus x_1 \oplus 1;$$

2) нулевой вектор — вершина a_i, $i = 1, 2$ (12 функций), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_3x_4 \oplus x_2x_4 \oplus x_4 \oplus x_3 \oplus x_2 \oplus 1;$$

3) нулевой вектор — вершина b_i, $i = 1, 2$ (12 функций), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_3x_4 \oplus x_4 \oplus x_3 \oplus x_2 \oplus 1.$$

2.2. Булевы функции веса 10 с $\mathrm{AI}(f) = 2$, $\mathrm{CI}(f) = 1$

Подграфы булева куба, индуцированные носителями функций веса 10 с $\mathrm{AI}(f) = 2$ и $\mathrm{CI}(f) = 1$, принимающих значение 1 на нулевом векторе, также изоморфны по вложению между собой; обозначим множество таких подграфов как G_{10} (рис. 2).

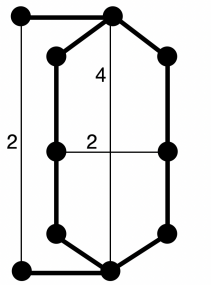


Рис. 2. Подграф из G_{10}

Утверждение 2. Булева функция f от четырёх переменных веса 10, принимающая значение 1 на нулевом векторе, имеет характеристики $\mathrm{AI}(f) = 2$ и $\mathrm{CI}(f) = 1$, если и только если индуцированный её носителем подграф булева куба $\mathbb{E}^4$ изоморфен по вложению графу на рис. 2. От того, какой вершиной индуцированного носителем подграфа является нулевой вектор булева куба, зависит вид алгебраической нормальной формы функции f:

1) нулевой вектор — вершина степени 1 (12 функций), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_1x_2 \oplus x_1x_3 \oplus x_2 \oplus x_3 \oplus x_4 \oplus 1;$$

2) нулевой вектор — вершина степени 2, равноудалённая от вершин степени 3 (12 функций), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_1x_2 \oplus x_1x_4 \oplus x_2x_3 \oplus x_3 \oplus x_4 \oplus 1;$$

3) нулевой вектор — вершина степени 2, не равноудалённая от вершин степени 3 (24 функции), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus x_2x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus$$

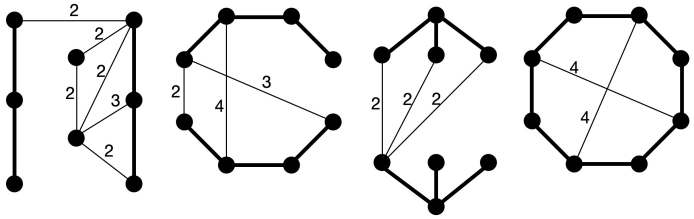
$$\oplus x_1 \oplus x_4 \oplus 1;$$

4) нулевой вектор — вершина степени 3 (12 функций), пример АНФ:

$$x_1x_2x_3 \oplus x_1x_2x_4 \oplus x_1x_3x_4 \oplus x_2x_3x_4 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus x_2x_3 \oplus x_2x_4 \oplus x_4 \oplus 1.$$

2.3. Булевы функции веса 8 с $\mathrm{AI}(f) = 2$, $\mathrm{CI}(f) = 1$

Индуцированные носителями функций веса 8 с $\mathrm{AI}(f) = 2$, $\mathrm{CI}(f) = 1$ подграфы имеют четыре возможных вида, обозначим их G_8^i, $i = 1, 2, 3, 4$ (рис. 3). Подграфы в каждом из G_8^i, $i = 1, 2, 3, 4$, изоморфны по вложению между собой.



а *б* *в* *г*

Рис. 3. Подграфы из G_8

Утверждение 3. Булева функция f от четырёх переменных веса 8, принимающая значение 1 на нулевом векторе, имеет характеристики $\mathrm{AI}(f) = 2$ и $\mathrm{CI}(f) = 1$, если и только если индуцированный её носителем подграф булева куба $\mathbb{E}^4$ изоморфен по вложению одному из графов на рис. 3. От того, какой вид имеет индуцированный носителем функции подграф и какой вершиной этого подграфа является нулевой вектор булева куба, зависит вид алгебраической нормальной формы функции f:

1. Два 2-пути и две изолированные вершины (рис. 3, *а*):
   а) нулевой вектор — вершина степени 2 (6 функций), пример АНФ:
   $$x_1x_2 \oplus x_3 \oplus x_4 \oplus 1;$$
   б) нулевой вектор — вершина степени 1 (12 функций), пример АНФ:
   $$x_1x_2 \oplus x_2 \oplus x_3 \oplus x_4 \oplus 1;$$
   в) нулевой вектор — изолированная вершина (6 функций), пример АНФ:
   $$x_3x_4 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus 1.$$
2. Два 3-пути (рис. 3, *б*):
   а) нулевой вектор — вершина степени 2 (24 функции), пример АНФ:
   $$x_1x_2 \oplus x_2x_3 \oplus x_3 \oplus x_4 \oplus 1;$$
   б) нулевой вектор — вершина степени 1н (24 функции), пример АНФ:
   $$x_1x_2 \oplus x_3x_4 \oplus x_2 \oplus x_3 \oplus x_4 \oplus 1.$$
3. Две вершины степени 3 (рис. 3, *в*):
   а) нулевой вектор — вершина степени 3 (4 функции), пример АНФ:
   $$x_1x_2 \oplus x_1x_3 \oplus x_2x_3 \oplus x_4 \oplus 1;$$
   б) нулевой вектор — вершина степени 1 (12 функций), пример АНФ:
   $$x_2x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus x_1 \oplus x_2 \oplus x_3 \oplus 1.$$
4. Цикл длины 8 (рис. 3, *г*), пример АНФ:
   $$x_1x_2 \oplus x_1x_4 \oplus x_2x_3 \oplus x_3x_4 \oplus x_3 \oplus x_4 \oplus 1.$$

3. Булевы функции от пяти переменных

При $n = 5$ получен полный список булевых функций с максимальной алгебраической иммунностью ($\mathrm{AI}(f) = 3$) — всего их 197 765 122. Из них 96 768 функций имеют корреляционную иммунность порядка 1. Функций с более высоким порядком корреляционной иммунности и с максимальной алгебраической иммунностью не существует.

4. Булевы функции от шести переменных

Для перехода к функциям от шести переменных мы заменяем каждый вектор булева куба $\mathbb{E}^4$ на грань размерности 2. В таком случае, если вектор булева куба $\mathbb{E}^4$ принадлежал носителю исходной функции от четырёх переменных, существуют пять вариантов расположения точек носителя в новой грани размерности 2:

— все векторы новой грани принадлежат носителю функции от шести переменных;
— три вектора новой грани принадлежат носителю;
— два вектора новой грани на расстоянии 1 друг от друга принадлежат носителю;
— два вектора новой грани на расстоянии 2 друг от друга принадлежат носителю;
— один вектор новой грани принадлежит носителю функции от шести переменных.

Исследован способ построения, при котором все вершины, принадлежащие носителю, заменяются на грань одного и того же вида. При этом необходимо учитывать, что для сохранения максимального порядка корреляционной иммунности 1 необходимо, чтобы каждой грани булева куба $\mathbb{E}^6$ размерности 5 принадлежало одинаковое число точек носителя. С помощью программы мы проверили алгебраическую иммунность всех полученных функций от шести переменных и выяснили, что все варианты (с учётом ограничений, накладываемых корреляционной иммунностью) позволяют построить функцию от шести переменных с сохранением показателей алгебраической и корреляционной иммунности исходной булевой функции от четырёх переменных. Однако ни в одном случае не наблюдался рост алгебраической иммунности до максимально возможного показателя $\mathrm{AI}(f) = 3$ для функций от шести переменных.

Заключение

В работе получена полная классификация булевых функций от трёх, четырёх и пяти переменных с максимальной алгебраической иммунностью и обладающих корреляционной иммунностью. Исследован способ построения булевых функций большей размерности на основе функций меньшей размерности с заданными свойствами с сохранением этих свойств. В дальнейшем планируется рассмотреть другие возможности построения функций от шести переменных на основе геометрического представления функций от четырёх переменных, добиться повышения показателя алгебраической иммунности.

ЛИТЕРАТУРА

1. *Тарасников Ю. В.* О корреляционно-иммунных и устойчивых булевых функциях // Математические вопросы кибернетики. Вып. 11. М.: Физматлит, 2002. С. 91–148.
2. *Courtois N. and Meier W.* Algebraic attack on stream ciphers with linear feedback // LNCS. 2003. V. 2656. P. 345–359.
3. *Mariot L., and Leporati A.* A genetic algorithm for evolving plateaued cryptographic Boolean functions // LNCS. 2015. V. 9477. P. 33–45.
4. *Sun L. and Fu F.-W.* Constructions of balanced odd-variable rotation symmetric Boolean functions with optimal algebraic immunity and high nonlinearity // Theor. Comput. Sci. 2018. V. 738. P. 13–24.

5. *Carlet C.* Boolean Functions for Cryptography and Coding Theory. Cambridge: Cambridge University Press, 2021.
6. *Siegenthaler T.* Correlation-immunity of nonlinear combining functions for cryptographic applications // IEEE Trans. Inform. Theory. 1984. V. 30. No. 5. P. 776–780.

УДК 519.7 DOI 10.17223/2226308X/15/10

О РАЗЛОЖЕНИИ БЕНТ-ФУНКЦИЙ ОТ ВОСЬМИ ПЕРЕМЕННЫХ В СУММУ ДВУХ БЕНТ-ФУНКЦИЙ[1]

А. С. Шапоренко

Максимально нелинейная булева функция от чётного числа переменных называется бент-функцией. Исследуется гипотеза о представлении произвольных булевых функций от n переменных степени не больше $n/2$ как суммы двух бент-функций. Доказано, что произвольная бент-функция от восьми переменных степени не больше 3 представляется как сумма двух бент-функций. Показано, что каждая квадратичная булева функция от чётного числа переменных $n \geqslant 4$ раскладывается в сумму двух бент-функций специального вида.

Ключевые слова: *бент-функции, булевы функции, разложение в сумму бент-функций.*

Булева функция $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ от чётного числа переменных n называется *бент-функцией*, если она находится на максимальном расстоянии Хэмминга от множества всех аффинных функций [1]. Обозначим через $\mathcal{B}_n$ множество бент-функций. Дальше полагаем, что n является чётным целым числом.

Преобразованием Уолша — Адамара булевой функции f от n переменных называется целочисленная функция, заданная на множестве $\mathbb{Z}_2^n$ равенством

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

Для любой бент-функции f от n переменных $W_f(y) = \pm 2^{n/2}$ [2]. Определим *дуальную бент-функцию* $\tilde{f}$ к $f \in \mathcal{B}_n$ равенством $W_f(y) = 2^{n/2}(-1)^{\tilde{f}(y)}$ для любого $y \in \mathbb{Z}_2^n$.

Шифры, в которых используются бент-функции, более устойчивы к линейному криптоанализу [3], потому что бент-функции крайне плохо аппроксимируются аффинными функциями. Бент-функции используются в структуре блочного шифра CAST как координатные функции S-блоков [4], а также для построения регистра сдвига с нелинейной обратной связью в поточном шифре Grain [5]. Бент-функции связаны также с некоторыми объектами теории кодирования, например с кодами Рида — Маллера [2].

В работе исследуется известная открытая проблема о разложении произвольной булевой функции в сумму двух бент-функций [6].

Гипотеза 1 (Н. Н. Токарева, [7]). Любая булева функция от n переменных степени не больше $n/2$ может быть представлена как сумма двух бент-функций от n переменных.

В [7] показано, что гипотеза 1 верна для $n \leqslant 6$. Известно, что если гипотеза 1 верна, то справедлива следующая нижняя оценка числа бент-функций [7]:

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

[1]Работа выполнена в рамках госзадания ИМ СО РАН (проект № FWNF-2022-0018).

Утверждение 1 [7, 8]. Пусть f_0, f_1 и f_2 — бент-функции от n переменных. Тогда функция g, определенная следующим образом:

$$g(x,0,0) = f_0(x), \qquad g(x,0,1) = f_1(x),$$
$$g(x,1,0) = f_2(x), \qquad g(x,1,1) = f_3(x),$$

является бент-функцией от $n+2$ переменных тогда и только тогда, когда f_3 — бент-функция от n переменных и $\tilde{f}_0 \oplus \tilde{f}_1 \oplus \tilde{f}_2 \oplus \tilde{f}_3 = 1$.

Бент-функции, которые получаются с помощью утверждения 1, называются *бент итеративными функциями.*

Следствие 1. Пусть g и ℓ являются бент-функцией и линейной функцией от n переменных соответственно. Тогда $f(x, x_{n+1}, x_{n+2}) = x_{n+2}(x_{n+1} \oplus \ell(x)) \oplus g(x)$ является бент-функцией от $n+2$ переменных.

Известно, что бент-функции от восьми переменных степени не больше 3 разбиваются на 10 классов аффинной эквивалентности [9]. В таблице представлены разложения каждого представителя класса аффинной эквивалентности в сумму двух бент итеративных функций от восьми переменных, которые имеют форму из следствия 1. Заметим, что в утверждении 1 и следствии 1 переменные x_{n+1}, x_{n+2} используются для разложения бент итеративных функций на подфункции. В таблице такие переменные для бент итеративных функций выделены жирным шрифтом. Для простоты мы используем обозначение 12 вместо x_1x_2.

Неэквивалентные бент-функции степени $\leqslant 3$	Разложение
12 + 34 + 56 + 78	13 + 25 + **48** + 67 + 12 + 56
	13 + 25 + 48 + 67 + 34 + **78**
123 + 14 + 25 + 36 + 78	**1**(**6** + 23) + 24 + 58 + 37 + 14 + 78
	16 + 24 + 58 + **3**(**7** + 6) + 25
123 + 245 + 34 + 26 + 17 + 58	**2**(**7** + 45 + 13 + 6) + 14 + 38 + 56 + 34
	7(**2** + 1) + 14 + 38 + 56 + 58
123 + 245 + 13 + 15 + 26 + 34 + 78	**2**(**7** + 45 + 13 + 6) + 14 + 38 + + 56 + 34 + 15
	7(**2** + 8) + 14 + 38 + 56 + 13
123 + 245 + 346 + 35 + 26 + 25 + 17 + 48	**2**(**7** + 45 + 6 + 5) + 14 + 38 + 56 + 48
	3(**8** + 46 + 12 + 5) + 27 + 14 + 56 + 17
123 + 245 + 346 + 35 + 13 + 14 + 27 + 68	**2**(**6** + 13 + 45 + 7) + 14 + 78 + 35 + 45
	3(**1** + 46) + 26 + 78 + 45 + 68
123 + 245 + 346 + 35 + 26 + 25 + 12 + 13 + 14 + 78	**2**(**7** + 13 + 45 + 5 + 1 + 6) + 14 + 56 + + 38 + 45 + 68
	3(**1** + 46 + 8 + 5) + 27 + 45 + 68 + + 56 + 78
123 + 245 + 346 + 35 + 16 + 27 + 48	**1**(**4** + 23 + 6) + 27 + 58 + 36 + 26 + 78
	4(**1** + 25 + 36 + 8) + 35 + 26 + 78 + + 36 + 58
127 + 347 + 567 + 14 + 36 + 25 + 45 + 78	**7**(**1** + 34 + 56 + 8 + 6) + 36 + + 45 + 28 + 35 + 56
	1(**4** + 27 + 7) + 28 + 35 + 67 + 25 + 56
123 + 245 + 346 + 147 + 35 + 27 + 15 + 16 + 48	**4**(**1** + 36 + 25) + 35 + 26 + 78 + 36
	1(**5** + 23 + 47 + 4 + 6) + 27 + + 48 + 36 + 78 + 26

Известно, что функции, аффинно эквивалентные бент-функции, также являются бент-функциями [2]. Следовательно, если функция f раскладывается в сумму двух

бент-функций, то функция, аффинно эквивалентная f, также представляется как сумма двух бент-функций.

Теорема 1. Произвольная бент-функция от восьми переменных степени не больше 3 раскладывается в сумму двух бент-функций от восьми переменных.

Известно, что каждая квадратичная функция от $n \geqslant 4$ переменных представляется как сумма двух бент-функций от n переменных [10]. Справедливо следующее утверждение:

Утверждение 2. Любая квадратичная булева функция от $n \geqslant 4$ переменных представляется как сумма двух бент итеративных функций.

Исследование разложения булевых функций в сумму двух бент итеративных функций может привести к интересным результатам, касающимся рассматриваемой гипотезы. В настоящее время ведётся работа по разложению кубических булевых функций от восьми переменных в сумму двух бент итеративных функций. Получены частичные результаты, но необходимо продолжение исследования.

ЛИТЕРАТУРА

1. *Rothaus O. S.* On "bent" functions // J. Combinat. Theory. Ser. A. 1976. V. 20. No. 3. P. 300–305.
2. *Tokareva N.* Bent Functions: Results and Applications to Cryptography. Acad. Press. Elsevier, 2015.
3. *Matsui M.* Linear cryptanalysis method for DES cipher // LNCS. 1994. V. 765. P. 386–397.
4. *Adams C.* Constructing symmetric ciphers using the CAST design procedure // Design, Codes, Cryptogr. 1997. V. 12. No. 3. P. 283–316.
5. *Hell M., Johansson T., Maximov A., and Meier W.* A stream cipher proposal: Grain-128 // IEEE Intern. Symp. Inform. Theory. 2006. P. 1614–1618.
6. *Carlet C.* Open questions on nonlinearity and on APN Functions // LNCS. 2015. V. 9061. P. 83–107.
7. *Tokareva 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.
8. *Canteaut A. and Charpin P.* Decomposing bent functions // IEEE Trans. Inform. Theory. 2003. V. 49. No. 8. P. 2004–2019.
9. *Hou X. D.* Cubic bent functions // Discr. Math. 1998. V. 189. Iss. 1–3. P. 149–161.
10. *Qu L. and Li C.* New results on the Boolean functions that can be expressed as the sum of two bent functions // IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 2016. V. 99-A. No. 8. P. 1584–1590.

Секция 3

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

УДК 519.7 DOI 10.17223/2226308X/15/11

РАЗРАБОТКА И СРАВНЕНИЕ МОДЕЛЕЙ КВАНТОВОГО ОРАКУЛА ДЛЯ ГИБРИДНОЙ АТАКИ НА ПОСТКВАНТОВЫЕ КРИПТОСИСТЕМЫ, ОСНОВАННЫЕ НА РЕШЁТКАХ[1]

А. О. Бахарев

Для предложенной ранее модели квантового оракула, используемого в гибридном квантово-классическом алгоритме решения задачи нахождения кратчайшего вектора в решётке, получены новые уточнённые оценки числа кубит и глубины схемы. Разработана и проанализирована новая модель квантового оракула, использующая классическую память для хранения списка векторов. Получены верхние оценки сложности реализации атаки на постквантовые криптосистемы, являющиеся финалистами конкурса NIST.

Ключевые слова: *квантовый поиск, криптография с открытым ключом, постквантовая криптография.*

1. Основные понятия и определения

С каждым годом квантовые вычисления развиваются с большей силой. Поэтому возникает необходимость в разработке и анализе криптосистем, которые будут устойчивы к атакам с использованием квантового компьютера. Стойкость многих известных постквантовых криптосистем, основанных на решётках, зависит от сложности решения задачи нахождения кратчайшего вектора в решётке.

В 2016 г. Национальный институт стандартов и технологий США (NIST) объявил конкурс «Post-Quantum Cryptography Competition», по завершении которого будет принят новый — квантово-устойчивый — стандарт асимметричного шифрования. И 22 июля 2020 г. начался третий этап конкурса, финалистами которого являются криптосистемы, основанные на теории решёток и кодах, исправляющих ошибки.

В данной работе рассмотрен подход на основе решёток.

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

$$\Lambda = \Big\{ \sum_{i=1}^{d} b_i u_i : b_i \in \mathbb{Z} \Big\}.$$

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

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

[1]Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования РФ № 075-15-2022-281.

В общем случае SVP является NP-трудной задачей. Стойкость систем, основанных на решётках, зависит от эффективности решения SVP, так как большинство известных атак сводятся к решению этой проблемы.

В 2015 г. опубликована работа [1], в которой представлен гибридный подход к поиску кратчайшего вектора решётки на основе алгоритма GaussSieve [2] (алгоритм 1) — одного из самых эффективных классических алгоритмов.

Алгоритм 1. Алгоритм GaussSieve (D. Micciancio and P. Voulgaris, 2010)

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

Инициализировать пустой неупорядоченный список L и пустой стек S.
Повторять
  получить вектор v из стека (или сгенерировать новый).
  Пока $w \leftarrow$ ПОИСК$\{w \in L : \|v \pm w\| \leqslant \|v\|\}$
    уменьшить v с помощью w $(v \leftarrow v \pm w)$.
  Пока $w \leftarrow$ ПОИСК$\{w \in L : \|w \pm v\| \leqslant \|w\|\}$
    удалить w из листа L;
    уменьшить w с помощью v $(w \leftarrow w \pm v)$;
    добавить w в стек S.
  Если v изменился, **то**
    добавить v в стек S,
  иначе
    добавить v в лист L.
Пока v не станет кратчайшим вектором.
Вернуть вектор v.

Так как длина неупорядоченного списка L целочисленных векторов фиксированной размерности из алгоритма GaussSieve увеличивается экспоненциально с ростом размерности решётки, самой трудозатратной операцией данного алгоритма является функция «ПОИСК», которая осуществляет перебор неупорядоченного списка L для нахождения элемента w, удовлетворяющего условию поиска: $\|v \pm w\| \leqslant \|v\|$ или $\|w \pm v\| \leqslant \|w\|$. В рамках предложенного в [1] подхода ускорение достигается за счёт использования в функции «ПОИСК» квантового алгоритма поиска. Задача, решаемая этим алгоритмом, называется *задачей поиска*. Предполагается, что есть неупорядоченный список из K элементов, и требуется найти один элемент, удовлетворяющий некоторому условию. Другими словами, определена булева функция f, которая по номеру элемента (его двоичному представлению x) определяет, является ли элемент подходящим. Если элемент подходящий, то $f(x) = 1$, иначе $f(x) = 0$. В такой постановке задача поиска сводится к нахождению решения уравнения $f(x) = 1$.

В классическом варианте при условии, что решение одно, требуется $\sim K/2$ обращений к функции f для нахождения решения. Квантовый алгоритм поиска элемента в неупорядоченном списке (алгоритм Гровера [3]) решает данную задачу за $\sim \frac{\pi}{4}\sqrt{K}$ обращений к *оракулу* — квантовому аналогу функции f. Известно, что любая булева функция может быть реализована на квантовом компьютере.

2. Квантовые вычисления

Квантовый компьютер, в отличие от обычного, оперирует *квантовыми битами* (*кубитами*) [4]. Подобно классическому биту, который может находиться в состоянии 0 или 1, кубит имеет возможные состояния $|0\rangle$ и $|1\rangle$. Символ «$|\rangle$» называется *дираковским обозначением*, он является стандартным обозначением состояния в квантовой механике. Различие между битами и кубитами в том, что кубит может находиться в состоянии, отличном от $|0\rangle$ или $|1\rangle$. Можно составить линейную комбинацию состояний (суперпозицию):

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

Числа α и β являются комплексными и $|\alpha|^2 + |\beta|^2 = 1$. Иначе говоря, состояние одного кубита можно представить как единичный вектор из $\mathbb{C}^2$. Однако мы не можем измерить кубит, чтобы определить его квантовое состояние, т. е. значения α и β. Из квантовой механики следует, что при измерении кубита мы получаем либо результат 0 с вероятностью $|\alpha|^2$, либо результат 1 с вероятностью $|\beta|^2$. Систему с произвольным количеством кубит описывает следующий

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

Изменения со временем состояния $|\psi\rangle$ квантово-механической системы описываются следующим постулатом.

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

Пусть $|\psi\rangle = |x_1, x_2, \ldots, x_k\rangle$ и $|\psi'\rangle = |y_1, y_2, \ldots, y_k\rangle$. Равенство $|\psi'\rangle = U|\psi\rangle$ в обозначениях квантовых схем представлено на рис. 1.

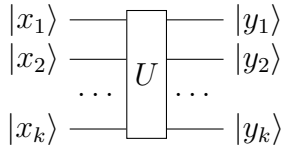
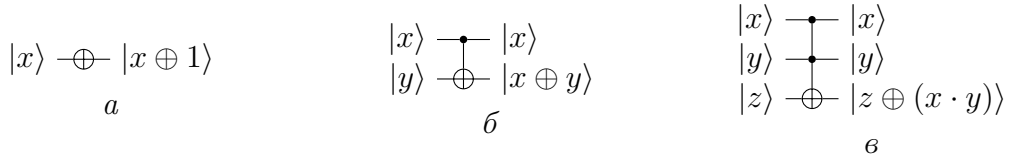


Рис. 1. Квантовая схема равенства $|\psi'\rangle = U|\psi\rangle$

Базовое преобразование квантового компьютера будем называть *вентилем*. В настоящей работе для построения всех операций и функций на квантовом компьютере используются базисные вентили, представленные на рис. 2 ($x, y, z \in \mathbb{F}_2$).



а *б* *в*

Рис. 2. Используемые вентили: *а* — вентиль Паули-X (NOT); *б* — вентиль CNOT; *в* — вентиль Тоффоли (CCNOT)

Определим *глубину квантовой схемы* [4] как количество слоёв, которые содержит схема. Один слой состоит из базисных вентилей, применённых к непересекающимся множествам кубит.

3. Модель оракула с квантовым списком

Рассмотрим предложенную в [5] модель оракула, при которой список L хранится в квантовой памяти. Работа модели происходит следующим образом:

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

В данной работе для построения квантового оракула рассматривается подход, минимизирующий количество кубит. *Постоянными* будем называть те кубиты, которые используются на протяжении всей работы оракула, а *временными* — те, которые нужны только во время проведения операции. В табл. 1 приведены количество кубит и глубина схемы в реализации операций, используемых в построении оракула.

Таблица 1

Количество кубит и глубина схемы, используемые в предложенной реализации, без учёта длины входа

Операция	Кубиты		Глубина
	постоянные	временные	
Сложение двух целых m-битных чисел, представленных в дополнительном коде	$m+1$	–	$3m$
Возведение в квадрат целого m-битного числа, представленного в прямом коде	$2m-2$	m^2-2m	$10m^2-33m+22$
Переключатель, где номер вектора представляется целым m-битным числом	2^m	–	3^m-1
Получение отрицания целого m-битного числа, представленного в дополнительном коде	m	–	$m+2$
Перевод целого m-битного числа из дополнительного кода в прямой	m	$m-2$	$2\lceil\log_2(m-1)\rceil+m+2$
Сравнение двух целых положительных m-битных чисел	1	$m+1$	$7m$
Функция $F:\mathbb{F}_2^n\to\mathbb{F}_2^m$	m	$n+\left\lceil\frac{m}{4}\right\rceil-3$	$2^{n+1}-2n+\lceil\log_2 m\rceil(2^{n+1}-3)-3$

С помощью данных, представленных в табл. 1, получается следующая

Теорема 1. Пусть имеется список длины K, состоящий из целочисленных векторов размерности $d\geqslant 2$, каждая координата которых кодируется битовой строкой длины $m\geqslant 3$. Тогда для реализации квантового оракула, при котором список L хранится в квантовой памяти, потребуется не более

$$\lceil\log_2 K\rceil+2^{\lceil\log_2 K\rceil}+Kdm+3dm^2+13dm+2d+6m+3\lceil\log_2 d\rceil+3$$

кубит. При этом глубина не превосходит

$$2 \cdot 3^{\lceil \log_2 K \rceil} + 2K(2\lceil \log_2 dm \rceil + 1) + 20m^2 + 4\lceil \log_2 m \rceil + \\ + \lceil \log_2 d \rceil (3\lceil \log_2 d \rceil + 12m + 5) + 21.$$

Как видно из теоремы 1, хранение списка L в квантовой памяти приводит к линейному росту используемого числа кубит от длины K, которая растёт экспоненциально с увеличением размерности решётки.

4. Модель оракула с классическим списком

Рассмотрим другую модель оракула, при которой список L хранится в классической памяти. Работа модели происходит следующим образом:

1) получение номера вектора на вход и передача его в функцию F;
2) получение в качестве выхода функции F вектора из списка L, соответствующего заданному номеру;
3) проверка скопированного вектора на условие поиска;
4) вывод ответа: 1 — если вектор удовлетворяет условию, 0 — если нет.

Данная модель помогает избежать линейного роста количества кубит, используемых оракулом, от увеличения размера списка L. Однако хранение списка L и построение векторной булевой функции F потребуют на стороне классической части памяти и вычислений, которые линейно зависят от длины списка L.

Теорема 2. Пусть имеется список длины K, состоящий из целочисленных векторов размерности $d \geqslant 2$, каждая координата которых кодируется битовой строкой длины $m \geqslant 3$. Тогда для реализации квантового оракула, при котором неупорядоченный список L хранится в классической памяти, потребуется не более

$$\lceil \log_2 K \rceil + 13dm + 5d + 6m + 3\lceil \log_2 d \rceil + 3 + \max\left(3d(m^2-1), \lceil \log_2 K \rceil + \left\lceil \frac{dm}{4} \right\rceil - 3\right)$$

кубит. При этом глубина не превосходит

$$2^{\lceil \log_2 K \rceil + 2} - 4\lceil \log_2 K \rceil + \lceil \log_2 dm \rceil (2^{\lceil \log_2 K \rceil + 2} - 6) + 20m^2 + \\ + \lceil \log_2 d \rceil (3\lceil \log_2 d \rceil + 12m + 5) + 4\lceil \log_2 m \rceil + 17.$$

Как видно из теоремы 2, число кубит, используемое для реализации квантового оракула с классическим списком, растёт логарифмически от длины списка L.

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

В табл. 2 приведены количество кубит и глубина схемы, достаточные для реализации квантовых оракулов с различными видами списка при атаке на постквантовые криптосистемы.

NTRU [6], SABER [7] и CRYSTALS-Kyber [8] являются постквантовыми криптосистемами, основанными на решётках и прошедшими в финальный раунд конкурса NIST. Как видно из табл. 2, экспоненциальная длина списка L накладывает ограничения на возможность реализации гибридных атак на полноразмерные постквантовые криптосистемы. Однако стоит заметить, что для атак на NTRU используются циклические решётки, исследование которых может помочь уменьшить длину списка L, что приведёт к меньшим верхним оценкам на сложность реализации квантового оракула.

Таблица 2

Вид списка	Уровень защищ.	NTRU		SABER		CRYSTALS-Kyber	
		Кубиты	Глубина	Кубиты	Глубина	Кубиты	Глубина
Квантовый	1	$2^{227,48}$	$2^{340,19}$	$2^{228,95}$	$2^{343,36}$	$2^{228,85}$	$2^{343,36}$
	3	$2^{298,45}$	$2^{452,72}$	$2^{337,06}$	$2^{512,95}$	$2^{336,96}$	$2^{512,95}$
	5	$2^{359,31}$	$2^{547,82}$	$2^{444,99}$	$2^{684,12}$	$2^{444,89}$	$2^{684,12}$
Классический	1	$2^{19,4}$	$2^{219,91}$	$2^{19,77}$	$2^{221,91}$	$2^{19,6}$	$2^{221,91}$
	3	$2^{19,81}$	2^{291}	$2^{20,36}$	2^{329}	$2^{20,18}$	2^{329}
	5	$2^{20,28}$	2^{351}	$2^{20,77}$	2^{437}	$2^{20,6}$	2^{437}

Заключение

Получены новые уточнённые верхние оценки на сложность реализации квантового оракула, использующего квантовый список L, из алгоритма Гровера для реализации гибридного квантово-классического алгоритма на основе GaussSieve, который может быть использован для атак на криптосистемы, стойкость которых зависит от решения задачи SVP. Предложена новая модель оракула с классическим списком L, на сложность реализации которой также получены верхние оценки. Проанализирована сложность реализации рассмотренных моделей оракула для атаки на постквантовые криптосистемы, основанные на решётках и являющиеся финалистами конкурса NIST.

ЛИТЕРАТУРА

1. *Laarhoven T., Mosca M., and van de Pol J.* Finding shortest lattice vectors faster using quantum search // Des. Codes Cryptogr. 2015. V. 77. No. 2. P. 375–400.
2. *Micciancio D. and Voulgaris P.* Faster exponential time algorithms for the shortest vector problem // Proc. 21 Ann. ACM-SIAM Symp. Discrete Algorithms. Society for Industrial and Applied Mathematics, USA, 2010. P. 1468–1480.
3. *Grover L. K.* A fast quantum mechanical algorithm for database search // Proc. 28 Ann. ACM Symp. Theory of Computing. N.Y.: ACM, 1996. P. 212–219.
4. *Nielsen M. A. and Chuang I. L.* Quantum Computation and Quantum Information. Cambridge: Cambridge University Press, 2010.
5. *Бахарев А. О.* Разработка и анализ оракула для гибридной атаки на криптографическую систему NTRU с использованием алгоритма квантового поиска // Прикладная дискретная математика. Приложение. 2021. № 14. С. 62–67.
6. *Chen C., Danba O., Hoffstein J., et al.* NTRU Algorithm Specifications and Supporting Documentation. `https://ntru.org/f/ntru-20190330.pdf`.
7. *D'Anvers J.-P., Karmakar A., Roy S. S., and Vercauteren F.* SABER: Mod-LWR based KEM. `https://www.esat.kuleuven.be/cosic/pqcrypto/saber/files/saberspecround1.pdf`.
8. *Avanzi R., Bos J., Ducas L., et al.* CRYSTALS-Kyber Algorithm Specifications and Supporting Documentation. `https://cryptojedi.org/papers/kybernist-20171130.pdf`.

УДК 519.7 DOI 10.17223/2226308X/15/12

О МНОЖЕСТВАХ НЕВОЗМОЖНЫХ РАЗНОСТЕЙ АЛГОРИТМОВ ШИФРОВАНИЯ ФЕЙСТЕЛЯ С НЕБИЕКТИВНОЙ ФУНКЦИЕЙ УСЛОЖНЕНИЯ

Д. А. Захаров, М. А. Пудовкина

Рассматривается семейство l-раундовых сбалансированных алгоритмов шифрования Фейстеля с небиективной функций усложнения. Для каждого из них доказано существование l-раундовых невозможных разностей для произвольного числа раундов l, а также получена нижняя оценка числа описанных невозможных разностей. Рассматриваемому семейству принадлежит алгоритм блочного шифрования GRANULE, для которого предложен новый подход поиска невозможных разностей. Показано, что он лучше других ранее известных способов. Получено как увеличение числа l раундов, для которых находятся невозможные разности, так и их количества. Приведены аналитические оценки числа невозможных разностей, которые подтверждены экспериментально.

Ключевые слова: *алгоритм шифрования Фейстеля, невозможные разности, небиективная функция усложнения, атака различением, алгоритм шифрования GRANULE.*

Метод невозможных разностей, независимо предложенный в работах [1, 2], является одним из наиболее распространённых подходов для оценки стойкости алгоритмов блочного шифрования. Он применялся, например, для анализа алгоритмов блочного шифрования Skipjack [1], DEAL [2], Present [3], CLEFIA, Simon, Camellia, Lblock [4] и AES [5], а также для анализа семейств XSL-алгоритмов [6], алгоритмов шифрования Фейстеля [7] и обобщений алгоритма Фейстеля [8]. Кроме того, невозможные разности применяются в атаках различения [8, 9], а также для отсеивания ложных ключей в атаках, основанных на методе невозможных разностей [4, 5]. Поиск таких разностей для наибольшего числа раундов — основная задача при анализе алгоритма шифрования относительно метода невозможных разностей.

Пусть $m \in \mathbb{N}$, $m \geqslant 2$, V_m — m-мерное векторное пространство над полем GF(2) с «естественной» операцией $+$ сложения векторов, $S(X)$ — симметрическая группа на множестве X.

Определение 1. Для l-раундовой функции зашифрования $g^{(l)}\colon V_m \times K \to V_m$ с множеством ключей шифрования K пара разностей $(\varepsilon, \delta) \in V_m^2$ называется l-раундовой невозможной разностью, если для всех $(\alpha, k) \in V_m \times K$ справедливо условие

$$g_k^{(l)}(\alpha + \varepsilon) \neq g_k^{(l)}(\alpha) + \delta,$$

где $g_k^{(l)}(\beta) = g^{(l)}(\beta, k)$ для всех $(\beta, k) \in V_m \times K$.

Определение 2. Невозможной тривиальной разностью будем называть такую невозможную разность $(\varepsilon, \delta) \in V_m \times V_m$, что $\varepsilon = 0$ или $\delta = 0$.

Опишем рассматриваемое семейство сбалансированных алгоритмов Фейстеля с небиективной функцией усложнения.

Пусть A — произвольная $(m \times m)$-матрица над полем GF(2), $\operatorname{rank}(A) = m - 1$. Зафиксируем отображения $f : V_m \to V_m$, $h^{(0)} : V_m \to V_m$, $h^{(1)} : V_m \to V_m$, заданные для каждого $\alpha \in V_m$ следующими условиями:

$$f : \alpha \mapsto h^{(1)}(h^{(0)}(\alpha)); \tag{1}$$

$$h^{(1)} : \alpha \mapsto \alpha A. \tag{2}$$

Рассмотрим также отображение $\nu : V_m^2 \times V_m \to V_m^2$ с частичной функцией $\nu_k \in S(V_m^2)$:

$$\nu_k : (\alpha_1, \alpha_0) \mapsto (\alpha_0 + f(\alpha_1) + k, \alpha_1) \text{ для всех } (\alpha_0, \alpha_1, k) \in V_m^3. \tag{3}$$

Ясно, что ν_k — частичная раундовая функция сбалансированного алгоритма Фейстеля. Отметим, что условию (3) удовлетворяет алгоритм блочного шифрования GRANULE [10].

В данной работе для семейства l-раундовых сбалансированных алгоритмов шифрования Фейстеля с функцией усложнения, удовлетворяющей условиям (1)–(3), предложен способ построения невозможных разностей для произвольного числа раундов, а также приведена аналитическая оценка числа таких разностей. Метод не зависит ни от биективных компонент функции усложнения, ни от алгоритма развёртывания ключа. Основой его является следующая теорема:

Теорема 1. Пусть A — произвольная $(m \times m)$-матрица над полем $\mathrm{GF}(2)$, $\mathrm{rank}(A) = m-1$, раундовая функция $\nu : V_m^2 \times V_m \to V_m^2$ задана условием (3). Тогда для каждого натурального $l > 3$ у l-раундового алгоритма шифрования с раундовой функцией ν существует не менее $3 \times 2^{2n-2} - 2^{n+1}$ невозможных l-раундовых нетривиальных разностей.

Предложенный в [9] алгоритм поиска невозможных разностей не учитывает ключевую особенность алгоритма шифрования GRANULE, а именно необратимость функции усложнения. Произведена его модификация путём изменения способа зашифрования разностей на описанный в [11]. Это позволило улучшить результаты [9] относительно числа найденных 7-раундовых различителей, а также получить экспериментальное подтверждение справедливости теоремы 1.

ЛИТЕРАТУРА

1. *Biham E., Birukov A., and Shamir A.* Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials // J. Cryptology. 2005. V. 18. P. 12–23.
2. *Knudsen L. R.* DEAL — a 128-bit cipher // Complexity. 1998. V. 258(2). P. 216–224.
3. *Tezcan C.* Improbable differential attacks on Present using undisturbed bits // J. Comput. Appl. Math. 2014. V. 259. P. 503–511.
4. *Boura C., Naya-Plasencia M., and Suder V.* Scrutinizing and improving impossible differential attacks: Applications to CLEFIA, Camellia, LBlock and Simon // LNCS. 2014. V. 8873. P. 179–199.
5. *Phan R. C. W.* Impossible differential cryptanalysis of 7-round Advanced Encryption Standard (AES) // Inform. Processing Lett. 2004. V. 91(1). P. 33–38.
6. *Li R., Sun B., and Li C.* Impossible differential cryptanalysis of SPN ciphers // IACR Cryptology ePrint Archive. 2010. V. 2010. P. 307–322.
7. *Wei Y., Li P., Sun B., and Li C.* Impossible differential cryptanalysis on Feistel ciphers with SP and SPS round functions // LNCS. 2010. V. 6123. P. 105–122.
8. *Cui T., Jin C., and Ma J.* A new method for finding impossible differentials of generalized Feistel structures // Chinese J. Electronics. 2018. No. 27(4). P. 728–733.
9. *Wu X., Li Y., Wei Y., and Sun Y.* Impossible differential distinguisher analysis of GRANULE and MANTRA algorithm // J. Communications. 2020. Iss. 1 P. 94–101.
10. *Bansod G., Pisharoty N., and Patil A.* GRANULE: An Ultra Lightweight Cipher Design for Embedded Security. IACR Cryptology ePrint Archive. 2018. `https://eprint.iacr.org/2018/600.pdf`.

11. *Shuying S. and Jun H.* Impossible differential cryptanalysis of GRANULE algorithm // Computer Engineering. 2019. V. 45(10). P. 134–138.

УДК 512.64, 519.21, 519.72 DOI 10.17223/2226308X/15/13

КРИТЕРИЙ МИНИМАЛЬНОСТИ ПО ВКЛЮЧЕНИЮ СОВЕРШЕННЫХ ШИФРОВ

Н. В. Медведева, С. С. Титов

Исследуются совершенные по Шеннону (абсолютно стойкие к атаке по шифртексту) шифры. Сформулирован и доказан критерий минимальности множества ключей совершенного шифра.

Ключевые слова: *совершенные шифры, эндоморфные шифры, неэндоморфные шифры.*

В рамках вероятностной модели шифра Σ_B [1] рассмотрим произвольный совершенный по Шеннону шифр. Пусть X, Y — конечные множества соответственно шифрвеличин и шифробозначений, с которыми оперирует некоторый шифр замены, K — множество ключей, $|X| = \lambda$, $|Y| = \mu$, $|K| = \pi$, где $\lambda > 1$, $\mu \geqslant \lambda$, $\pi \geqslant \mu$. Это означает, что открытые и шифрованные тексты представляются словами (ℓ-граммами, $\ell \geqslant 1$) в алфавитах X и Y соответственно. Согласно [2, 3], под шифром Σ_B будем понимать совокупность множеств правил зашифрования и правил расшифрования с заданными распределениями вероятностей на множествах открытых текстов и ключей. Шифры, для которых апостериорные вероятности открытых текстов совпадают с их априорными вероятностями, называются совершенными.

Описание эндоморфных ($\lambda = \mu$) с минимально возможным числом ключей ($|K| = |Y|$) совершенных шифров даётся теоремой Шеннона, таблица зашифрования таких шифров — это латинский квадрат из равновероятных подстановок зашифрования [1]. При описании неэндоморфных ($\lambda < \mu$) совершенных шифров, как показано в [4], возникает естественная задача описания минимальных по включению (т. е. содержащих минимально возможное множество ключей зашифрования с ненулевыми вероятностями) совершенных шифров, не сводящихся к латинским прямоугольникам размера $\mu \times \lambda$, которые можно рассматривать как непосредственное обобщение теоремы Шеннона.

На первом этапе реализации данного подхода в работах [4, 5] получены достаточные условия того, что в таблице зашифрования как неэндоморфных, так и эндоморфных совершенных шифров с равновероятными ключами отсутствуют латинские соответственно прямоугольники и квадраты. В [5] на основе графового подхода к исследованию и описанию совершенных шифров, их аналогов и обобщений сформулировано достаточное условие минимальности шифра по включению.

Пусть дана таблица зашифрования совершенного шифра Σ_B с λ столбцами, π строками и μ шифробозначениями y_j, $j = 1, 2, \ldots, \mu$, где $\pi \geqslant \mu \geqslant \lambda > 1$. Обозначим через $\mathsf{P} = (\mathsf{P}_1, \mathsf{P}_2, \ldots, \mathsf{P}_\pi)^{\mathrm{T}}$ вектор-столбец вероятностей P_k ключей k ($k = 1, 2, \ldots, \pi$), через p_j — априорные вероятности шифробозначений y_j ($j = 1, 2, \ldots, \mu$) данного шифра. Для исходной таблицы зашифрования построим соответствующую ей $(0, 1)$-матрицу A с π строками и $\lambda\mu$ столбцами следующим образом:

1) каждому столбцу x_i таблицы зашифрования поставим в соответствие μ столбцов и занумеруем все получившиеся $\lambda\mu$ столбцы индексами (i, j), где $i = 1, 2, \ldots, \lambda$, $j = 1, 2, \ldots, \mu$;

2) на пересечении строки k ($k = 1, 2, \ldots, \pi$) и столбца (i, j) поставим единицу тогда и только тогда, когда $e_k(x_i) = y_j$, т. е. если шифрвеличина x_i на ключе k зашифровывается в шифробозначение y_j. В противном случае ставим нуль.

Соответствие таблицы зашифрования и матрицы A, можно считать, получено посредством замены каждого i-го столбца в таблице зашифрования на μ столбцов матрицы A, индексированных парами (i, j), где $i = 1, 2, \ldots, \lambda$, $j = 1, 2, \ldots, \mu$. Столбцы матрицы A можно упорядочить и занумеровать, например, естественным образом, присвоив индексу (i, j) номер $m = (i-1)\mu + j$, так что $\lambda\mu \geqslant m \geqslant 1$.

Используя $(0, 1)$-матрицу A, для которой $A_{k,(i,j)} = 1 \Leftrightarrow e_k(x_i) = y_j$, условие совершенности шифра можно записать в виде системы линейных уравнений

$$\sum_{k=1}^{\pi} \mathsf{P}_k \, A_{k,(i,j)} = p_j \quad (i = 1, 2, \ldots, \lambda, \ j = 1, 2, \ldots, \mu) \tag{1}$$

с дополнительным условием

$$\sum_{k=1}^{\pi} \mathsf{P}_k = 1. \tag{2}$$

К матрице A добавим столбец из π единиц. При предлагаемом способе упорядочивания столбцов этому столбцу естественно присвоить номер $m = 0$ и считать его добавленным к матрице A слева. В вектор правых частей добавим 1 (в соответствии с уравнением (2)). Получившуюся матрицу размером π строк на $1 + \lambda\mu$ столбцов обозначим через $\tilde{A}$, а вектор правых частей — через $\tilde{p} = (1, p_1, \ldots, p_\mu, \ldots, p_1, \ldots, p_\mu)^{\mathrm{T}}$, при этом $\operatorname{rank} \tilde{A} \leqslant \pi$.

Теорема 1. Множество ключей шифра минимально тогда и только тогда, когда ранг матрицы $\tilde{A}$ максимальный (равный π).

Доказательство. Рассматривая вероятности P_k ключей как неизвестные ненулевые искомые величины уравнений (1), а вероятности p_j — как их правые части, по теореме Кронекера — Капелли заключаем, что, поскольку система имеет решение, ранг матрицы $\tilde{A}$ равен рангу расширенной (посредством вектора $\tilde{p} = (1, p_1, \ldots, p_\mu, \ldots, p_1, \ldots, p_\mu)^{\mathrm{T}}$) матрицы. Максимально возможный ранг матрицы $\tilde{A}$ равен количеству её строк — π.

Предположим, что данный шифр не является минимальным по включению. Докажем, что тогда ранг матрицы $\tilde{A}$ меньше π. Действительно, пусть существует шифр с вероятностями ключей P'_k с теми же правыми частями p_j, в котором отсутствуют некоторые ключи из данного набора. Этот факт можно отразить в системе уравнений (1), (2), положив вероятности отсутствующих ключей равными нулю. Так, пусть без ограничения общности ключ $k = \pi$ отсутствует, тогда система уравнений (1), (2) имеет решение, в котором $\mathsf{P}'_\pi = 0$. Заметим, что неизвестные $\mathsf{P}''_k = \mathsf{P}_k - \mathsf{P}'_k$ ($k = 1, \ldots, \pi$) не все равны нулю (например, $\mathsf{P}''_\pi \neq 0$) и удовлетворяют однородной системе уравнений, откуда следует, что в матрице $\tilde{A}$ не может быть невырожденной квадратной подматрицы максимального размера π на π, т. е. её ранг строго меньше π.

Обратно, пусть ранг матрицы $\tilde{A}$ строго меньше π. Докажем, что данный шифр не является минимальным по включению. Действительно, в этом случае одна из строк матрицы $\tilde{A}$ (без ограничения общности пусть это π-я строка) есть линейная комбинация остальных. Пусть

$$A_{\pi,(i,j)} = \sum_{k=1}^{\pi-1} q_k A_{k,(i,j)}, \quad q_k \in \mathbb{R} \ (i = 1, 2, \ldots, \lambda, \ j = 1, 2, \ldots, \mu).$$

Для каждого $t \in [0,1]$ введём новые неизвестные $\mathsf{P}'_k = \mathsf{P}_k + q_k t\,\mathsf{P}_\pi$ $(k = 1, \ldots, \pi-1)$, $\mathsf{P}'_\pi = (1-t)\,\mathsf{P}_\pi$, где $q_1 + q_2 + \ldots + q_{\pi-1} = 1$ из-за столбца единиц в матрице $\tilde{A}$. В силу тождества

$$\begin{aligned}\sum_{k=1}^{\pi} \mathsf{P}_k\, A_{k,(i,j)} &= \sum_{k=1}^{\pi-1} \mathsf{P}_k\, A_{k,(i,j)} + \mathsf{P}_\pi\, A_{\pi,(i,j)} = \sum_{k=1}^{\pi-1} \mathsf{P}_k\, A_{k,(i,j)} + [t\,\mathsf{P}_\pi + (1-t)\,\mathsf{P}_\pi] A_{\pi,(i,j)} = \\ &= \left(\sum_{k=1}^{\pi-1} \mathsf{P}_k\, A_{k,(i,j)} + t\,\mathsf{P}_\pi \sum_{k=1}^{\pi-1} q_k A_{k,(i,j)}\right) + (1-t)\,\mathsf{P}_\pi\, A_{\pi,(i,j)} = \\ &= \sum_{k=1}^{\pi-1} (\mathsf{P}_k + t q_k\,\mathsf{P}_\pi) A_{k,(i,j)} + (1-t)\,\mathsf{P}_\pi\, A_{\pi,(i,j)} = \sum_{k=1}^{\pi} \mathsf{P}'_k\, A_{k,(i,j)}\end{aligned}$$

эти новые неизвестные P'_k удовлетворяют той же системе уравнений. Поскольку $\mathsf{P}'_k = \mathsf{P}_k > 0$ при $t = 0$ и линейная функция непрерывна, имеем $\mathsf{P}'_k > 0$ при всех достаточно малых $t > 0$. Ввиду того, что $\tilde{A}$ является $(0,1)$-матрицей и все координаты вектора $\tilde{p}$ положительны, среди чисел q_k должны быть и положительные; для таких ключей k имеем $\mathsf{P}'_k > 0$ для всех $t > 0$. Если для всех $k = 1, \ldots, \pi-1$ имеем $\mathsf{P}'_k > 0$ при всех $t \in [0,1]$, то при $t = 1$ получаем $\mathsf{P}'_\pi = 0$, $\mathsf{P}'_k > 0$ $(k = 1, \ldots, \pi-1)$, причём

$$\sum_{k=1}^{\pi-1} \mathsf{P}'_k = \sum_{k=1}^{\pi-1} \mathsf{P}_k = 1$$

для любого t ввиду равенства

$$\sum_{k=1}^{\pi-1} q_k = 1.$$

Следовательно, величины P'_k $(k = 1, \ldots, \pi-1)$ могут рассматриваться как вероятности ключей собственного подмножества ключей данного шифра, так как $\mathsf{P}'_\pi = 0$. Если же $\mathsf{P}'_k = 0$ при некотором $k = k_0$ и $t = t_0$, $0 < t_0 < 1$, то, выбрав t_0 наименьшим по всем таким k, получим тоже собственное подмножество ключей с ненулевыми вероятностями P'_k, что доказывает неминимальность данного шифра по включению. ■

Отметим, что для эндоморфного шифра равенство $q_1 + \ldots + q_{\pi-1} = 1$ выполняется уже при рассмотрении матрицы A без использования столбца из единиц.

Замечание 1. Из критерия минимальности по включению совершенных шифров следуют необходимые условия:

1) для минимальности по включению множества ключей шифра необходимо выполнение неравенства $\pi \leqslant \lambda\mu$;
2) для эндоморфного минимального по включению совершенного шифра выполняется неравенство $\pi \leqslant \lambda(\lambda - 1)$.

Полученные теоретические результаты могут быть положены в основу классификации минимальных совершенных шифров и исследований почти совершенных шифров [6, 7].

Таким образом, сформулирован и доказан критерий минимальности множества ключей совершенных шифров: множество ключей минимально, если и только если ранг бинарной матрицы, состоящей из π столбцов и $1 + \lambda\mu$ столбцов, максимален и равен количеству ключей, содержащихся в таблице зашифрования. Получены также необходимые условия минимальности по включению совершенных шифров.

ЛИТЕРАТУРА

1. *Шеннон К.* Теория связи в секретных системах // Работы по теории информации и кибернетике. М.: Наука, 1963. С. 333–402.
2. *Алферов А. П., Зубов А. Ю., Кузьмин А. С., Черемушкин А. В.* Основы криптографии. М.: Гелиос АРВ, 2001.
3. *Зубов А. Ю.* Совершенные шифры. М.: Гелиос АРВ, 2003.
4. *Медведева Н. В., Титов С. С.* Конструкции неэндоморфных совершенных шифров // Прикладная дискретная математика. Приложение. 2020. № 13. С. 51–54.
5. *Медведева Н. В., Титов С. С.* К задаче описания минимальных по включению совершенных шифров // Прикладная дискретная математика. Приложение. 2021. № 14. С. 91–95.
6. *Зубов А. Ю.* Почти совершенные шифры и коды аутентификации // Прикладная дискретная математика. 2011. № 4(14). С. 28–33.
7. *Зубов А. Ю.* О понятии ε-совершенного шифра // Прикладная дискретная математика. 2016. № 3(33). С. 45–52.

УДК 519.7 DOI 10.17223/2226308X/15/14

ВЫЧИСЛЕНИЕ РАЗНОСТНЫХ ХАРАКТЕРИСТИК ДЛЯ СЛОЖЕНИЯ k ЧИСЕЛ ПО МОДУЛЮ 2^n [1]

А. С. Мокроусов

Рассматривается разностная характеристика $\mathrm{xdp}^+_{\mathrm{k}}(\alpha^1, \ldots, \alpha^k \to \alpha^0)$, где $\alpha^0, \alpha^1, \ldots, \alpha^k \in \mathbb{Z}_2^n$, которая определяет вероятность преобразования разностей $\alpha^1, \ldots, \alpha^k$ в разность α^0 (относительно побитового «исключающего или») функцией $f(x_1, \ldots, x_k) = x_1 + \ldots + x_k \mod 2^n$. Данная величина используется при разностном криптоанализе криптографических примитивов, содержащих «исключающее или» и сложение по модулю 2^n, например ARX-конструкций. Предложены аналитические выражения для матриц, используемых для вычисления $\mathrm{xdp}^+_{\mathrm{k}}$. Кроме того, рассмотрена разностная характеристика $\mathrm{adp}^{\oplus}(\alpha, \beta \to \gamma)$, где $\alpha, \beta, \gamma \in \mathbb{Z}_2^n$, определяющая вероятность преобразования разностей α, β в разность γ (относительно сложения по модулю 2^n) функцией $x \oplus y$, и получены все тройки разностей, вероятность которых больше $1/4$.

Ключевые слова: *ARX, исключающее или, сложение по модулю, разностный криптоанализ, разностные характеристики.*

Одним из подходов к построению криптографических примитивов является комбинирование сложения по модулю 2^n ($\boxplus$), побитовых операций (например, «исключающего или» — $\oplus$), битовых сдвигов ($\ll$), циклических сдвигов ($\lll$). Это позволяет получить очень быстрые в программной реализации алгоритмы. Особый интерес представляют ARX-конструкции, использующие только операции $\boxplus$, $\oplus$ и $\lll$. Примерами таких шифров являются FEAL [1], TEA [2], Salsa20 [3], Speck [4].

Хорошие шифры должны быть стойкими к различным видам криптоанализа, в частности к разностному криптоанализу [5]. Это один из основных статистических методов, основанный на исследовании того, в какие разности шифртекстов могут переходить разности открытых текстов. Важным шагом при реализации метода является вычисление разностных характеристик и их максимальных значений. Для базовых операций архитектуры ARX данные характеристики определяются следующим образом [6]:

[1] Работа выполнена в рамках госзадания ИМ СО РАН (проект № FWNF-2022-0018).

$$\mathrm{xdp}^{+}(\alpha,\beta\to\gamma)=\frac{1}{4^n}\big|\{x,y\in\mathbb{Z}_2^n:\ (x\oplus\alpha)\boxplus(y\oplus\beta)=\gamma\oplus(x\boxplus y)\}\big|,$$
$$\mathrm{adp}^{\oplus}(\alpha,\beta\to\gamma)=\frac{1}{4^n}\big|\{x,y\in\mathbb{Z}_2^n:\ (x\boxplus\alpha)\oplus(y\boxplus\beta)=\gamma\boxplus(x\oplus y)\}\big|.$$

С вектором $x=(x_0,\ldots,x_{n-1})\in\mathbb{Z}_2^n$ мы ассоциируем целое число $\sum\limits_{i=0}^{n-1}x_i2^i$, тогда $x\boxplus\alpha$ означает сложение ассоциированных с x и α чисел по модулю 2^n.

Недостатком ARX-шифров является сложность вычисления разностных характеристик для композиций операций. Существует подход с использованием S-функций [6], позволяющий вычислить разностные характеристики как произведение специальных матриц, построенных на основе рассматриваемого преобразования. Однако его алгоритмическое применение в большинстве случаев не позволяет получить аналитические выражения для данных матриц.

1. Матричный способ вычисления $\mathrm{xdp}^{+}_{\mathrm{k}}(\alpha^1,\ldots,\alpha^k\to\alpha^0)$

Рассмотрим функцию $f(x_1,\ldots,x_k)=x_1+\ldots+x_k$. Разностная характеристика $\mathrm{xdp}^{+}_{\mathrm{k}}$ для неё определяется следующим образом:

$$\mathrm{xdp}^{+}_{\mathrm{k}}(\alpha^1,\ldots,\alpha^k\to\alpha^0)=\frac{1}{2^{nk}}\left|\left\{x^1,\ldots,x^k\in\mathbb{Z}_2^n:\ \boxplus_{i=1}^{k}(x^i\oplus\alpha^i)=\alpha^0\oplus\boxplus_{i=1}^{k}x^i\right\}\right|.$$

Подход с использованием S-функций подразумевает построение матриц на основе преобразования, через произведения которых можно подсчитать значения разностной характеристики [5]. Для характеристики $\mathrm{xdp}^{+}_{\mathrm{k}}$ далее предлагаются явные выражения для вычисления всех ненулевых элементов матриц.

Обозначим через $\mathrm{wt}(x)$ вес Хэмминга вектора x. Определим $N(a,b)=a+2kb$, где $0\leqslant a,b<2k$. Мы будем использовать матрицы размера $4k^2\times4k^2$. Заметим, что $0\leqslant N(a,b)<4k^2$. Таким образом, через $N(a,b)$ будем представлять номер строки или столбца матрицы с помощью пары чисел (a,b).

Зададим матрицы A_m размера $4k^2\times4k^2$ для всех $m\in\mathbb{Z}_2^{k+1}$ следующим образом. Рассмотрим любую четвёрку целых чисел x, y, x', y', таких, что $0\leqslant x,y,x',y'<2k$. Пусть $c=\mathrm{wt}(m_1,\ldots,m_k)$,

$$\Delta x=x'-\left\lfloor\frac{x}{2}\right\rfloor,\quad a=\left\lfloor\frac{\Delta x+\Delta y-c}{2}\right\rfloor,$$
$$\Delta y=y'-\left\lfloor\frac{y}{2}\right\rfloor,\quad b=\left\lfloor\frac{\Delta x-\Delta y+c}{2}\right\rfloor.$$

Тогда элемент матрицы A_m в $N(x',y')$-й строке и $N(x,y)$-м столбце определяется следующим образом:

1) 0, если выполнено одно из следующих условий:
   - хотя бы одно из чисел Δx, Δy, a или b меньше нуля,
   - $x_1+y_1+c+m_0$ — нечётное,
   - $\Delta x+\Delta y+c$ — нечётное;
2) $\binom{k-c}{a}\binom{c}{b}$ в противном случае.

С использованием матриц A_m, а также матриц $L = \begin{pmatrix}1 & 1 & \dots & 1\end{pmatrix}$ размера $1 \times 4k^2$ и $C = \begin{pmatrix}1 & 0 & \dots & 0\end{pmatrix}^{\mathrm{T}}$ размера $4k^2 \times 1$ можно вычислить значения характеристики $\mathrm{xdp}^+_{\mathrm{k}}$.

Теорема 1. Пусть $\alpha^0, \alpha^1, \dots, \alpha^k \in \mathbb{Z}_2^n$. Тогда

$$\mathrm{xdp}^+_{\mathrm{k}}(\alpha^1, \dots, \alpha^k \to \alpha^0) = \frac{1}{2^{nk}} L A_{w_{n-1}} \dots A_{w_1} A_{w_0} C, \text{ где } w_j = (\alpha^0_j, \dots, \alpha^k_j) \in \mathbb{Z}_2^{k+1}.$$

Заметим, что элементы матрицы A_m зависят только от $\mathrm{wt}(m_1, \dots, m_k)$ и m_0.

Следствие 1. В последовательности матриц A_m, где $m \in \mathbb{Z}_2^{k+1}$, существует лишь $2(k+1)$ различных матриц.

Переобозначим эти матрицы как $A_{\mathrm{wt}(m_1,\dots,m_k),m_0}$. Алгоритм 1 позволяет вычислять все матрицы $A_{0,0}, \dots, A_{k,0}$ и $A_{0,1}, \dots, A_{k,1}$ одновременно за $O(k^6)$ операций.

Алгоритм 1. Алгоритм одновременного вычисления всех матриц $A_{i,j}$

Для всех целых m, i, j, таких, что $0 \leqslant m \leqslant k, 0 \leqslant i \leqslant k, 0 \leqslant j \leqslant 1$:
  $B^m_{i,j}$ — матрица размера $4k^2 \times 4k^2$, изначально заполненная нулями.
Для всех целых x, y, таких, что $0 \leqslant x, y < 2k$:
  $c := x_1 \oplus y_1$;
  $B^0_{0,c}[N(\lfloor x/2 \rfloor, \lfloor y/2 \rfloor)][N(x,y)] := 1$.
Для всех m от 1 до k:
  Для всех x, y, x', y', таких, что $0 \leqslant x, y, x', y' < 2k$:
    Для всех i от 0 до $m-1$, j от 0 до 1:
      $P := B^{m-1}_{i,j}[N(x',y')][N(x,y)]$;
      $B^m_{i,j}[N(x',y')][N(x,y)] \mathrel{+}= P$.
      Если $x' + 1 < 2k$ и $y' + 1 < 2k$, **то**
        $B^m_{i,j}[N(x'+1,y'+1)][N(x,y)] \mathrel{+}= P$.
    Для всех j от 0 до 1:
      $P := B^{m-1}_{m-1,j}[N(x',y')][N(x,y)]$;
      $j' := j \oplus 1$.
      Если $y' + 1 < 2k$, **то**
        $B^m_{m,j'}[N(x',y'+1)][N(x,y)] \mathrel{+}= P$.
      Если $x' + 1 < 2k$, **то**
        $B^m_{m,j'}[N(x'+1,y')][N(x,y)] \mathrel{+}= P$.
Для всех i, j, таких, что $0 \leqslant i \leqslant k$, $0 \leqslant j \leqslant 1$:
  $A_{i,j} := B^k_{i,j}$.
Вернуть $A_{i,j}$ для всех i, j, таких, что $0 \leqslant i \leqslant k$, $0 \leqslant j \leqslant 1$.

Отметим, что для разностной характеристики xdp^+, которая является частным случаем $\mathrm{xdp}^+_{\mathrm{k}}$ при $k = 2$, известен более простой способ вычисления без использования матриц [7]. Однако на случай $\mathrm{xdp}^+_{\mathrm{k}}$ он, по всей видимости, не обобщается.

2. Максимумы $\mathrm{adp}^{\oplus}(\alpha, \beta \to \gamma)$

Рассмотрим характеристику $\mathrm{adp}^{\oplus}(\alpha, \beta, \gamma)$. В [8] изучены максимумы при фиксированном значении γ. В данной работе мы рассматриваем максимумы без фиксации γ, по всем возможным тройкам $\alpha, \beta, \gamma \in \mathbb{Z}_2^n$. В результате обнаружено, что n максимальных значений данной характеристики имеют достаточно простые выражения.

Теорема 2. Пусть $p_1, \dots, p_n$ — n различных максимальных значений $\mathrm{adp}^{\oplus}(\alpha, \beta, \gamma)$, где $\alpha, \beta, \gamma \in \mathbb{Z}_2^n$, $p_1 > p_2 > \dots > p_n$. Тогда

1) $p_1 = 1$ и $p_i = p_{i-1} - \dfrac{1}{2 \cdot 4^{i-2}}$ при $2 \leqslant i \leqslant n$;
2) $\mathrm{adp}^{\oplus}(\alpha, \beta, \gamma) = p_i \iff$ тройка (α, β, γ) получается из тройки $(0, 2^{n-i}, 2^{n-i})$ композицией следующих преобразований, сохраняющих значение $\mathrm{adp}^{\oplus}$ [8]:
   а) перестановка элементов тройки;
   б) $(\alpha, \beta, \gamma) \mapsto (\alpha \oplus 2^{n-1}, \beta \oplus 2^{n-1}, \gamma)$;
   в) $(\alpha, \beta, \gamma) \mapsto (\pm\alpha, \pm\beta, \pm\gamma)$, где $\pm\alpha$ обозначает α или $-\alpha \bmod 2^n$.

Замечание 1. Если $\mathrm{adp}^{\oplus}(\alpha, \beta, \gamma) \neq p_i$ для $1 \leqslant i \leqslant n$ из теоремы 2, то

$$\mathrm{adp}^{\oplus}(\alpha, \beta, \gamma) \leqslant 1/4.$$

Из теоремы 2 нетрудно получить количество разностей, на которых достигаются данные максимальные значения. Обозначим количество разностей (α, β, γ), $\alpha, \beta, \gamma \in \in \mathbb{Z}_2^n$, на которых достигается $\mathrm{adp}^{\oplus}(\alpha, \beta, \gamma) = p_i$, как C_i.

Следствие 2. Для C_i верны следующие утверждения:
— $C_1 = 4$, $C_2 = 24$;
— $C_3 = C_4 = \ldots = C_n = 48$.

ЛИТЕРАТУРА

1. *Shimizu A. and Miyaguchi S.* Fast data encipherment algorithm FEAL // LNCS. 1988. V. 304. P. 267–278.
2. *Wheeler D. J. and Needham R. M.* TEA, a tiny encryption algorithm // LNCS. 1995. V. 1008. P. 363–366.
3. *Bernstein D. J.* Salsa20 specification. eSTREAM Project algorithm description. `http://www.ecrypt.eu.org/stream/salsa20pf.html`. 2005.
4. *Beaulieu R., Shors D., Smith J., et al.* The SIMON and SPECK Families of Lightweight Block Ciphers. `https://eprint.iacr.org/2013/404`.
5. *Biham E. and Shamir A.* Differential cryptanalysis of DES-like cryptosystems // J. Cryptology. 1991. No. 4. P. 3–72.
6. *Mouha N., Velichkov V., De Cannière C., and Preneel B.* The differential analysis of S-functions // LNCS. 2011. V. 6544. P. 36–56.
7. *Lipmaa H. and Moriai S.* Efficient algorithms for computing differential properties of addition // LNCS. 2002. V. 2355. P. 336–350.
8. *Mouha N., Kolomeec N., Akhtiamov D., et al.* Maximums of the additive differential probability of Exclusive-Or // IACR Trans. Symmetric Cryptology. 2021. No. 2. P. 292–313.

УДК 519.7 DOI 10.17223/2226308X/15/15

НЕКОТОРЫЕ УСЛОВИЯ ПРИМЕНИМОСТИ ИНТЕГРАЛЬНОГО МЕТОДА К ЧЕТЫРЁМ РАУНДАМ AES-ПОДОБНЫХ АЛГОРИТМОВ

К. Н. Панков

Получен ряд необходимых и одно достаточное условие того, что к блочным алгоритмам, построенным аналогично алгоритму AES (например, SQUARE, Rijndael, Crypton) с уменьшенным до четырёх числом раундов может быть применён интегральный метод криптоанализа. Приведены данные экспериментов о применении интегрального метода к алгоритму Rijndael.

Ключевые слова: *блочные алгоритмы, AES, SQUARE, Rijndael, Crypton, спектральные коэффициенты, интегральный метод.*

Современное состояние систем защиты информации характеризуется наличием так называемого квантового вызова. Появление квантового компьютера с соответствующим набором характеристик, которые, согласно мнению ряда экспертов, будут достигнуты в ближайшее десятилетие, приведёт к потере некоторыми существующими системами защиты информации или отдельными их элементами своего практического значения. Таким образом, актуальной становится задача разработки, исследования и программно-аппаратной реализации криптографических примитивов, которые смогут противостоять квантовому вызову. В настоящее время существует несколько подходов к решению данной проблемы, одним из которых является использование симметричных криптографических систем, стойкость которых понизится с появлением гипотетического квантового вычислителя незначительно. Поэтому продолжает оставаться актуальным вопрос об оценке криптографической стойкости симметричных шифрсистем, применяемых как в современных информационно-телекоммуникационных системах, так и в системах интернета вещей (IoT). Отметим, что, согласно информации, приведённой в [1, 2], в условиях ограниченных ресурсов в системах IoT наиболее рационально использовать алгоритмы блочного шифрования. В соответствии с [3] одним из основных методов криптоанализа подобных алгоритмов является интегральный метод.

Интегральный метод, впервые названный так в [4], предложен в работе [5] и является развитием метода из [6]. Этот метод ещё называют square-атака [5] или saturation-атака [7] (метод квадрата и метод статистического насыщения). Он аналогичен по схеме применения известному дифференциальному методу, также являясь атакой на основе адаптивно подобранного открытого текста [8].

Общую схему метода можно описать следующим образом. Преобразование F_k, реализуемое блочным алгоритмом шифрования и зависящее от ключа k, представляется как композиция двух отображений $F_k = F_k^2 \circ F_k^1$. Допустим, что имеется такое множество $I \subset X$ блоков открытого текста, что «интеграл» $\sum\limits_{x \in I} F_k^1(x)$ обладает определённым легко проверяемым «отличительным» свойством (например, равен нулевому вектору). Предположим, что известны пары «открытый — шифрованный текст» $(x, F_k(x))$ для всех $x \in I$. Тогда для любого k можно определить элемент $\sum\limits_{x \in I} \left(F_k^2\right)^{-1}(F_k(x))$, который при истинном варианте k должен обладать «отличительным» свойством «интеграла».

Применению различных модификаций интегрального метода к ряду блочных шифрсистем, поиску путей развития этого метода и изучению связанных с ним характеристик посвящён ряд работ [9 – 12].

Интегральный метод в основном применяется к блочным алгоритмам с малым числом раундов, например к редуцированным вариантам AES [13] или Кузнечика [14], что является актуальным именно в условиях низкоресурсной криптографии.

Рассмотрим вопрос об условиях возможности применения интегрального метода к системам шифрования типа AES (алгоритмы SQUARE, Rijndael, Crypton) с четырьмя итерациями $G = f_4^* \circ f_3 \circ f_2 \circ f_1$, в которой первые три итерации f_1, f_2, f_3 обычные (но, например, в первую итерацию Rijndael мы включаем прибавление к блоку открытого текста начального ключа), а последняя итерация f_4^* может отличаться (например, в Rijndael она не содержит преобразования MixColumn). Схемы примения интегрального метода к перечисленным алгоритмам рассматриваются в работах [5, 15, 16].

Промежуточный блок после третьей итерации обозначим через $E(x, k_0) = (E_0(x, k_0), \ldots, E_{15}(x, k_0))$, где k_0 — ключ начального преобразования в терминах [17]. При фиксированном k_0 его можно рассматривать в обозначениях [18] как вектор-функцию $E(x, k_0) = E_{k_0}(x) : V_{128} \to V_{128}$, $E(x, k_0) = f_3 \circ f_2 \circ f_1(x, k_0)$.

Пусть подвектор $B_i = z$, $i \in \{0, \ldots, 15\}$, принимает все возможные значения из V_8, а остальные подвекторы блока данных фиксированы. Полученное множество обозначим $I_i = \{(B_0, \ldots, B_{i-1}, z, B_{i+1}, \ldots, B_{15}) : z \in V_8\}$. Рассмотрим для каждого индекса $m \in \in \{0, \ldots, 15\}$ при фиксированном $k_0 \in V_{128}$ значение интеграла $\sum\limits_{x \in I_i} E_m(x, k_0)$.

Теорема 1. В AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций для любого непустого множества $J \subset \{1 + 8s, \ldots, 8 + 8s\}$, $s \in \{0, \ldots, 15\}$, и любого множества $I = \{1, \ldots, 128\} \setminus \{8l + k : k = 1, \ldots, 8\}$, $l \in \{0, \ldots, 15\}$, для отображения $E(\cdot, k_0)$ при фиксированном ключе $k_0 \in V_{128}$ в обозначениях [18] верно, что

$$\sum_{x \in I_i} E_m(x, k_0) = 0.$$

Следствие 1. В AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций для любого непустого множества $J \subset \{1 + 8s, \ldots, 8 + 8s\}$, $s \in \{0, \ldots, 15\}$, и любого множества $I = \{1, \ldots, 128\} \setminus \{8l + k : k = 1, \ldots, 8\}$, $l \in \{0, \ldots, 15\}$, для отображения $E(\cdot, k_0)$ при фиксированном ключе $k_0 \in V_{128}$ в обозначениях [18] верно, что

$$w_I^J(E_{k_0}(x)) \equiv 0 \pmod 2.$$

Следствие 2. В условиях следствия 1 в обозначениях [19] выполняется

$$\Delta_\varnothing^J(E_{k_0}(x)) \equiv 0 \pmod 2.$$

Предположим, что нам известно 256 пар открытого и шифрованного текстов (x, y), $x \in I_i$, $y = y(x) = G(x, k_0^*) = f_4^*(E(x, k_0), k_4) = s(E(x, k_0)) \oplus k_4$, где k_0^* — истинный ключ, а s — некоторое нелинейное преобразование, зависящее от конкретного алгоритма; ключ j-й итерации вычисляется из ключа $(j-1)$-й с помощью функции ψ: $k_j = \psi(k_{j-1})$ — нелинейной подстановки на множестве двоичных векторов V_{128}; $k_4 = \psi^4(k_0^*)$. Требуется определить ключ четвёртой итерации $k_4 = (k_{4,0}, \ldots, k_{4,15})$. Опробование ключа будем осуществлять по байтам $k_{4,0}, \ldots, k_{4,15} \in V_8$. Блок шифртекста разобьём на компоненты по 8 битов: $y(x) = (y_0(x), \ldots, y_{15}(x))$, $y_m(x) \in V_8$. Для каждого индекса $m \in \{0, \ldots, 15\}$ для каждого варианта $k_{4,m}$ вычисляем $\sum\limits_{x \in I_i} s^{-1}(y_m(x) \oplus k_{4,m})$. Если эта сумма оказывается не равной нулю, то опробуемый вариант для $k_{4,m}$ отбраковываем. Если в результате ключ определился неоднозначно, то алгоритм можно повторить для другого i.

Итак, пусть у нас имеется отображение $G : V_{128} \times V_{128} \to V_{128}$, $G = f_1 f_2 f_3 f_4^*$. При фиксированном $k_0 \in V_{128}$ отображение $G_{k_0}(x) = G(x, k_0) : V_{128} \to V_{128}$ является биективным. Очевидно, что $G(x, k_0) = s \circ E(x, k_0) \oplus \psi^4(k_0)$.

Теорема 2. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$, является то, что при любом фиксированном ключе $k_0 \in V_{128}$ для отображения $E(x, k_0) = (g_1(x), \ldots, g_{128}(x))$, где $g_m(x) : V_{128} \to V_1$, $m \in \{1, \ldots, 128\}$, существует вектор $\alpha \in V_8 \setminus \{0\}$, такой, что для любого $(\beta_1, \ldots, \beta_{120}) \in V_{120}$ в обозначениях [18] выполняется

$$\left\| (\alpha, (g_{1+8j}, \ldots, g_{8+8j}))_{i_1, \ldots, i_{120}}^{\beta_1, \ldots, \beta_{120}} \right\| \neq 128$$

для некоторого $I = \{1, \ldots, 128\} \setminus \{8j + k : k = 1, \ldots, 8\} = \{i_1, \ldots, i_{120}\}$.

Следствие 3. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$, является то, что в условиях теоремы 2 существует непустое множество $J \subset \{1+8j, \ldots, 8+8j\}$, такое, что $w_I^J\left(E(x, k_0)\right) \neq 128$.

Следствие 4. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$, является то, что в условиях теоремы 2 существует непустое множество $J \subset \{1+8j, \ldots, 8+8j\}$, такое, что $\Delta_{\varnothing}^J(E(x, k_0)) \neq 0$.

Нас интересует, в каком случае выполняется равенство

$$\mathrm{Int}\,(k_{4,j}, j) = \sum_{x \in I_i} s^{-1}(y_j(x) \oplus k_{4,j}) = 0.$$

Очевидно, что это произойдёт, если байт ключа угадан верно.

Следствие 5. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$, является то, что существуют такие $x_1, x_2 \in I_i$, $x_1 \neq x_2$, что $y_j(x_1) = y_j(x_2)$.

Теорема 3. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$ для некоторого $j \in \{0, \ldots, 15\}$, является то, что отображение s является нелинейной подстановкой.

Рассмотрим мультимножество $\{y_j(x) : x \in I_i\}$, где $(y_0(x), \ldots, y_{15}(x)) = G(x, k_0^*)$, состоящее из 256 элементов, и его подмножество (уже не являющееся мультимножеством)

$$Y_j^* = \left\{\alpha \in V_8 : |\{x \in I_i : y_j(x) = \alpha\}| = 2k-1, k \in \mathbb{N}\right\}.$$

Теорема 4. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$, является то, что множество Y_j^* не пусто.

Пусть I^* — произвольное подмножество I, такое, что $\{y_j(x) : x \in I^*\} = Y_j^*$. Очевидно, что

$$\mathrm{Int}\,(k_{4,j}, j) = \sum_{x \in I_i} s^{-1}(y_j(x) \oplus k_{4,j}) = \sum_{x \in I^*} s^{-1}(y_j(x) \oplus k_{4,j}).$$

Следовательно, можно модифицировать технику применения интегрального метода тем, что интеграл будет вычисляться по меньшему множеству I^*.

Рассмотрим отображение

$$\tau_j(k_4) = \sum_{x \in I^*} s^{-1}(y_j(x) \oplus k_{4,j}), \quad \tau_j(k_4) : V_8 \to V_8.$$

Теорема 5. Необходимым условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций получить информацию о $k_{4,j}$, является то, что существует такой элемент $\alpha \in V_8$, что $\tau_j(\alpha) \neq 0$.

Теорема 6. Достаточным условием для того, чтобы интегральный метод позволял в AES-подобных схемах (SQUARE, Rijndael, Crypton) для четырёх итераций найти $k_{4,j}$, является то, что существует единственный элемент $\beta \in V_8$, такой, что $\tau_j(\beta) = 0$.

В рамках исследования интегрального метода проведено 16 наборов экспериментов для редуцированного до четырёх раундов алгоритма Rijndael по схеме из [20], в которых в качестве набора открытых текстов взято множество из 256 векторов $I_0 = \{x = (x_0, \ldots, x_{15}) \in V_{128}\}$, где x_0 принимает все значения из V_8, а остальные x_j равны нулю. На фиксированных множествах из 256 ключей $\{k_0^*\}_t$, $t \in \{1, \ldots, 16\}$, вычислялось множество шифртекстов $\{y = (y_0(x), \ldots, y_{15}(x)) = G(x, k_0^*) : x \in I_0\}$, а затем для заданного номера байта $j = j(t)$ для всех $k_{4,j} \in V_8$ вычислялся интеграл $\mathrm{Int}(k_{4,j}, j, I_0) = \sum\limits_{x \in I_0} s^{-1}(y_j(x) \oplus k_{4,j})$. Среди тех $k_{4,j}$, для которых $\mathrm{Int}(k_{4,j}, j, I_0)$ обращался в ноль, обязательно содержится истинный байт четвёртой итерации ключа, но он далеко не всегда единственен. В каждом из наборов экспериментов для каждого ключа из $\{k_0^*\}_t$ подсчитывалось, на скольких байтах $k_{4,j}$ интеграл $\mathrm{Int}(k_{4,j}, j, I_0)$ обращался в ноль. В качестве множеств $\{k_0^*\}_t = \left\{(k_{0,0}^*, \ldots, k_{0,15}^*) \in V_{128} : k_{0,l}^* \in V_8, l \in \{0, \ldots, 15\}\right\}_t$ брались векторы, у которых байт $k_{0,m}^*$ принимал все значения из V_8 при фиксированном $m = m(t) = j(t)$, а остальные байты равны нулю. Данные экспериментов сведены в таблицу, в которой приведено количество ключей из множества $\{k_0^*\}_t$, для которых интеграл $\mathrm{Int}(k_{4,j}, j, I_0)$ равен нулю на v байтах при всех $v \in \{1, \ldots, 7\}$.

t	$j(t)$	v						
		1	2	3	4	5	6	7
1	0	97	88	49	15	7	0	0
2	1	102	97	37	11	7	2	0
3	2	102	89	38	21	4	2	0
4	3	86	97	51	17	4	1	0
5	4	104	95	42	9	5	1	0
6	5	95	93	47	17	3	1	0
7	6	82	109	47	11	5	2	0
8	7	80	105	45	18	8	0	0
9	8	102	86	42	22	3	1	0
10	9	110	85	43	16	2	0	0
11	10	95	93	54	10	2	1	1
12	11	102	85	52	12	5	0	0
13	12	106	89	48	11	2	0	0
14	13	86	106	44	16	4	0	0
15	14	91	105	40	17	3	0	0
16	15	83	97	52	12	12	0	0

ЛИТЕРАТУРА

1. *Жуков А. Е.* Легковесная криптография. Ч. 1 // Вопросы кибербезопасности. 2015. № 1(9). С. 26–43.
2. *Жуков А. Е.* Легковесная криптография. Ч. 2 // Вопросы кибербезопасности. 2015. № 2(10). С. 2–10.
3. *Романченко А. М.* Метод оценивания результатов криптоанализа блочного шифра // Труды СПИИРАН. 2015. № 2(39). С. 101–108.
4. *Hu Y., Zhang Y., and Xiao G.* Integral cryptanalysis of SAFER+ // IET Electronics Let. 1999. V. 35. No. 17. P. 1458–1459.
5. *Daemen J., Knudsen L. R., and Rijmen V.* The block cipher Square // LNCS. 2002. V. 2365. P. 112–127.
6. *Lai X.* Higher order derivatives and differential cryptanalysis // Communications and Cryptography. Springer Intern. Ser. Engin. Comput. Sci. 1994. V. 276. P. 227–233.

7. *Collard B. and Standaert F.-X.* A statistical saturation attack against the block cipher PRESENT // LNCS. 2009. V. 5473. P. 195–210.
8. *Шнайер Б.* Прикладная криптография: протоколы, алгоритмы и исходный код. М.: Альфа-книга, 2019. 1024 с.
9. *Alda F., Aragona R., Nicolodi L., and Sala M.* Implementation and improvement of the partial sum attack on 6-round AES // Physical and Data-Link Security Techniques for Future Communication Systems. Lecture Notes in Electr. Eng. 2016. V. 358. P. 181–195.
10. *Сорокин М. А., Пудовкина М. А.* О почти совершенных нелинейных преобразованиях и разделяющем свойстве мультимножеств // Прикладная дискретная математика. Приложение. 2019. № 12. С. 237–239.
11. *ElSheikh M. and Youssef A. M.* Integral cryptanalysis of reduced-round tweakable TWINE // LNCS. 2020. V. 12579. P. 485–504.
12. *Hebborn P., Lambin B., Leander G., and Todo Y.* Strong and tight security guarantees against integral distinguishers // LNCS. 2021. V. 13090. P. 362–391.
13. *Knudsen L. R. and Wagner D.* Integral cryptanalysis (extended abstract) // LNCS. 2002. V. 2365. P. 112–127.
14. *Kiryukhin V. A.* Related-key attack on 5-round Kuznyechik // Математические вопросы криптографии. 2020. Т. 11. № 2. С. 53–67.
15. *Ferguson N., Kelsey J., Schneier B., et al.* Improved cryptanalysis of Rijndael // LNCS. 2000. V. 1978. P. 213–230.
16. *D'Halluin C., Bijnens G., Rijmen V., and Preneel B.* Attack on six rounds of Crypton // LNCS. 1999. V. 1636. P. 46–59.
17. *Лось А. Б., Нестеренко А. Ю., Рожков М. И.* Криптографические методы защиты информации. М.: Юрайт, 2018, 473 с.
18. *Панков К. Н.* Оценки скорости сходимости в предельных теоремах для совместных распределений части характеристик случайных двоичных отображений // Прикладная дискретная математика. 2012. № 4(18). С. 14–30.
19. *Панков К. Н.* Уточнённые асимптотические оценки для числа (n, m, k)-устойчивых двоичных отображений // Прикладная дискретная математика. Приложение. 2017. № 10. С. 46–49.
20. *Бабенко Л. К., Ищукова Е. А.* Современные алгоритмы блочного шифрования и методы их анализа. М.: Гелиос АРВ, 2006. 376 с.

УДК 519.7 + 004.056.55 DOI 10.17223/2226308X/15/16

СВОЙСТВА XS-СХЕМ, СВЯЗАННЫЕ С ГАРАНТИРОВАННЫМ ЧИСЛОМ АКТИВАЦИЙ[1]

Д. Р. Парфенов, А. О. Бахарев, А. В. Куценко, А. Р. Белов, Н. Д. Атутова

Гарантированное число активаций является важной криптографической характеристикой, позволяющей получить оценку стойкости блочного шифра к разностному криптоанализу. В работе исследован один из алгоритмов (Агиевич, 2020) поиска числа гарантированных активаций XS-схем. Предложен подход к оптимизации существующего решения с помощью метода ветвей и границ, а также анализа специальных матриц, характеризующих XS-схему. Для нескольких шифров проведены вычислительные эксперименты, которые демонстрируют существенное

[1]Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования РФ № 075-15-2022-281.

ускорение вычисления гарантированного числа активаций по сравнению с известными подходами. С помощью оптимизированной версии алгоритма проведены численные эксперименты. На основе полученных данных выдвинуто несколько гипотез, часть из которых доказана. Например, обнаружен класс XS-схем, обладающих наименьшими гарантированными числами активации, а также доказано равенство гарантированного числа линейных и разностных активаций.

Ключевые слова: *гарантированное число активаций, XS-схемы, разностный криптоанализ, линейный криптоанализ, метод ветвей и границ.*

Введение

На сегодняшний день разностный (дифференциальный) криптоанализ является одним из наиболее эффективных статистических методов анализа симметричных блочных шифров. Данный подход основывается на анализе разностей открытых текстов и разностей соответствующих им шифртекстов.

XS-схемы представляют собой конструкции блочных шифров, основанные на использовании двух операций: X (поразрядное сложение двоичных слов по модулю 2) и S (подстановка слов с помощью нелинейных взаимно-однозначных отображений). Известно, что модели XS-схем покрывают достаточно широкий спектр блочных шифров, включая SM4, Skipjack и схему Фейстеля.

В данной работе рассматривается задача оптимизации поиска гарантированного числа активаций в XS-схемах, что позволит получить оценку эффективности разностного криптоанализа таких шифров.

Рассмотрим поле $\mathbb{F}$ характеристики 2. Раундовое преобразование XS-схемы задаётся следующим образом:

$$(a, B, c)[S] : \mathbb{F}^n \to \mathbb{F}^n, (x_1, x_2, \ldots, x_n) \longmapsto (x_2, x_3, \ldots, x_n, x_{n+1}),$$
$$x_{n+1} = (x_1, x_2, \ldots, x_n)B + S((x_1, x_2, \ldots, x_n)a)c,$$

где $a, c \in \mathbb{F}_2^n$; B — булева матрица.

Множество регулярных XS-схем распадается на классы эквивалентности относительно отношения подобия [1]. Выберем в качестве представителей классов XS-схемы, находящиеся в *первой канонической форме* [1]. Класс XS-схем однозначно определяется парой векторов a и b, где b — последний столбец матрицы B, которая является фробениусовой клеткой, когда XS-схема находится в первой канонической форме.

Для каждого шифра можно найти число активаций. Данная характеристика позволяет получить нижнюю оценку сложности разностного криптоанализа шифра.

Определение 1. *Активация* — получение на вход S-блока векторов с ненулевой разностью при различных входных векторах раунда.

Определение 2. *Гарантированное число активаций* — наименьшее число активаций по всем ненулевым разностям между входными векторами за заданное число раундов.

Отметим, что поиск связан с исследованием линейного кода определенного вида, который строится на основе рассматриваемого шифра. Для каждого шифра из класса XS-схем (схема находится в первой канонической форме) строится матрица $G = G(n, a, b, t)$ размерности $(t + n) \times 2t$ (подробнее см. в [1]).

Одним из подходов к поиску гарантированного числа активаций является алгоритм GNA [2]. Основная идея алгоритма — полный перебор разбиений матрицы G на G_0 и G_1 так, чтобы:

1) матрица G_0 содержала $k+1$ (k изначально известно) пар столбцов из G;
2) $\mathrm{rank}(G_0) < t+n-1$, где t — число раундов, n — размерность векторов a и b;
3) в матрице G_1 не было ни одного столбца, линейно зависимого от столбцов в G_0.

1. Оптимизация алгоритма вычисления гарантированного числа активаций

Существующий алгоритм при увеличении t и n работает весьма длительное время. Предложен способ оптимизации разбиения матрицы G, который существенно ускоряет алгоритм. Его основная идея строится на подходе, основанном на методе ветвей и границ, который является развитием метода полного перебора с отсечением подмножеств допустимых решений, заведомо не содержащих оптимальных решений.

Предлагается следующая интерпретация полного перебора. Рассмотрим двоичное дерево, в котором каждый лист соответствует некоторому разбиению пар столбцов матрицы G на матрицы G_0 и G_1. Корень дерева соответствует пустому множеству пар столбцов матрицы G. При переходе с i-го уровня на $(i+1)$-й переход к правому потомку соответствует добавлению $(i+1)$-й пары столбцов в матрицу G_0, а переход влево — в G_1. Таким образом, каждый узел дерева соответствует некоторому разбиению подмножества пар столбцов матрицы G на матрицы G_0 и G_1.

Если при обходе дерева обрабатываемый узел соответствует разбиению, которое не позволяет увеличить имеющуюся оценку k (в G_1 существует столбец, линейно зависимый от столбцов G_0), то все потомки данного узла также будут соответствовать разбиениям, в которых в G_1 существует столбец, линейно зависимый от столбцов G_0, и поэтому их можно не обрабатывать.

Кроме того, дополнительные критерии отсечения могут быть получены на основе следующего свойства матрицы G:

Лемма 1. Рассмотрим n подряд идущих пар столбцов из G, где n — размерность XS-схемы. Тогда первый столбец из $(n+1)$-й пары линейно зависим от столбцов из этих пар.

Для краткости формулировок введём следующее

Определение 3. Будем называть разбиение пар столбцов матрицы G на матрицы G_0 и G_1 «*неподходящим*», если в G_1 содержится столбец, линейно зависящий от столбцов матрицы G_0.

Сформулируем достаточные условия того, что разбиение является «неподходящим»:

Теорема 1. Пусть G_0 содержит n подряд идущих в G пар столбцов. Если выполнено одно из следующих условий, то такое разбиение является «неподходящим»:

1) в G_1 содержится пара с бóльшим порядковым номером, чем у последней из n подряд идущих пар;
2) a_1 и b_1 одновременно не равны 1.

Более того, возможно заранее обнаружить, что обрабатываемая ветвь неизбежно приведёт к «неподходящему» разбиению.

Следствие 1. Если для разбиения, заданного обрабатываемым узлом дерева, невозможно дополнить матрицу G_0 до $k+1$ пары столбцов без появления n подряд идущих в G пар столбцов, то все листья, являющиеся потомками этого узла, соответствуют «неподходящим» разбиениям.

Поскольку «неподходящее» разбиение не позволяет увеличить имеющуюся оценку k, все его потомки также заведомо не содержат оптимального решения. Это позволяет использовать теорему 1 и следствие из неё в качестве дополнительных критериев отсечения.

2. Тестирование

Для проверки предложенного решения на практике было произведено несколько тестовых запусков. Референсная реализация GNA [3] выполнена на Python. Для корректности сравнения оптимизированный вариант также выполнен на Python и использует те же библиотеки, что и референсная реализация GNA. В таблице приведено время работы существующего и предложенного алгоритма на XS-схемах, описывающих блочные шифры SMS4 (стандарт КНР GB/T 32907-2016), skipjackg-4 (стандарт США FIPS 185 EES) и beltWBL-4 (стандарт Республики Беларусь СТБ 34.101.31-2020). Для старого алгоритма приведено время работы до 25 раундов, поскольку для большего числа раундов вычисления заняли бы существенно больше времени.

Время работы для SMS4, skipjackg-4 и beltWBL-4, с

Кол-во раундов	SMS4		skipjackg-4		beltWBL-4	
	Сущ. алг.	Предл. алг.	Сущ. алг.	Предл. алг.	Сущ. алг.	Предл. алг.
20	60,0129	1,2929	131,3367	0,2681	67,303	0,8394
21	89,1743	0,8057	231,5267	0,2155	176,574	2,5539
22	132,9802	0,4689	644,076	0,5805	391,1274	3,4973
23	197,0892	0,264	1589,793	1,4337	994,2484	6,6221
24	735,1466	1,4642	2916,3251	1,117	1677,1016	4,5253
25	2555,2445	7,1559	7348,0468	2,6989	3183,5355	2,9505
26	—	4,3972	—	2,1177	—	11,3771
29	—	8,0172	—	8,2899	—	26,1846
32	—	14,217	—	12,983	—	196,2868
35	—	169,632	—	37,8486	—	388,0823

3. Приложение к линейному криптоанализу

В [1] упомянуто, что алгоритм может быть применён к оценке стойкости к линейному криптоанализу. Для того чтобы вычислить гарантированное число линейных активаций, необходимо вычислить гарантированное число разностных активаций дуальной схемы. Матрица схемы, дуальной к заданной схеме (a, B, c), получается с помощью транспонирования матрицы B и обмена местами значений векторов a и c — $(c^{\mathrm{T}}, B^{\mathrm{T}}, a^{\mathrm{T}})$.

Интересно посмотреть, каким образом могут быть взаимосвязаны гарантированные числа разностных и линейных активаций. При анализе результатов численных экспериментов мы заметили, что они совпадают. В свою очередь, это означает, что совпадают и первые канонические формы XS-схем, что в конечном итоге было доказано.

Теорема 2. Пусть (a, B, c) — XS-схема в первой канонической форме. Тогда первая каноническая форма дуальной схемы $(c^{\mathrm{T}}, B^{\mathrm{T}}, a^{\mathrm{T}})$ совпадает с (a, B, c).

4. Схемы с экстремальным гарантированным числом активаций

С помощью усовершенствованной версии алгоритма вычисления гарантированного числа активаций осуществлён полный перебор представителей классов регулярных XS-схем для различных n и вычислены максимальные и минимальные гарантированные числа активаций для различного числа раундов. При анализе результатов выявлен

класс схем, которые обладают минимальными гарантированными числами активации. Для них установлена и доказана зависимость гарантированного числа активаций от размерности и числа раундов.

Утверждение 1. Пусть (a, B, c) — XS-схема в первой канонической форме размерности n и $a + b = (1, 0, \ldots, 0)$. Тогда k-я активация происходит на раунде kn.

Данные схемы образуют класс потенциально наименее стойких схем.

Кроме того, исходя из результатов численных экспериментов, можно выдвинуть следующие гипотезы:

Гипотеза 1. У всех схем размерности n, кроме схем с минимальным гарантированным числом активаций, число раундов, необходимое для k активаций, не превышает $kn - (k - 1)$.

Гипотеза 2. XS-схемы, которые задаются парами векторов (a, b) и (b, a), имеют одинаковые гарантированные числа активации.

Заключение

В работе проанализирован алгоритм GNA [2] и предложен подход к оптимизации существующего решения. Предлагаемые изменения позволяют существенно ускорить вычисление гарантированного числа активаций, а также вычислять гарантированное число активаций для большего количества раундов, чем референсная реализация. Проведены численные эксперименты, вычислены гарантированные числа активации для всех схем размерности 7 и меньше. На основе полученных данных выдвинуто несколько гипотез, часть из которых доказана, в том числе равенство гарантированного числа разностных и линейных активаций.

В дальнейшем планируется продолжать исследования с целью поиска принципов построения криптографических примитивов на основе XS-схем с оптимальными гарантированными числами активаций. Возможными применениями являются шифры, хэш-функции и генераторы псевдослучайных чисел. Авторы заинтересованы также в исследовании возможности использовать sponge-функции для построения примитивов с переменной размерностью.

ЛИТЕРАТУРА

1. *Агиевич С. В.* XS-circuits in block ciphers // Матем. вопр. криптогр. 2019. Т. 10. № 2. С. 7–30.
2. *Агиевич С. В.* On the guaranteed number of activations in XS-circuits // Матем. вопр. криптогр. 2021. Т. 12. № 2. С. 7–20.
3. Реализация алгоритмов поиска гарантированного числа активаций. `https://github.com/agievich/xs`.

УДК 519.7 DOI 10.17223/2226308X/15/17

О РАЗНОСТНЫХ ХАРАКТЕРИСТИКАХ КОМПОЗИЦИЙ ПОБИТОВЫХ XOR ПО МОДУЛЮ 2^n [1]

И. А. Сутормин

Рассматривается разностная характеристика $\mathrm{adp}_k^{\oplus}$ композиции побитовых XOR относительно сложения по модулю 2^n. Эта величина используется при анализе примитивов, имеющих конструкцию Addition-Rotation-XOR (ARX). Получены рекуррентные формулы, позволяющие найти значение $\mathrm{adp}_k^{\oplus}$ от аргументов размерности $n+1$ при помощи набора значений $\mathrm{adp}_k^{\oplus}$ от аргументов размерности n. Изучены симметрии и нули характеристики. В случае чётного k найден максимум $\mathrm{adp}_k^{\oplus}$ при одном фиксированном аргументе.

Ключевые слова: *разностный криптоанализ, ARX, XOR, сложение по модулю.*

Существуют различные подходы для разработки алгоритмов симметричной криптографии. Один из них — ARX. Во всех примитивах этой архитектуры используются только три операции: сложение по модулю 2^n ($\boxplus$), циклический сдвиг битов и побитовое сложение по модулю 2 ($\oplus$, XOR). ARX-шифры могут быть различных назначений, например блочные шифры FEAL [1], Threefish [2], поточные шифры Salsa20 [3] и его модификация ChaCha [4], хэш-функции BLAKE [5] и Skein [2]. Разностный криптоанализ — один из современных методов криптоанализа, предложенный в [6]. Для проведения разностного криптоанализа ARX-шифров при выборе в качестве разности сложения по модулю 2^n необходима разностная характеристика $\mathrm{adp}^{\oplus}$:

$$\mathrm{adp}^{\oplus}(\alpha, \beta \to \gamma) = \frac{1}{4^n}\big|\{x, y \in \mathbb{Z}_2^n : (x \boxplus \alpha) \oplus (y \boxplus \beta) = (x \oplus y) \boxplus \gamma\}\big|.$$

Здесь и далее с вектором $x \in \mathbb{Z}_2^n$ ассоциируется целое число $x_n + x_{n-1}2^1 + \ldots + x_1 2^{n-1}$.

Многие свойства $\mathrm{adp}^{\oplus}$ изучены в работах [7, 8]. Однако в некоторых ARX-шифрах присутствует применение композиции побитовых XOR. Так, например, в хэш-функции EDON-R [9] используется XOR трёх векторов. В этом случае использование характеристики $\mathrm{adp}^{\oplus}$ может привести к неверным оценкам. Более точные оценки можно получить при прямом использовании аналогичной характеристики для XOR нескольких векторов, которая определяется как

$$\mathrm{adp}_k^{\oplus}(\alpha^1, \ldots, \alpha^k \to \alpha^{k+1}) = \frac{1}{2^{kn}}\Big|\{x^1, \ldots, x^k \in \mathbb{Z}_2^n : \bigoplus_{i=1}^{k}(x^i \boxplus \alpha^i) = \alpha^{k+1} \boxplus \bigoplus_{i=1}^{k} x^i\}\Big|.$$

Здесь и далее $k \geqslant 2$.

Многие свойства $\mathrm{adp}_k^{\oplus}$ и $\mathrm{adp}^{\oplus}$ схожи. Так, в частности, симметрии аргументов $\mathrm{adp}_k^{\oplus}$ аналогичны симметриям $\mathrm{adp}^{\oplus}$, описанным в [8, разд. 4]. Однако случай замены аргументов на обратные относительно сложения по модулю 2^n в случае нечётного k отличается от чётного k. В нечётном случае на обратный можно заменить только пару элементов одновременно.

Теорема 1. Для любого набора аргументов характеристика $\mathrm{adp}_k^{\oplus}$ обладает следующими свойствами:

[1]Работа выполнена при поддержке Математического центра в Академгородке, соглашение с Министерством науки и высшего образования РФ № 075-15-2022-281.

1) $\mathrm{adp}_k^{\oplus}$ — симметрическая функция, то есть её значение не изменится при перестановке аргументов. Например, для любых $\alpha^1, \ldots, \alpha^4 \in \mathbb{Z}_2^n$ справедливо

$$\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3 \to \alpha^4) = \mathrm{adp}_3^{\oplus}(\alpha^2, \alpha^1, \alpha^3 \to \alpha^4).$$

2) Значение $\mathrm{adp}_k^{\oplus}$ не изменится, если к любым двум аргументам прибавить 2^{n-1} по модулю 2^n. Например, для любых $\alpha^1, \ldots, \alpha^4 \in \mathbb{Z}_2^n$ справедливо

$$\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3 \to \alpha^4) = \mathrm{adp}_3^{\oplus}(\alpha^1 \boxplus 2^{n-1}, \alpha^2 \boxplus 2^{n-1}, \alpha^3 \to \alpha^4).$$

3) Значение $\mathrm{adp}_k^{\oplus}$ не изменится, если два аргумента одновременно заменить на обратные относительно сложения по модулю 2^n. Например, для любых $\alpha^1, \ldots, \alpha^4 \in \mathbb{Z}_2^n$ справедливо

$$\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3, \to \alpha^4) = \mathrm{adp}_3^{\oplus}(-\alpha^1, -\alpha^2, \alpha^3, \to \alpha^4).$$

4) При чётном k значение $\mathrm{adp}_k^{\oplus}$ не изменится, если любой из аргументов заменить на обратный относительно сложения по модулю 2^n. Например, для любых $\alpha^1, \ldots, \alpha^5 \in \mathbb{Z}_2^n$ справедливо

$$\mathrm{adp}_4^{\oplus}(\alpha^1, \alpha^2, \alpha^3, \alpha^4 \to \alpha^5) = \mathrm{adp}_4^{\oplus}(-\alpha^1, \alpha^2, \alpha^3, \alpha^4 \to \alpha^5).$$

Для вектора $\alpha \in \mathbb{Z}_2^n$ обозначим через $\alpha||1$ и $\alpha||0$ векторы $(\alpha_1, \ldots \alpha_n, 1)$ и $(\alpha_1, \ldots \alpha_n, 0)$ из $\mathbb{Z}_2^{n+1}$ соответственно; вес Хэмминга $\mathrm{wt}(\alpha) = \sum\limits_{i=1}^{n} \alpha_i$. Запись $b \preceq a$ обозначает, что для векторов $a, b \in \mathbb{Z}_2^{k+1}$ выполнено $b_i \leqslant a_i$, $i = 1, \ldots, k+1$. Тогда для $\mathrm{adp}_k^{\oplus}$ можно доказать рекуррентные формулы, аналогичные [8, теорема 3] и позволяющие получить значение $\mathrm{adp}_k^{\oplus}$ от аргументов размерности $n+1$ при помощи набора значений $\mathrm{adp}_k^{\oplus}$ от аргументов размерности n.

Теорема 2. Для любого набора векторов $\alpha^1, \ldots, \alpha^{k+1} \in \mathbb{Z}_2^n$ и вектора $a \in \mathbb{Z}_2^{k+1}$, составленного из младших бит аргументов, выполняются следующие равенства:

1) если $\mathrm{wt}(a)$ нечётный, то $\mathrm{adp}_k^{\oplus}(\alpha^1||a_1, \ldots \alpha^k||a_k \to \alpha^{k+1}||a_{k+1}) = 0$;
2) если k нечётное и $a = (1, \ldots, 1)$, то

$$\mathrm{adp}_k^{\oplus}(\alpha^1||1, \ldots \alpha^k||1 \to \alpha^{k+1}||1) =$$
$$= \frac{1}{2^k} \sum_{\substack{b \preceq a, \\ \mathrm{wt}(b) - \text{четн.}}} \mathrm{adp}_k^{\oplus}(\alpha^1 \boxplus b_1, \ldots, \alpha^k \boxplus b_k \to \alpha^{k+1} \boxplus b_{k+1});$$

3) во всех остальных случаях

$$\mathrm{adp}_k^{\oplus}(\alpha^1||a_1, \ldots \alpha^k||a_k \to \alpha^{k+1}||a_{k+1}) =$$
$$= \frac{1}{2^{\mathrm{wt}(a)}} \sum_{b \preceq a} \mathrm{adp}_k^{\oplus}(\alpha^1 \boxplus b_1, \ldots, \alpha^k \boxplus b_k \to \alpha^{k+1} \boxplus b_{k+1}).$$

Так, например, для любых $\alpha^1, \ldots, \alpha^4 \in \mathbb{Z}_2^n$, согласно п. 1, справедливо

$$\mathrm{adp}_3^{\oplus}(\alpha^1||0, \alpha^2||0, \alpha^3||1 \to \alpha^4||0) = 0.$$

Согласно п. 3, справедливо

$$\mathrm{adp}_3^{\oplus}(\alpha^1||0, \alpha^2||0, \alpha^3||1 \to \alpha^4||1) = \frac{1}{4}\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3 \to \alpha^4) + \frac{1}{4}\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3 \boxplus 1 \to \alpha^4 \boxplus 1) +$$
$$+ \frac{1}{4}\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3 \boxplus 1 \to \alpha^4) + \frac{1}{4}\mathrm{adp}_3^{\oplus}(\alpha^1, \alpha^2, \alpha^3 \to \alpha^4 \boxplus 1).$$

Заметим, что ранее известные формулы для $\mathrm{adp}^{\oplus}$ описываются пп. 1 и 3.

Для вектора $\alpha \in \mathbb{Z}_2^n$ обозначим через $\overline{\alpha}$ вектор $(\alpha_1 \oplus 1, \ldots, \alpha_n \oplus 1)$. Тогда симметрии из теоремы 1 позволяют в некоторых случаях записать рекуррентные формулы при помощи операции инверсии, анализировать которую проще, чем сложение по модулю. Например,

$$\mathrm{adp}_3^{\oplus}(\alpha||1, \alpha||1, \alpha||1 \to \alpha||1) = \frac{3}{4}\mathrm{adp}_3^{\oplus}(\overline{\alpha}, \overline{\alpha}, \alpha \to \alpha) + \frac{1}{8}\mathrm{adp}_3^{\oplus}(\overline{\alpha}, \overline{\alpha}, \overline{\alpha} \to \overline{\alpha}) + \\ + \frac{1}{8}\mathrm{adp}_3^{\oplus}(\alpha, \alpha, \alpha \to \alpha).$$

Рекуррентные формулы позволяют также найти максимум $\mathrm{adp}_k^{\oplus}$ при чётном k, аналогичный максимуму при $k = 2$, доказанному в [8, теорема 2].

Теорема 3. Для любого $\gamma \in \mathbb{Z}_2^n$ и любого чётного k выполняется

$$\max_{\alpha^1, \ldots, \alpha^k \in \mathbb{Z}_2^n} \mathrm{adp}_k^{\oplus}(\alpha^1, \ldots, \alpha^k \to \gamma) = \mathrm{adp}_k^{\oplus}(0, \ldots, 0, \gamma \to \gamma).$$

Однако при нечётном k данное утверждение неверно и аналогичный максимум $\mathrm{adp}_k^{\oplus}$ выглядит иначе. Мы предполагаем, что он выглядит так:

Гипотеза 1. Для любого $\gamma \in \mathbb{Z}_2^n$ и любого нечётного k выполняется

$$\max_{\alpha^1, \ldots, \alpha^k \in \mathbb{Z}_2^n} \mathrm{adp}_k^{\oplus}(\alpha^1, \ldots, \alpha^k \to \gamma) = \mathrm{adp}_k^{\oplus}(\gamma, \ldots, \gamma \to \gamma).$$

Гипотеза подтверждается вычислительными экспериментами и разбором некоторых частных случаев. В случае $k = 3$ для этого полезны следующие неравенства:

Теорема 4. Для любого $\gamma \in \mathbb{Z}_2^n$ выполняется

$$\mathrm{adp}_3^{\oplus}(\gamma, \gamma, \overline{\gamma} \to \overline{\gamma}) \leqslant \mathrm{adp}_3^{\oplus}(\gamma, \gamma, \gamma \to \gamma) \leqslant 3\,\mathrm{adp}_3^{\oplus}(\gamma, \gamma, \overline{\gamma} \to \overline{\gamma}).$$

Отметим, что для разностного криптоанализа важно различать наборы аргументов, на которых $\mathrm{adp}_k^{\oplus}$ равно нулю.

Теорема 5. При любом k и любом наборе аргументов $\alpha^1, \ldots, \alpha^{k+1} \in \mathbb{Z}_2^n$ значение $\mathrm{adp}_k^{\oplus}(\alpha^1, \ldots, \alpha^k \to \alpha^{k+1}) = 0$ тогда и только тогда, когда существует позиция i, такая, что вектор $(\alpha_i^1, \ldots, \alpha_i^{k+1}) \neq (0, \ldots, 0)$, а для любой позиции j, $n \geqslant j > i$ верно $(\alpha_j^1, \ldots, \alpha_j^{k+1}) = (0, \ldots, 0)$, и выполняется одно из следующих условий:

1) вектор $(\alpha_i^1, \ldots, \alpha_i^{k+1})$ имеет нечётный вес;
2) k нечётное, $i > 1$, вектор $(\alpha_i^1, \ldots, \alpha_i^{k+1})$ равен $(1, \ldots, 1)$ и вектор битов на разряд выше $(\alpha_{i-1}^1, \ldots, \alpha_{i-1}^{k+1})$ имеет нечётный вес.

Заметим, что нули функции в случае чётного k выглядят аналогично нулям для $\mathrm{adp}^{\oplus}$. Случай 2 появляется только при нечётном k и порождает дополнительное множество нулей характеристики.

ЛИТЕРАТУРА

1. *Shimizu A. and Miyaguchi S.* Fast data encipherment algorithm FEAL // LNCS. 1988. V. 304. P. 267–278.
2. *Ferguson N., Lucks S., Schneier B., et al.* The Skein Hash Function Family. `http://www.skein-hash.info`. 2009.
3. *Bernstein D. J.* Salsa20 Specification. `https://cr.yp.to/snuffle/spec.pdf`. 2005.

4. *Bernstein D. J.* ChaCha, a Variant of Salsa20. `https://cr.yp.to/chacha/chacha-20080128.pdf`. 2008.
5. *Aumasson J.-P., Meier W., Phan R. C.-W., and Henzen L.* The Hash Function BLAKE. `https://www.researchgate.net/publication/316806226_The_Hash_Function_BLAKE`. 2014.
6. *Biham E. and Shamir A.* Differential cryptanalysis of DES-like cryptosystems // J. Cryptology. 1991. No. 4. P. 3–72.
7. *Lipmaa H., Wallen J., and Dumas P.* On the additive differential probability of exclusive-or // LNCS. 2004. V. 3017. P. 317–331.
8. *Mouha N., Kolomeec N., Akhtiamov D., et al.* Maximums of the additive differential probability of Exclusive-Or // IACR Trans. Symmetric Cryptology. 2021. No. 2. P. 292–313.
9. *Gligoroski D., Odegard R. S., Mihova M., et al.* Cryptographic hash function Edon-R // Proc. IWSCN. 2009. P. 1–9.

UDC 519.17 DOI 10.17223/2226308X/15/18

KEY SCHEDULE BASED ON A MODIFIED ADDITIVE GENERATOR[1]

V. M. Fomichev, D. A. Bobrovskiy, R. R. Sotov

A method of round key generation for iterated block ciphers based on a modified additive generator (MAG), and, in addition, on MAG and a linear congruent generator in a series circuit is proposed. The bijectivity of the generating transformation is demonstrated. Using the matrix-graph approach the number of iterations necessary for achieving enhanced cryptographic properties is experimentally evaluated. This number depends on the generator characteristics.

Keywords: *key scheduling algorithm, iterative block ciphers, matrix-graph approach, modified additive generator, mixing properties, nonlinearity.*

1. Introduction

The key schedule is an important component of any iterated block cipher. The first versions of key schedules (DES, GOST 28147-89) involved bit sampling from the cipher key which gives the cryptanalyst grounds for attacks such as differential analysis. In AES, the generation of round keys is more complex and requires a non-stationary recurrence relation over a set of binary vectors. The Kuznechik algorithm provides a complex key dependency using the Feistel network. The goal of key schedule algorithms is to combine a complex functional relationship between the bits of the cipher key and the round keys with a relatively low computational complexity of key generation.

This paper proposes a round key generator (RKG) based on a modified additive generator and, in addition, on MAG and a linear congruent generator (LCG) in a series circuit.

2. Additive generator

The additive generator (AG) is a shift register of length n with feedback $f(z_0, \ldots, z_{n-1})$ over the space of binary r-dimensional vectors, i.e., a register transformation φ of the set $V_{nr} = \{(z_0, \ldots, z_{n-1}) : z_0, \ldots, z_{n-1} \in V_r\}$:

$$\varphi(z_0, \ldots, z_{n-1}) = (z_1, \ldots, z_{n-1}, f(z_0, \ldots, z_{n-1})), \tag{1}$$

where the function $f\colon V_{nr} \to V_r$ is the shift register feedback function.

The AG feedback has the following form: $f(z_0, \ldots, z_{n-1}) = z_0 \boxplus z_2 \boxplus z_4 \boxplus z_6$, where $\boxplus$ is addition modulo 2^r, that is, f is bijective on the variable z_0, hence the transformation φ is also bijective [1]. It has insufficient mixing: the leading bits of all vectors depend only on the least significant ones. Therefore, a transformation φ^g of the modified AG is proposed, in which the vector value of the feedback is transformed by permutation g of the set V_r. The feedback of the MAG is denoted by f^g. It is proved that the transformation $\varphi^g(z_0, \ldots, z_{n-1})$ is bijective if and only if the function φ is bijective and g is a permutation [2].

The MAG is studied given $r = 32$, where the transformation $g(k)$ is a cyclic shift permutation of binary vectors by k bits towards the leading bits. Hence $\varphi^{g(k)}$ is a bijection of the set V_{nr}, and the feedback function has the form

$$f^{g(k)}(z_0, \ldots, z_{n-1}) = \mathrm{Int}_{32}(g(k)(\mathrm{Vec}_{32}(z_0 \boxplus z_2 \boxplus z_4 \boxplus z_6))), \tag{2}$$

where $\mathrm{Vec}_{32}\colon \mathbb{Z}_{2^{32}} \to V_{32}$ is a bijection that maps a number $X \in \mathbb{Z}_{2^{32}}$ to its binary representation, $\mathrm{Int}_{32} = \mathrm{Vec}_{32}^{-1}$ is the inverse function.

Given $t = 0, 1, 2, \ldots$, we denote:

- $g(k)$ — left cyclic shift by 1 bit ($k = 1$);
- $X_j^{(t)}$ — the state of the j-th MAG cell at t, $j = 0, 1, \ldots, 6$;
- $X^{(t)} = (X_0^{(t)}, X_1^{(t)}, \ldots, X_6^{(t)})$ — the state of the MAG at t;
- $X^{(0)}$ — the initial state of MAG.

The key of MAG is its initial state. From (1) and (2) we get:

$$X^{(t+1)} = (X_1^{(t)}, \ldots, X_6^{(t)}, g(k)(X_0^{(t)} \boxplus X_2^{(t)} \boxplus X_4^{(t)} \boxplus X_6^{(t)})). \tag{3}$$

A round key sequence is formed as an irregular sample from the sequence $\{X_0^{(t)}\}$, $t = 0, 1, 2, \ldots$

3. Cyclic structure of the MAG state digraph

To avoid repetitions in the sequence of round keys, short cycles of length less than 300 are undesirable in the cyclic transformation structure (this limit is determined by the required number of round keys in a number of block ciphers). The structure of the MAG state digraph was studied.

We denote by $\Gamma(\varphi^{g(k)})$ the $\varphi^{g(k)}$ transformation digraph, i.e., $\Gamma(\varphi^{g(k)}) = (V_{224}, E)$, where E is the set of arcs. The arc (z^i, z^j) exists if $\varphi^{g(k)}(z^i) = z^j$.

The digraph $\Gamma(\varphi^{g(k)})$ has one self-loop generated by the zero fill of MAG.

The cyclic structure of the digraph $\Gamma(\varphi^{g(k)})$ was studied given $k = 1$, $r = 4, 5, 6$ (as r increases, the computational complexity increases rapidly). Using an algorithm that generates a sequence of MAG states, 11 cycles at $r = 4$, more than 18 cycles at $r = 5$, and more than two cycles at $r = 6$ are found in the corresponding digraphs. In Table 1, the lengths of the found cycles are provided.

At $r = 32$, the lengths of the cycles are also experimentally evaluated. We assume 10^8 generated pseudorandom numbers as initial values. For each of the numbers assumed 1000 clock cycles of the generator are implemented (which is enough to generate all the round keys). It is obtained that the length of all cycles exceeds 1000, which excludes repetitions in the sequence of round keys.

The results of the experiment given $r = 32$ suggest that a randomly chosen state of MAG with a probability close to 1 belongs to a cycle of the length greater than 1000.

Table 1

Cycles lengths

Cycle number	Length, $r = 4$	Length, $r = 5$	Length, $r = 6$
1	234 711 845	16 871 058 994	837 124 439 025
2	17 076 802	13 808 636 426	117 617 876 965
3	15 925 876	1 965 696 526	
4	305 050	1 122 723 601	
5	208 004	233 097 005	
6	91 195	151 954 479	
7	67 889	65 351 609	
8	28 603	43 458 018	
9	11 552	41 627 677	
10	7 497	29 128 117	
11	1 142	14 671 598	
12		10 296 293	
13		1 134 118	
14		460 091	
15		212 519	
16		120 918	
17		62 980	
18		22 785	

4. Round keys generator using LCG

In a RKG based on the MAG and LCG series circuit, the key is the initial state of LCG and MAG. The recurrence of the LCG can be represented as

$$X_{n+1} = (aX_n + c) \bmod m, \quad n \geqslant 0,$$

where a is a multiplier, m is a modulus, c is a shift and X_0 is an initial value.

Recommended LCG parameters: $a = 1$, $m = 2^{32}$, c is an odd number, guaranteeing the full-cycle permutation of LCG [3].

The MAG+LCG automaton model's transition function is injective regarding the input variable, hence the period length of the sequence of GRK states is a multiple of the period length of LCG, that is a multiple of 2^{32} [4].

From (3) we get

$$X^{(t+1)} = (X_1^{(t)}, \ldots, X_6^{(t)}, g(k)(X_0^{(t)} \boxplus X_2^{(t)} \boxplus X_4^{(t)} \boxplus X_6^{(t)}) \boxplus (K_0 \boxplus c(t+1))),$$

where K_0 is the lowest 32 bits of the initial key, $c \in \mathbb{Z}_{2^{32}}$ is an odd number, $t = 1, 2, 3, \ldots$ is the sequence number of iteration of the RKG. Consequently, the period length of the sequence $\{X_0^{(t)}\}$ is guaranteed to be at least 2^{32}.

5. Mixing properties and nonlinearity

The RKG parameter k influences the key schedule properties of nonlinearity and mixing. These properties are evaluated using the local exponent of the mixing digraph for RKG state permutations (according to the matrix-graph approach [5]). After evaluation, the properties are determined experimentally.

The experiment results are presented here given different k. The least number of the RKG clock cycles is found after which each vector $\{X_0^{(t)}\}$ coordinate depends essentially and nonlinearly on each initial state bit. In Table 2, the results for $k = 1, 3, 5$ are provided.

Table 2

Experimental evaluation of total mixing and nonlinearity characteristics

k	Round t of total mixing	Round t of nonlinearity
1	30	33
3	18	20
5	16	18

6. Conclusion

Advanced characteristics of RKG based on MAG are shown both with and without the use of LCG. In the first case, the structural properties of the permutation states of RKG are guaranteed by the LCG parameters. In the second case, they are justified experimentally. The computational complexity of the round key generation method is low, which can be explained by uncomplicated implementation of MAG and LCG.

The presented method of key schedule generation can be used in many iterated block ciphers, in particular, the method is recommended for wide-block algorithm KB-256.

REFERENCES

1. *Fomichev V. M.* Metody diskretnoi matematiki v kriptologii [Methods of Discrete Mathematics in Cryptology]. Moscow, Dialog-MEPHI, 2012. 424 p. (in Russian)
2. *Koreneva A. M. and Fomichev V. M.* The mixing properties of modified additive generators. J. Appl. Industr. Math., 2017, vol. 11, no. 2, pp. 215–226.
3. *Knuth D. E.* The Art of Computer Programming. Vol. 2. Seminumerical Algorithms. Third ed. Reading, Massachusetts, Addison-Wesley, 1997. xiv+762 p.
4. *Fomichev V. M. and Melnikov D. A.* Kriptograficheskie metody zashchity informatsii. Ch. 1. Matematicheskie aspekty [Cryptographic Methods of Information Protection. P. 1. Mathematical Aspects]. Moscow, Urait Publ., 2016. 209 p. (in Russian)
5. *Fomichev V. M. and Koreneva A. M.* Encryption performance and security of certain wide block ciphers. J. Comput. Virol. Hack. Tech., 2020, vol. 16, pp. 197–216.

УДК 519.17 DOI 10.17223/2226308X/15/19

THE DIFFERENCE RELATIONS AND IMPOSSIBLE DIFFERENTIALS CONSTRUCTION FOR THE KB-256 ALGORITHM

V. M. Fomichev, A. V. Kurochkin, A. B. Chukno

In this paper, new results of the analysis of the KB 256-3 block cipher algorithm are outlined. We set up a difference relation with probability 1 for the six-round algorithm under study and propose a key recovery method using this difference relation for the nine-round KB 256-3 algorithm. We construct an impossible differential for the full-round algorithm.

Keywords: *differential cryptanalysis, impossible differentials.*

1. Introduction

The existence of a difference relation for a block cipher algorithm may indicate the possibility of developing efficient key recovering methods. We show that difference relations discovered for a block cipher algorithm can be efficiently used for key recovery computation (as compared to exhaustive key search) for the nine-round KB 256-3 algorithm. The

existence of an impossible differential for a block cipher algorithm enables cryptanalysts to recover information about encrypted blocks.

2. Description of the KB 256-3 encryption algorithm

The KB 256-3 encryption algorithm, based on the generalized Feistel network, was proposed in [1, 2]. Next, the algorithm description is provided.

We introduce notations as follows:

— $\boxplus$ — the addition modulo 2^{32};
— $\oplus$ — the XOR of two binary strings of the same length;
— V_n — a set of binary strings of length $n \in \mathbb{N}$, where $V = \{0, 1\}$;
— $K = (K_0, K_1, \ldots, K_7)$, where $K_j \in V_{32}$, $j = 0, \ldots, 7$, — an encryption key;
— $Q_i = (q_{0.i}, q_{1.i}, q_{2.i})$, where $q_{0.i}, q_{1.i}, q_{2.i} \in V_{32}$, $i = 1, \ldots, 16$, — round keys, derived from the encryption key.

Encryption of a 256-bit block $X = (X_0, X_1, X_2, \ldots, X_7)$, where $X_j^0 \in V_{32}$, $j = 0, \ldots, 7$, with an encryption key K can be performed by applying 16-round functions R, in sequence. Each of these functions depends on three 32-bit round keys (q_0^i, q_1^i, q_2^i), $i = 1, \ldots, 16$ (i.e., each round of encryption uses three round keys). We denote the round transformation of the KB 256-3 encryption algorithm by $R : V_{256} \times V_{96} \to V_{256}$.

As a result, after the round $i \in \{1, \ldots, 16\}$, the block $X \in V_{256}$ encrypted with the key K can be written as

$$R(\ldots R(R(X^0, Q_1), Q_2)), \ldots, Q_i) = X^i = (X_0^i, X_1^i, \ldots, X_7^i).$$

We introduce the additional notation:

$$F(X, K) = R(\ldots R(R(X, Q_1), Q_2)), \ldots, Q_{16}).$$

3. Round transformation

We define a round transformation. We use notations as follows:

1) $\Sigma(A_0, A_1, \ldots, A_7) = A_1 \boxplus A_3 \boxplus A_4 \boxplus A_6 \boxplus A_7$, where $A_i \in V_{32}$, $i = 0, \ldots, 7$;
2) $f(a_0, a_1, \ldots, a_7) = T(s_0(a_0), s_1(a_1), \ldots, s_7(a_7))$, where 4-bit permutations $s_0, s_1, \ldots, s_7$ are taken from [3], T is the left cyclic shift of a 32-bit string by 19 positions, $a_i \in V_4$, $i = 0, \ldots, 7$.

Hence, the round transformation can be written as

$$R\,(A, (b_0, b_1, b_2)) =$$
$$= (A_1, A_2 \oplus f(\Sigma(A) \boxplus b_0), A_3, A_4, A_5 \oplus f(\Sigma(A) \boxplus b_1), A_6, A_7, A_0 \oplus f(\Sigma(A) \boxplus b_2)).$$

4. Round key sequence

To construct a sequence q_j based on the key $K = (K_0, K_1, \ldots, K_7)$, $K_j \in V_{32}$, $j = 0, \ldots, 7$, we use the non-linear shift register with $\alpha \in V_{32}$ as a parameter. The initial state of the register is:

$$q_1 = K_0,\ q_2 = K_1,\ q_3 = K_2,\ q_4 = K_3,\ q_5 = K_4,\ q_6 = K_5,\ q_7 = K_6;$$
$$q_i = T_1\,[q_{i-1} \boxplus q_{i-3} \boxplus q_{i-5} \boxplus q_{i-7}] \boxplus K_7 \boxplus (i-7)\alpha,$$

where $i \in \{8, \ldots, 123\}$ and T_1 is a left cyclic shift of a string from V_{32}.

5. Difference relation

We define the difference relation for the algorithm under study. Let $X^0, \underline{X}^0$ be plaintexts:

$$X^0 = \left(X_0^0, X_1^0, X_2^0, X_3^0, X_4^0, X_5^0, X_6^0, X_7^0\right),$$
$$\underline{X}^0 = \left(X_0^0, X_1^0 \oplus 2^{31}, X_2^0, X_3^0, X_4^0, X_5^0, X_6^0 \oplus 2^{31}, X_7^0\right).$$

It is evident that $X^0 \oplus \underline{X}^0 = (0, 2^{31}, 0, 0, 0, 0, 2^{31}, 0)$. For plaintexts X^0 и $\underline{X}^0$ and any round keys $Q_1, Q_2, \ldots, Q_6 \in V_{96}$ the following equations hold:

$$R(\ldots R(X^0, Q_1), \ldots, Q_i) \oplus R(\ldots R(\underline{X}^0, Q_1), \ldots, Q_i) = C_i,$$

where $i = 1, \ldots, 6$ and constant $C_1, C_2, \ldots, C_6$ are

$$C_1 = (2^{31}, 0, 0, 0, 0, 2^{31}, 0, 0); \quad C_2 = (0, 0, 0, 0, 2^{31}, 0, 0, 2^{31}); \quad C_3 = (0, 0, 0, 2^{31}, 0, 0, 2^{31}, 0);$$
$$C_4 = (0, 0, 2^{31}, 0, 0, 2^{31}, 0, 0); \quad C_5 = (0, 2^{31}, 0, 0, 2^{31}, 0, 0, 0); \quad C_6 = (2^{31}, 0, 0, 2^{31}, 0, 0, 0, 0).$$

Thus, the difference relation with probability 1 for the six-round algorithm is provided.

6. Difference relation attack on 9 rounds

We consider the truncated KB-256 algorithm which comprises 9 encryption rounds. The algorithm structure besides the number of rounds is similar to that of the original algorithm.

Let X^0 and $\underline{X}^0$ be plaintexts such that $X^0 \oplus \underline{X}^0 = C_0$. The encrypted plaintexts $X^9, \underline{X^9}$ are known to the cryptanalyst.

It is also known that $X^6 \oplus \underline{X}^6 = (2^{31}, 0, 0, 2^{31}, 0, 0, 0, 0)$. Due to the algorithm functioning principles, the following equations hold:

$$X_4^6 = X_2^8; \quad X_7^6 = X_5^8.$$

The equations are easy to verify.

Next, we demonstrate how round keys q_1^9, q_2^9 can be recovered. For ease, we denote $a = q_1^9$, $b = q_2^9$. The cryptanalyst derives:

$$\begin{aligned} Y_1 &= X_1^9 \oplus f(X_0^9 \boxplus X_2^9 \boxplus X_3^9 \boxplus X_5^9 \boxplus X_6^9 \boxplus a); \\ Y_2 &= \underline{X}_1^9 \oplus f(\underline{X}_0^9 \boxplus \underline{X}_2^9 \boxplus \underline{X}_3^9 \boxplus \underline{X}_5^9 \boxplus \underline{X}_6^9 \boxplus a). \end{aligned} \tag{1}$$

The Y_1, Y_2 values potentially coincide X_2^8 and $\underline{X}_2^8$ respectively. It is known that $X_2^8 = \underline{X}_2^8$. So for the key a the following equation holds: $Y_1 = Y_2$.

The b key can be recovered in the same way. Generally, it is possible that for several a values equations 1 hold. In this section, we study the KB-256 algorithm properties without delving into the key recovery algorithm. Therefore, for ease, we assume that having a single a, the equations 1 hold. Obviously, recovering a round key allows recovering the key within approximately 2^{224} operations. By operation we assume encryption of a block using KB-256.

7. Finding an impossible differential for the KB-256-3 algorithm

In this section, we prove that an impossible differential exists for the KB-256 algorithm. We assume that an impossible differential for the encryption algorithm $E : V_n \times V_k \to V_n$ is the pair $D_1, D_2 \in V_n$ such that for any key $K \in V_k$ and for any $X, \underline{X} \in V_n$ such that $X \oplus \underline{X} = D_1$ the inequality holds:

$$E_K(X) \oplus E_K(\underline{X}) \neq D_2.$$

If an impossible differential exists, in some cases, it is possible to design an effective attack on block algorithms [4]. In general, this property enables a cryptanalyst to gain some information about the plaintext from the ciphertext.

We demonstrate that there exists an impossible differential D_1, D_2 for the KB-256 algorithm, where

$$D_1 = (0, 2^{31}, 0, 0, 0, 0, 2^{31}, 0), \quad D_2 = (2^{31}, 0, 0, 2^{31}, 0, 0, 0, 0).$$

By verification, a cryptanalyst can make sure that the text pair $X, \underline{X}$ such that $X \oplus \underline{X} = (0, 2^{31}, 0, 0, 0, 0, 2^{31}, 0)$, after 8 rounds, becomes the pair $X^8, \underline{X^8}$ such that $X^8 \oplus \underline{X^8} = (t_1, t_2, 0, t_3, t_4, 0, t_5, t_6)$ for some non-zero vectors $t_1, t_2, t_3, t_4, t_5, t_6 \in V_{32}$. We note that $t_1, t_2, t_3, t_4, t_5, t_6$ depend on each pair $X, \underline{X}$.

In Table 1, the differences between texts after each of 8 rounds are presented.

Table 1

Round no.	Difference
1	$(2^{31}, 0, 0, 0, 0, 2^{31}, 0, 0)$
2	$(0, 0, 0, 0, 2^{31}, 0, 0, 2^{31})$
3	$(0, 0, 0, 2^{31}, 0, 0, 2^{31}, 0)$
4	$(0, 0, 2^{31}, 0, 0, 2^{31}, 0, 0)$
5	$(0, 2^{31}, 0, 0, 2^{31}, 0, 0, 0)$
6	$(2^{31}, 0, 0, 0, 0, 2^{31}, 0, 0)$
7	$(0, \circ, 2^{31}, 0, \circ, 0, 0, \circ)$
8	$(t_1, t_2, 0, t_3, t_4, 0, t_5, t_6)$

By $\circ$ we denote non-zero differences. By verification, the cryptanalyst can make sure that the pair $Y^{16}, \underline{Y^{16}}$ such that $Y^{16} \oplus \underline{Y^{16}} = (2^{31}, 0, 0, 2^{31}, 0, 0, 0, 0)$ after 8 reverse rounds, becomes the pair $Y^8, \underline{Y^8}$ such that $Y^8 \oplus \underline{Y^8} = (t'_1, t'_2, t'_3, t'_4, 0, t'_5, t'_6, 0)$ for some non-zero vectors $t'_1, t'_2, t'_3, t'_4, t'_5, t'_6 \in V_{32}$. We note that $t'_1, t'_2, t'_3, t'_4, t'_5, t'_6$ depend on each pair $Y^{16}, \underline{Y^{16}}$.

In Table 2, the differences between texts after each of 8 rounds are presented in reverse order.

Table 2

Round no.	Difference
15	$(0, 2^{31}, 0, 0, 2^{31}, 0, 0, 0)$
14	$(0, 0, 2^{31}, 0, 0, 2^{31}, 0, 0)$
13	$(0, 0, 0, 2^{31}, 0, 0, 2^{31}, 0)$
12	$(0, 0, 0, 0, 2^{31}, 0, 0, 2^{31})$
11	$(2^{31}, 0, 0, 0, 0, 2^{31}, 0, 0)$
10	$(0, 2^{31}, 0, 0, 0, 0, 2^{31}, 0)$
9	$(\circ, 0, \circ, 0, 0, \circ, 0, 2^{31})$
8	$(t'_1, t'_2, t'_3, t'_4, 0, t'_5, t'_6, 0)$

An impossible differential exists if for text pairs $X, \underline{X}$ and $Y^{16}, \underline{Y^{16}}$ such that $X \oplus \underline{X} = (0, 2^{31}, 0, 0, 0, 0, 2^{31}, 0)$, $Y^{16} \oplus \underline{Y^{16}} = (2^{31}, 0, 0, 2^{31}, 0, 0, 0, 0)$, the sets $(t_1, t_2, 0, t_3, t_4, 0, t_5, t_6)$ and $(t'_1, t'_2, t'_3, t'_4, 0, t'_5, t'_6, 0)$ never coincide. As a result, since we have $t'_5 \neq 0$, we derive that an impossible differential exists.

8. Conclusion

In this paper, the KB-256 properties that may influence the overall cipher strength are provided. However, no key recovery method has been found more efficient than exhaustive key searching for the full-round algorithm.

REFERENCES

1. *Fomichev V. M., Koreneva A. M., Miftakhutdinova A. R., and Zadorozhny D. I.* Ocenki predelnoy proizvoditelnosti algoritmov blochnogo shifrovaniya [Evaluation of the maximum performance of block encryption algorithms]. Matematicheskie Voprosy Kriptografii, 2019, vol. 10, no. 2, pp. 181–191.
2. *Fomichev V. M. and Koreneva A. M.* Encryption performance and security of certain wide block ciphers. J. Comput. Virol. Hack. Tech., 2020, vol. 16, pp. 197–216.
3. GOST 34.12-2018. Informatsionnaya tekhnologiya. Kriptograficheskaya zashchita informatsii. Blochnye shifry. [GOST 34.12-2018. Information Technology. Cryptographic data security. Block ciphers]. `https://docs.cntd.ru/document/1200161708`, 2018.
4. *Biham E., Biryukov A., and Shamir A.* Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. J. Cryptology, 2005, vol. 18, pp. 291–311.

Секция 4

МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ, ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

УДК 519.682 DOI 10.17223/2226308X/15/20

О ПОЛИНОМИАЛЬНЫХ ГРАММАТИКАХ, ПОРОЖДАЮЩИХ БЕСКОНЕЧНОЕ МНОЖЕСТВО ЯЗЫКОВ

О. И. Егорушкин, И. В. Колбасина, К. В. Сафонов

Исследуются формальные грамматики — системы полиномиальных уравнений относительно некоммутативных переменных, которые решаются в виде формальных степенных рядов, выражающих нетерминальные символы алфавита через терминальные; первая компонента решения является формальным языком. Рассмотрено определение грамматики, имеющей бесконечно много решений (порождающей бесконечное множество языков). Такие грамматики могут возникать в ситуации, когда якобиан коммутативного образа грамматики тождественно равен нулю. Показано, что в этом случае описание множества решений грамматики сложнее, чем для аналогичных полиномиальных систем с вещественными или комплексными переменными, поскольку могут реализовываться все возможные ситуации: такая грамматика может иметь бесконечно много решений, любое конечное число решений либо не иметь решений вовсе.

Ключевые слова: *полиномиальные грамматики, некоммутативные переменные, формальный степенной ряд, коммутативный образ, якобиан.*

Как известно, теория формальных языков имеет фундаментальное значение как для лингвистики, так и программирования. Многочисленные приложения используют взаимосвязи языка (как множества возможных текстов) с грамматикой (сводом формальных правил, определяющих языковые конструкции и их равнозначимость). В этой связи нужны быстрые качественные алгоритмы формальных построений грамматики по языку и языка по грамматике, а также синтаксического анализа конструкций, что невозможно без серьёзного теоретического обоснования.

Рассмотрим систему полиномиальных уравнений

$$P_j(z, x) = 0, \quad P_j(0, 0) = 0, \quad j = 1, \ldots, n, \tag{1}$$

которая решается относительно символов $z = (z_1, \ldots, z_n)$ в виде формальных степенных рядов (ФСР), зависящих от символов $x = (x_1, \ldots, x_m)$.

Системы такого вида обобщают важные классы формальных грамматик [1, 2] и называются полиномиальными грамматиками [3, 4]. Одним из достоинств полиномиальных, в частности контекстно-свободных, грамматик является возможность задания широкого класса языков при сохранении относительной компактности представления [1 – 4].

Символы $x_1, \ldots, x_m$ называются терминальными, они образуют словарь языка, а символы $z_1, \ldots, z_n$ — нетерминальными, они необходимы для задания грамматических

правил. Над всеми символами определена некоммутативная операция конкатенации и коммутативная операция формальной суммы, а также коммутативная операция умножения на числа, что позволяет рассматривать ФСР с числовыми коэффициентами. Мономы от терминальных символов рассматриваются как предложения языка, а каждый ФСР (сумма всех «правильных» мономов) — компонент решения системы (1) понимается как порождённый грамматикой язык [1, 2].

Для полиномиальных грамматик актуальны вопросы существования, единственности и бесконечности решений, причём для понимания последней ситуации необходимо сделать уточнения. Дадим следующее

Определение 1. Будем говорить, что полиномиальная грамматика (1) имеет бесконечно много решений (порождает бесконечное множество языков), если множество решений системы (1) зависит хотя бы от одного произвольного ФСР от символов $x_1, \ldots, x_m$.

Так, система из двух одинаковых уравнений

$$x_1 z_1 - z_2 x_2 = 0$$

имеет тождественно равный нулю якобиан и бесконечно много решений, поскольку решения можно записать в виде

$$z_1 = s x_2, \;\; z_2 = x_1 s,$$

где s — произвольный ФСР от $x_1, \; x_2$.

Поскольку исследовать системы с некоммутативными символами трудно, в работах [3 – 5] предложено использовать коммутативный образ системы (1), который получается, если считать все переменные коммутативными. Обозначая коммутативный образ ФСР s через $\mathrm{ci}(s)$, рассмотрим коммутативный образ

$$\mathrm{ci}(P_j(z, x)) = 0, \quad j = 1, \ldots, n, \tag{2}$$

системы уравнений (1). Отметим, что из совместности некоммутативной системы (1) следует совместность коммутативной системы (2), а обратное утверждение неверно, что подчёркивает актуальность вопросов, связанных с совместностью системы уравнений (1). Используем для их решения такой инструмент, как якобиан.

Пусть

$$J(z, x) = \det\left(\frac{\partial(\mathrm{ci}(P_i(z, x)))}{\partial z_j}\right)$$

— якобиан системы уравнений (2) относительно переменных $z_1, \ldots, z_n$.

Для систем уравнений с вещественными либо комплексными переменными хорошо известна следующая

Теорема 1. Пусть выполнено равенство

$$J(z, x) \equiv 0,$$

тогда система уравнений (2) либо не имеет решения для каждого x в пространстве $\mathbb{C}_z^n$, либо все её решения в этом пространстве неизолированные.

Таким образом, суть теоремы состоит в том, что такие системы не могут иметь изолированных решений.

Для систем с некоммутативными переменными ситуация с описанием множества решений сложнее, а именно получена следующая

Теорема 2. Пусть для якобиана коммутативной системы (2) выполнено равенство

$$J(z, x) \equiv 0,$$

тогда некоммутативная система уравнений (1) либо не имеет решений (в виде ФСР $z = z(x)$), либо имеет любое конечное число решений, либо бесконечно много решений.

Суть теоремы 2 состоит в том, что равенство нулю якобиана не ограничивает свойств некоммутативной системы уравнений.

Учитывая, что система

$$f = 0, \ \ldots, \ f = 0,$$

из n одинаковых уравнений с n некоммутативными неизвестными $z_1, \ldots, z_n$, имеющая тождественно равный нулю якобиан, эквивалентна одному уравнению $f = 0$, сформулируем следствие: одно уравнение с некоммутативными неизвестными $P_1(z, x) = 0$ может не иметь решений, а также иметь конечное и бесконечное число решений.

В этом состоит фундаментальное отличие от одного уравнения над полем комплексных чисел, которое всегда имеет решения в виде аналитических функций.

ЛИТЕРАТУРА

1. *Глушков В. М., Цейтлин Г. Е., Ющенко Е. Л.* Алгебра. Языки. Программирование. Киев: Наукова думка, 1973.
2. *Salomaa A. and Soitolla M.* Automata-Theoretic Aspects of Formal Power Series. N.Y.: Springer Verlag, 1978.
3. *Егорушкин О. И., Колбасина И. В., Сафонов К. В.* О совместности систем символьных полиномиальных уравнений и их приложении // Прикладная дискретная математика. Приложение. 2016. № 9. С. 119–121.
4. *Egorushkin O. I., Kolbasina I. V., and Safonov K. V.* On solvability of systems of symbolic polynomial equations // Журн. СФУ. Сер. Матем. и физ. 2016. Т. 9. Вып. 2. С. 166–172.
5. *Семёнов А. Л.* Алгоритмические проблемы для степенных рядов и контекстно-свободных грамматик // Доклады АН СССР. 1973. № 212. С. 50–52.

УДК 004.056.5, 004.94 DOI 10.17223/2226308X/15/21

ПРИЕМЫ ДЕДУКТИВНОЙ ВЕРИФИКАЦИИ ПРОГРАММНОГО КОДА С ИСПОЛЬЗОВАНИЕМ AstraVer Toolset

А. О. Кокорин, С. Д. Тиевский, П. Н. Девянин

Описывается ряд практических приёмов дедуктивной верификации программного кода на языке Си на соответствие спецификациям его функций, заданных на языке ACSL. Для такой верификации используется основанный на платформе Frama-C набор инструментов AstraVer Toolset. Апробация этих приёмов осуществлена при верификации программного кода модуля управления доступом, реализованного в подсистеме безопасности PARSEC отечественной защищённой операционной системы специального назначения Astra Linux Special Edition. Благодаря использованию этих приёмов удалось упростить спецификации функций PARSEC, уменьшить трудоёмкость и ускорить процесс их дедуктивной верификации.

Ключевые слова: *дедуктивная верификация программного кода, ACSL, Frama-C, AstraVer Toolset, Astra Linux.*

Введение

При разработке отечественной защищённой сертифицированной по высшим классам защиты (уровням доверия) операционной системы специального назначения (ОССН) Astra Linux Special Edition [1, 2] применяется широкий спектр технологий верификации, статического и динамического анализа её программного кода. При этом особое внимание уделяется коду подсистемы безопасности PARSEC, которая в ОССН реализует механизм управления доступом и строится на основе мандатной сущностно-ролевой ДП-модели управления доступом и информационными потоками в ОС семейства Linux (МРОСЛ ДП-модели) [3]. В этой связи, во-первых, верифицируется сама модель, для чего её описание переводится с математического языка на машиночитаемый язык формального метода Event-B [4, 5]. После этого осуществляется дедуктивная верификация модели с применением инструмента Rodin и её верификация по методу проверки моделей инструментом ProB [6]. Во-вторых, наличие развитой модели, достаточно детально описывающей на математическом и формализованном языках механизм управления доступом ОССН, создаёт условия для верификации реализации модели непосредственно в программном коде подсистемы безопасности PARSEC. В результате для ОССН обеспечивается выполнение требования «Методики выявления уязвимостей и недекларированных возможностей в программном обеспечении» [7] в части разработки формальных (математических) описаний модулей, реализующих функции безопасности, и верификации их согласованности с моделью.

В настоящей работе рассмотрена только дедуктивная верификация, которая позволяет получить гарантию корректности исходного кода относительно его спецификаций, т. е. дедуктивная верификация заключается в разработке спецификаций функций (в первую очередь их контрактов, включающих, как минимум, предусловия и постусловия функций) программного кода PARSEC на языке ANSI/ISO C Specification Language (ACSL) [8] и доказательстве с применением набора инструментов AstraVer Toolset [5, 9] того, что в предположении о получении на вход каждой функции допустимых данных (предусловие выполнено) она возвращает соответствующий результат (постусловие выполнено). При этом также доказывается, что функции завершают работу корректно, т. е. не «зацикливаются» и не содержат ошибок вида неопределённого поведения.

Результатом является описание нескольких приёмов дедуктивной верификации программного кода, разработанных (либо развитых и адаптированных на основе существующих) и апробированных авторами.

1. Порядок верификации функций

Для дедуктивной верификации программного кода подсистемы безопасности PARSEC используется набор инструментов AstraVer Toolset. Он основан на открытой платформе верификации Frama-C [5, 10] и инструменте дедуктивной верификации Why3 [11]. В его состав входит доработанный плагин Jessie [12], задачей которого является преобразование кода на языке Си и спецификаций на языке ACSL в множество утверждений для доказательства (англ. *Proof Obligation — PO*). Эти утверждения получает инструмент Why3, который далее передаёт их на проверку выполнимости с помощью различных инструментов — автоматических, таких, как SMT-решатели CVC4, Alt-Ergo, Z3 и др., или интерактивных, таких, как инструмент Coq.

Спецификации на языке ACSL разрабатываются вручную экспертом и представляют собой набор формальных утверждений относительно поведения функций программного кода. Эти спецификации состоят из предусловий (условий нормального вы-

полнения функции) и постусловий, истинность которых гарантируется после выполнения функции. Они записываются перед каждой специфицируемой функцией в комментарии, начинающемся с символов /*@. Предусловия задаются с помощью ключевого слова requires, постусловия — с помощью ключевого слова ensures.

Пример 1. Пусть необходимо задать спецификации функции pdpl_get подсистемы безопасности PARSEC, предназначенной для увеличения счётчика использований метки безопасности в поле ucnt структуры типа PDPL_T, на которую указывает входной параметр l.

```
/*@ requires \valid(l);
requires l->ucnt.counter < INT_MAX;
assigns l->ucnt.counter;
ensures \result == l;
*/

const PDPL_T* pdpl_get(const PDPL_T *l) {
PDPL_T *ncl = (PDPL_T*)l;
atomic_inc(&(ncl->ucnt));
return l;
}
```

Для этого в предусловия функции, во-первых, включено требование корректности указателя l, которое определяется с помощью ключевого слова \valid. Во-вторых, в него добавлено ограничение на значение счётчика, которое должно быть меньше INT_MAX, что гарантирует отсутствие целочисленного переполнения (константы INT_MAX и INT_MIN, определённые в стандартной библиотеке языка Си в заголовочном файле limits.h, задают максимальное и минимальное значения переменных целого типа int соответственно). В-третьих, в предусловии указан разрешённый для изменения в функции участок памяти. Это осуществляется с помощью рамочного условия assigns, которое разрешает изменение только переменной l–>ucnt.counter (при этом в условии assigns не требуется указывать изменяющиеся локальные переменные). В завершении спецификаций в постусловие с помощью ключевого слова \result задано возвращаемое значение функции.

2. Использование глобальных переменных для отслеживания вызовов функций захвата/освобождения разделяемой памяти

В программах на языке Си необходимо обеспечить корректное управление разделяемой памятью [13] на всех путях исполнения. При верификации кода необходимо задавать спецификации функций, захватывающих и освобождающих такую память. Однако некоторые функции захвата/освобождения разделяемой памяти в ядре ОС семейства Linux определены с использованием сложных конструкций из макросов, что затрудняет написание спецификаций в полном соответствии с реализацией функции. Это также усложняется тем, что в программном коде применяются операторы условного перехода «if then else», а иногда безусловного перехода «goto», которые при верификации затрудняют отслеживание операций захвата/освобождения разделяемой памяти на всех путях выполнения кода. Кроме того, модель памяти, используемая набором инструментов AstraVer Toolset [9], не поддерживает верификацию параллельно исполняемого программного кода.

Для верификации функций захвата/освобождения разделяемой памяти предлагается следующий приём: формировать спецификации, только моделирующие поведение этих функций, для чего использовать «теневой» код на языке ACSL [8]. Переменные «теневого» кода задаются с помощью ключевого слова ghost. Такой код не обрабатывается компилятором Си, а предназначен для использования только инструментами верификации, которые работают с ним как с программным кодом. Это удобно во многих случаях для описания в спецификациях сложных свойств поведения функций, в том числе работающих с разделяемой памятью.

Пример 2. Пусть необходимо отслеживание доступов к критической секции для обращения на чтение к памяти в ядре ОС семейства Linux, осуществляемого с помощью функций rcu_read_lock (вход в критическую секцию для чтения памяти) и rcu_read_unlock (выход из критической секции). В этом случае необходимо обеспечить равенство числа вызовов rcu_read_lock и rcu_read_unlock.

```
// «Теневая» глобальная переменная - счетчик обращений к памяти
//@ ghost int ghost_rcu_lock;

// Предусловие для предотвращения целочисленного переполнения счетчика
/*@ requires ghost_rcu_lock < INT_MAX;
assigns ghost_rcu_lock;
// Увеличение счетчика ghost_rcu_lock на единицу
ensures ghost_rcu_lock == \old(ghost_rcu_lock) + 1;
*/
static inline void rcu_read_lock(void);

// Предусловие для предотвращения целочисленного переполнения счетчика
/*@ requires ghost_rcu_lock > INT_MIN;
assigns ghost_rcu_lock;
// Уменьшение счетчика ghost_rcu_lock на единицу
ensures ghost_rcu_lock == \old(ghost_rcu_lock) - 1;
*/
static inline void rcu_read_unlock(void);
```

Предположим, что в некоторой функции f вызываются функции захвата/освобождения критической секции и в спецификациях этой функции указано assigns ghost_rcu_lock, т. е. внешние по отношению к функции области памяти не должны изменяться. При этом в коде функции f при использовании критической секции допущена ошибка.

```
/*@ requires ghost_rcu_lock < INT_MAX;
assigns ghost_rcu_lock;
ensures ghost_rcu_lock == \old(ghost_rcu_lock);
*/
void f(int a) {
rcu_read_lock();
if (a == 0) {
goto exit;
}
```

```
rcu_read_unlock();
exit:
}
```

В функции f обеспечен вход в критическую секцию с помощью функции rcu_read_lock, но если выполняется условие «a == 0», то выход из критической секции не осуществляется. Для этой функции утверждение «ensures ghost_rcu_lock == \old(ghost_rcu_lock)» не пройдёт верификацию, так как оно требует неизменность счётчика ghost_rcu_lock, что позволяет контролировать равенство числа входов и выходов из критической секции. Недостаток предложенного приёма в примере 2 заключается в том, что в каждой функции, вызывающей rcu_read_lock, необходимо указывать условие ghost_rcu_lock $<$ INT_MAX. Без него верификация не будет завершаться, так как функция rcu_read_lock временно увеличивает значение ghost_rcu_lock, что без проверки указанного условия может привести к целочисленному переполнению.

Таким образом, приём, основанный на применении «теневого» кода на языке ACSL, позволяет моделировать поведение функций захвата/освобождения разделяемой памяти, что снижает риск появления ошибок в использующем их программном коде. При этом он не предполагает задания спецификаций таких функций, в точности им соответствующих. В итоге достигается компромисс между сложностью верификации и её результативностью.

3. Проверка корректности спецификаций вызываемых функций

При дедуктивной верификации как программного кода с применением спецификаций функций на языке ACSL, так и моделей на других формализованных языках, например метода Event-B [5, 6], существует опасность доказательства произвольных утверждений из некоторого ошибочного ложного условия или совокупности ложных условий. Такие условия могут не иметь явных признаков ошибки и при верификации могут рассматриваться экспертом как истинные.

В примере 2 счётчик числа входов и выходов из критической секции ghost_rcu_lock имеет целочисленный тип int (целое число в диапазоне между INT_MIN и INT_MAX), а не integer (произвольное целое число), так как «теневые» переменные (ghost) на языке ACSL могут иметь только типы, используемые в языке Си. Поэтому в спецификациях необходимо указывать в качестве предусловий функций rcu_read_lock и rcu_read_unlock ограничения для предотвращения переполнения этого счётчика, иначе в вызывающих функциях появится возможность доказывать произвольные утверждения. Поэтому необходимо контролировать корректность спецификаций тех функций, которые полностью не верифицируются, а только моделируются.

Для этого предлагается использовать следующий приём: добавлять заведомо ложное постусловие в спецификациях вызывающих функций, например «ensures 1 == 0». Если при этом верификация будет успешно завершаться, то в спецификациях допущена ошибка. Рассмотрим пример.

Пример 3. Пусть, аналогично примеру 2, необходимо отслеживание доступов к критической секции для обращения на чтение к памяти в ядре ОС семейства Linux. При этом в спецификациях функции входа в критическую секцию rcu_read_lock пропущено условие assigns ghost_rcu_lock.

```
// «Теневая» глобальная переменная - счетчик обращений к памяти
//@ ghost int ghost_rcu_lock;
```

```
/*@
// Увеличение счетчика ghost_rcu_lock на единицу
ensures ghost_rcu_lock == \old(ghost_rcu_lock) + 1;
*/
static inline void rcu_read_lock(void);

// Заведомо ложное условие
/*@ ensures 1 == 0; */
void f() {
rcu_read_lock();
}
```

Условие «ensures $1 == 0$» в спецификациях функции f проходит доказательство, хотя очевидно, что оно ложно. Это стало возможным из-за некорректной спецификации функции rcu_read_lock. На самом деле, постусловие функции rcu_read_lock гарантирует увеличение счётчика ghost_rcu_lock, в то время как отсутствует условие assigns. По умолчанию инструменты верификации считают, что если контракт без assigns написан для функции без тела (объявления функции), функция может изменять «всё». На практике функции, которые вызывают данную функцию, невозможно верифицировать верно, что делает написание контрактов для подобных функций обязательным. Если у функции есть тело, но нет assigns, то инструменты попытаются аппроксимировать множества изменяемых мест в памяти [8]. Инструменты верификации принимают это ложное условие и, исходя из него, доказывают произвольные утверждения. После проведения верификации при заведомо ложном постусловии «ensures $1 == 0$» его необходимо удалить, так как в противном случае уже в функциях, вызывающих f, будут верифицироваться произвольные утверждения.

4. Обход ограничений инструментов верификации при сравнении значений целочисленных типов и указателей

В языке программирования Си указатели представляют собой целые числа заданного для конкретной аппаратной платформы размера. Поэтому перетипизация целого числа в указатель (и наоборот) не требует дополнительных машинных инструкций, что позволяет сравнивать переменные таких типов друг с другом. Так, часто функции ядра ОС семейства Linux в качестве возвращаемого значения могут выдавать либо указатель на переменную, либо приведённый к типу указателя целочисленный код ошибки.

Однако модель памяти, используемая в наборе инструментов AstraVer Toolset, не допускает сравнений переменных целочисленного типа и произвольных указателей. Это затрудняет верификацию функций, в которых такие сравнения производятся.

Чтобы обойти это ограничение, предлагается следующий приём: использовать сравнение указателя с \null вместо сравнения с приведённой к типу указателя целочисленной переменной. Рассмотрим пример.

Пример 4. Проанализируем функции ядра ОС семейства Linux: ERR_PTR (преобразует целочисленный код ошибки в тип указателя) и IS_ERR (проверяет, возвращается ли в указателе код ошибки). Первая функция имеет следующий программный код:

```
static inline void * __must_check ERR_PTR(long error)
{
return (void *) error;
}
```

Пусть вызывающая функцию ERR_PTR функция use_err_ptr в случае успешного выделения памяти функцией get_int в переменной *pointer* возвращает указатель на неё, в противном случае — приведённый к типу указателя код ошибки -EFAULT:

```
#define EFAULT 13

/*@ assigns \nothing;
ensures \result == \null || \valid(\result);
*/
int* get_int();

int* use_err_ptr()
{
int* pointer = get_int();
if (!pointer) {
return ERR_PTR(-EFAULT);
} else {
return pointer;
}
}
```

В наборе инструментов AstraVer Toolset целочисленный тип и тип указателя несовместимы, что не позволяет записать в спецификациях постусловие функции use_err_ptr как «ensures \valid(\result) || (\result == -EFAULT). Чтобы решить эту задачу, предлагается моделировать ERR_PTR как функцию, возвращающую нулевой указатель, задав её следующие спецификации:

```
/*@ assigns \nothing;
ensures \result == \null;
*/
static inline void *ERR_PTR(long error);
```

Сравнение \result == \null поддерживается AstraVer Toolset. При этом теряется информация о коде ошибки, однако становится возможным описание в спецификациях поведения функций, использующих ERR_PTR, в том числе функции use_err_ptr:

```
/*@ assigns \nothing;
ensures \result == \null || \valid(\result);
*/
int* use_err_ptr();
```

Тогда спецификации функции IS_ERR, проверяющей, содержит ли возвращаемый указатель код ошибки, можно описать так:

```
/*@ assigns \nothing;
ensures \result == (ptr == \null);
*/
static inline bool IS_ERR(const void *ptr);
```

Таким образом, использование в спецификациях функций сравнения указателя с \null вместо сравнения с приведённой к типу указателя целочисленной переменной не полностью соответствует поведению функций, однако этот приём позволяет верифицировать большую часть программного кода таких функций.

5. Леммы, показывающие корректность предикатов

С помощью языка ACSL [8] для описания сложных свойств поведения функций возможно определение лемм, аксиом и предикатов. Леммы задаются ключевым словом lemma. Они определяются экспертом и могут помочь средствам верификации при доказательстве утверждений. Например, лемма mean_property (о том, что среднее арифметическое двух целых чисел находится между их значениями) может являться «подсказкой» при верификации программы бинарного поиска:

```
/*@ lemma mean_property:
\forall integer x,y; x <= y ==> x <= (x+y)/2 <= y;
*/
```

Аксиомы аналогичны леммам, но принимаются средствами верификации без попытки доказательства. Они задаются с помощью ключевого слова axiom.

Предикаты являются выражениями с результатом логического типа boolean. Они задаются с помощью ключевого слова predicate. Например, предикат is_positive определяет, является ли параметр x положительным целым числом:

```
//@ predicate is_positive(integer x) = x > 0;
```

При верификации нетривиальных программ часто необходимо описывать сложные предикаты. Одним из возможных подходов здесь является задание предиката через аксиомы. Рассмотрим пример.

Пример 5. Зададим предикат чётности целого положительного числа через аксиомы Even (для чётных чисел) и NotEven (для нечётных чисел):

```
/*@ axiomatic Even {
predicate is_even(integer x);
axiom Even: is_even(0) &&
(\forall integer x; is_even(x) ==> is_even(x + 2));
axiom NotEven: !is_even(1) &&
(\forall integer x; !is_even(x) ==> !is_even(x + 2));
}*/
```

Предикат is_even определяется в блоке axiomatic Even. Ключевое слово axiomatic не означает, что всё, что находится внутри блока, является утверждениями, принимаемыми без доказательства, а лишь группирует находящиеся внутри леммы, предикаты или логические функции.

Однако при определении предиката через аксиомы, корректность которых не доказывается, возрастает риск ошибки в спецификациях. В некоторых случаях определение предиката как функции на языке Си проще, чем через аксиомы (например, для предиката is_even можно использовать выражение на языке Си «x % 2 == 0»).

Предлагается следующий приём: задавать предикат так же, как реализована функция на языке Си, а затем с помощью введения дополнительных лемм доказывать, что этот предикат обладает требуемыми свойствами. Рассмотрим пример.

Пример 6. Для задания множеств небольшой мощности (до 32 элементов) может быть использована битовая маска, которая представлена переменной типа unsigned int языка Си. Тогда функция «множество a является подмножеством множества b» на языке Си может быть определена следующим образом:

```
int is_subset(unsigned int a, unsigned int b) {
return (a & b) == a;
}
```

Функция is_subset принимает две задающие множества маски бит в виде параметров типа unsigned int. В соответствии с предложенным приёмом, для функции is_subset спецификации могут быть заданы следующим образом:

```
/*@ // Предикат "является подмножеством"
predicate isSubset(unsigned int a, unsigned int b) =
(a & b) == a; // соответствует реализации функции
// на языке Си is_subset
*/
/*@ axiomatic isSubsetProperties {
// Проверяемые свойства предиката isSubset
// Проверка для конкретных значений множеств
lemma testIsSubset_1:
isSubset(5, 13); // 5 == 0b0101; 13 == 0b1101
// Проверка для конкретных значений множеств
lemma testIsSubset_2:
!isSubset(13, 1); // 13 == 0b1101; 1 == 0b0001
// Проверка: пустое множество подмножество любого множества
lemma emptyIsAlwaysSubset: // (1)
\forall unsigned int x;
isSubset(0, x);
// Проверка: подмножество пустого множества - пустое множество
lemma onlyEmptyIsSubsetOfEmpty: // (2)
\forall unsigned int x;
isSubset(x, 0) ==> (x == 0);
// Проверка: отношение быть подмножеством рефлексивно
lemma isSubsetReflexive: // (3)
\forall unsigned int x;
isSubset(x, x);
// Проверка: отношение быть подмножеством транзитивно
lemma isSubsetTransitive: // (4)
\forall unsigned int x, y, z;
```

```
isSubset(x, y) && isSubset(y, z) ==> isSubset(x, z);
}*/
/*@ // Постусловие:  задание предиката isSubset на языке ACSL
ensures \result == (isSubset(a, b) ? 1 : 0);
*/
int is_subset(unsigned int a, unsigned int b);
```

В блоке axiomatic заданы леммы. Для верификации этих лемм достаточно использовать автоматические средства доказательства. После того, как доказано, что предикат обладает требуемыми свойствами, несущественные леммы целесообразно удалить (в примере это леммы testIsSubset_1 и testIsSubset_2), так как большое число лемм затрудняет функционирование средств автоматического доказательства.

Таким образом, предложенный приём позволяет избежать излишнего использования аксиом для определения предикатов, однако при этом необходимо вводить леммы, с помощью которых проверяется корректность задания предикатов.

Заключение

Изложены некоторые практические приёмы дедуктивной верификации программного кода на языке Си на соответствие спецификациям его функций, заданных на языке ACSL. Эти приёмы успешно апробированы при верификации программного кода подсистемы безопасности PARSEC ОССН Astra Linux Special Edition. Они во многих случаях позволяют упростить и ускорить верификацию программного кода на языке Си, увеличить число функций, которые могут быть верифицированы, и повысить уверенность в корректности написанных спецификаций. В дальнейшем предполагается расширять состав таких приёмов, адаптированных для применения при верификации программного кода системного программного обеспечения, в особенности ОССН. Всё перечисленное должно быть полезно при дедуктивной верификации программного кода других сертифицированных средств защиты информации.

ЛИТЕРАТУРА

1. `https://astralinux.ru/products/astra-linux-special-edition/` — Операционная система специального назначения Astra Linux Special Edition. 2022.
2. *Буренин П. В., Девянин П. Н., Лебеденко Е. В. и др.* Безопасность операционной системы специального назначения Astra Linux Special Edition: учеб. пособие для вузов. 3-е изд., перераб. и доп. М.: Горячая линия — Телеком, 2019. 404 с.
3. *Девянин П. Н.* Модели безопасности компьютерных систем. Управление доступом и информационными потоками: учеб. пособие для вузов. 3-е изд., перераб. и доп. М.: Горячая линия — Телеком, 2020. 352 с.
4. *Abrial J.-R., Butler M., Hallerstede S., et al.* Rodin: An open toolset for modelling and reasoning in Event-B // Intern. J. Software Tools for Technology Transfer. 2010. V. 12. No. 6. P. 447–466.
5. *Девянин П. Н., Ефремов Д. В., Кулямин В. В. и др.* Моделирование и верификация политик безопасности управления доступом в операционных системах. М.: Горячая линия — Телеком, 2019. 214 с.
6. *Девянин П. Н., Леонова М. А.* Приёмы по доработке описания модели управления доступом ОССН Astra Linux Special Edition на формализованном языке метода Event-B для обеспечения её автоматизированной верификации с применением инструментов Rodin и ProB // Прикладная дискретная математика. 2021. № 52. С. 83–96.

7. https://fstec.ru/normotvorcheskaya/informatsionnye-i-analiticheskie-materialy/2171-informatsionnoe-soobshchenie-fstek-rossii-ot-10-fevralya-2021-g-n-240-24-647. Информационное сообщение ФСТЭК России от 10.02.2021 № 240/24/647.
8. https://frama-c.com/html/acsl.html — ANSI/ISO C Specification Language. 2022.
9. https://www.ispras.ru/technologies/astraver_toolset/ — Система верификации AstraVer Toolset. 2022.
10. *Kirchner F., Kosmatov N., Prevosto V., et al.* Frama-C: a software analysis perspective // Formal Aspects of Computing. 2015. No. 27(3). P. 573–609.
11. *Filliatre J.-C. and Paskevich A.* Why3 — where programs meet provers // LNCS. 2013. V. 7792. P. 125–128.
12. *Marche C. and Moy Y.* The Jessie Plugin for Deductive Verification in Frama-C. INRIA Saclay Ile-de-France and LRI, CNRS UMR, 2012.
13. *Колисниченко Д. Н.* Разработка Linux-приложений. СПб.: БХВ-Петербург, 2012. 432 с.

УДК 004.056.5, 004.94 DOI 10.17223/2226308X/15/22

СРАВНЕНИЕ СПОСОБОВ МОДЕЛИРОВАНИЯ МЕХАНИЗМОВ УПРАВЛЕНИЯ ДОСТУПОМ ОС И СУБД НА ФОРМАЛИЗОВАННОМ ЯЗЫКЕ МЕТОДА Event-B С ЦЕЛЬЮ ИХ ВЕРИФИКАЦИИ ИНСТРУМЕНТАМИ Rodin И ProB

М. А. Леонова, П. Н. Девянин

В результате перевода описания формальной модели управления доступом промышленной ОССН Astra Linux Special Edition (МРОСЛ ДП-модели) из математической в формализованную нотацию на языке метода Event-B, её автоматизированной верификации инструментами Rodin и ProB и проведённых в ГК Astra Linux научных исследований разработаны два способа моделирования взаимодействующих между собой систем, самостоятельно реализующих развитые механизмы управления доступом, такие, как ОС и СУБД. Эти способы основаны на использовании различных вариантов построения иерархии спецификаций МРОСЛ ДП-модели в формализованной нотации с применением техники пошагового уточнения. Сравнение способов показало как их достоинства, так и недостатки в части сложности написания спецификаций, необходимости повторения доказательств при верификации инструментом Rodin, возможности устранения эффекта «комбинаторного взрыва» при верификации инструментом ProB. По результатам сравнения сделан вывод, что рассмотренные способы могут быть полезны при разработке других формальных моделей управления доступом и их верификации с применением соответствующих средств.

Ключевые слова: *формальная модель управления доступом, МРОСЛ ДП-модель, Event-B, верификация, дедуктивная верификация, Rodin, метод проверки модели, ProB.*

Введение

В средствах защиты информации (СЗИ), таких, как ОС и СУБД, механизм управления доступом выполняет одну из основных функций по обеспечению их безопасности. Для достижения доверия к корректности этого механизма, создания условий для научного обоснования выполнения им заданных для СЗИ требований безопасности уже многие десятилетия разрабатываются формальные модели управления доступом [1, 2].

Мандатная сущностно-ролевая ДП-модель управления доступом и информационными потоками в ОС семейства Linux (МРОСЛ ДП-модель) [2] изначально создана для реализации в защищённой операционной системе специального назначения Astra Linux Special Edition (ОССН) [3, 4]. Данная модель, изложенная на математическом языке (в математической нотации), имеет иерархическое представление и состоит из восьми уровней (рис. 1) — четырёх уровней, моделирующих механизм управления доступом самой ОССН, и четырёх аналогичных уровней для штатной СУБД PostgreSQL. Уровни ОССН уточняются (наследуются и дополняются) строго последовательно (линейно) от первого к четвёртому, но иерархия уровней СУБД имеет более сложный вид: каждый последующий уровень СУБД уточняет предыдущий, а также соответствующий ему уровень ОССН, что делается для моделирования взаимодействия механизмов управления доступом данных систем.

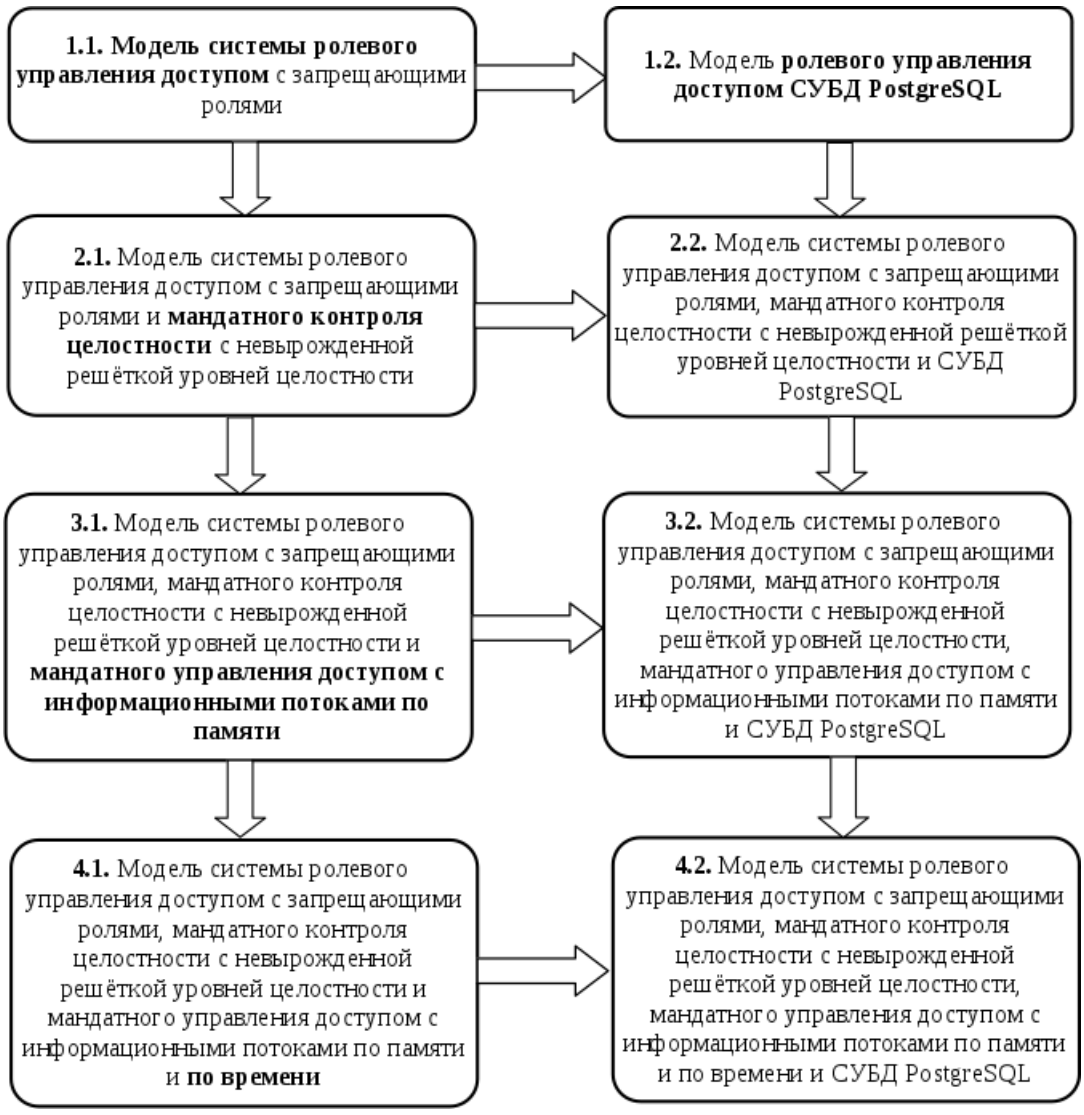


Рис. 1. Схема иерархического представления МРОСЛ ДП-модели

Хотя изначально модель формировалась в математической нотации, по мере увеличения объёма её описания (сейчас около 600 страниц) для повышения эффективности разработки, проверки корректности и отсутствия ошибок в самой модели для её описания стал применяться формализованный (машиночитаемый) язык метода Event-B [5],

т. е. представление модели в формализованной нотации [6, 7]. Для автоматизированной проверки корректности и верификации описания модели [8] в этой нотации используется инструмент дедуктивной верификации Rodin [9]. Кроме того, для повышения качества верификации МРОСЛ ДП-модели, расширения спектра применяемых для этого методов и инструментов, моделирования и в перспективе автоматизированного тестирования на соответствие этой модели её реализации непосредственно в программном коде и настройках конфигурации механизмов управления доступом ОССН и СУБД [10] осуществляется верификация модели с использованием инструмента проверки моделей ProB [11, 12].

Описание МРОСЛ ДП-модели в формализованной нотации на языке метода Event-B представляет собой последовательность связанных между собой спецификаций двух видов: контекстов (context) и машин (machine). Контексты определяют неизменяемую, базовую часть модели, в них задаются несущие множества (sets), константы (constants) и аксиомы (axioms). Машины являются динамической частью, в которых описываются переменные (variables), инварианты на них (invariants) и события (events), в свою очередь состоящие из параметров (parameters), охранных условий (guards) и действий (actions), изменяющих значения переменных машин. Для каждого уровня модели создаются машина и, при необходимости, контекст, после написания которых Rodin автоматически генерирует утверждения для доказательства (proof obligations). Модель считается дедуктивно верифицированной с применением инструмента Rodin, если для всех её уровней выполнены доказательства сгенерированных утверждений.

Последовательности спецификаций в инструменте Rodin задаются следующим образом: контексты могут расширяться (extends) несколькими контекстами, а для машин используется техника пошагового уточнения (refinement) [13], которая позволяет машине уточнять не более одной машины, другими словами, возможно составить только линейную последовательность машин. Ещё одной особенностью является невозможность изменения значений определяемых инвариантами функций на любых уровнях спецификаций, кроме того, на котором они задаются. Хотя, следует отметить, это может быть необходимо для корректного моделирования взаимодействия систем (например, возникновения информационных потоков между элементами ОС и СУБД).

При переводе описания модели из математической в формализованную нотацию накладываемые refinement ограничения не препятствовали моделированию иерархии уровней для ОССН, так как их последовательность линейна. Но при моделировании нелинейной иерархии динамической части уровней СУБД потребовался поиск способов обхода данных ограничений ввиду того, что машина каждого уровня (кроме первого) должна уточнять две другие машины. При этом необходимо на данных уровнях изменять значения функций, задаваемых на уровнях ОССН.

Предлагаются два способа моделирования нелинейной иерархии МРОСЛ ДП-модели. Первый способ заключается в непосредственном использовании техники refinement и построении линейной последовательности всех спецификаций восьми уровней, когда первый уровень СУБД уточняет четвёртый уровень ОССН. Второй способ также состоит в использовании техники refinement, но с построением двух ветвей уточнений, корнем которых является машина, описывающая первый уровень ОССН. Каждый из способов обладает достоинствами и недостатками как самого процесса моделирования, так и при осуществлении дедуктивной верификации инструментом Rodin и верификации по методу проверки моделей инструментом ProB получаемого представления

модели в формализованной нотации. Анализу и сравнению этих двух способов моделирования посвящена настоящая работа.

1. Первый способ

Первый способ описания нелинейной иерархии МРОСЛ ДП-модели в формализованной нотации заключается в прямом использовании техники refinement и построении линейной последовательности всех спецификаций восьми уровней, согласно которой первый уровень СУБД уточняет четвёртый уровень ОССН (рис. 2), но с переопределением на них необходимых для логического согласования модели в математической и формализованной нотациях функций и событий.

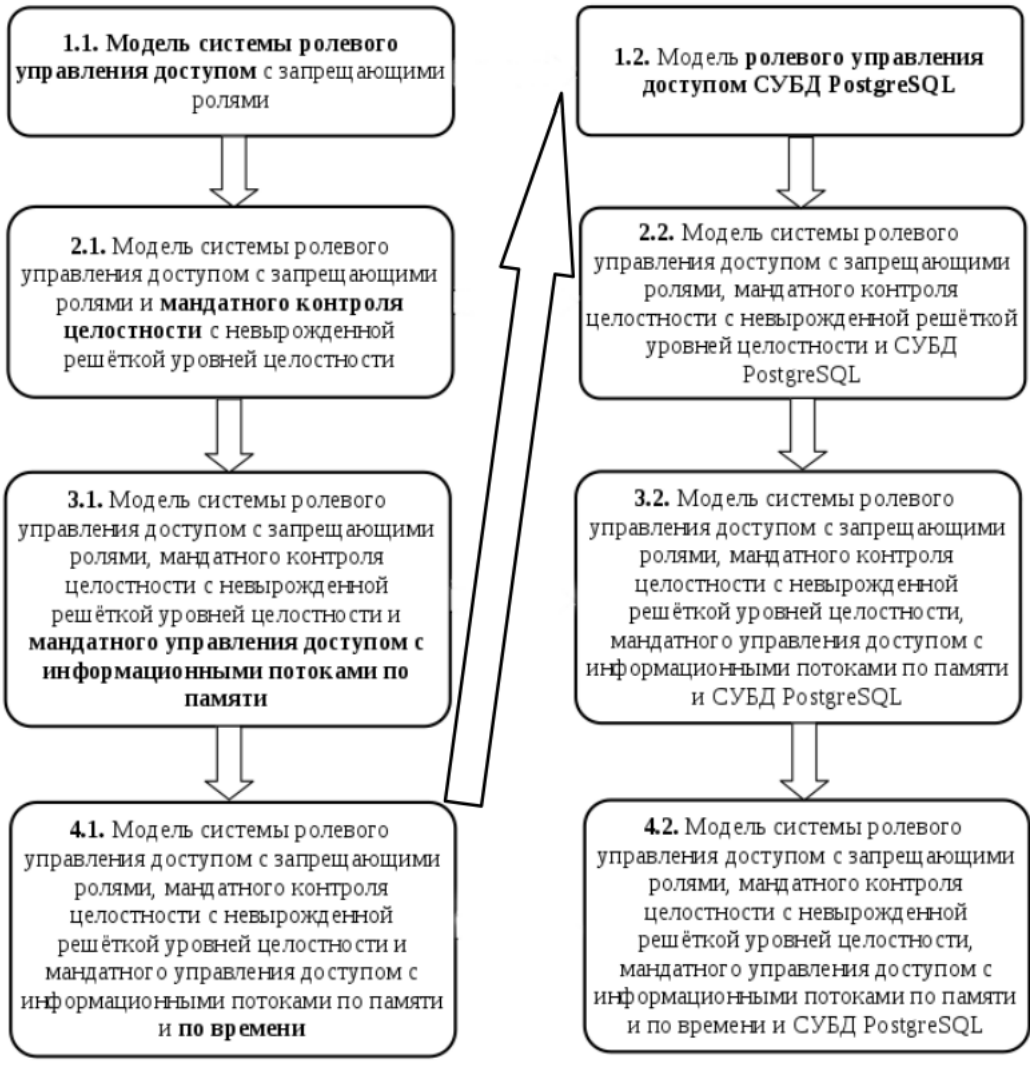


Рис. 2. Последовательность уточнения спецификаций уровней модели при использовании первого способа

Например, при описании машины второго уровня СУБД (шестого в последовательности) необходимо иметь возможность изменять значение функции информационных потоков по памяти от субъектов к сущностям ОССН SEMFlows при выполнении дефакто события создания информационного потока по памяти от субъекта к сущности ОССН при наличии субъекта-посредника find_entity_m. Однако данная функция определяется в машине аналогичного уровня ОССН. В результате использования

первого способа в машине второго уровня СУБД задаются функция dbSEFlows (переопределяющая функцию SEMFlows) и изменяющее её значение де-факто событие os_find_entity (переопределяющее событие find_entity_m). При этом для анализа информационных потоков по памяти от субъектов к сущностям СУБД определяются новые функция S_DBEFlows и событие db_find_entity (листинг 1).

```
os_find_entity
ANY
x, y, z, flow

WHERE
grd1: x ∈ Subjects
grd2: y ∈ Subjects
grd3: z ∈ Entities
grd4: flow ∈ ℙ1({1,2})
//‘‘1’’ - информационный поток по памяти, ‘‘2’’ - информационный поток по времени
grd5: 1 ∈ flow ⇒ z ∉ EHole
grd6: 1 ∈ flow ⇒ y ↦ 1 ∈ dbSSFlows(x) ∧ z ↦ 1 ∈ dbSEFlows(y)

THEN
act1: dbSEFlows(x) := dbSEFlows(x) ∪ ({z} × flow)
END

db_find_entity
ANY
x, y, z, flow

WHERE
grd1: x ∈ Subjects
grd2: y ∈ Subjects
grd3: z ∈ Entities
grd4: flow ∈ ℙ1({1,2})
grd5: 1 ∈ flow ⇒ z ∉ DBEHole
grd6: 1 ∈ flow ⇒ y ↦ 1 ∈ dbSSFlows(x) ∧ z ↦ 1 ∈ S_DBEFlows(y)

THEN
act1: S_DBEFlows(x) := S_DBEFlows(x) ∪ ({z} × flow)
END
```

Листинг 1. Общие для ОССН и СУБД события при моделировании первым способом

Описание машины на каждом уровне СУБД состоит из двух частей. Первая (и основная) часть описывает его отличия от предыдущего уровня, касающиеся моделируемого управления доступом (при этом необходимый уровень ОССН уже наследуется данным уровнем СУБД ввиду линейной последовательности всех спецификаций). Так, второй уровень СУБД (шестой в последовательности) задаёт мандатный контроль целостности, а значит, на третьем уровне СУБД (седьмом в последовательности) добавится описание мандатного управления доступом с информационными потоками по памяти с учётом наследования третьего уровня ОССН. Вторая часть ввиду особенностей техники refinement содержит переопределения некоторых функций и событий аналогичного уровня ОССН. Например, для третьего уровня СУБД (седьмого в последовательности) это затронет несколько функций и событий третьего уровня ОССН. Однако в целом по сравнению с общим объёмом кода описания каждого уровня СУБД объём такого переопределяемого («избыточного») кода невелик — приблизительно десятая часть. Соответственно столь же невелико число «избыточных» (повторяемых

для переопределенных функций и событий на уровне СУБД, уже выполненных на уровне ОССН) доказательств относительно их общего числа для каждого уровня, что является основным достоинством первого способа. Для примера на втором уровне СУБД переопределены 5 функций и 13 событий из общего числа описанных на нём, соответственно 20 функций и 63 событий, а также выполнены 156 «избыточных» доказательств из 1099, что не так много по сравнению с описанным далее вторым способом.

Однако у включения на каждом уровне СУБД всех уровней ОССН есть существенные недостатки. Во-первых, несоответствие сути постепенного для обеих систем добавления механизмов защиты, таких, как мандатный контроль целостности и мандатное управление доступом. Во-вторых, трудности непосредственного применения инструментов Rodin и ProB, одной из причин этого является большой объём описания уровней модели для СУБД в её формализованной нотации в результате наследования всего описания модели для ОССН. Если для Rodin это не так критично, то для ProB зачастую делает невозможным даже начальную инициализацию модели при верификации.

При использовании инструмента Rodin с каждым последующим уровнем увеличивается сложность верификации с точки зрения автоматизации ввиду того, что при доказательстве утверждения в качестве гипотез используются все аксиомы, инварианты и охранные условия спецификаций всех предыдущих уровней. Это затрудняет встроенным пруверам (provers) и солверам (solvers) поиск необходимых гипотез для вывода гипотезы-цели, являющейся смыслом каждого доказательства. Данную задачу приходится решать ручным удалением избыточных гипотез.

При верификации Rodin определённого уровня требуется выполнить также доказательства утверждений, сгенерированных только для этого уровня, считая, что уточнённые им уровни верифицированы (на них выполнены доказательства всех утверждений), даже если это на данный момент не так (например, определённый набор утверждений остался недоказанным). Другими словами, уровни верифицируются отдельно друг от друга, что при выявлении впоследствии ошибок в спецификациях предыдущего уровня может потребовать переработку спецификаций и повторное доказательство утверждений последующего уровня.

В свою очередь, применяемый для верификации по методу проверки моделей инструмент ProB позволяет для заданной модели с конечным (для МРОСЛ ДП-модели очень большим) числом состояний проверить, выполняются ли в рассматриваемых состояниях модели условия их корректности (инварианты). Однако основная трудность, которую приходится здесь преодолевать, связана с эффектом «комбинаторного взрыва» в пространстве состояний. Это характерно для моделирования систем (в том числе ОС и СУБД), состоящих из многих компонент, взаимодействующих друг с другом, и описываемых структурами данных, способными принимать большое число значений.

При работе ProB спецификации верифицируемого уровня включают спецификации всех предыдущих уровней. Соответственно при инициализации модели и переходе её в каждое следующее состояние необходимо проверить выполнение инвариантов для значений переменных в спецификациях текущего и всех предыдущих уровней. В результате для верификации, например, второго уровня СУБД (шестого в последовательности) необходимо также по сути избыточно верифицировать третий и четвёртый уровни ОССН, что вследствие проявления «комбинаторного взрыва» завершало работу ProB с ошибкой вида timeout, вызванной превышением допустимого интервала времени, установленного для выполнения переборных алгоритмов.

Таким образом, первый способ, основанный на построении линейной последовательности спецификаций уровней ОССН и СУБД МРОСЛ ДП-модели в формализованной нотации, обладает рядом отмеченных недостатков, которые сделали практически невозможной верификацию с использованием инструмента ProB уровней СУБД. Для устранения этих недостатков разработан альтернативный способ представления модели, предполагающий построение двух ветвей уточнений.

2. Второй способ

Второй способ описания нелинейной иерархии МРОСЛ ДП-модели в формализованной нотации заключается также в использовании техники refinement, но при этом строятся две ветви (последовательности) уточнений, корнем которых является машина, описывающая первый уровень ОССН (рис. 3):

— первая ветвь — линейно уточняющие друг друга спецификации четырёх уровней ОССН;
— вторая ветвь — линейно уточняющие друг друга машины первого уровня ОССН и четырёх уровней СУБД, но с ручным полным дублированием кода машин уровней ОССН в соответствующие им машины уровней СУБД и повторным доказательством сгенерированных утверждений для ОССН, начиная со второго уровня.

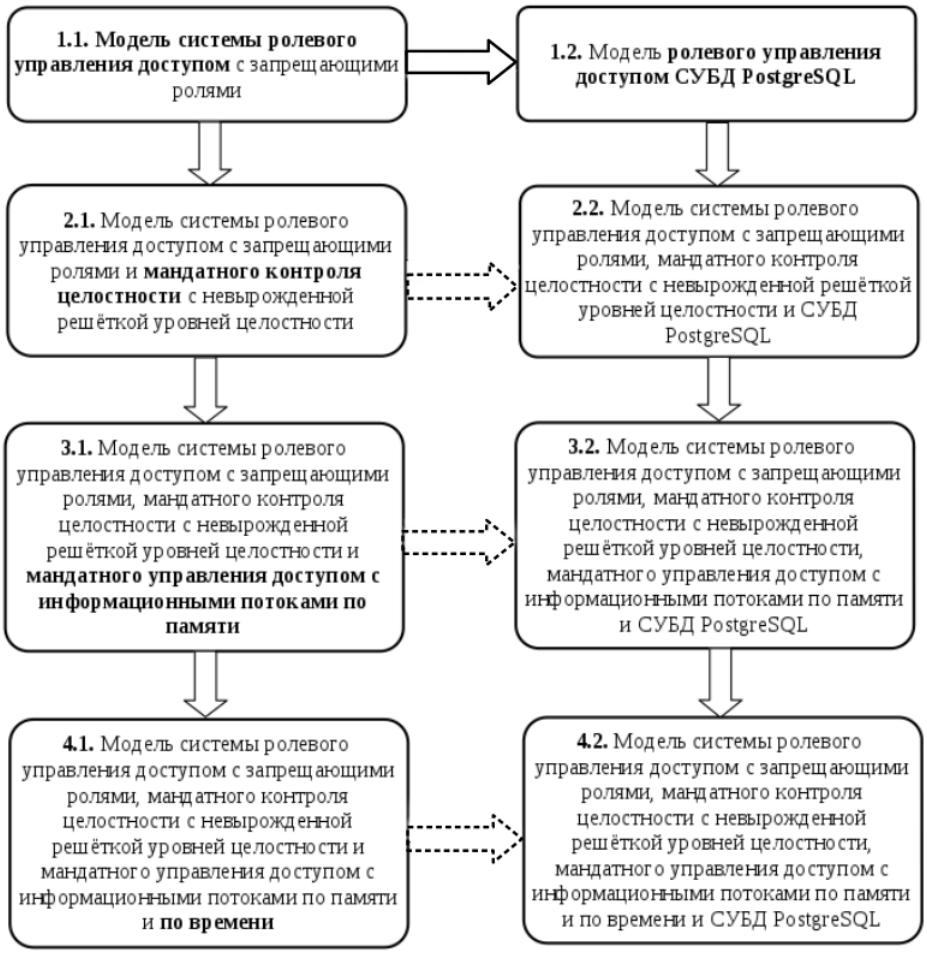


Рис. 3. Ветви (последовательности) уточнения спецификаций уровней модели при использовании второго способа

Построение второй ветви по сути является «ручным» уточнением машин первой ветви для ОССН соответствующими машинами второй ветви для СУБД. В отличие от машин, вследствие поддержки Rodin расширения контекстов несколькими контекстами иерархия спецификаций контекстов уровней полностью идентична исходной нелинейной иерархии МРОСЛ ДП-модели.

Для примера в листинге 2 приведены два события из машины второго уровня СУБД 2-M-DBMS-MIC: access_read_entity (получение субъектом доступа на чтение к сущности ОССН) и access_read_db_entity (получение субъектом доступа на чтение к сущности СУБД). Событие access_read_entity уточняет охранными условиями grd5–grd7 одноимённое событие машины первого уровня ОССН, при этом данные предикаты дублируются из машины второго уровня ОССН, а событие access_read_db_entity уточняет охранными условиями grd6, grd7 одноимённое событие машины первого уровня СУБД, при этом данные предикаты являются новыми в рамках описания модели в формализованной нотации.

```
    access_read_entity
    ANY
    subject, entity

    WHERE
    grd1: subject ∈ Subjects
    grd2: entity ∈ Entities
    grd3: entity ↦ Read ∈ CheckRightE(subject)
    grd4: entity ∈ ExecuteContainer(subject)
    grd5: entity ↦ Read ∈ CheckRightEInt(subject)
    grd6: entity ∈ ExecuteContainerInt(subject)
    grd7: WithoutCoop = TRUE ⇒ SubjectInt(subject) ≠ HighI
    THEN
    act1: SubjectAccesses(subject) := SubjectAccesses(subject) ∪ {entity ↦ ReadA}
    END

    access_read_db_entity
    ANY
    subject, element, privilege

    WHERE
    grd1: subject ∈ Subjects
    grd2: elements ∈ DBElements
    grd3: privilege ∈ DBPrivileges
    grd4: Read ∈ dbRights(privilege)
    grd5: PostgresAdmRole ↦ ReadA ∈ DBSubjectAdmAccesses(subject) ∨
(dbEntity(element) ↦ Read ∈ dbCheckRightE(subject) ∧
dbEntity(element) ∈ dbExecuteContainer(subject))
    grd6: PostgresAdmRole ↦ ReadA ∈ DBSubjectAdmAccesses(subject) ∨
(dbEntity(element) ↦ Read ∈ dbCheckRightEInt(subject) ∧
dbEntity(element) ∈ dbExecuteContainerInt(subject))
    grd7: WithoutCoop = TRUE
    THEN
    act1: DBSubjectAccesses(subject) := DBSubjectAccesses(subject) ∪
{dbEntity(element) ↦ ReadA}
    END
```

Листинг 2. Примеры событий второго уровня СУБД, дублирующих события ОССН или являющихся новыми, при моделировании вторым способом

При использовании данного способа моделирования в формализованной нотации по аналогии с математической реализуется идея постепенного для обеих систем включения механизмов защиты, таких, как мандатный контроль целостности и мандатное управление доступом. Также упрощаются автоматизированные дедуктивная верификация модели с применением инструмента Rodin (устраняется избыточность используемых в доказательстве гипотез) и верификация по методу проверки моделей с применением инструмента ProB (устраняется избыточность необходимых для верификации машин), так как каждая машина уровня СУБД уточняет по технике refinement машину предыдущего уровня СУБД и «вручную» соответствующую ей машину уровня ОССН.

Недостатком второго способа моделирования относительно первого является его трудоёмкость. Вместо описания восьми уровней МРОСЛ ДП-модели на формализованном языке Event-B и их автоматизированной дедуктивной верификации необходимо по сути описать и верифицировать одиннадцать уровней — четыре уровня ОССН и четыре уровня СУБД, внутри себя содержащих полное дублирование трёх уровней ОССН (второго, третьего и четвёртого). Сравнительная статистика объёма избыточного кода при описании модели и доли её избыточной верификации в формализованной нотации каждым из двух способов представлены в таблице. Под «ручным» кодом здесь понимается код, который необходимо написать на каждом уровне модели без учёта кода, наследуемого в результате использования техники refinement с предыдущих уровней модели. Кроме того, в таблице отдельно приведена статистика для де-юре событий (используемых для моделирования функций механизмов защиты ОССН и СУБД) и де-факто событий (используемых для анализа информационных потоков или условий получения управления одним субъектом над другим).

Восемь уровней модели для управления доступом в ОССН и СУБД	Количество де-юре событий, шт.	Количество де-факто событий, шт.	Объём «ручного» кода, тыс. строк	Объём избыточного кода, тыс. строк	Доля избыточной верификации, %
Первый способ	76	80	12,5	1,2	0,1
Второй способ	74	64	15,4	4,2	26,2

Сравнивая объёмы «ручного» и избыточного кода, полученные каждым из способов, можно сделать вывод, что на описание МРОСЛ ДП-модели в формализованной нотации с применением второго способа уходит больше времени, при этом доля избыточного кода по отношению к общему объёму кода, написанного «вручную», составляет около 27 % против 10 для первого способа. Длительность верификации модели с использованием второго способа также возрастает, так как доля избыточной верификации становится равной 26 % против 0,1 для первого способа. Однако использование второго способа значительно упрощает верификацию с применением инструмента ProB, которая при использовании первого способа для уровней СУБД практически невозможна.

Заключение

Проанализированы два способа перевода описания модели управления доступом промышленной ОССН Astra Linux Special Edition (МРОСЛ ДП-модели) из математической в формализованную нотацию на языке метода Event-B и её автоматизированной верификации инструментами Rodin и ProB. Оба способа основаны на применении техники пошагового уточнения refinement для представления иерархии уровней

МРОСЛ ДП-модели, соответствующих взаимодействующим между собой системам, самостоятельно реализующим развитые механизмы управления доступом, такие, как сама ОССН и штатная для неё СУБД PostgreSQL. Несмотря на отмеченные недостатки, каждый из предложенных способов имеет существенные достоинства и вместе они могут быть полезны при разработке других формальных моделей управления доступом и их верификации с применением соответствующих инструментов.

ЛИТЕРАТУРА

1. ГОСТ Р 59453.1-2021 «Защита информации. Формальная модель управления доступом. Ч. 1. Общие положения». М.: Стандартинформ, 2021. 16 с.
2. *Девянин П. Н.* Модели безопасности компьютерных систем. Управление доступом и информационными потоками: учеб. пособие для вузов. 3-е изд., перераб. и доп. М.: Горячая линия — Телеком, 2020. 352 с.
3. `https://astralinux.ru/products/astra-linux-special-edition/` — Операционная система специального назначения Astra Linux Special Edition. 2022.
4. *Буренин П. В., Девянин П. Н., Лебеденко Е. В. и др.* Безопасность операционной системы специального назначения Astra Linux Special Edition: учеб. пособие для вузов. 3-е изд., перераб. и доп. М.: Горячая линия — Телеком, 2019. 404 с.
5. *Abrial J.-R.* Modeling in Event-B: System and Software Engineering. Cambridge University Press, 2010.
6. *Девянин П. Н., Ефремов Д. В., Кулямин В. В. и др.* Моделирование и верификация политик безопасности управления доступом в операционных системах. М.: Горячая линия — Телеком, 2019. 214 с.
7. *Девянин П. Н., Леонова М. А.* Применение подтипов и тотальных функций формального метода Event-B для описания и верификации МРОСЛ ДП-модели // Программная инженерия. 2020. Т. 11. № 4. С. 230–241.
8. ГОСТ Р 59453.2-2021 «Защита информации. Формальная модель управления доступом. Ч. 2. Рекомендации по верификация формальной модели управления доступом». М.: Стандартинформ, 2021. 12 с.
9. *Abrial J.-R., Butler M., Hallerstede S., et al.* Rodin: An open toolset for modelling and reasoning in Event-B // Intern. J. Software Tools for Technology Transfer. 2010. V. 12. No. 6. P. 447–466.
10. *Девянин П. Н., Тележников В. Ю., Хорошилов А. В.* Формирование методологии разработки безопасного системного программного обеспечения на примере операционных систем // Труды ИСП РАН. 2021. Т. 33. № 5. С. 25–40.
11. *Leuschel M. and Butler M.* ProB: an automated analysis toolset for the B method // Int. J. Softw. Tools Technol. Transf. 2008. No. 10(2). P. 185–203.
12. *Девянин П. Н., Леонова М. А.* Приёмы по доработке описания модели управления доступом ОССН Astra Linux Special Edition на формализованном языке метода Event-B для обеспечения её автоматизированной верификации с применением инструментов Rodin и ProB // Прикладная дискретная математика. 2021. № 52. С. 83–96.
13. *Abrial J.-R. and Hallerstede S.* Refinement, decomposition, and instantiation of discrete models: Application to Event-B // Fundamenta Informaticae. 2007. V. 77. Iss. 1–2. P. 1–28.

УДК 519.7 DOI 10.17223/2226308X/15/23

О СКРЫТЫХ УПРОЩАЮЩИХ СТРУКТУРАХ В КОМБИНАТОРНЫХ ЗАДАЧАХ И ИХ ВЕРОЯТНОСТНЫХ ОБОБЩЕНИЯХ[1]

А. А. Семёнов

Дан обзор некоторых недавних результатов, связанных со структурами, за которыми в англоязычной литературе закрепился термин “Backdoor”. Наиболее близким аналогом в русском, по-видимому, является термин «лазейка». Лазейка — это такое множество переменных в произвольной задаче удовлетворения ограничений, знание которого существенно упрощает рассматриваемую задачу либо даёт верхнюю оценку трудности её решения, которая лучше трудности тривиальной переборной стратегии. Лазейки в последние годы являются популярным объектом исследований как в прикладных областях, так и в теоретических (главным образом, в параметризованной сложности). Обсуждается применение лазеек для повышения эффективности решения конкретных комбинаторных задач из семейств SAT (проблема булевой выполнимости) и 0-1-ILP (0-1-целочисленное линейное программирование).

Ключевые слова: *лазейки в комбинаторных задачах, проблема булевой выполнимости (SAT), 0-1-целочисленное линейное программирование.*

Понятие «лазейка» в применении к произвольной задаче удовлетворения ограничений (Constraint Satisfaction Problem, CSP) впервые дано в работе [1]. Нас в дальнейшем будут, в первую очередь, интересовать «сильные лазейки» (Strong Backdoor Sets), также введённые в [1].

Рассмотрим проблему булевой выполнимости (SAT) как один из простейших и наиболее наглядных примеров CSP. В произвольном экземпляре SAT фигурирует булева формула C (обычно в конъюнктивной нормальной форме, КНФ). Множество встречающихся в C переменных со значениями в $\{0,1\}$ обозначим через X. Элементарными ограничениями являются дизъюнкты, входящие в C. Для произвольного $B \subseteq X$ обозначим через $\{0,1\}^{|B|}$ множество всех наборов значений переменных из B. Подстановка значения $\alpha \in \{0,1\}$ произвольной переменной $x \in X$ в формулу C определяется стандартным образом [2]. Обозначим через $C\,[\beta/B]$ КНФ, которая получается в результате подстановки в C набора β значений переменных из B. Пусть O — некоторый полиномиальный алгоритм.

Определение 1 [1]. Сильной лазейкой (Strong Backdoor Set, SBS) для КНФ C относительно алгоритма O называется такое множество $B \subseteq X$, что для любого $\beta \in \{0,1\}^{|B|}$ задача выполнимости КНФ $C\,[\beta/B]$ разрешима алгоритмом O.

Таким образом, если B — сильная лазейка для КНФ C, то задача выполнимости C решается за время $\mathrm{poly}(|C|)\,2^{|B|}$, где $\mathrm{poly}(\cdot)$ — некоторый полином; $|C|$ — длина двоичного описания C. Можно привести массу примеров, когда C содержит сильную лазейку, число переменных в которой может не превосходить нескольких процентов (или даже долей процента) от общего числа переменных в X. Таковы, например, КНФ, кодирующие задачи криптоанализа. В этих случаях множество переменных, кодирующих вход рассматриваемой криптографической функции, образует SBS относитель-

[1]Исследование выполнено в рамках госзадания Минобрнауки России по проекту «Теоретические основы, методы и высокопроизводительные алгоритмы непрерывной и дискретной оптимизации для поддержки междисциплинарных научных исследований», № гос. регистрации 121041300065-9.

но простейшего правила распространения булевых ограничений — правила единичного дизъюнкта (Unit Propagation rule, UP). Например, КНФ, кодирующая задачу поиска прообраза криптографической функции SHA-256, то есть задачу обращения функции $f_{\text{SHA-256}} : \{0,1\}^{512} \to \{0,1\}^{256}$, имеет SBS относительно правила UP, которое состоит из 512 переменных, при том что общее число переменных в данной КНФ равно 49 098. Легко понять, что сильная лазейка размерности существенно меньше 512, если бы она существовала, давала бы нетривиальную атаку на данную функцию. Очевидный интерес представляют сильные лазейки наименьшей мощности, поскольку они дают лучшие верхние границы в рассматриваемом классе таких границ.

Определение 2. Для произвольной КНФ C и полиномиального алгоритма O сильную лазейку относительно O наименьшей мощности будем называть минимальной.

Теорема 1 [1]. Если КНФ C содержит сильную лазейку размера $|B| \leqslant |X|/2$, то существует алгоритм, который решает SAT для C за время

$$\text{poly}(|C|) \left(\frac{2\,|X|}{\sqrt{|B|}} \right)^{|B|}. \tag{1}$$

Следствие 1 [1]. Для формул с сильной лазейкой размера не более $n/4{,}404$ проблема SAT может быть решена детерминированным образом за время $\mathcal{O}\left(c^n\right)$, где $c < 2$.

Алгоритм из [1] (далее — «алгоритм A»), который подразумевается в формулировке теоремы 1, очень прост. Сначала проверяются на свойство быть сильной лазейкой все множества, состоящие из одной переменной — их $n = |X|$, затем — все множества из двух переменных — их $\binom{n}{2}$, и так далее. Очевидно, что первая сильная лазейка, найденная алгоритмом A, минимальная.

С другой стороны, в худшем случае алгоритм A пройдёт все подмножества множества X и переберёт все 2^n наборов значений переменных из X. Таким образом, в худшем случае сложность данного алгоритма есть следующая величина с точностью до некоторого полиномиального от $|C|$ сомножителя:

$$\sum_{i=1}^{n} \binom{n}{i} 2^i = \sum_{i=0}^{n} \binom{n}{i} 2^i - 1 = 3^n - 1. \tag{2}$$

Следствие 1 означает, что если существует сильная лазейка мощности $\leqslant |X|/2$, то даже алгоритм A может быть асимптотически (на соответствующем семействе формул) лучше полного перебора. Оценку (1) можно переписать в более наглядной форме. Это делается за счёт анализа биномиальных коэффициентов и использования формулы Стирлинга. Заметим, что в случае существования лазейки мощности K имеет место $T_A \leqslant p(|C|) \sum\limits_{i=1}^{K} \binom{n}{i} 2^i$, где $p(\cdot)$ — некоторый полином, что для $K \leqslant n/q$ и $q \geqslant 2$ с учётом свойств биномиальных коэффициентов даёт следующую оценку:

$$T_A \leqslant p(|C|)\, 2^K \binom{n}{K}. \tag{3}$$

Преобразуя (3) с использованием формулы Стирлинга, имеем

Следствие 2. Если КНФ C содержит сильную лазейку мощности $K = \lfloor n/q \rfloor$, $q \geqslant 2$, то существует алгоритм, решающий SAT для C за время T_A, где

$$T_A \leqslant p(|C|)\, 2^{n(1/q + \log q - (q-1)\log(q-1)/q)}. \tag{4}$$

К сожалению, использование (4) не даёт существенного улучшения оценки из [1]. Конкретно, (4) даёт следующий результат (аналог следствия 4.2 из [1]), который получен за счёт решения пакетом WolframAlpha уравнения $\frac{1}{q} + \log q - \frac{q-1}{q}\log(q-1) = 1$:

Следствие 3. Для формул с сильной лазейкой размера не более $n/4{,}40349$ проблема SAT может быть решена детерминированным образом за время $\mathcal{O}(c^n)$, где $c < 2$.

Соотношение (2) означает, что алгоритм A крайне неэффективен в применении к поиску минимальной сильной лазейки и вряд ли может использоваться для решения практических задач. При детальном рассмотрении становится понятно, что основной фактор неэффективности A — это необходимость проверять свойство произвольного $B \subseteq X$ быть сильной лазейкой: для этого требуется проверить разрешимость $C[\beta/B]$ алгоритмом O для каждого $\beta \in \{0,1\}^{|B|}$ (делая, таким образом, каждый раз $2^{|B|}$ проверок).

В работе [3] предложено вероятностное обобщение понятия сильной лазейки. Соответствующее определение выглядит следующим образом.

Определение 3. Пусть C — произвольная КНФ, X — множество переменных в C и O — некоторый полиномиальный алгоритм. Произвольное множество $B \subseteq X$ назовём ρ-лазейкой, $\rho \in [0,1]$, относительно алгоритма O, если среди всех задач вида $C[\beta/B]$, $\beta \in \{0,1\}^{|B|}$, доля задач, решаемых алгоритмом O, составляет не менее ρ.

Очевидно, что ρ-лазейка с $\rho = 1$ — это сильная лазейка. Если $\rho < 1$, то с практической точки зрения важно, чтобы ρ было как можно ближе к 1. Действительно, если B — такая ρ-лазейка, то мы можем алгоритмом O решить $\rho \cdot 2^{|B|}$ задач вида $C[\beta/B]$, а к оставшимся $(1-\rho)\,2^{|B|}$ задачам применить полный алгоритм решения SAT. Если считать ρ, проверяя, решается ли алгоритмом O каждая задача вида $C[\beta/B]$, то соответствующая процедура, как и в случае сильных лазеек, неэффективна. Однако, как показано в [3], можно эффективно оценить ρ, используя простой вероятностный тест.

Для произвольных C над X, $B \subseteq X$ и полиномиального алгоритма O зададим на $\{0,1\}^{|B|}$ равномерное распределение и определим случайную величину $\xi_B : \{0,1\}^{|B|} \to \{0,1\}$, которая принимает на $\beta \in \{0,1\}^{|B|}$ значение 1, если $C[\beta/B]$ решается алгоритмом O и значение 0 в противном случае. Очевидно, что если B — ρ-лазейка, то вероятность успеха в наблюдении величины ξ_B составляет не менее ρ и, таким образом, $E[\xi_B] \geqslant \rho$. Последнее означает, что можно оценивать параметр ρ произвольного $B \subseteq X$, оценивая величину $E[\xi_B]$. Поскольку ξ_B — случайная величина Бернулли, то для оценки $E[\xi_B]$ можно использовать следующее неравенство (являющееся вариантом известной границы Чернова):

$$\mathsf{P}\left[\left|\frac{1}{N}\sum_{j=1}^{N}\xi_j - E[\xi_B]\right| < \varepsilon\right] \geqslant 1 - 2e^{-N\varepsilon^2/4}. \tag{5}$$

Таким образом, мы можем оценить $E[\xi_B]$ с точностью ε и уровнем значимости δ за счёт использования N независимых наблюдений величины ξ_B, обозначенных в (5) через $\xi_1, \ldots, \xi_N$. Соответствующей оценкой является среднее арифметическое наблюдаемых значений. Алгоритмы оценивания такого типа относятся к методу Монте-Карло [4].

Для произвольного $B \subseteq X$ определим следующий тест Монте-Карло: зафиксируем $\varepsilon, \delta \in (0,1)$ и $N = \lceil 16\ln(2/\delta)/\varepsilon^2 \rceil$, пронаблюдаем ξ_B независимо N раз, пусть $\xi_1, \ldots, \xi_N$ — результаты этих наблюдений, вычислим $\widetilde{\rho} = \frac{1}{N}\sum_{j=1}^{N}\xi_j$. Скажем, что B про-

ходит тест, если $\widetilde{\rho} \in [1-\varepsilon/2, 1]$. Используя (5), нетрудно показать справедливость следующего факта.

Следствие 4. Если B проходит описанный тест Монте-Карло, то заключение, что $E[\xi_B] \in [1-\varepsilon, 1]$, верно с вероятностью $\geqslant 1-\delta$.

Теперь мы можем определить вероятностный аналог сильной лазейки.

Определение 4 [3]. Назовём B сильной (ε, δ)-лазейкой, если B проходит описанный тест Монте-Карло.

Например: если B проходит тест при $\varepsilon = \delta = 0{,}01$, то с вероятностью 99 % можно заключить, что B — это ρ-лазейка с $\rho \geqslant 0{,}98$ (то есть ρ очень близко к 1). Нетрудно заметить, что сложность описанного алгоритма оценивания $E[\xi_B]$ при фиксированных $\varepsilon, \delta \in (0,1)$ ограничена сверху величиной $p(|C|)16\ln(2/\delta)/\varepsilon^2$ и, следовательно, данный алгоритм эффективный.

Таким образом, в противовес алгоритму A, перебирающему множества B последовательно возрастающей мощности, мы можем искать сильные (ε, δ)-лазейки, минимизируя некоторую целевую функцию при помощи метаэвристических алгоритмов на множестве всех подмножеств X. Более конкретно, с произвольным $B \subseteq X$ в [3] связывается значение следующей функции:

$$\Phi_{C,A,N}(B) = \widetilde{\rho}_B \cdot 2^{|B|} + G_{C,A,N}(B). \tag{6}$$

В (6) часть $G_{C,A,N}(B)$ — это штрафная функция: её значение должно резко возрастать при уменьшении величины $\widetilde{\rho}_B$. В [3] использована штрафная функция вида

$$G_{C,A,N} = \begin{cases} (1-\widetilde{\rho}_B)2^{\omega|X|}, & \text{если } \widetilde{\rho}_B > 0; \\ \infty, & \text{если } \widetilde{\rho}_B = 0, \end{cases}$$

где $\omega \in (0,1)$ — параметр, подбираемый эмпирически для каждой конкретной КНФ C.

Как сказано ранее, для решения задач вида $C[\beta/B]$, не разрешимых полиномиальным алгоритмом O, можно использовать полный SAT-решатель. Для решения SAT в отношении трудной C можно использовать также несколько найденных сильных (ε, δ)-лазеек. Пусть $B_1, \ldots, B_s$ — такие лазейки. Для каждого B_r, $r \in \{1, \ldots, s\}$, построим множество Δ_r, состоящее из всех таких $\beta \in \{0,1\}^{|B_r|}$, что $C[\beta/B_r]$ не разрешима алгоритмом O. Заметим, что мощность каждого Δ_r, $r \in \{1, \ldots, s\}$, может составлять доли процента от $2^{|B_r|}$. Рассмотрим декартово произведение $\Delta = \Delta_1 \times \ldots \times \Delta_r$. Заметим, что каждый вектор из Δ состоит из $\sum\limits_{r=1}^{s} |B_r|$ бит и, следовательно, его подстановка в C может существенно упростить задачу. При этом мощность Δ равна

$$\left(\prod_{r=1}^{s}(1-\rho_r^*)\right) 2^{\sum\limits_{r=1}^{s}|B_r|}, \tag{7}$$

где ρ_r^* — доля таких $\beta \in \{0,1\}^{|B_r|}$, что $C[\beta/B]$ решается алгоритмом O. При близких к 1 величинах ρ_r^*, $r \in \{1, \ldots, s\}$, величина (7) может быть разумной, и соответствующие SAT-задачи оказываются простыми для современных SAT-решателей.

В [3] проведены вычислительные эксперименты по поиску сильных (ε, δ)-лазеек, которые подтвердили, что во многих комбинаторных задачах часто встречаются такие лазейки, имеющие малую мощность. Иногда их использование позволяет ускорить

решение исходной SAT-проблемы в десятки и сотни раз. Поиск сильных (ε, δ)-лазеек может рассматриваться как задача минимизации функции вида (6), для чего можно применять любой алгоритм псевдобулевой оптимизации. В [3] для минимизации функций вида (6) использован генетический алгоритм. Эксперименты проводились на вычислительном кластере с помощью программной системы EvoGuess [5]. В роли алгоритма O выступало простейшее правило единичного дизъюнкта [6].

Следует отметить, что концепция сильных (ε, δ)-лазеек может быть естественным образом перенесена на любую задачу удовлетворения ограничений. Например, пусть I — произвольная задача целочисленного линейного программирования (ILP, Integer Linear Programming) и X — множество фигурирующих в постановке I переменных со значениями в $\{0, 1\}$ (таким образом, речь идет о 0-1-ILP). Напомним, что в стандартной подстановке для решения I требуется минимизировать (или максимизировать) некоторую функцию при ограничениях, заданных системой целочисленных неравенств следующего вида:

$$Ax \geqslant b, \tag{8}$$

где A — целочисленная матрица размера $m \times n$; b — целочисленный вектор длины m; $x = (x_1, \ldots, x_n)$ — вектор переменных, принимающих значения в $\{0, 1\}$.

По аналогии мы можем ввести множество булевых переменных $X = \{x_1, \ldots, x_n\}$, рассмотреть произвольное B, $B \subseteq X$, а в качестве полиномиального алгоритма взять алгоритм преобразования системы неравенств (8) после подстановки в неё произвольного $\beta \in \{0, 1\}^{|B|}$. Далее можем искать ρ-лазейки и, в частности, сильные (ε, δ)-лазейки, используя шаги, аналогичные описанным выше. Первые вычислительные эксперименты в этом направлении (с использованием ILP-решателей Gurobi и SCIP) показали работоспособность концепции вероятностных лазеек: для некоторых трудных 0-1-ILP-задач найденные лазейки дают ускорение в десятки раз.

ЛИТЕРАТУРА

1. *Williams R., Gomes C. P., and Selman B.* Backdoors to typical case complexity // Proc. IJCAI'03. August 2003. P. 1173–1178.
2. *Чень Ч., Ли Р.* Математическая логика и автоматическое доказательство теорем. М.: Наука, 1983.
3. *Semenov A., Pavlenko A., Chivilikhin D., and Kochemazov S.* On probabilistic generalization of backdoors in Boolean satisfiability // Proc. AAAI-2022. `https://www.aaai.org/AAAI22Papers/AAAI-8477.SemenovA.pdf`.
4. *Metropolis N. and Ulam S.* The Monte Carlo method // J. Amer. Statistical Assoc. 1949. No. 44(247). P. 335–341.
5. `https://github.com/ctlab/EvoGuess/releases/tag/v2.0.0`.
6. *Dowling W. F. and Gallier J. H.* Linear-time algorithms for testing the satisfiability of propositional horn formulae // J. Logic Programming. 1984. V. 1. Iss. 3. P. 267–284.

Секция 5

ПРИКЛАДНАЯ ТЕОРИЯ КОДИРОВАНИЯ И ГРАФОВ

УДК 519.1, 004.05 DOI 10.17223/2226308X/15/24

О КОНЕЧНОЙ ДИНАМИЧЕСКОЙ СИСТЕМЕ ВСЕХ ВОЗМОЖНЫХ ОРИЕНТАЦИЙ ДАННОГО ГРАФА СО ВСЕМИ ДОСТИЖИМЫМИ И НЕДОСТИЖИМЫМИ СОСТОЯНИЯМИ

А. В. Жаркова

Рассматривается конечная динамическая система, состояниями которой являются все возможные ориентации данного графа, 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. Характеризуются системы, все состояния которых являются достижимыми и в которых есть недостижимые состояния; подсчитывается количество графов, образующих системы со всеми достижимыми состояниями; приводится таблица с количеством графов с числом вершин от одной до двенадцати, образующих системы со всеми достижимыми и недостижимыми состояниями.

Ключевые слова: *граф, достижимое состояние, конечная динамическая система, недостижимое состояние, ориентированный граф, отказоустойчивость, эволюционная функция.*

Графовые модели, в которых отказы процессоров интерпретируются как удаление соответствующих вершин, а отказы сетевых каналов — как удаление дуг, занимают важное место в задачах, связанных с отказоустойчивостью компьютерных сетей. При изучении модельных графов можно применять идеи и методы теории конечных динамических систем [1 – 3]. В модели [1] в качестве механизма восстановления работоспособности сети предлагается так называемая SER-динамика бесконтурных связных ориентированных графов. В настоящей работе графы изучаются с точки зрения динамического подхода к отказоустойчивости графовых систем.

Под *ориентированным графом* (*орграфом*) понимается пара $\overrightarrow{G} = (V, \beta)$, где V — конечное непустое множество вершин; $\beta \subseteq V \times V$ — отношение смежности на множестве V (пара $(u, v) \in \beta$ называется *дугой* орграфа). *Неориентированным графом* (или, для краткости, *графом*) называется пара $G = (V, \beta)$, где β — симметричное и антирефлексивное отношение на множестве вершин V. Дуги неориентированного графа называют *рёбрами*. Граф $G = (V, \beta)$ называется *полным*, если любые две его вершины соединены ребром. Полный граф с n вершинами обозначается K_n. Вершины u и v графа G называются *связанными*, если в G существует проходящий через них путь. Отношение связанности является эквивалентностью на множестве вершин графа. Классы этого отношения называются *компонентами связности* (или просто *компонентами*) графа. Говорят, что вершина v *достижима* из вершины u, если в орграфе существует путь из u в v. Вершина орграфа, недостижимая из других его вершин, называется *источником*, а вершина, из которой недостижима никакая другая вершина, — *стоком* [4].

Под *конечной динамической системой* понимается пара (S, δ), где S — конечное непустое множество *состояний* системы; $\delta : S \to S$ — отображение множества состояний в себя, называемое *эволюционной функцией* системы. Конечной динамической системе сопоставляется карта, представляющая собой ориентированный граф с множеством вершин S и дугами, проведёнными из каждой вершины $s \in S$ в вершину $\delta(s)$.

Основными проблемами теории конечных динамических систем являются задачи отыскания эволюционных параметров системы без проведения динамики. К числу основных характеристик состояний динамических систем относятся свойства достижимости и недостижимости: состояние, не имеющее непосредственных предшественников, называется *недостижимым* или *начальным* состоянием системы, иначе состояние называется *достижимым*. В [5] описаны недостижимые состояния конечных динамических систем всех возможных ориентаций графов. В данной работе характеризуются конечные динамические системы всех возможных ориентаций графов, все состояния которых являются достижимыми и в которых есть недостижимые состояния.

Пусть дан некоторый граф G. Пометим его вершины и придадим его рёбрам произвольную ориентацию, тем самым получив ориентированный граф $\overrightarrow{G}$. 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 чего получаем орграф $\alpha(\overrightarrow{G})$. Если проделать эти действия со всеми возможными ориентациями данного графа, то получим карту динамической системы. Такая динамика для бесконтурных связных орграфов введена в [1]. Итак, будем рассматривать конечную динамическую систему (Γ_G, α), где через Γ_G обозначим множество всех возможных ориентаций графа G, а эволюционная функция α задаётся следующим образом: если дан некоторый орграф $\overrightarrow{G} \in \Gamma_G$, 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 G и карта конечной динамической системы (Γ_G, α).

Теорема 1. В конечной динамической системе (Γ_G, α) все состояния являются достижимыми тогда и только тогда, когда в графе G компонентами связности являются полные графы с n вершинами при $1 \leqslant n \leqslant 3$, и только они.

На рис. 1 граф G имеет следующие три компоненты связности: один граф K_1, один граф K_2 и один граф K_3; все состояния конечной динамической системы (Γ_G, α) являются достижимыми.

Теорема 2. В конечной динамической системе (Γ_G, α) есть недостижимые состояния тогда и только тогда, когда в графе G есть компоненты связности, отличные от полных графов с n вершинами при $1 \leqslant n \leqslant 3$.

Теорема 3. Количество графов G с n вершинами, образующих конечные динамические системы (Γ_G, α), все состояния которых являются достижимыми, равно

$$\text{КСД}_{G_n} = 1 + \left\lfloor \frac{n}{2} \right\rfloor + \left\lfloor \frac{n}{3} \right\rfloor + \sum_{i=1}^{\lfloor (n-3)/2 \rfloor} \left\lfloor \frac{n-2i}{3} \right\rfloor .$$

В таблице приведены данные о количестве графов с n вершинами, $1 \leqslant n \leqslant 12$, образующих конечные динамические системы (Γ_G, α) со всеми достижимыми (граф Д) и недостижимыми (граф Н) состояниями, полученные с помощью вычислительных экспериментов.

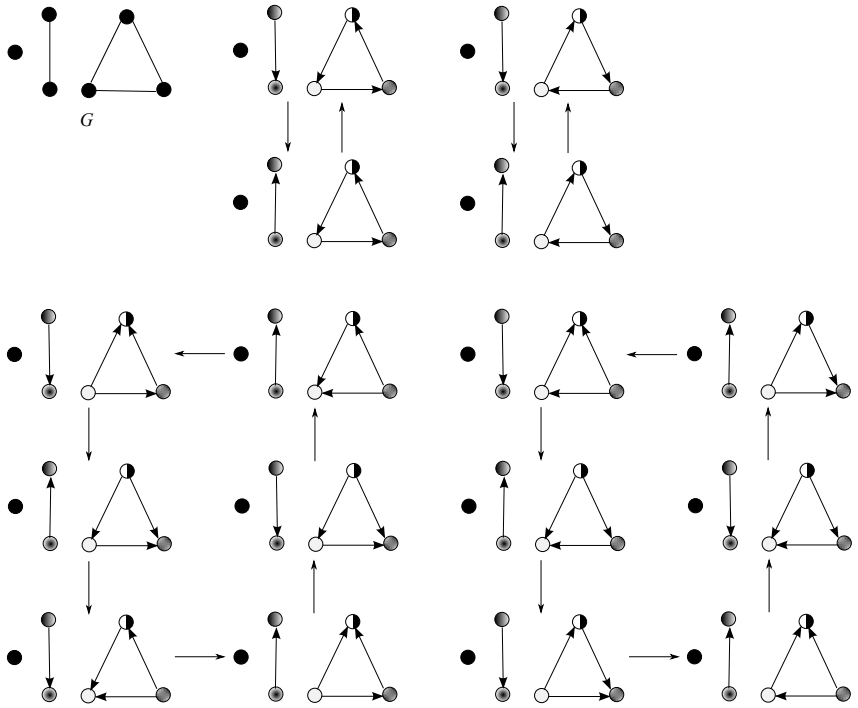


Рис. 1. Граф G и карта конечной динамической системы (Γ_G, α)

n	Количество графов Д	%	Количество графов Н	%
1	1	100	0	0
2	2	100	0	0
3	3	75	1	25
4	4	≈ 36	7	≈ 64
5	5	≈ 15	29	≈ 85
6	7	≈ 4	149	≈ 96
7	8	≈ 1	1036	≈ 99
8	10	$\approx 0{,}1$	12336	$\approx 99{,}9$
9	12	$\approx 0{,}004$	274656	$\approx 99{,}996$
10	14	$\approx 0{,}0001$	12005154	$\approx 99{,}9999$
11	16	$\approx 0{,}000002$	1018997848	$\approx 99{,}999998$
12	19	$\approx 0{,}00000001$	165091172573	$\approx 99{,}99999999$

Можно заметить, что при увеличении количества вершин в графе G в большинстве конечных динамических систем (Γ_G, α) есть недостижимые состояния.

ЛИТЕРАТУРА

1. *Barbosa V. C.* An Atlas of Edge-Reversal Dynamics. London: Chapman&Hall/CRC, 2001.
2. *Салий В. Н.* Об одном классе конечных динамических систем // Вестник Томского госуниверситета. Приложение. 2005. № 14. С. 23–26.
3. *Anashin V. and Khrennikov A.* Applied Algebraic Dynamics. Walter De Gruyter, 2009.
4. *Богомолов А. М., Салий В. Н.* Алгебраические основы теории дискретных систем. М.: Наука, Физматлит, 1997.
5. *Жаркова А. В.* О ветвлении и непосредственных предшественниках состояний в конечной динамической системе всех возможных ориентаций графа // Прикладная дискретная математика. Приложение. 2013. № 6. С. 76–78.

УДК 519.72 DOI 10.17223/2226308X/15/25

СЕРИЯ ФОРМУЛ ДЛЯ ПАРАМЕТРОВ БХАТТАЧАРИЯ В ТЕОРИИ ПОЛЯРНЫХ КОДОВ[1]

С. Г. Колесников, В. М. Леонтьев

В теории полярных кодов для определения позиций замороженных и информационных битов используются параметры Бхаттачария. Они характеризуют скорость поляризации каналов $W_N^{(i)}$, где $1 \leqslant i \leqslant N$ и $N = 2^n$, $n = 1, 2, \ldots$, — длина кода, специальным образом построенных из исходного канала W. Предполагается, что i-й бит сообщения передаётся по каналу $W_N^{(i)}$, а параметр Бхаттачария $Z(W_N^{(i)})$ можно интерпретировать как степень зашумлённости $W_N^{(i)}$. W является моделью физического канала передачи. В случае, когда W есть классический двоичный симметричный канал без памяти, известные в настоящее время формулы для параметров Бхаттачария содержат порядка $2^N = 2^{2^n}$ слагаемых. Для серии каналов $W_N^{(N-2^k+1)}$, $k = 0, 1, \ldots, n-1$, найдены формулы, которые содержат порядка $2^{(n-k+1)2^k}$ слагаемых. Также высказан ряд предположений о том, как ещё можно упростить полученные формулы.

Ключевые слова: *полярный код, параметр Бхаттачария.*

Пусть W — двоичный симметричный канал без памяти с входным алфавитом $X = \{0, 1\}$, выходным алфавитом $Y = \{0, 1\}$ и переходными вероятностями $W(y \mid x) = p$, если $x \neq y$, иначе $W(y \mid x) = 1 - p$. Через W^N, $N = 2^n$, $n = 1, 2, \ldots$, обозначим N-ю декартову степень W. Для каждого i, $1 \leqslant i \leqslant N$, определим канал $W_N^{(i)} : X \to Y^N \times X^{i-1}$ с переходными вероятностями

$$W_N^{(i)}(y, u' \mid u_i) = \frac{1}{2^{N-1}} \sum_{u'' \in X^{N-i}} W^N(y \mid uG_N),$$

где $y \in Y^N$; $u' \in X^{i-1}$; $u = u'u_iu''$ — конкатенация векторов $u', (u_i), u''$; G_N — порождающая матрица полярного кода с ядром Арикана

$$F = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Тогда

$$Z(W_N^{(i)}) = \sum_{y \in Y^N} \sum_{u' \in X^{i-1}} \sqrt{W_N^{(i)}(y, u \mid 0)\ W_N^{(i)}(y, u \mid 1)}. \tag{1}$$

Таким образом, формула для вычисления параметра Бхаттачария содержит 2^{N+i-1} слагаемых. Более того, суммы под знаком корня содержат по 2^{N-i} слагаемых. Поэтому расчёт $Z(W_N^{(i)})$ невозможен уже при $N \geqslant 32$. В [1, 2] формула (1) упрощена до следующей:

$$Z(W_N^{(i)}) = 2^{N-1} \sum_{y \in L} \sqrt{W_N^{(i)}(y, u \mid 0)\ W_N^{(i)}(y, u \mid 1)},$$

где L есть система представителей некоторого фактор-множества и $|L| = 2^i$. Там же установлено равенство

$$Z(W_N^{(2i)}) = \left(Z(W_{N/2}^{(i)})\right)^2,$$

[1]Работа поддержана Красноярским математическим центром, финансируемым Минобрнауки РФ (соглашение № 075-02-2022-876).

с учётом которого достаточно найти эффективные формулы для нечётных i.

Доказана следующая

Теорема 1. Пусть $n, k \in \mathbb{N}$, $k < n$, $N = 2^n$, $m = 2^k$, $M = N/m - 1$. Тогда

$$Z(W_N^{(N-2^k+1)}) = \sum_{s_1=0}^{M} \cdots \sum_{s_m=0}^{M} \binom{M}{s_1} \cdots \binom{M}{s_m} \sqrt{Q_1(s_1, \ldots, s_m) Q_2(s_1, \ldots, s_m)}, \qquad (2)$$

где

$$Q_i(s_1, \ldots, s_m) = \\ = \sum_{\substack{y \in Y^m, \\ \omega(y) \in 1+i+2\mathbb{Z}}} p^{\omega(y)N/m + (\overline{y}_1 - y_1)s_1 + \ldots + (\overline{y}_m - y_m)s_m} (1-p)^{N - \omega(y)N/m + (y_1 - \overline{y}_1)s_1 + \ldots + (y_m - \overline{y}_m)s_m}.$$

Здесь $\omega(y) = y_1 + \ldots + y_m$ — вес Хэмминга вектора y, $\overline{0} = 1$, $\overline{1} = 0$.

Замечание 1. Для различных наборов чисел $s_1, \ldots, s_m$ произведения

$$Q_1(s_1, \ldots, s_m) Q_2(s_1, \ldots, s_m)$$

могут быть равны (как полиномы от p). Поэтому интересно найти условия на $s_1, \ldots, s_m$ и $s'_1, \ldots, s'_m$, при которых

$$Q_1(s_1, \ldots, s_m) Q_2(s_1, \ldots, s_m) = Q_1(s'_1, \ldots, s'_m) Q_2(s'_1, \ldots, s'_m),$$

что, возможно, существенно упростит полученную формулу.

Замечание 2. Выражение для Q_i в общем случае является полиномом от p следующего вида:

$$Q_i(s_1, \ldots, s_m) = \sum_{j=0}^{N} \alpha_j^i(s_1, \ldots, s_m) p^j (1-p)^{N-j}, \qquad i = 1, 2.$$

Явное вычисление коэффициентов $\alpha_j(s_1, \ldots, s_m)$ также значительно упростит (2). Легко видеть, что

$$\alpha_j^i(s_1, \ldots, s_m) = \\ = \left| \{ y \in Y^m : \omega(y) \in 1 + i + 2\mathbb{Z},\ \omega(y)(N/m) + (\overline{y}_1 - y_1)s_1 + \ldots + (\overline{y}_m - y_m)s_m = j \} \right|.$$

Следующая формула (не из указанной в теореме 1 серии) найдена с учётом замечаний 1 и 2:

$$Z(W_N^{(1)}) = 2 \sqrt{\left(\sum_{k=0}^{N/2} \binom{N}{2k} p^{2k} (1-p)^{N-2k} \right) \left(\sum_{k=1}^{N/2} \binom{N}{2k-1} p^{2k-1} (1-p)^{N-2k+1} \right)}.$$

ЛИТЕРАТУРА

1. *Arikan E.* Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels. `https://arxiv.org/abs/0807.3917`. 2009.
2. *Arikan E.* Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels // IEEE Trans. Inform. Theory. 2009. V. 55. No. 7. P. 3051–3073.

УДК 519.17 DOI 10.17223/2226308X/15/26

О ЕДИНСТВЕННОСТИ МИНИМАЛЬНОГО РЁБЕРНОГО 1-РАСШИРЕНИЯ ГИПЕРКУБА[1]

А. А. Лобов, М. Б. Абросимов

Одним из важных свойств надёжных вычислительных систем является их отказоустойчивость. Для её исследования можно использовать аппарат теории графов. Рассматриваются минимальные рёберные расширения графа, которые являются моделью для исследования отказа связей вычислительной системы. Граф $G^* = (V^*, \alpha^*)$ с n вершинами называется минимальным рёберным k-расширением n-вершинного графа $G = (V, \alpha)$, если граф G вкладывается в каждый граф, получающийся из G^* удалением любых его k рёбер, и имеет при этом минимально возможное число рёбер. Гиперкуб Q_n — это регулярный 2^n-вершинный граф порядка n, представляющий собой декартово произведение n полных 2-вершинных графов K_2. Гиперкуб является распространённой топологией для построения вычислительных систем. Ранее было описано семейство графов Q_n^*, представители которого при $n > 1$ являются минимальными рёберными 1-расширениями соответствующих гиперкубов. Проведённый вычислительный эксперимент показал, что при $n \leqslant 4$ эти расширения являются единственными с точностью до изоморфизма. Получено аналитическое доказательство единственности минимальных рёберных 1-расширений гиперкубов при $n \leqslant 4$ и установлено одно общее свойство произвольного минимального рёберного 1-расширения гиперкуба при $n > 2$.

Ключевые слова: *граф, гиперкуб, рёберная отказоустойчивость, минимальное рёберное 1-расширение.*

Введение

Топология гиперкуба является популярной схемой соединения параллельных процессоров [1], в том числе в отказоустойчивых системах типа IBM Blue Gene/Q [2]. Особый интерес с точки зрения отказоустойчивости представляет 4-куб Q_4 или тессеракт. Для исследования полной отказоустойчивости элементов Дж. Хейз предложил модель, основанную на графах [3], которая позднее была перенесена и на случай отказов связей [4]. Оптимальные вершинные k-отказоустойчивые реализации гиперкубов Q_n при $n > 3$ неизвестны. На практике используются тривиальные отказоустойчивые реализации с одним избыточным элементом, соединённым со всеми остальными. В работе [5] описывается вершинное 1-расширение гиперкуба Q_4, которое имеет на одно ребро меньше, чем тривиальное. Однако не доказано, что оно является минимальным. Ранее была предложена оптимальная рёберная 1-отказоустойчивая реализация гиперкуба Q_4 [4], обобщённая на произвольное $n > 1$ [6]. В данной работе удалось доказать, что при $n \leqslant 4$ гиперкуб Q_n имеет единственную с точностью до изоморфизма оптимальную рёберную 1-отказоустойчивую реализацию (или, в другой терминологии, минимальное рёберное 1-расширение).

Определение 1. Декартовым произведением $G_1 \times G_2$ двух графов $G_1 = (V_1, \alpha_1)$ и $G_2 = (V_2, \alpha_2)$ называется граф $G = (V, \alpha)$, такой, что $V = V_1 \times V_2$ и вершины (u_1, u_2) и (v_1, v_2) смежны в G тогда и только тогда, когда либо $u_1 = v_1$, а u_2 и v_2 смежны в G_2, либо $u_2 = v_2$, а u_1 и v_1 смежны в G_1.

[1]Работа выполнена при поддержке Минобрнауки России в рамках госзадания (проект №FSRR-2020-0006).

Определение 2. Гиперкубом Q_n (n-кубом) называется граф, являющийся декартовым произведением n полных 2-вершинных графов K_2.

Гиперкуб Q_n — это регулярный 2^n-вершинный граф порядка n. Семейство гиперкубов достаточно хорошо изучено [7].

Вершины гиперкуба можно пометить двоичными векторами таким образом, чтобы расстояние между каждыми двумя вершинами равнялось дистанции Хэмминга между их метками. Это свойство непосредственно следует из построения гиперкуба.

1. Рёберные расширения

Определение 3. Граф $G^* = (V^*, \alpha^*)$ называется *минимальным рёберным k-расширением* n-вершинного графа $G = (V, \alpha)$, если выполняются следующие условия:

1) граф G^* является рёберным k-расширением графа G, то есть граф G вкладывается в каждый граф, получающийся из G^* удалением любых его k рёбер;
2) граф G^* содержит n вершин, то есть $|V^*| = |V|$;
3) α^* имеет минимальную мощность при выполнении условий 1 и 2.

Если рассматривать простые графы, то минимальное рёберное k-расширение существует не у всех графов. Например, полные графы K_n не имеют минимальных ребёрных k-расширений ни при каких натуральных k. Гиперкуб Q_1 соответственно тоже не имеет минимального рёберного k-расширения ни при каких натуральных k. У графа может быть и несколько неизоморфных минимальных рёберных k-расширений. Задача поиска минимальных рёберных k-расширений является вычислительно сложной [8].

В [2] доказана лемма, которая позволяет охарактеризовать вид минимального рёберного k-расширения и оценить минимально возможное количество дополнительных рёбер в нём.

Лемма 1 [2]. Если минимальная степень вершины графа G равна $d > 0$, то его минимальное рёберное k-расширение не содержит вершин степени ниже $d + k$.

2. Минимальное рёберное 1-расширение гиперкуба

Опишем схему построения минимального рёберного 1-расширения для любого гиперкуба Q_n при $n > 1$.

Определим семейство графов Q_n^*. Граф Q_n^* при $n > 1$ получается путём соединения каждой вершины гиперкуба Q_n с наиболее удалённой от неё вершиной. Если вершина имеет код k, то она соединяется с вершиной, код которой получается из k поразрядной инверсией.

Теорема 1. Для n-мерного гиперкуба Q_n при $n > 1$ граф Q_n^* является минимальным рёберным 1-расширением.

Легко убедиться, что минимальное рёберное 1-расширение для гиперкуба Q_2 единственно с точностью до изоморфизма: граф Q_2^* изоморфен графу K_4, и это единственный с точностью до изоморфизма регулярный 4-вершинный граф порядка 3. Для произвольного минимального рёберного 1-расширения гиперкуба Q_n при $n > 2$ удалось установить следующее общее свойство.

Лемма 2. Минимальное рёберное 1-расширение гиперкуба Q_n при $n > 2$ не содержит рёбер, соединяющих вершины, расстояние между которыми равно 2.

Следствие 1. Гиперкуб Q_3 имеет единственное с точностью до изоморфизма минимальное рёберное 1-расширение.

Аналогичный результат удалось доказать и для гиперкуба Q_4, что является основным результатом данной работы.

Теорема 2. Гиперкуб Q_4 имеет единственное с точностью до изоморфизма минимальное рёберное 1-расширение.

Единственные с точностью до изоморфизма минимальные рёберные 1-расширения гиперкубов Q_2 и Q_3 изображены на рис. 1.

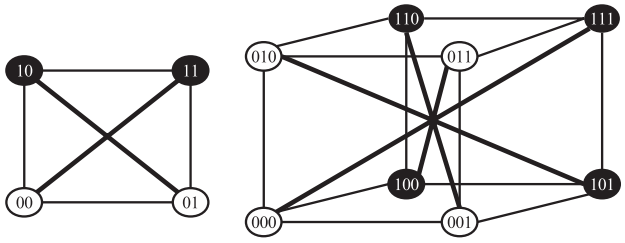


Рис. 1. Минимальные рёберные 1-расширения для Q_2 и Q_3

ЛИТЕРАТУРА

1. *Padua D. A.* Encyclopedia of Parallel Computing. N.Y.: Springer, 2011.
2. *Абросимов М. Б.* Графовые модели отказоустойчивости. Саратов: Изд-во Сарат. ун-та, 2012. 192 с.
3. *Hayes J. P.* A graph model for fault-tolerant computing system // IEEE Trans. Comput. 1976. V. C.25. No. 9. P. 875–884.
4. *Harary F. and Hayes J. P.* Edge fault tolerance in graphs // Networks. 1993. V. 23. P. 135–142.
5. *Лобов А. А., Абросимов М. Б.* О вершинном 1-расширении гиперкуба // Компьютерные науки и информационные технологии: Материалы Междунар. науч. конф. Саратов: Издат. центр «Наука», 2018. С. 249–251.
6. *Лобов А. А., Абросимов М. Б.* О минимальном рёберном 1-расширении гиперкуба // Прикладная дискретная математика. Приложение. 2018. № 11. С. 109–111.
7. *Harary F., Hayes J. P., and Wu H.-J.* A survey of the theory of hypercube graphs // Computers & Math. with Appl. 1988. V. 15. Iss. 4. P. 277–289.
8. *Абросимов М. Б.* О сложности некоторых задач, связанных с расширениями графов // Матем. заметки. 2010. № 5(88). С. 643–650.

УДК 519.17 DOI 10.17223/2226308X/15/27

О ВЕРХНЕЙ И НИЖНЕЙ ОЦЕНКАХ ЧИСЛА ДОПОЛНИТЕЛЬНЫХ ДУГ МИНИМАЛЬНОГО РЁБЕРНОГО 1-РАСШИРЕНИЯ ОРИЕНТАЦИИ ЦИКЛА

О. В. Моденова, М. Б. Абросимов

Исследуются верхняя и нижняя оценки числа дополнительных дуг $\mathrm{ec}(\overrightarrow{C_n})$ минимального рёберного 1-расширения ориентации $\overrightarrow{C_n}$ цикла C_n. Основной результат работы: $\lceil n/2 \rceil \leqslant \mathrm{ec}(\overrightarrow{C_n}) \leqslant n$. Приводятся примеры ориентаций циклов, на которых оценки достигаются.

Ключевые слова: *минимальное рёберное 1-расширение, ориентация цикла, отказоустойчивость.*

Введение

Рассмотрим неориентированные и ориентированные графы, основные определения даются согласно работам [1 – 3]. Неориентированным циклом (далее — просто циклом) C_n называется n-вершинный граф, состоящий из единственного цикла, содержащего все вершины. Очевидно, что число вершин любого цикла $n \geqslant 3$. Цикл C_n является связным однородным графом порядка 2. Для нас представляют интерес ориентации цикла C_n, которые получаются заменой каждого ребра цикла на дугу. Простые циклические пути в ориентированном графе называются контурами. Особым случаем ориентации цикла C_n является контур $\overrightarrow{C_n}$, то есть ориентированный граф, состоящий из единственного контура, содержащего все вершины. В контуре $\overrightarrow{C_n}$ все вершины имеют степени исхода и захода, равные 1.

Граф $G^* = (V^*, \alpha^*)$ называется *минимальным вершинным k-расширением* (МВ-kР) n-вершинного графа $G = (V, \alpha)$, если выполняются следующие условия:

1) граф G^* является вершинным k-расширением графа G, то есть G вкладывается в каждый подграф графа G^*, получающийся удалением любых его k вершин;
2) граф G^* содержит $n + k$ вершин, то есть $|V^*| = |V| + k$;
3) α^* имеет минимальную мощность при выполнении условий 1 и 2.

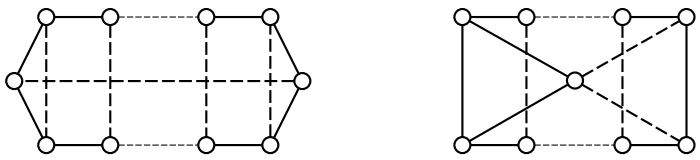
Понятие минимального вершинного k-расширения появилось в работе [1] как модель для исследования отказоустойчивости элементов дискретных систем. Позднее в [2] введена модель для исследования отказов связей между элементами.

Граф $G^* = (V^*, \alpha^*)$ называется *минимальным рёберным k-расширением* (МР-kР) n-вершинного графа $G = (V, \alpha)$, если выполняются следующие условия:

1) граф G^* является рёберным k-расширением графа G, то есть G вкладывается в каждый граф, получающийся из G^* удалением любых его k рёбер (дуг);
2) граф G^* содержит n вершин, то есть $|V^*| = |V|$;
3) α^* имеет минимальную мощность при выполнении условий 1 и 2.

Задача поиска минимального вершинного или рёберного k-расширения для произвольного графа является вычислительно сложной [4], и в общем виде решение удалось получить лишь для некоторых классов графов. Обзор основных результатов можно найти в [3]. В работе [2] предлагаются схемы построения минимальных рёберных 1-расширений для циклов.

Теорема 1. Графы, представленные на рис. 1, являются минимальными рёберными 1-расширениями для цикла C_n при чётном числе вершин (a) и нечётном числе вершин ($б$).

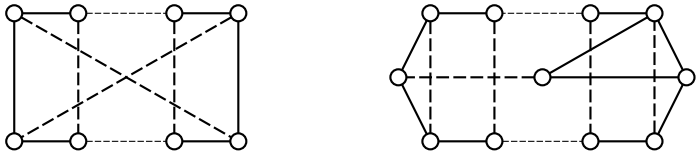


а *б*

Рис. 1. МР-1Р цикла C_n из теоремы 1

Заметим, что число дополнительных рёбер в этих расширениях равно $\lceil n/2 \rceil$. В [3] предлагаются другие схемы построения минимальных рёберных 1-расширений циклов и доказывается, что при $n > 5$ построенные по ним расширения неизоморфны расширениям из теоремы 1.

Теорема 2. Графы, представленные на рис. 2, являются минимальными рёберными 1-расширениями для цикла C_n при чётном числе вершин (a) и нечётном числе вершин ($б$); при $n > 5$ они неизоморфны расширениям, построенным по теореме 1.



а *б*

Рис. 2. МР-1Р цикла C_n из теоремы 2

1. Ориентации циклов

Рассмотрим ориентации цикла. Ранее были получены оценки для числа дополнительных дуг в МВ-1Р ориентации цикла, а также схемы построения для двух ориентаций цикла, на которых достигается нижняя оценка. Напомним, что расширение (вершинное или рёберное) G^* графа G называется неприводимым, если никакая его собственная часть не является расширением (вершинным или рёберным) графа G. Заметим, что неориентированный цикл можно рассматривать как ориентированный граф, в котором каждое ребро является парой встречных дуг.

Теорема 3. Цикл C_n является неприводимым рёберным 1-расширением для произвольной ориентации $\overrightarrow{C_n}$ цикла C_n.

Теорема 1 даёт оценку сверху для числа дополнительных дуг в минимальном рёберном 1-расширении ориентации цикла. Следующая теорема показывает, что оценка является достижимой.

Теорема 4. Цикл C_n является минимальным рёберным 1-расширением контура $\overrightarrow{C_n}$.

Отметим, что в общем случае цикл C_n является не единственным минимальным рёберным 1-расширением контура $\overrightarrow{C_n}$.

Для получения нижней оценки заметим, что в цикле каждая вершина имеет степень 2, соответственно в ориентации цикла в каждой вершине будет две дуги (входящие или исходящие). Тогда в минимальном рёберном 1-расширении в каждой вершине будет не менее трёх дуг [3]. Это даёт нижнюю оценку числа дополнительных дуг $\lceil n/2 \rceil$. Получаем итоговую оценку:

Теорема 5. Для числа дополнительных дуг минимального рёберного 1-расширения любой ориентации $\overrightarrow{C_n}$ цикла C_n справедливо следующее неравенство:

$$\lceil n/2 \rceil \leqslant \mathrm{ec}(\overrightarrow{C_n}) \leqslant n.$$

Далее представлены схемы построения минимальных рёберных 1-расширений для некоторых ориентаций циклов, которые показывают, что нижняя оценка также является достижимой. С этой целью рассмотрим возможность ориентации минимальных рёберных 1-расширений циклов из теоремы 2.

2. Циклы с чётным числом вершин

Рассмотрим цикл C_n с чётным числом вершин, который ориентируем по схеме «сток — источник». Обозначим такую ориентацию $\overrightarrow{CST_n}$. Очевидно, что в минимальном рёберном 1-расширении орграфа $\overrightarrow{CST_n}$ все вершины, в которых есть три дуги, могут быть также только источниками или стоками. По этой причине минимальное рёберное 1-расширение для $\overrightarrow{CST_n}$ не может быть получено никакой ориентацией минимального рёберного 1-расширения цикла из теоремы 1. Можно заметить, что это расширение имеет два цикла длины 3. Любая его ориентация приведёт к тому, что появится вершина, не являющаяся ни стоком, ни источником. Граф из теоремы 2 не имеет циклов длины 3 и для него подобную ориентацию построить возможно.

Теорема 6. Граф, представленный на рис. 3, является минимальным рёберным 1-расширением для ориентации $\overrightarrow{CST_n}$ при $n = 4k + 2$.

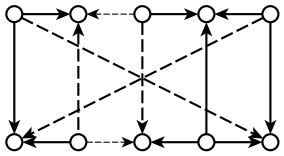


Рис. 3. MP-1P ориентации цикла из теоремы 6

Пунктирными линиями на рис. 3 показаны дополнительные дуги. Заметим, что в ориентации направление дуг выбирается естественным образом, чтобы сохранить источники и стоки. Очевидно, что такая ориентация невозможна при $n = 4k$, так как в этом случае вершины в противоположных углах прямоугольника будут иметь одинаковый тип (сток — сток или источник — источник). Любая ориентация диагонали не сможет сохранить источники и стоки.

3. Циклы с нечётным числом вершин

Рассмотрим цикл C_n с нечётным числом вершин. Его нельзя ориентировать по схеме «сток — источник», поэтому предложим другую ориентацию. В одной вершине ориентируем рёбра так, чтобы одно ребро было исходящим, а другое — входящим. Остальные вершины ориентируем по схеме «сток — источник». Обозначим такую ориентацию $\overrightarrow{CSTN_n}$. Для неё можно ориентировать минимальные рёберные 1-расширения как из теоремы 1, так и из теоремы 2.

Теорема 7. Граф, представленный на рис. 4, является минимальным рёберным 1-расширением для ориентации $\overrightarrow{CSTN_n}$.

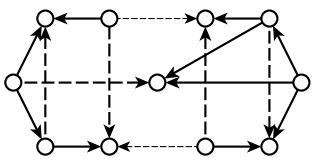


Рис. 4. MP-1P ориентации цикла из теоремы 7

ЛИТЕРАТУРА

1. *Hayes J. P.* A graph model for fault-tolerant computing system // IEEE Trans. Comput. 1976. V. C25. No. 9. P. 875–884.
2. *Harary F. and Hayes J. P.* Edge fault tolerance in graphs // Networks. 1993. V. 23. P. 135–142.
3. *Абросимов М. Б.* Графовые модели отказоустойчивости. Саратов: Изд-во Сарат. ун-та, 2012. 192 с.
4. *Абросимов М. Б.* О сложности некоторых задач, связанных с расширениями графов // Матем. заметки. 2010. Т. 88. Вып. 5. С. 643–650.

УДК 519.17 DOI 10.17223/2226308X/15/28

ОБ ОДНОМ СЕМЕЙСТВЕ ОПТИМАЛЬНЫХ ГРАФОВ С ЗАДАННЫМИ МЕРАМИ СВЯЗНОСТИ[1]

Б. А. Теребин, М. Б. Абросимов

Вершинной связностью k называется наименьшее число вершин, удаление которых приводит к несвязному или тривиальному графу. Рёберной связностью λ нетривиального графа называется наименьшее число рёбер, удаление которых приводит к несвязному графу. Исследуются минимальные по числу рёбер n-вершинные графы, которые имеют заданные значения вершинной и рёберной связности. Помимо теоретического интереса, графы с заданными значениями вершинной или рёберной связности представляют и прикладной интерес как модели отказоустойчивых сетей. Основной результат состоит в том, что для определённой области значений k и λ удалось описать графы, которые при заданном n имеют минимальное число рёбер.

Ключевые слова: *граф, вершинная связность, рёберная связность, отказоустойчивость.*

Введение

Изучение графов с заданной вершинной или рёберной связностью представляет интерес как с теоретической, так и с прикладной точек зрения. В теоретическом плане эти исследования восходят к работам [1 – 3], в прикладном — к работе [4], в которой исследуется построение сетей минимальной стоимости с заданнной связностью. Большой интерес представляют графы Харари, которые имеют минимальное число рёбер при заданном значении вершинной связности [2, 5].

Рассмотрим простые неориентированные графы и их основные меры связности. Понятия из теории графов используются в соответствии с [6, 7]. Напомним, что *связным* называется граф, любая пара вершин которого соединена путём. В противном случае граф называется *несвязным*. *Тривиальным* называется одновершинный граф. Граф, любые две вершины которого смежны, называется *полным*.

Определение 1. *Вершинной связностью* k графа G называется наименьшее число вершин, удаление которых приводит к несвязному или тривиальному графу.

Определение 2. *Рёберная связность* λ нетривиального графа G определяется как наименьшее количество рёбер, удаление которых приводит к несвязному графу.

[1]Работа выполнена при поддержке Минобрнауки России в рамках госзадания (проект № FSRR-2020-0006).

Например, деревья имеют вершинную и рёберную связности 1. Полный n-вершинный граф имеет вершинную и рёберную связности $n-1$. Далее будем рассматривать только связные графы. Обозначим минимальную степень вершины в графе через δ.

Вершинная связность k, рёберная связность λ и минимальная степень вершины δ произвольного графа связаны следующим неравенством:

Теорема 1 [1]. Для любого графа G справедливо неравенство $k \leqslant \lambda \leqslant \delta$.

Доказано, что для любых подходящих значений k, λ и δ существует соответствующий граф:

Теорема 2 [3]. Для любых натуральных чисел a, b, c, таких, что $0 < a \leqslant b \leqslant c$, существует граф G, у которого $k = a$, $\lambda = b$, $c = \delta$.

В работах [8, 9] рассматривается задача о поиске графов с минимальным числом вершин и рёбер для любых a, b, c из теоремы 2. В данной работе решается задача описания графов с заданным числом вершин n и с минимальным числом рёбер для пар возможных значений k и λ.

Обозначим $N_{k,\lambda}$ минимальное число вершин, которое может содержать граф с заданной вершинной связностью k и рёберной связностью λ.

Теорема 3.

$$N_{k,\lambda} = \begin{cases} 2(\lambda+1) - k & \text{при } \lambda > k, \\ \lambda + 1 & \text{при } \lambda = k. \end{cases}$$

Обозначим $E_{k,\lambda}$ минимальное число рёбер, которое может содержать граф с заданной вершинной связностью k и рёберной связностью λ.

Теорема 4.

$$E_{k,\lambda} = \begin{cases} \lambda^2 - k^2 + k + \lambda + \sigma & \text{при } \lambda > k, \\ \lambda(\lambda+1)/2 & \text{при } \lambda = k, \end{cases}$$

где $\sigma = \begin{cases} 0, & \text{если } \lceil (2k^2 - k\lambda - 2k)/2 \rceil \leqslant 0, \\ \lceil (2k^2 - k\lambda - 2k)/2 \rceil & \text{иначе.} \end{cases}$

Очевидно, что построить граф, содержащий заданное число вершин n, с минимальным числом рёбер для заданных значений k и λ можно только при $n \geqslant N_{k,\lambda}$. Если $k = \lambda = 1$, то $N_{1,1} = 2$. Легко видеть, что граф с минимальным числом рёбер для заданного числа вершин n с $k = \lambda = 1$ — дерево с числом рёбер $n - 1$.

Основной результат

Определение 3. *Диагональю порядка i* назовём множество пар (k, λ), удовлетворяющих следующим условиям:

1) $\lambda - k = i$;
2) для заданных k и λ можно построить граф с вершинной связностью k и рёберной связностью λ;
3) граф из условия 2 является либо λ-регулярным, либо одна из его вершин имеет степень $\lambda + 1$, а остальные вершины имеют степени λ;
4) условие 3 должно выполняться для графов с любым числом вершин $n \geqslant N_{k,\lambda}$.

Определение 4. Под парой значений $(k_{\min(i)}, \lambda_{\min(i)})$ будем понимать образующий элемент диагонали порядка i. Образующий элемент — это такая пара значений,

которая удовлетворяет условиям из определения 3 и является наименьшим значением (k, λ) для соответствующей диагонали.

Если есть образующий элемент диагонали порядка i, то можно получить образующий элемент диагонали $i+1$ следующим образом:

$$k_{\min(i+1)} = k_{\min(i)} + 1, \quad \lambda_{\min(i+1)} = \lambda_{in(i)} + 2.$$

По аналогии для $i-1$ диагонали:

$$k_{\min(i-1)} = k_{\min(i)} - 1, \quad \lambda_{\min(i-1)} = \lambda_{\min(i)} - 2.$$

Определение 5. Пару $(2, 2)$ назовём корневым образующим элементом и обозначим (k_r, λ_r).

Корневой образующий элемент является образующим элементом диагонали порядка 0. Остальные образующие элементы диагоналей можно найти по следующей формуле:

$$k_{\min(i)} = k_r + i, \quad \lambda_{\min(i)} = \lambda_r + 2i.$$

Обозначим множество диагоналей через D. Схематично множество D представлено на рис. 1.

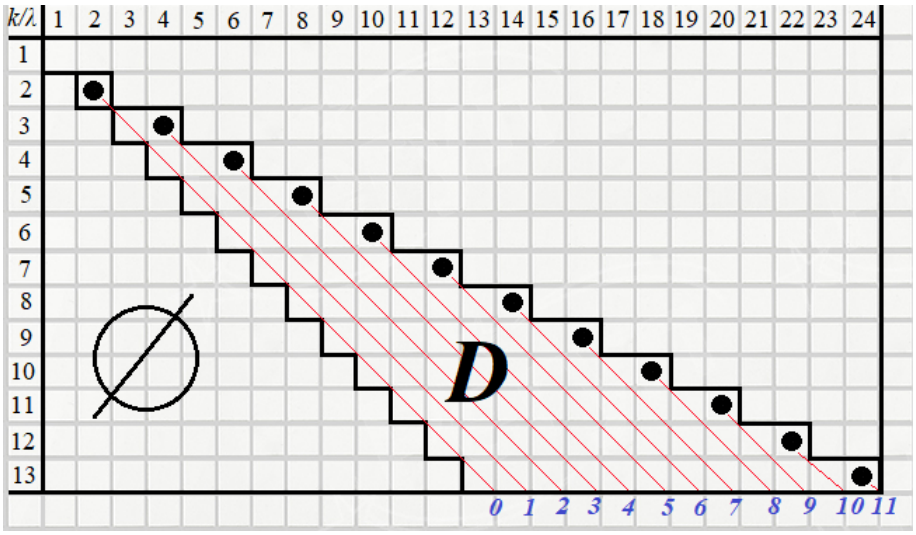


Рис. 1. Схема множества диагоналей D

На рис. 1 точками помечены образующие элементы соответствующих диагоналей. Линиями выделены сами диагонали. Цифрами снизу обозначены порядки диагоналей. В таблице по вертикали идут значения вершинной связности, по горизонтали — рёберной связности. Графы Харари образуют диагональ порядка 0. Напомним, что графом Харари $H_{t,n}$ называется n-вершинный граф с минимальным числом рёбер, у которого $k = \lambda = t$ [2]. Далее приводится основной результат работы, в котором описываются графы из множества D.

Теорема 5. Пусть $k \geqslant k_{\min(i)}$, $\lambda \geqslant \lambda_{\min(i)}$, $\lambda - k = i$, $i \geqslant 0$. Тогда для любого $n \geqslant N_{k,\lambda}$ существует граф G с заданными k и λ, содержащий n вершин, такой, что:

— если λ или n чётные, то G — λ-регулярный граф;

— если λ и n нечётные, то одна из вершин графа G имеет степень $(\lambda+1)$, остальные вершины имеют степени λ.

При этом граф G является оптимальным по рёбрам, то есть состоит из наименьшего возможного числа рёбер, равного $\lceil \lambda n/2 \rceil$.

ЛИТЕРАТУРА

1. *Whitney H.* Congruent graphs and the connectivity of graphs // Amer. J. Math. 1932. V. 54. Iss. 1. P. 150–168.
2. *Harary F.* The maximum connectivity of a graph // Proc. NAS USA. 1962. V. 48. P. 1142–1146.
3. *Chartrand G. and Harary F.* Graphs with prescribed connectivities // Theory of Graphs. N.Y.: Academic Press, 1968. P. 61–63.
4. *Steiglitz K., Weiner P., and Kleitman D.* The design of minimum-cost survivable networks // IEEE Trans. Circuit Theory. 1969. V. 16. No. 4. P. 455–460.
5. *Jafarpour M., Shekaramiz M., Javan A., and Moeini A.* Building graphs with maximum connectivity // Proc. IETS. 2020. P. 1–5.
6. *Харари Ф.* Теория графов М.: Мир, 1973.
7. *Богомолов А. М., Салий В. Н.* Алгебраические основы теории дискретных систем. М.: Наука, 1997.
8. *Теребин Б. А., Абросимов М. Б.* Об оптимальности реализации графов с заданными мерами связности // Прикладная дискретная математика. Приложение. 2020. № 13. С. 103–105.
9. *Теребин Б. А., Абросимов М. Б.* О минимальном числе рёбер в реализациях графов с заданными мерами связности // Компьютерные науки и информационные технологии: Материалы Междунар. науч. конф. 2021. С. 159–161.

СВЕДЕНИЯ ОБ АВТОРАХ

АБРОСИМОВ Михаил Борисович — доктор физико-математических наук, доцент, заведующий кафедрой Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского, г. Саратов. E-mail: **mic@rambler.ru**

АТУТОВА Наталья Дмитриевна — лаборант Института математики им. С. Л. Соболева СО РАН, студентка ММФ НГУ, г. Новосибирск. E-mail: **n.atutova@g.nsu.ru**

БАХАРЕВ Александр Олегович — лаборант Института математики им. С. Л. Соболева СО РАН, студент Новосибирского государственного университета, г. Новосибирск.
E-mail: **a.bakharev@g.nsu.ru**

БЕЛОВ Александр Романович — аспирант Ярославского государственного университета, г. Ярославль. E-mail: **ashmedey@gmail.com**

БОБРОВСКИЙ Дмитрий Александрович — аспирант Финансового университета при Правительстве РФ, старший системный аналитик ООО «Код Безопасности», г. Москва.
E-mail: **dabobrovskiy@gmail.com**

БЫКОВ Денис Александрович — лаборант Института математики им. С. Л. Соболева СО РАН, студент Новосибирского государственного университета, г. Новосибирск.
E-mail: **den.bykov.2000i@gmail.com**

ДЕВЯНИН Петр Николаевич — доктор технических наук, профессор, член-корреспондент Академии криптографии РФ, научный руководитель ООО «РусБИТех-Астра», г. Москва.
E-mail: **pdevyanin@astralinux.ru**

ЕГОРУШКИН Олег Игоревич — кандидат физико-математических наук, доцент Сибирского государственного университета науки и технологий имени академика М. Ф. Решетнёва, г. Красноярск.
E-mail: **olegegoruschkin@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**

ЗАХАРОВ Дмитрий Александрович — студент НИЯУ МИФИ, г. Москва.
E-mail: **zakhar343@yandex.ru**

ЗЮБИНА Дарья Александровна — инженер Института математики им. С. Л. Соболева СО РАН, студентка факультета информационных технологий НГУ, г. Новосибирск.
E-mail: **zyubinadarya@gmail.com**

КОКОРИН Артем Олегович — научный сотрудник ООО «РусБИТех-Астра», г. Москва.
E-mail: **akokorin@astralinux.ru**

КОЛБАСИНА Ирина Валерьевна — старший преподаватель Сибирского государственного университета науки и технологий имени академика М. Ф. Решетнёва, г. Красноярск.
E-mail: **kabaskina@yandex.ru**

КОЛЕСНИКОВ Сергей Геннадьевич — доктор физико-математических наук, доцент, профессор Сибирского государственного университета науки и технологий, профессор Сибирского федерального университета, г. Красноярск. E-mail: **sklsnkv@mail.ru**

КОЛОМЕЕЦ Николай Александрович — кандидат физико-математических наук, научный сотрудник Института математики им. С. Л. Соболева СО РАН, г. Новосибирск.
E-mail: **kolomeec@math.nsc.ru**

КРУГЛОВ Василий Игоревич — кандидат физико-математических наук, научный сотрудник Математического института им. В. А. Стеклова РАН, г. Москва. E-mail: **kruglov@mi-ras.ru**

КУРОЧКИН Алексей Вячеславович — преподаватель МФТИ, сотрудник ООО «Код Безопасности», г. Москва. E-mail: **kurochkin.av@phystech.edu**

КУЦЕНКО Александр Владимирович — кандидат физико-математических наук, научный сотрудник Института математики им. С. Л. Соболева СО РАН, ассистент механико-математического факультета Новосибирского государственного университета, г. Новосибирск.
E-mail: **alexandrkutsenko@bk.ru**
ЛЕОНОВА Мария Александровна — старший научный сотрудник ООО «РусБИТех-Астра», г. Москва. E-mail: **mleonova@astralinux.ru**
ЛЕОНТЬЕВ Владимир Маркович — аспирант Сибирского федерального университета, г. Красноярск. E-mail: **v.m.leontiev@outlook.com**
ЛОБОВ Александр Андреевич — аспирант Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского, г. Саратов. E-mail: **aisanekai@mail.ru**
МЕДВЕДЕВА Наталья Валерьевна — кандидат физико-математических наук, доцент, доцент Уральского государственного университета путей сообщения, г. Екатеринбург.
E-mail: **medvedeva_n_v@mail.ru**
МЕЖЕННАЯ Наталья Михайловна — кандидат физико-математических наук, доцент, доцент кафедры прикладной математики Московского государственного университета им. Н. Э. Баумана, г. Москва. E-mail: **natalia.mezhennaya@gmail.com**
МИХАЙЛОВ Владимир Гаврилович — доктор физико-математических наук, ведущий научный сотрудник отдела дискретной математики Математического института им. В. А. Стеклова Российской академии наук, г. Москва. E-mail: **mikhail@mi-ras.ru**
МОДЕНОВА Ольга Владимировна — доцент Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского, г. Саратов. E-mail: **oginiel@rambler.ru**
МОКРОУСОВ Антон Сергеевич — магистрант факультета информационных технологий Новосибирского государственного университета, г. Новосибирск. E-mail: **settingx@mail.ru**
ПАНКОВ Константин Николаевич — кандидат физико-математических наук, ведущий научный сотрудник НИЛ-24, ВРИО заведующего кафедрой Московского технического университета связи и информатики, эксперт ТК-159 и ISO 307, г. Москва. E-mail: **k.n.pankov@yandex.ru**
ПАНКРАТОВА Ирина Анатольевна — кандидат физико-математических наук, доцент, заведующая лабораторией компьютерной криптографии Национального исследовательского Томского государственного университета, г. Томск. E-mail: **pank@mail.tsu.ru**
ПАРФЕНОВ Денис Романович — студент ФИТ НГУ, г. Новосибирск.
E-mail: **d.parfenov@g.nsu.ru**
ПОГОРЕЛОВ Борис Александрович — доктор физико-математических наук, профессор, действительный член Академии криптографии Российской Федерации, г. Москва.
ПУДОВКИНА Марина Александровна — доктор физико-математических наук, профессор НИЯУ МИФИ, г. Москва. E-mail: **maricap@rambler.ru**
РУБАН Егор Алексеевич — студент Национального исследовательского Томского государственного университета, г. Томск. E-mail: **egor-ruban@mail.ru**
САФОНОВ Константин Владимирович — доктор физико-математических наук, профессор, заведующий кафедрой Сибирского государственного университета науки и технологий имени академика М. Ф. Решетнёва, г. Красноярск. E-mail: **safonovkv@rambler.ru**
СЕМЁНОВ Александр Анатольевич — кандидат технических наук, доцент, ведущий научный сотрудник Института динамики систем и теории управления им. В. М. Матросова СО РАН, г. Иркутск. E-mail: **biclop.rambler@yandex.ru**
СОТОВ Роман Русланович — системный аналитик ООО «Код Безопасности», г. Москва.
E-mail: **imbirnyale@yandex.ru**
СУТОРМИН Иван Александрович — студент Новосибирского государственного университета, младший научный сотрудник Института математики им. С. Л. Соболева СО РАН, г. Новосибирск. E-mail: **ivan.sutormin@gmail.com**

ТЕРЕБИН Богдан Андреевич — аспирант Саратовского национального исследовательского государственного университета имени Н. Г. Чернышевского, г. Саратов.
E-mail: **bogdan.terebin@yandex.ru**
ТИЕВСКИЙ Станислав Дмитриевич — научный сотрудник ООО «РусБИТех-Астра», г. Москва.
E-mail: **stievskiy@astralinux.ru**
ТИТОВ Сергей Сергеевич — доктор физико-математических наук, профессор, профессор кафедры Уральского государственного университета путей сообщения, г. Екатеринбург.
E-mail: **stitov@usaaa.ru**
ТОКАРЕВА Наталья Николаевна — кандидат физико-математических наук, старший научный сотрудник Института математики им. С. Л. Соболева СО РАН, доцент НГУ, г. Новосибирск.
E-mail: **tokareva@math.nsc.ru**
ФОМИЧЕВ Владимир Михайлович — доктор физико-математических наук, профессор, профессор Финансового университета при Правительстве РФ, научный консультант ООО «Код Безопасности», ведущий научный сотрудник ФИЦ ИУ РАН, г. Москва. E-mail: **fomichev.2016@yandex.ru**
ХИЛЬЧУК Ирина Сергеевна — студентка механико-математического факультета НГУ, г. Новосибирск. E-mail: **irina.khilchuk@gmail.com**
ЧИКАЛОВА Светлана Викторовна — студентка Национального исследовательского Томского государственного университета, г. Томск. E-mail: **chikalova.sveta@mail.ru**
ЧУХНО Андрей Борисович — преподаватель кафедры компьютерной безопасности МИЭМ НИУ ВШЭ, эксперт ООО «Код Безопасности», г. Москва. E-mail: **achuhno@hse.ru**
ШАПОРЕНКО Александр Сергеевич — младший научный сотрудник Института математики им. С. Л. Соболева СО РАН, аспирант Новосибирского государственного университета, г. Новосибирск. E-mail: **a.shaporenko@g.nsu.ru**

АННОТАЦИИ ДОКЛАДОВ НА АНГЛИЙСКОМ ЯЗЫКЕ

SECTION 1

Kolomeec N. A., Bykov D. A. **INVARIANT SUBSPACES OF FUNCTIONS AFFINE EQUIVALENT TO THE FINITE FIELD INVERSION.** In the paper, we consider affine $\mathbb{F}_p$-subspaces of a finite field $\mathbb{F}_{p^n}$, p is prime, such that the function x^{-1} which inverses a field element x (we assume that $0^{-1} = 0$) maps them to affine subspaces. It is proven that the image of an affine subspace U, $|U| > 2$, is an affine subspace as well if and only if $U = q\mathbb{F}_{p^k}$, where $q \in \mathbb{F}_{p^n}^*$ and $k|n$. In other words, these subspaces can be expressed using subfields of $\mathbb{F}_{p^n}$. As a consequence, we propose a sufficent condition providing that a function $A(x^{-1}) + b$ has no invariant affine subspaces U of cardinality $2 < |U| < p^n$, where $A : \mathbb{F}_{p^n} \to \mathbb{F}_{p^n}$ is an invertible $\mathbb{F}_p$-linear transformation, $b \in \mathbb{F}_{p^n}^*$. Also, we give examples of functions which have no invariant affine subspaces except for $\mathbb{F}_{p^n}$.
Keywords: *finite fields, inversion, affine subspaces, invariant subspaces.*

Mikhailov V. G., Kruglov V. I. **ASYMPTOTIC NORMALITY OF NUMBER OF MULTIPLE COINCIDENCES OF CHAINS IN COMPLETE q-ARY TREES AND FORESTS WITH RANDOMLY MARKED VERTICES.** We consider complete q-ary trees of height H with vertices marked by random independent marks taking values from some the set $\{1, 2, \ldots, N\}$. The object of our research is the number of tuples of $r \geqslant 2$ paths having the same length s and identical sequences of vertex marks. We propose a theorem on sufficient conditions of asymptotic normality of considered random values for the case when height of the tree tends to infinity. We also investigate repetitions of chains in forest of trees and suppose that there are r trees which may have different heights $H_1, \ldots, H_r$ and vertices of these trees are marked in the same way. We consider the number of tuples of r paths of the same length s and suppose that a tuple includes one path from each tree. For such numbers of tuples, we also propose similar theorem on sufficient conditions of asymptotic normality.
Keywords: *marked trees, chains of marks, chains on a tree, repetitions of chains, conditions of asymptotic normality*

Mikhailov V. G., Mezhennaya N. M. **THE RATE OF NORMAL APPROXIMATION FOR THE DISTRIBUTION OF THE NUMBER OF MULTIPLE REPETITIONS OF CHARACTERS IN A STATIONARY RANDOM SEQUENCE.** We study the asymptotic normality of the number of r-fold characters repetitions in a segment of length n of a strictly stationary random sequence with values in a finite set that satisfies the uniformly strong mixing condition. It is shown that if there exists a number $\alpha > 0$ such that the uniformly strong mixing coefficient $\varphi(t)$ decreases as $t^{-6-\alpha}$, then the distance in the uniform metric between the distribution function of the standardized number of repetitions of multiplicity r and the distribution function of the standard normal law decreases at a rate of $O(n^{-\delta})$ for any $\delta \in (0, \alpha(32 + 4\alpha)^{-1})$ with increasing of segment length n.
Keywords: *multiple repetitions, dependent random variables, uniformly strong mixing, normal approximation, convergence rate estimate.*

Pogorelov B. A., Pudovkina M. A. **DIFFUSION PROPERTIES OF GENERALIZED QUASI-HADAMARD TRANSFORMATIONS ON FINITE ABELIAN GROUPS.** In this paper, we introduce a generalization of quasi-Hadamard transformations on a finite abelian group X. For $X = \mathbb{Z}_{2^m}$, it includes the pseudo-Hadamard transformation employed in block ciphers Safer and Twofish, and the quasi-Hadamard transformations proposed by H. Lipmaa. For bijective generalized quasi-Hadamard transformations, we describe diffusion properties of imprimitivity systems of regular permutation representations of additive groups $\mathbb{Z}_{2^m}^2$ and $\mathbb{Z}_{2^{2m}}$. We describe a set of generalized quasi-Hadamard transformations having the best diffusion properties of the imprimitivity systems. We also give conditions such that some generalized quasi-Hadamard transformations have bad diffusion properties.

Keywords: *Safer block cipher family, Twofish block cipher, pseudo-Hadamard transformation, quasi-Hadamard transformation, imprimitivity system, regular permutation representation, primitive group.*

SECTION 2

Atutova N. D. **APPLICATION OF HEURISTIC METHODS TO SEARCH FOR BOOLEAN FUNCTIONS WITH GOOD CRYPTOGRAPHIC CHARACTERISTICS.** Currently, one of the most promising and developing methods of cipher analysis is linear and algebraic cryptanalysis. To ensure resistance to this type of attack, it is necessary to use Boolean functions with high nonlinearity and algebraic immunity when constructing components of block and stream ciphers. We propose a combined approach to the search for Boolean functions with nonlinearity and algebraic immunity based on heuristic methods, in particular, a genetic algorithm and a hill climbing algorithm. Computational experiments have been carried out for Boolean functions in $n \leqslant 8$ variables, which demonstrated the effectiveness of the proposed approach, as well as a comparative analysis of the results obtained by random search. On the basis of the obtained Boolean functions, vector Boolean functions are constructed and among them the number of functions with the component algebraic immunity and nonlinearity is calculated.

Keywords: *genetic algorithm, Hill Climbing algorithm, algebraic immunity, nonlinearity, heuristics.*

Bykov D. A. **LOWER BOUND FOR THE NUMBER OF BENT FUNCTIONS AT THE MINIMUM DISTANCE FROM MAJORANA — MCFARLAND BENT FUNCTIONS.** The construction of bent functions at a certain distance from a given bent function is investigated. The criterion that the function obtained from the bent function f by adding an indicator of an affine subspace of dimension n is a bent function is proven, where f belongs to the Maiorana — McFarland class $\mathcal{M}_{2n}$. It is shown that the lower bound $2^{2n+1} - 2^n$ for the number of bent functions at the minimum distance from a bent function from the class $\mathcal{M}_{2n}$ is attained for prime $n \geqslant 5$. Bent functions are found for which the lower bound is attainable. It is shown that this lower bound is not attained for bent functions from the class $\mathcal{M}_{2n}$, where the permutation is not an APN function. For some distances, in particular 2^{2n-1}, lower bounds for the number of bent functions in the class $\mathcal{M}_{2n}$ at these distances from bent functions in the class $\mathcal{C}$ are obtained.

Keywords: *bent functions, boolean functions, minimum distance, Maiorana — McFarland class, lower bounds.*

Kutsenko A. V. **PROPERTIES OF SUBFUNCTIONS OF SELF-DUAL BENT FUNCTIONS.** Boolean functions in an even number of variables with flat Walsh —

Hadamard spectrum are called bent functions. For every bent function, say f, its dual bent function, denoted by $\widetilde{f}$, is uniquely defined. If $\widetilde{f} = f$, then f is called *self-dual bent*, and in the case $\widetilde{f} = f \oplus 1$ it is called an *anti-self-dual bent*. In this paper, we study subfunctions of self-dual bent functions obtained by a fixation of the first and the first two coordinates. We characterize subfunctions in $n-1$ variables considering their Rayleigh quotients. A sufficient condition for all subfunctions in $n-2$ variables to be bent is obtained. We propose new iterative constructions of self-dual bent functions in n variables comprising the usage of bent functions in $n-4$ variables. Based on them, a new iterative lower bound on the cardinality of the set of self-dual bent functions is obtained.
Keywords: *self-dual bent function, subfunction, near-bent function, Rayleigh quotient of the Sylvester Hadamard matrix.*

Pankratova I. A., Ruban E. A., Chikalova S. V. **CONSTRUCTING VECTOR BOOLEAN FUNCTIONS WITH NON-DEGENERATE COORDINATE FUNCTIONS.** An algorithm for constructing a bijection on $\mathbb{Z}_2^n$ with coordinate functions depending essentially on all variables is proposed. The algorithm consists of three steps: generation of a random bijection $F : \mathbb{Z}_2^n \to \mathbb{Z}_2^n$, analysis of the degeneracy of its coordinates, and one transposition of the values of $F(x)$ and $F(y)$ such that they differ only in the positions corresponding to the degenerate coordinate functions.
Keywords: *vector Boolean function, bijection, essential dependence of a function on a variable.*

Khilchuk I. S., Zyubina D. A., Tokareva N. N. **CORRELATION-IMMUNE FUNCTIONS WITH OPTIMAL ALGEBRAIC IMMUNITY.** Boolean functions are the main components of symmetric ciphers, and their properties ensure the cipher's resistance to various types of cryptanalysis. An important problem is to combine several cryptographic properties in one function, since the properties may contradict each other. Also, an interesting way to build Boolean functions is an iterative construction, i.e., constructing functions in a larger number of variables based on functions in a smaller number while preserving cryptographic properties. In this paper, we intersect sets of functions with maximal algebraic immunity and functions with the maximal order of correlation immunity equal to one, of a small number of variables. There are no correlation-immune Boolean functions in 3 variables with maximal algebraic immunity. There are 392 functions in 4 variables with the maximal order of correlation immunity 1 and maximal algebraic immunity, and for the case of 5 variables there are 96 768 such functions. For functions in 4 variables, a classification is obtained based on their Hamming weight and the type of their geometric representation. The construction of functions in 6 variables has been studied on the basis of functions in 4 variables, in which each vertex of the Boolean cube $\mathbb{E}^4$ is replaced by a face of dimension 2 containing elements of the support of the 6-variable function only if the original vertex belonged to the support. It has been programmatically verified that this construction preserves the indices of algebraic and correlation immunity.
Keywords: *Boolean functions, algebraic immunity, correlation immunity, Boolean cube.*

Shaporenko A. S. **ON DECOMPOSITION OF BENT FUNCTIONS IN 8 VARIABLES INTO THE SUM OF TWO BENT FUNCTIONS.** A Boolean function in an even number of variables is called bent if it has maximal nonlinearity. We study the well-known hypothesis about the representation of arbitrary Boolean functions in n variables of degree at most $n/2$ as the sum of two bent functions. We prove that bent functions in 8 variables of degree at most 3 can be represented as the sum of two bent functions in 8 variables. It was shown that all quadratic Boolean functions in an even number of

variables $n \geqslant 4$ can be represented as the sum of two bent functions of a special form.
Keywords: *Boolean functions, bent functions, decomposition into sum of bent functions.*

SECTION 3

Bakharev A. O. **DEVELOPMENT AND COMPARISON OF QUANTUM ORACLE MODELS FOR THE HYBRID ATTACK ON POST-QUANTUM LATTICE-BASED CRYPTOSYSTEMS.** Every year, quantum computing is developing with increasing force. Therefore, there is a need to design and analyze cryptosystems that will be resistant to attacks using a quantum computer. The security of many well-known post-quantum lattice-based cryptosystems depends on the complexity of solving the shortest vector problem (SVP). One of the most efficient algorithms to solve this problem is the GaussSieve algorithm. For the previously proposed model of the quantum oracle used in the hybrid quantum-classical algorithm for solving SVP, new updated bounds of the number of qubits and the depth of the scheme are obtained. This improvement is achieved by optimizing all quantum operations with integers used in the model. A new quantum oracle model is proposed and analyzed, which uses classical memory to store a list of vectors. This model requests the number of qubits that is logarithmic to the length of the list in the GaussSieve algorithm. Upper bounds on the complexity of the attack of post-quantum cryptosystems that have become finalists of the NIST competition have been obtained. These upper bounds indicate that the exponential length of the list from the GaussSieve algorithm imposes limitations on the possibility of implementing hybrid attacks on the known standards of post-quantum cryptosystems.
Keywords: *quantum search, public-key cryptography, post-quantum cryptography.*

Zakharov D. A., Pudovkina M. A. **ON A SET OF IMPOSSIBLE DIFFERENCES OF FEISTEL CIPHERS WITH A NON-BIJECTIVE TRANSFORM OF A ROUND FUNCTION.** In this paper, a family of l-round balanced Feistel ciphers with non-bijective combining functions is being considered. For any such cipher, the existence of impossible differentials for an arbitrary number of rounds l is proved, and a lower estimate of the number of described impossible differentials is obtained. The GRANULE block cipher belongs to the family under consideration, for which a new approach for finding impossible differences is proposed. Its superiority, in comparison with other previously known approaches, is shown both in terms of the number of impossible differences found and in terms of the number of rounds. Experimental confirmation of the theoretical estimate of the number of impossible differences has been obtained.
Keywords: *balanced Feistel cipher, impossible differentials, non-bijective function, distinguish attack, GRANULE block cipher.*

Medvedeva N. V., Titov S. S. **THE CRITERION OF MINIMUM PERFECT CIPHERS WITH RESPECT TO INCLUSION.** The paper deals with the problem of Shannon perfect ciphers description (which are absolutely immune against the attack on ciphertext, according to Shannon), minimal by inclusion. The criterion of minimum non-endomorphic (endomorphic) perfect ciphers by inclusion is formulated and proved. The table of encryption of a perfect cipher with $\lambda > 1$ ciphers, $\mu \geqslant \lambda$ ciphers and $\pi \geqslant \mu$ keys is considered. For a given cipher, $(0, 1)$-matrix with π rows and $1 + \lambda\mu$ columns is constructed in a natural way. It is shown that the set of encryption keys is minimal if and only if the matrix rank is maximal and equals to π. The necessary conditions for perfect ciphers of minimum by inclusion have been obtained.
Keywords: *perfect ciphers, endomorphic ciphers, non-endomorphic ciphers.*

Mokrousov A. S. **CALCULATION OF THE DIFFERENTIAL PROBABILITIES FOR THE SUM OF k NUMBERS MODULO 2^n.** We study the differential probabilities $\mathrm{xdp}^+_{\mathrm{k}}(\alpha^1, \ldots, \alpha^k \to \alpha^0)$ of the function $f(x_1, \ldots, x_k) = x_1 + \cdots + x_k \mod 2^n$, $\alpha^0, \alpha^1, \ldots, \alpha^k \in \mathbb{Z}_2^n$, where differences are expressed using bitwise "exclusive or". These values are used in differential cryptanalysis of cryptographic primitives which contain bitwise "exclusive or" and addition modulo 2^n, such as ARX-constructions. We propose analytic expressions of matrices that are used for calculating $\mathrm{xdp}^+_{\mathrm{k}}$. We also study the differential probability $\mathrm{adp}^{\oplus}(\alpha, \beta \to \gamma)$ of the function $x \oplus y$, $\alpha, \beta, \gamma \in \mathbb{Z}_2^n$, where differences are expressed using addition modulo 2^n, and describe all triples of differences whose probabilities are greater than $1/4$.
Keywords: *ARX, exclusive or, modular addition, differential cryptanalysis, differential probabilities.*

Pankov K. N. **SOME CONDITIONS FOR THE APPLICABILITY OF THE INTEGRAL CRYPTANALYSIS TO 4-ROUNDS OF AES-LIKE CIPHERS.** A number of necessary conditions and one sufficient condition are obtained so that the integral cryptanalysis can be applied to block algorithms constructed similarly to the AES algorithm (i.e., SQUARE, Rijndael, Crypton) with a reduced number of rounds to four, which are denoted as f_1, f_2, f_3, f_4. For example, it was proved that if we consider the multiset $\{y_j(x) \in V_8 : x \in I_i\}$, where $I_i = \{(B_0, \ldots, B_{i-1}, z, B_{i+1}, \ldots, B_{15}) : z \in V_8\}$, the subvector $B_i = z$, $i \in \{0, \ldots, 15\}$, takes all possible values from V_8, and the other data block subvectors are fixed, $(y_0(x), \ldots, y_{15}(x)) = f_4 \circ f_3 \circ f_2 \circ f_1(x, k_0^*)$, k_0^* is the true key, then a necessary condition for obtaining information about the fourth round key $k_{4,j}$ by the integral method is: the subset $Y_j^* = \{\alpha \in V_8 : |\{x \in I_i : y_j(x) = \alpha\}| = 2k-1, k \in \mathbb{N}\}$ is not empty. The experimental data on the application of the integral method to the Rijndael algorithm are presented.
Keywords: *block cipher, AES, SQUARE, Rijndael, Crypton, spectral coefficient, integral cryptanalysis.*

Parfenov D. R., Bakharev A. O., Kutsenko A. V., Belov A. R., Atutova N. D. **XS-CIRCUITS' PROPERTIES RELATED TO THE GUARANTEED NUMBER OF ACTIVATIONS.** The guaranteed number of activations (GNA) is an important characteristic that determines the efficiency of differential cryptanalysis of a given XS-circuit. In the paper, we propose an approach to optimize the known GNA calculation algorithm based on the branch and bound method and the analysis of special matrices that define the XS-circuit. Now, it is possible to compute GNA for more than 30 rounds, which would take significantly longer if the original algorithm were used. The optimized algorithm was used for exhaustive enumeration of low-dimensional XS-schemes. We prove that the canonical forms of the XS-circuit and its dual coincide, which provides a strict connection between the guaranteed number of linear and differential activations. Based on computational experiments, several hypotheses have been proposed. One of the hypotheses is that there are no XS-circuits of dimension greater than two that achieve an optimal GNA in every round.
Keywords: *guaranteed number of activations, XS-circuit, differential cryptanalysis, linear cryptanalysis, branch and bound method.*

Sutormin I. A. **THE ADDITIVE DIFFERENTIAL PROBABILITY OF k SUCCESSIVE EXCLUSIVE-OR.** We study the additive differential probability of a com-

position of bitwise XOR — $\mathrm{adp}_k^{\oplus}$. For a set of vectors $\alpha^1 \dots \alpha^{k+1} \in \mathbb{Z}_2^n$, it is defined as

$$\mathrm{adp}_k^{\oplus}(\alpha^1, \dots, \alpha^k \to \alpha^{k+1}) = \frac{1}{2^{kn}} \Big| \{x^1, \dots, x^k \in \mathbb{Z}_2^n : \bigoplus_{i=1}^{k} (x^i \boxplus \alpha^i) = \alpha^{k+1} \boxplus \bigoplus_{i=1}^{k} x^i\} \Big|.$$

It is used when analyzing symmetric key primitives, such as Addition-Rotation-XOR constructions. It is proven that: 1) $\mathrm{adp}_k^{\oplus}$ is a symmetric function; 2) the value of $\mathrm{adp}_k^{\oplus}$ does not change if 2^{n-1} is added to any two arguments; 3) the value of $\mathrm{adp}_k^{\oplus}$ does not change if any two arguments are replaced by the opposite element of $\mathbb{Z}_2^n$, and if k is even, then the value of $\mathrm{adp}_k^{\oplus}$ does not change if any argument is replaced. Recurrence formulas are obtained allowing to calculate the value $\mathrm{adp}_k^{\oplus}$ for arguments of dimension $n+1$ using the set of values $\mathrm{adp}_k^{\oplus}$ for arguments of dimension n. Using recurrent formulas we prove that $\mathrm{adp}_k^{\oplus}$ is equal to zero if and only if there exists a position i such that $(\alpha_i^1, \dots, \alpha_i^{k+1}) \neq (0, \dots, 0)$; $(\alpha_j^1, \dots, \alpha_j^{k+1}) = (0, \dots, 0)$ for any j, $n \geqslant j > i$, and either Hamming weight of the vector $(\alpha_i^1, \dots, \alpha_i^{k+1})$ is odd, or k is odd, $i > 1$, vector $(\alpha_i^1, \dots, \alpha_i^{k+1})$ is equal to the $(1, \dots, 1)$ and Hamming weight of the vector $(\alpha_{i-1}^1, \dots, \alpha_{i-1}^{k+1})$ is odd. In the case of even k, it is proved that $\mathrm{adp}_k^{\oplus}(0, \dots, 0, \gamma \to \gamma)$ is maximal when the last argument is fixed. In the case of odd k, it is conjectured that $\mathrm{adp}_k^{\oplus}(\gamma, \dots, \gamma \to \gamma)$ is maximum when the last argument is fixed.
Keywords: *differential cryptanalysis, ARX, XOR, modular addition.*

Fomichev V. M., Bobrovskiy D. A., Sotov R. R. **KEY SCHEDULE BASED ON A MODIFIED ADDITIVE GENERATOR.** A method of round key generation for iterated block ciphers based on a modified additive generator (MAG), and, in addition, on MAG and a linear congruent generator in a series circuit is proposed. The bijectivity of the generating transformation is demonstrated. Using the matrix-graph approach the number of iterations necessary for achieving enhanced cryptographic properties is experimentally evaluated. This number depends on the generator characteristics.
Keywords: *key scheduling algorithm, iterative block ciphers, modified additive generator, mixing properties, nonlinearity.*

Fomichev V. M., Kurochkin A. V., Chukno A. B. **THE DIFFERENCE RELATIONS AND IMPOSSIBLE DIFFERENTIALS CONSTRUCTION FOR THE KB-256 ALGORITHM.** New results of the analysis of the KB 256-3 block cipher algorithm are outlined. We set up a difference relation with probability 1 for the six-round algorithm under study and propose a key recovery method using this difference relation for the nine-round KB 256-3 algorithm. We construct an impossible differential for the full-round algorithm.
Keywords: *differential cryptanalysis, impossible differentials.*

SECTION 4

Egorushkin O. I., Kolbasina I. V., Safonov K. V. **POLYNOMIAL GRAMMARS GENERATING AN INFINITE SET OF LANGUAGES.** We continue the study of formal grammars, by which we mean systems of polynomial equations with respect to noncommutative variables, which are solved in the form of formal power series expressing nonterminal alphabet symbols through terminal alphabet symbols. The first component of the solution is a formal language. The definition of a grammar having infinitely many solutions (generating an infinite number of languages) is considered. Such grammars can arise in a situation, where the Jacobian of the commutative image of the grammar is identically equal to zero. It is shown that in the case of the Jacobian, which is identically equal to zero, it is more difficult to describe the set of grammatical solutions than for similar polynomial

systems with real or complex variables, since all possible situations can be realized: such a grammar may have infinitely many solutions, any finite number of solutions, or no solutions at all.
Keywords: *polynomial grammars, noncommutative variables, formal power series, commutative image, Jacobian.*

Kokorin A. O., Tievskiy S. D., Devyanin P. N. **METHODS FOR DEDUCTIVE VERIFICATION OF C CODE USING AstraVer Toolset.** Some practical methods for deductive verification of C code for compliance with ACSL specifications are described. For verification, Astraver Toolset based on the Frama-C platform is used. Approbation of these methods was carried out during the verification of access control module in PARSEC security subsystem of secure operating system Astra Linux Special Edition. For example, the method of global ghost variables allows monitoring shared memory, this is helpful for verification of certain functions. There is also a specification validation method that can help find out if the written specification is lacking. Thanks to these methods, it is possible to simplify function specifications, reduce labour intensity and speed up deductive verification. Examples of the use of the proposed methods are given.
Keywords: *deductive software verification, ACSL, Frama-C, AstraVer Toolset, Astra Linux.*

Leonova M. A., Devyanin P. N. **COMPARISON OF METHODS FOR MODELING ACCESS CONTROL IN OS AND DBMS IN Event-B FOR THE PURPOSE OF THEIR VERIFICATION WITH Rodin AND ProB TOOLS.** The paper presents two methods of modeling interacting systems with developed access control mechanisms, such as OS and DBMS. These methods are the result of translating the description of the formal access control model of Astra Linux Special Edition OS (MROSL DP-model) from mathematical to formalized notation using the formal Event-B method, as well as its automatic verification using Rodin and ProB tools. The considered methods are based on the use of various options for constructing a hierarchy of specifications of the MROSL DP-model in the formalized notation using the technique “refinement”. We compare these methods showing their advantages and disadvantages. They consist in the complexity of writing specifications, the need to repeat proofs during the verification with the Rodin tool, the possibility of eliminating the effect of “combinatorial explosion” during the verification with the ProB tool. Based on the results of the comparison, it is concluded that considered methods can be useful in the development of other formal access control models and their verification using appropriate tools.
Keywords: *formal access control model, Event-B, verification, deductive verification, Rodin, model checking, ProB.*

Semenov A. A. **BACKDOORS IN COMBINATORIAL PROBLEMS AND THEIR PROBABILISTIC GENERALIZATIONS.** We present some recent results about the structures which arise in many combinatorial problems under the term “Backdoors”. A backdoor for a constraint satisfaction problem is a set of variables, the knowledge of which makes it possible to solve the original problem more efficiently than without knowing a backdoor. In the past few years, backdoors have become quite a popular subject of research. They are actively investigated in both theoretical and practical areas of computer science. We will start from some simple modifications of known results about backdoors. In particular, we present one refinement of the well-known result from the paper by R. Williams, C. Gomes and B. Selman (2003) about the worst-case complexity estimation of SAT using strong backdoors. Also, we discuss a probabilistic generalization of a

strong backdoor notion and show that this concept can improve the efficiency of algorithms applied to hard instances (both industrial and crafted ones) from Boolean Satisfiability (SAT) and 0-1-Integer Linear Programming (0-1-ILP). In particular, we present the results of computational experiments which demonstrate the ability of probabilistic backdoors to significantly speed up the solution of the aforementioned hard combinatorial problems.
Keywords: *backdoors in combinatorial problems, Boolean Satisfiability Problem (SAT), 0-1-Integer Linear Programming.*

SECTION 5

Zharkova A. V. **THE FINITE DYNAMIC SYSTEM OF ALL POSSIBLE ORIENTATIONS OF A GIVEN GRAPH WITH ALL ACCESSIBLE STATES AND WITH INACCESIBLE STATES.** The finite dynamic systems (Γ_G, α) is considered, the states of which are all possible orientations of a given graph G, and the evolutionary function α transforms a given state $\overrightarrow{G}$ by reversing all arcs in $\overrightarrow{G}$ that enter into sinks, and there are no other differences between the given $\overrightarrow{G}$ and the next $\alpha(\overrightarrow{G})$ states. We characterize the systems in which all states are accessible and in which there are inaccessible states. Namely, in a finite dynamic system (Γ_G, α) all states are accessible if and only if the (connected) components in the graph G are complete graphs with n vertices for $1 \leqslant n \leqslant 3$ and only they. Otherwise, the system under consideration has inaccessible states. The number of graphs forming systems with all accessible states is counted. Namely, the number of graphs G with n vertices that form finite dynamic systems (Γ_G, α), all states of which are accessible, is equal to $1 + \lfloor n/2 \rfloor + \lfloor n/3 \rfloor + \sum\limits_{i=1}^{\lfloor (n-3)/2 \rfloor} \lfloor (n-2i)/3 \rfloor$. The table is given with the number of graphs with the number of vertices from 1 to 12, forming systems with all accessible states and with inaccessible states.
Keywords: *accessible state, directed graph, evolutionary function, finite dynamic system, fault-tolerance, graph, inaccessible state.*

Kolesnikov S. G., Leontiev V. M. **A SERIES OF FORMULAS FOR BHATTACHARYA PARAMETERS IN THE THEORY OF POLAR CODES.** In the theory of polar codes, the Bhattacharya parameters are used to determine the positions of frozen and information bits. The parameters characterize the polarization rate of the channels $W_N^{(i)}$ constructed in a special way from the original channel W, here $1 \leqslant i \leqslant N$, $N = 2^n$, and $n = 1, 2, \ldots$ is the length of the code. It is assumed that the i-th bit of a message is transmitted over the channel $W_N^{(i)}$, and the Bhattacharya parameter $Z(W_N^{(i)})$ can be interpreted as the noise level of $W_N^{(i)}$. W is a model of a physical transmission channel. If W is a classical binary memoryless symmetric channel, the currently known formulas for the Bhattacharya parameters contain $2^N = 2^{2^n}$ terms. We have obtained the formulas for the series of channels $W_N^{(N-2^k+1)}$, $k = 0, 1, \ldots, n-1$, that contain $2^{(n-k+1)2^k}$ terms. Some assumptions are also given for further simplification of the obtained formulas.
Keywords: *polar code, Bhattacharya parameter.*

Lobov A. A., Abrosimov M. B. **ABOUT THE UNIQUENESS OF THE MINIMAL 1-EDGE EXTENSION OF A HYPERCUBE.** A graph G^* is a k-edge extension of a graph G if every graph obtained by removing any k edges from G^* contains G. A k-edge extention G^* of a graph G is said to be minimal if it contains n vertices, where n is the number of vertices in G, and G^* has the minimum number of edges among all k-edge extensions of the graph G with n vertices. The hypercube Q_n is a regular 2^n-vertex graph

of order n, which is the Cartesian product of n complete 2-vertex graphs K_2. We propose a family of graphs Q_n^* whose representatives for $n > 1$ are minimal 1-edge extensions of the corresponding hypercubes. The computational experiment showed that for $n \leqslant 4$ these extensions are unique up to isomorphism. In this paper, we succeeded in obtaining an analytical proof of the uniqueness of minimal 1-edge extensions of hypercubes for $n \leqslant 4$, as well as establishing one general property of an arbitrary minimal 1-edge extension of a hypercube for $n > 2$.

Keywords: *graph, hypercube, edge fault tolerance, minimal 1-edge extension.*

Modenova O. V., Abrosimov M. B. **THE UPPER AND LOWER BOUNDS FOR THE NUMBER OF ADDITIONAL ARCS IN A MINIMAL EDGE 1-EXTENSION OF ORIENTED CYCLE.** A k-edge extension of a graph G with n vertices is minimal if it has n vertices and contains the minimum number of edges or arcs among all k-edge extensions of G with n vertices. Minimal edge 1-extensions of cycles are well studied. In this paper, we consider minimal edge 1-extensions of cycle orientations. We study the upper and lower bounds for the number of additional arcs $\mathrm{ec}(C_n)$ of a minimal edge 1-extension of the oriented cycle C_n. The main result is an estimate for the number of additional arcs: $\lceil n/2 \rceil \leqslant \mathrm{ec}(C_n) \leqslant n$. Examples of cycle orientations on which the upper and lower bounds are achieved are given.

Keywords: *minimal edge extension, cycle orientation, fault-tolerance.*

Terebin B. A., Abrosimov M. B. **ONE FAMILY OF OPTIMAL GRAPHS WITH PRESCRIBED CONNECTIVITIES.** The vertex connectivity k is the smallest number of vertices whose removal leads to a disconnected or trivial graph. The edge connectivity λ of a nontrivial graph is the smallest number of edges whose removal leads to a disconnected graph. In this paper, we study n-vertex graphs that are minimal in terms of the number of edges and have given values of vertex and edge connectivity. In addition to theoretical interest, graphs with given values of vertex or edge connectivity are also of applied interest as models of fault-tolerant networks. The main result is that, for a certain range of values of k and λ, we describe the graphs that, for a given n, have the minimum number of edges $\lceil \lambda n/2 \rceil$. The corresponding graph is either regular of order λ or has one vertex of degree $\lambda + 1$, and the remaining vertices of degree λ.

Keywords: *graph, vertex connectivity, edge connectivity, fault tolerance.*