МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

# ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА

*Научный журнал*

2012 №3(17)

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

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**Адрес редакции:** 634050, г. Томск, пр. Ленина, 36
**E-mail:** vestnik_pdm@mail.tsu.ru

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

Периодичность выхода журнала: 4 номера в год.

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

---

Подписано к печати 10.09.2012.
Формат $60 \times 84\frac{1}{8}$. Усл. п. л. 13. Уч.-изд. л. 14,5. Тираж 300 экз.

---

Издательство ТГУ. 634029, Томск, ул. Никитина, 4
Отпечатано в типографии ТПУ.

# СОДЕРЖАНИЕ

## ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ



## МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ



## МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ



## МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ



## ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ



## ПРИКЛАДНАЯ ТЕОРИЯ ГРАФОВ



## ДИСКРЕТНЫЕ МОДЕЛИ РЕАЛЬНЫХ ПРОЦЕССОВ

# CONTENTS

# ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

DOI 10.17223/20710410/17/1

УДК 519.7

## ЛИНЕЙНАЯ СЛОЖНОСТЬ ОБОБЩЁННЫХ ЦИКЛОТОМИЧЕСКИХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ С ПЕРИОДОМ $2^m p^n$

В. А. Едемский, О. В. Антонова

*Новгородский государственный университет, г. Великий Новгород, Россия*

**E-mail:** Vladimir.Edemsky@novsu.ru

Предлагается метод анализа линейной сложности обобщённых циклотомических последовательностей с периодом $2^m p^n$, позволяющий выделять последовательности с высокой линейной сложностью. Вычисляется линейная сложность ряда последовательностей на основе классов квадратичных и биквадратичных вычетов.

**Ключевые слова:** *обобщённые циклотомические последовательности, линейная сложность.*

### Введение

Линейная сложность бинарной последовательности является важным показателем её качества и определяется как длина самого короткого линейного регистра сдвига с обратной связью, который может воссоздать последовательность, т. е. если $X = \{x_i : i = 0, 1, \ldots\}$ — последовательность с периодом $N$ над полем $\mathrm{GF}(2)$, то её линейная сложность определяется как наименьшее натуральное $L$, для которого существуют константы $c_1, \ldots, c_L \in \mathrm{GF}(2)$, такие, что $x_i = c_1 x_{i-1} + c_2 x_{i-2} + \ldots + c_L x_{i-L}$ для всех $i$, $L \leqslant i < N$ [1]. Последовательности, обладающие высокой линейной сложностью, важны для криптографических приложений. Известны алгоритмы определения линейной сложности последовательности, например алгоритм Берлекэмпа — Месси [1]. В то же время для ряда периодических последовательностей, сформированных на основе классов степенных вычетов, линейная сложность определяется видом периода последовательности [2].

Построение последовательностей на основе классов степенных вычетов (циклотомических классов) по модулю $N$ является одним из широко применяемых методов выработки последовательностей. Такие последовательности называются обобщёнными циклотомическими, если $N$ составное [2]. В настоящей работе предлагается метод анализа линейной сложности ряда обобщённых циклотомических последовательностей с периодом $N = 2^m p^n$, где $p$ — нечётное простое число, а $m, n$ — натуральные числа. Вычисляется линейная сложность последовательностей на основе классов квадратичных и биквадратичных вычетов. Приведённые примеры показывают, что разработанный метод позволяет выделять последовательности с высокой линейной сложностью. Ранее линейная сложность отдельных последовательностей на основе классов квадратичных вычетов с периодом $2p^n$ и биквадратичных вычетов с периодом $2p$ исследовалась в [3 – 6].

## 1. Основные определения

Пусть $p$ — нечётное простое число и $d$ — натуральный делитель $p-1$, $d \geqslant 2$. Хорошо известно, что всегда существует первообразный корень $\theta$ по модулю $p^n$ [7]. Обозначим через $H_0 = \{\theta^{td} \bmod p^n : t = 0, \ldots, p^{n-1}(p-1)/d - 1\}$ класс вычетов степени $d$ по модулю $p^n$. Если $A$ — подмножество $\mathbb{Z}_{p^n}$, то через $tA$ будем обозначать множество $tA = \{ta \bmod p^n : a \in A\}$, где $t \in \mathbb{Z}$.

Пусть $H_k = \theta^k H_0$ для $k = 0, 1, \ldots, d-1$. Классы вычетов $H_k$ образуют разбиение множества обратимых элементов кольца $\mathbb{Z}_{p^n}$ и являются обобщёнными циклотомическими классами.

Положим $C_k = \bigcup\limits_{j=0}^{n-1} p^j H_k$, тогда

$$\mathbb{Z}_{p^n} = \bigcup_{k=0}^{d-1} C_k \cup \{0\}. \tag{1}$$

Кольцо классов вычетов $\mathbb{Z}_N$, $N = 2^m p^n$, изоморфно прямому произведению $\mathbb{Z}_{2^m} \otimes \mathbb{Z}_{p^n}$ относительно изоморфизма $\varphi(a) = (a \bmod 2^m, a \bmod p^n)$ [7]. Пусть $H_{l,k} = \varphi^{-1}(\{l\} \otimes C_k)$ для $l = 0, 1, \ldots, 2^m - 1$, $k = 0, 1, \ldots, d-1$, тогда, согласно (1), справедливо разбиение

$$\mathbb{Z}_N = \bigcup_{l=0}^{2^m-1} \bigcup_{k=0}^{d-1} H_{l,k} \cup \{0, p^n, \ldots, (2^m-1)p^n\}.$$

Цель работы заключается в исследовании линейной сложности характеристических последовательностей различных объединений классов вычетов $H_{l,k}$. Для каждого $l = 0, 1, \ldots, 2^m - 1$ зададим $I_l$ — подмножества индексов, элементы которых могут принимать значения от 0 до $d-1$, то есть $I_l \subset \{0, 1, \ldots, d-1\}$. Введем множество $C = \bigcup\limits_{l=0}^{2^m-1} \bigcup\limits_{k \in I_l} H_{l,k} \cup \{0, 2p^n, \ldots, (2^m-2)p^n\}$ и рассмотрим последовательность $X$, определяемую следующим образом:

$$x_i = \begin{cases} 1, & \text{если } i \bmod N \in C, \\ 0 & \text{иначе.} \end{cases} \tag{2}$$

Так как, согласно определению, $x_i = x_{i \bmod N}$, то $N$ является периодом последовательности $X$.

Далее рассмотрим метод вычисления линейной сложности последовательностей, определяемых формулой (2), и проиллюстрируем его на примерах.

## 2. Метод анализа линейной сложности последовательностей с периодом $2^m p^n$

Хорошо известно (см., например, [1]), что если $S(t)$ — производящая функция цикла последовательности, то есть $S(t) = x_0 + x_1 t + \ldots + x_{N-1} t^{N-1}$, $S(t) \in \mathrm{GF}(2)[t]$, то её минимальный многочлен $m(t)$ и линейную сложность $L$ можно определить по следующим формулам:

$$m(t) = (t^N - 1)/\mathrm{НОД}(t^N - 1, S(t)), \quad L = N - \deg[\mathrm{НОД}(t^N - 1, S(t))]. \tag{3}$$

В нашем случае, в кольце $\mathrm{GF}(2)[t]$, справедливо разложение $t^{2^m p^n} - 1 = (t^{p^n} - 1)^{2^m}$, то есть

$$L = N - \deg[\mathrm{НОД}\left((t^{p^n} - 1)^{2^m}, S(t)\right)]. \tag{4}$$

Обозначим через $\alpha$ примитивный корень степени $p^n$ из единицы в расширении поля GF(2); тогда, согласно формулам (3) и (4), для вычисления минимального многочлена и линейной сложности последовательности $X$ достаточно найти корни многочлена $S(t)$ в множестве $\{\alpha^v : v = 0, 1, \ldots, p^n - 1\}$ и определить их кратность.

Согласно определению последовательности, имеем

$$S(\alpha^v) = \sum_{l=0}^{2^m-1} \sum_{k \in I_l} \sum_{u \in H_{l,k}} \alpha^{vu} + \sum_{j=0}^{2^{m-1}-1} \alpha^{jv2p^n}. \tag{5}$$

**Лемма 1.** Для любых $l = 0, 1, \ldots, 2^m - 1$ и $k = 0, 1, \ldots, d - 1$ справедливо равенство $\sum\limits_{u \in H_{l,k}} \alpha^u = \sum\limits_{w \in C_k} \alpha^w$.

***Доказательство.*** По определению $\alpha$ имеем, что $\alpha^u = \alpha^{u \bmod p^n}$, а так как $\{u \bmod p^n : u \in H_{l,k}\} = C_k$, то это и доказывает лемму 1. ■

Из леммы 1 и формулы (5) видно, что если один и тот же индекс $k$ принадлежит двум разным подмножествам $I_s$ и $I_f$, то соответствующий вклад классов вычетов $H_{s,k}$ и $H_{f,k}$ при вычислении значений $S(\alpha^v)$ будет нулевой. Введём два новых подмножества индексов $I = \{k : k = 0, 1, \ldots, d - 1$ и $\sum\limits_{j=0}^{2^{m-1}-1} |\{k\} \cap I_{2j}| -$ нечётное число$\}$ и $J = \{k : k = 0, 1, \ldots, d - 1$ и $\sum\limits_{j=0}^{2^{m-1}-1} |\{k\} \cap I_{2j+1}| -$ нечётное число$\}$.

Далее, метод вычисления значений производящей функции последовательности с периодом $p^n$, то есть сумм $\sum\limits_{w \in C_l} \alpha^w$, предложен в [8]. Введём дополнительно следующие обозначения. Пусть $\beta = \alpha^{p^{n-1}}$ — первообразный корень степени $p$ из единицы в расширении поля GF(2) и $S_d(t) = \sum\limits_{u \in H_0} t^{u \bmod p}$, то есть $S_d(t)$ — многочлен, соответствующий классу вычетов степени $d$ по модулю $p$. Тогда, согласно [8], если $v \in p^f H_j$, то $\sum\limits_{u \in C_k} \alpha^{uv} = S_d(\beta^{\theta^{j+k}}) + f(p-1)/d$.

Отсюда, воспользовавшись леммой 1, формулой (5) и определением множеств $I, J$, получаем следующее утверждение (если $I$ или $J$ — пустое множество, то соответствующую сумму считаем равной нулю).

**Лемма 2.** Если $v \in p^f H_j$ для $f = 0, 1, \ldots, n - 1$ и $j = 0, 1, \ldots, d - 1$, то

$$S(\alpha^v) = \sum_{k \in I} S_d(\beta^{\theta^{j+k}}) + \sum_{k \in J} S_d(\beta^{\theta^{j+k}}) + (|I| + |J|) f(p-1)/d + \delta, \tag{6}$$

где

$$\delta = \begin{cases} 1, & \text{если } m = 1, \\ 0, & \text{если } m > 1. \end{cases}$$

Таким образом, задача вычисления значений $S(\alpha^v)$ свелась к расчёту значений многочлена $S_d(t)$. Исследуем вопрос о кратных корнях многочлена $S(t)$.

**Лемма 3.** Если $v \in p^f H_j$ для $f = 0, 1, \ldots, n - 1$ и $j = 0, 1, \ldots, d - 1$, то $\alpha^v$ — кратный корень многочлена $S(t)$ тогда и только тогда, когда

$$\sum_{k \in I} S_d(\beta^{\theta^{j+k}}) = f|I|(p-1)/d + \delta, \quad \sum_{k \in J} S_d(\beta^{\theta^{j+k}}) = f|J|(p-1)/d.$$

***Доказательство.*** Исследуем производную многочлена $S(t)$. Если $u \in H_{l,k}$, то для $l \equiv 0 \pmod 2$ всегда $u \equiv 0 \pmod 2$, следовательно, в кольце $\mathrm{GF}(2)[t]$

$$\left(\sum_{u\in H_{l,k}} t^u\right)' = \begin{cases} 0, \text{ если } l - \text{ чётное,} \\ \sum\limits_{u\in H_{k,l}} t^{u-1}, \text{ если } l - \text{ нечётное.} \end{cases}$$

Таким образом, $S'(t) = \sum\limits_{l=0}^{2^{m-1}-1} \sum\limits_{k\in I_{2l+1}} \sum\limits_{u\in H_{2l+1,k}} t^{u-1}$. Тогда по формуле (6) и лемме 1 имеем

$$S'(\alpha^v) = \alpha^{-v} \sum_{l=0}^{2^{m-1}-1} \sum_{k\in I_{2l+1}} \left(S_d(\beta^{\theta^{j+k}}) + f(p-1)/d\right) =$$

$$= \alpha^{-v} \left(\sum_{k\in J} S_d(\beta^{\theta^{j+k}}) + \textstyle\sum_{l=0}^{2^{m-1}-1} |I_{2l+1}| f(p-1)/d\right),$$

или

$$S'(\alpha^v) = \alpha^{-v} \left(\sum_{k\in J} S_d(\beta^{\theta^{j+k}}) + |J| f(p-1)/d\right). \tag{7}$$

Таким образом, если $\alpha^v$ — кратный корень многочлена $S(t)$, то $S(\alpha^v) = S'(\alpha^v) = 0$ и утверждение леммы 3 следует из формул (7) и (8). Наоборот, если выполняется условие леммы 3, то по (7) и (8) имеем, что $S(\alpha^v) = S'(\alpha^v) = 0$, то есть $\alpha^v$ — кратный корень многочлена $S(t)$. ■

Леммы 2 и 3 показывают, что для определения корней многочлена $S(t)$ в множестве $\{\alpha^v : v = 0, 1, \ldots, N-1\}$ и выделения среди них кратных достаточно знать числа $S_d(\beta^{\theta^l})$, $l = 0, 1, \ldots, d-1$. При этом если $v, u \in p^f H_j$, то $S(\alpha^v) = S(\alpha^u)$ и $S'(\alpha^v) = S'(\alpha^u)$ для любых значений $f = 0, 1, \ldots, n-1$, $j = 0, 1, \ldots, d-1$.

Пусть $\mathbf{S}_d(x) = \left(S_d(x), S_d(x^\theta), \ldots, S_d(x^{\theta^{d-1}})\right)$, $\mathbf{R}(x) = \sum\limits_{k\in I} \mathbf{S}_d(x^{\theta^k})$ и $\mathbf{Q}(x) = \sum\limits_{k\in J} \mathbf{S}_d(x^{\theta^k})$. Обозначим координаты вектор-функций $\mathbf{R}(x)$ и $\mathbf{Q}(x)$ при $x = \beta$ через $r_i, q_i$ для $i = 0, 1, \ldots, d-1$.

Непосредственно из лемм 2 и 3 получаем следующее утверждение.

**Теорема 1.** Если $v \in p^f H_j$, $f = 0, 1, \ldots, n-1$, $j = 0, 1, \ldots, d-1$, то $\alpha^v$ — корень многочлена $S(t)$ тогда и только тогда, когда $r_j + q_j = (|I| + |J|) f(p-1)/d + \delta$, и корень $\alpha^v$ многочлена $S(t)$ кратный тогда и только тогда, когда $r_j = |I| f(p-1)/d + \delta$.

Таким образом, теорема 1 и формулы (3), (4) показывают, что известные значения $\mathbf{R}(\beta)$, $\mathbf{Q}(\beta)$, а фактически $\mathbf{S}_d(\beta)$, позволяют оценить линейную сложность последовательности $X$. Метод вычисления значений $\mathbf{S}_d(x)$ предложен в [9] и развит в [10], там же найдены значения $\mathbf{S}_d(\beta^{\theta^j})$ при $d = 2, 4, 6, 8$. Следовательно, теорема 1 и результаты, представленные в [10], определяют метод анализа линейной сложности последовательностей с периодом $2^m p^n$, сформированных по правилу (2). Причём если $m = 1$, то можно явно рассчитать линейную сложность любой последовательности при $d = 2, 3, 4, 6, 8$, а если $m > 1$ или $d$ отлично от перечисленных, то можно подобрать такое правило построения последовательностей, при котором они заведомо будут обладать высокой линейной сложностью. Более того, вычисленные значения $S(\alpha^v)$, $v = 0, 1, \ldots, p^n - 1$, позволяют рассчитать минимальный многочлен последовательности по формуле (3).

Далее проиллюстрируем предложенный метод на примерах.

### 3. Линейная сложность последовательностей с периодом $2p^n$

Исследуем линейную сложность ряда обобщённых циклотомических последовательностей с периодом $2p^n$, ограничившись при этом вариантом, когда число нулей и единиц на периоде последовательности одинаково, то есть последовательности являются сбалансированными. В этом случае $S(1) = p^n$, то есть $S(1) \bmod 2 = 1$.

Не нарушая общности, всегда можно считать, что $0 \in I_0$ и ноль является общим элементом пересечения множеств $I_0$, $I_1$, если оно непусто. Рассмотрим возможные варианты для множеств $I_0, I_1$ при $d = 2, 4$. Для $m = 1$ всегда $I = I_0$, $J = I_1$, по определению множеств $I, J$, и $\delta = 1$.

Пусть $d = 2$, тогда, как это показано в [3] (см. также [10]), справедливо соотношение

$$\mathbf{S}_d(\beta) = \begin{cases} (1,0), \text{ если } p \equiv \pm 1 \pmod 8, \\ (\mu, \mu+1), \text{ если } p \equiv \pm 3 \pmod 8. \end{cases} \tag{8}$$

Здесь $\mu$ — корень уравнения $x^2 + x + 1 = 0$ в расширении поля GF(2).

**Лемма 4.** Если $I_0 = I_1 = \{0\}$, то для последовательности $X$, сформированной по правилу (2), линейная сложность $L = 2p^n$, а её минимальный многочлен $m(t) = t^{2p^n} - 1$.

***Доказательство.*** По условию $I = \{0\}$, $J = \{0\}$, тогда $\mathbf{R}(\beta) = \mathbf{Q}(\beta) = \mathbf{S}_2(\beta)$. Следовательно, $\mathbf{R}(\beta) + \mathbf{Q}(\beta) = (0,0)$ и $r_j + q_j = 0$ для любого $j = 0, 1$. А так как $(|I|+|J|)(p-1)/2$ — чётное число и $\delta = 1$, то по теореме 1 $S(\alpha^v) \neq 0$ при всех значениях $v = 0, 1, \ldots, 2p^n - 1$. Утверждение леммы 4 следует из формул (3) и (4). ■

**Лемма 5.** Если $I_0 = \{0\}, I_1 = \{1\}$, то для последовательности $X$, сформированной по правилу (2), линейная сложность

$$L = \begin{cases} p^n + 1, \text{ если } p \equiv \pm 3 \pmod 8, \\ (p^n + 3)/2, \text{ если } p \equiv \pm 1 \pmod 8, \end{cases}$$

а её минимальный многочлен

$$m(t) = \begin{cases} (t^{p^n} - 1)(t-1), \text{ если } p \equiv \pm 3 \pmod 8, \\ (t-1)^2 \prod\limits_{v \in C_1} (t - \alpha^v), \text{ если } p \equiv 1 \pmod 8, \\ (t-1)^2 \prod\limits_{v \in D} (t - \alpha^v), \text{ если } p \equiv -1 \pmod 8, \end{cases}$$

где $D = H_1 \cup pH_0 \cup \ldots \cup p^{n-1}H_{n \bmod 2}$.

***Доказательство.*** В условиях леммы 5 $\mathbf{R}(\beta) = \mathbf{S}_2(\beta)$, а $\mathbf{Q}(\beta) = \mathbf{S}_2(\beta^\theta)$, следовательно, $\mathbf{R}(\beta) + \mathbf{Q}(\beta) = (1,1)$ и $r_j + q_j = 1$ для $j = 0, 1$. А так как $(|I| + |J|)(p-1)/2$ — четное число и $\delta = 1$, то по теореме 1 $S(\alpha^v) = 0$ при всех значениях $v \in C_0 \cup C_1$, то есть при $v = 1, \ldots, p^n - 1$.

Далее, если $v \in p^f H_j$ и $\alpha^v$ — корень многочлена $S(x)$, то по теореме 1 он кратный корень, если $r_j = f(p-1)/2 + 1$.

Таким образом, если $p \equiv \pm 3 \pmod 8$, то по формуле (9) $\alpha^v$ для $v = 1, \ldots, p^n - 1$ не является кратным корнем многочлена $S(t)$ и $L = 2p^n - (p^n - 1) = p^n + 1$, а $m(t) = (t^{2p^n} - 1)/((t^{p^n} - 1)(t-1)) = (t^{p^n} - 1)(t-1)$.

Пусть $p \equiv \pm 1 \pmod 8$, тогда $\mathbf{R}(\beta) = (1,0)$ по формуле (9). Если $p \equiv 1 \pmod 8$, то $f(p-1)/2 + \delta \equiv 1 \pmod 2$, то есть $\alpha^v$ — кратный корень многочлена $S(t)$ для любого $v \in C_0$. Если же $p \equiv -1 \pmod 8$, то соответственно получаем, что $\alpha^v$ — кратный корень многочлена $S(t)$, когда $r_j = f + 1$, то есть при $v \in H_0 \cup pH_1 \cup \ldots \cup p^{n-1}H_{(n-1) \bmod 2}$.

В обоих случаях $L = 2p^n-(p^n-1)-(p^n-1)/2 = (p^n+3)/2$, но минимальные многочлены последовательностей, вычисленные по формуле(3), различаются. ■

В частном случае, когда $2 \in H_0$, утверждения лемм 4 и 5 были получены в [5, 6] другим способом.

Пусть $d = 4$, тогда справедливо разложение $p = x^2 + 4y^2$, где $x \equiv 1 \pmod 4$, $x$ и $y$ — целые числа [7]. Обозначим через $\left(\frac{a}{p}\right)$ символ Лежандра, а через $\left(\frac{a}{p}\right)_4$ — символ 4-степенного вычета [7], $\left(\frac{a}{p}\right)_4 = 1$ тогда и только тогда, когда сравнение $z^4 \equiv a \pmod p$ разрешимо в целых числах. Согласно [9, 10], имеем

1) если $\left(\frac{2}{p}\right)_4 = 1$, то $\mathbf{S}_4(\beta) = (1, 1, 0, 1)$ для $x \equiv 5 \pmod 8$ и $\mathbf{S}_4(\beta) = (1, 0, 0, 0)$ для $x \equiv 1 \pmod 8$;

2) если $\left(\frac{2}{p}\right) = 1$ и $\left(\frac{2}{p}\right)_4 \neq 1$, то $\mathbf{S}_4(\beta) = (\mu, 0, \mu + 1, 0)$ для $x \equiv 5 \pmod 8$ и $\mathbf{S}_4(\beta) = (\mu, 1, \mu + 1, 1)$ для $x \equiv 1 \pmod 8$, где $\mu$ — корень уравнения $t^2 + t + 1 = 0$ в расширении поля $\mathrm{GF}(2)$;

3) если $\left(\frac{2}{p}\right) \neq 1$, то $\mathbf{S}_4(\beta) = (\zeta, \zeta^2, \zeta^4, \zeta^8)$ или $\mathbf{S}_4(\beta) = (\zeta, \zeta^8, \zeta^4, \zeta^2)$, где $\zeta$ удовлетворяет условию $\zeta^8 + \zeta^4 + \zeta^2 + \zeta + 1 = 0$.

Воспользуемся этими соотношениями для расчёта линейной сложности последовательностей, сформированных на основе классов биквадратичных вычетов.

**Лемма 6.** Если $I_0 = \{0, 1\}$, то для линейной сложности последовательности $X$, сформированной по правилу (2), справедливо следующее:

1) $L = 2p^n$, если $I_1 = \{0, 1\}$, или $I_1 = \{0, 2\}$ и $\left(\frac{2}{p}\right)_4 \neq 1$, или $I_1 = \{0, 3\}$ и $\left(\frac{2}{p}\right) \neq 1$;

2) $L = (3p^n + 1)/2$, если $I_1 = \{0, 3\}$ и $\left(\frac{2}{p}\right) = 1$, $\left(\frac{2}{p}\right)_4 \neq 1$;

3) $L = (5p^n + 1)/4$, если $\left(\frac{2}{p}\right)_4 = 1$ и $I_1 = \{0, 2\}$ или $I_1 = \{0, 3\}$.

***Доказательство.*** Докажем лемму только для $I_1 = \{0, 3\}$, другие два варианта исследуются аналогично. Так как по условию $|I| = |J| = 2$, то, согласно теореме 1, $S(\alpha^v) = 0$ при всех значениях $v \in C_k$ тогда и только тогда, когда $r_k + q_k = 1$ и при этом $\alpha^v$ — кратный корень для любого $v \in C_k$ тогда и только тогда, когда $r_k = 1$. Если $I_0 = \{0, 1\}$ и $I_1 = \{0, 3\}$, то $\mathbf{R}(\beta) = \mathbf{S}_4(\beta) + \mathbf{S}_4(\beta^\theta)$, а $\mathbf{R}(\beta) + \mathbf{Q}(\beta) = \mathbf{S}_4(\beta^\theta) + \mathbf{S}_4(\beta^{\theta^3})$. Из приведённых выше соотношений для $\mathbf{S}_4(\beta)$ получаем, что $r_k + q_k \neq 1$ для $k = 0, 1, 2, 3$ при $\left(\frac{2}{p}\right) \neq 1$, следовательно, $L = 2p^n$. Когда же $\left(\frac{2}{p}\right) = 1$, то $\mathbf{R}(\beta)+\mathbf{Q}(\beta) = (0, 1, 0, 1)$, кроме этого, $r_k \neq 1$, $k = 0, 1, 2, 3$, при $\left(\frac{2}{p}\right)_4 \neq 1$, то есть в этом варианте нет кратных корней и $L = 2p^n - 2|C_0| = (3p^n + 1)/2$. В то же время если $\left(\frac{2}{p}\right)_4 = 1$, то $\mathbf{R}(\beta) = (0, 1, 1, 0)\,((1, 0, 0, 1))$, тогда $\alpha^v$ являются кратными корнями при $v \in C_1(C_0)$ и $L = 2p^n - 3|C_0| = (5p^n + 3)/4$. ■

В частном случае, для $n = 1$ и $I_0 = \{0, 1\}$, $I_1 = \{0, 2\}$, линейная сложность последовательности $X$ была исследована в [4].

**Лемма 7.** Если $I_0 = \{0, 2\}$, $I_1 = \{0, 3\}$, то для линейной сложности последовательности $X$, сформированной по правилу (2), справедливы следующие равенства:

1) $L = 2p^n$, если $\left(\frac{2}{p}\right)_4 \neq 1$;

2) $L = p^n + 1$, если $\left(\frac{2}{p}\right)_4 = 1$.

Лемма 7 доказывается подобно лемме 6. Все другие варианты для $I_0, I_1$ сводятся к уже рассмотренным.

## 4. Примеры оценки линейной сложности последовательностей с периодом $2^m p^n$

Рассмотрим три примера оценки линейной сложности последовательности. В [10, 11] предложено несколько правил построения последовательностей с периодами $4p, 8p$, обладающих хорошей периодической автокорреляционной функцией. Оценим линейную сложность этих последовательностей в общем случае. Предварительно отметим, что для всех рассматриваемых последовательностей единица является корнем многочлена $S(x)$ при $m > 1$ и её кратность не превосходит $m$.

**Лемма 8.** Если $d = 2$ и последовательность $X$ с периодом $4p^n$ определена правилом (2) при $I_0 = I_1 = I_2 = \{0\}, I_3 = \{1\}$, то $L \geqslant 4p^n - 4$.

***Доказательство.*** Согласно условию леммы и определению множеств $I$ и $J$, имеем $I = \varnothing$, $J = \{0, 1\}$, тогда $r_j + q_j = 1$, $j = 0, 1$, так как сумма $\sum\limits_{i=0}^{d-1} S_d(\beta^{\theta^i}) = 1$ [10], то есть, по теореме 1, $S(\alpha^v) = 1$ для $v = 1, \ldots, p^n - 1$. Таким образом, утверждение леммы следует из формулы (4).■

**Лемма 9.** Если $d = 4$ и последовательность $X$ с периодом $4p^n$ определена правилом (2) при $I_0 = I_1 = \{0, 1\}$ и $I_2 = \{0, 3\}, I_3 = \{1, 2\}$ или $I_2 = \{1, 2\}, I_3 = \{0, 3\}$, то $L \geqslant 4p^n - 4$.

Лемма 9 доказывается аналогично лемме 8.

**Лемма 10.** Если $d = 4$ и последовательность $X$ с периодом $8p^n$ определена правилом (2) при $I_0 = I_1 = I_2 = I_5 = \{0\}$, $I_3 = \{1\}$, $I_4 = I_6 = \{2\}$ и $I_7 = \{3\}$, то $L \geqslant 8p^n - 8$, если $\left(\frac{2}{p}\right) \neq 1$, и $L \geqslant 4p^n - 8$, если $\left(\frac{2}{p}\right) = 1$.

***Доказательство.*** В условиях леммы $S(1) = 0$ и $I = \varnothing$, $J = \{1, 3\}$, тогда $\mathbf{R}(\beta) = \mathbf{0}$. Если $\left(\frac{2}{p}\right) \neq 1$, то $q_i \neq 1$, следовательно, по теореме 1 $S(\alpha^v) \neq 0$ для $v = 1, \ldots, p^n - 1$. Если же $\left(\frac{2}{p}\right) = 1$, то $\mathbf{Q}(\beta) = (0, 1, 0, 1)$ и по теореме 1 $S(\alpha^v) = 0$ для $v \in C_0 \cup C_2$, причём каждый корень является кратным. Применение формулы (4) завершает доказательство леммы 10. ■

## Заключение

Предложен метод анализа линейной сложности последовательностей с периодом $2^m p^n$, построенных на основе обобщённых циклотомических классов. Метод позволяет как явно рассчитать линейную сложность и минимальный многочлен рассматриваемых последовательностей, так и оценить её, а также определить характеристики последовательностей, обладающих заведомо высокой линейной сложностью. Вычисле-

на линейная сложность ряда последовательностей на основе классов квадратичных и биквадратичных вычетов.

## ЛИТЕРАТУРА


1. *Лидл Р., Нидеррайтер Г.* Конечные поля. М.: Мир, 1988. 820 с.
2. *Cusick T. W., Ding C., and Renvall A.* Stream Ciphers and Number Theory. North-Holland Mathematical Library. V. 55. Amsterdam: Elsevier, 1998.
3. *Ding C., Helleseth T., and Shan W.* On the Linear Complexity of Legendre Sequences // IEEE Trans. Info Theory. 1998. V. IT-44. P. 1276–1278.
4. *Ding C., Helleseth T., and Martinsen H.* New families of binary sequences with optimal three-level autocorrelation // IEEE Trans. Info Theory. 2001. V. 47. P. 428–433.
5. *Zhang J., Zhao C.-A., and Ma X.* Linear complexity of generalized cyclotomic binary sequences of length $2p^m$ //Appl. Algebra Eng. Commun. Comput. 2010. V. 21. No. 2. P. 93–108.
6. *Zhang J., Zhao C.-A., and Ma X.* On the Linear Complexity of Generalized Cyclotomic Binary Sequences with Length $2p^2$ // IEICE Trans. Fundament. Electron., Commun. Comput. Sci. 2010. V. E93.A. Iss. 1. P. 302–308.
7. *Айерлэнд К., Роузен М.* Классическое введение в современную теорию чисел. М.: Мир, 1987. 416 с.
8. *Edemskiy V. A.* About computation of the linear complexity of generalized cyclotomic sequences with period $p^{n+1}$ // Designs, Codes and Cryptography. 2011. V. 61. No. 3. P. 251–260.
9. *Едемский В. А.* О линейной сложности двоичных последовательностей на основе классов биквадратичных и шестеричных вычетов // Дискретная математика. 2010. Т. 22. № 1. С. 74–82.
10. *Едемский В. А, Гантмахер В. Е.* Синтез двоичных и троичных последовательностей с заданными ограничениями на их характеристики. Великий Новгород: НовГУ, 2009. 189 с.
11. *Zhang Y., Lei J. G.,and Zhang S. P.* A new family of almost differences sets and some necessary conditions // IEEE Trans. Info. Theory. 2006. V. 52. P. 2052–2061.

2012 Теоретические основы прикладной дискретной математики №3(17)

DOI 10.17223/20710410/17/2

УДК 512.62

# О ПОСТРОЕНИИ ВОЗМОЖНО ОДНОСТОРОННИХ ФУНКЦИЙ НА ОСНОВЕ АЛГОРИТМИЧЕСКОЙ НЕРАЗРЕШИМОСТИ ПРОБЛЕМЫ ЭНДОМОРФНОЙ СВОДИМОСТИ В ГРУППАХ

С. Ю. Ерофеев, В. А. Романьков

*Омский государственный университет им. Ф. М. Достоевского, г. Омск, Россия*

**E-mail:** stepan.erofeev@gmail.com, romankov48@mail.ru

Рассматривается схема построения возможно односторонней функции в группе с разрешимой проблемой равенства и неразрешимой проблемой эндоморфной сводимости. Анализируются предпосылки криптографической стойкости предлагаемой схемы. В качестве приложения предлагается схема аутентификации с нулевым разглашением пользователей в системе. Отмечается, что для её криптостойкости требуется неразрешимость более сильной проблемы двукратной эндоморфной сводимости.

## Введение

Работа относится к «криптографии, основанной на группах» (group-based cryptography), — современному направлению, возникшему на рубеже XX и XXI столетий. Основными объектами в нём являются абстрактные бесконечные группы, а основной целью — построение на этих группах криптографических схем и протоколов. Исследования ведутся методами теории групп, теории сложности и теории вычислений.

Современное состояние полученных в этой области результатов отражено в монографиях [1, 2] (см. также обзор второго автора [3]).

Основной целью настоящей работы является представление конструкции возможно односторонней функции на основе алгоритмической неразрешимости проблемы эндоморфной сводимости в группах.

*Односторонние* (в другой терминологии *однонаправленные*) функции являются неотъемлемой частью криптографических схем и протоколов. Теоретически их существование при формальном определении до сих пор не установлено. В то же время односторонние функции являются основным инструментом во многих разделах и приложениях криптографии, в частности, они применяются в электронных подписях, протоколах аутентификации, алгоритмах генерации псевдослучайных последовательностей и т. п. Конечно, используемые с указанной целью функции можно назвать только *возможно односторонними*.

Относительно общей теории односторонних функций см. монографии [4 – 6]. В работах Л. А. Левина [7, 8] приведена универсальная функция, которая автоматически является односторонней, если существует хотя бы одна формально односторонняя функция. Такие функции названы *полными односторонними* функциями. Для их построения Л. А. Левин использовал нумерацию всех машин Тьюринга [7] и комбинаторный тайлинг [8]. Отсюда видно, что такое построение имеет чисто теоретическое значение. В работе А. А. Кожевникова и С. И. Николенко [9] приведены другие примеры полных односторонних функций.

В данной работе предлагается новый кандидат на роль односторонней функции. В качестве платформы для неё рассматривается бесконечная группа с разрешимой

проблемой равенства и неразрешимой проблемой эндоморфной сводимости. Конкретное предложение — свободная метабелева группа $M_n$ достаточно большого ранга $n$.

Теоретическая база в данном случае была заложена в работе В. А. Романькова [10], где доказана неразрешимость проблемы эндоморфной сводимости в свободных метабелевых группах достаточно большого ранга. Заметим, что разрешимость проблемы равенства в свободных метабелевых группах ко времени появления работы [10] была не только хорошо известна, но уже существовали вполне эффективные с практической точки зрения алгоритмы её решения.

Более точно, В. А. Романьков в работах [10, 11] ввёл в рассмотрение интерпретацию диофантовых уравнений в свободных нильпотентных группах ступени $\geqslant 9$ и в свободных метабелевых группах достаточно большого ранга, позволяющую перенести алгоритмическую неразрешимость 10-й Проблемы Гильберта, установленную Ю. В. Матиясевичем в [12, 13] (см. также [14]), на алгоритмическую неразрешимость проблемы эндоморфной сводимости в рассматриваемых группах. Оценка на ранг следовала из результатов Матиясевича.

В работе [3] дан обзор исследований по криптографии, основанной на группах. Объяснены возможности использования неразрешимых и трудноразрешимых алгоритмических проблем теории групп в качестве основы для построения криптографических схем и протоколов. Об этих возможностях и их реализациях см. также [1, 2, 15 – 19].

В работе [3] введено понятие диофантовой криптографии. Показана универсальность диофантова языка, позволяющая записывать на нём функции многих известных схем и протоколов криптографии. Подчёркнута объединяющая роль диофантовой криптографии. Описаны её возможности, вытекающие из неразрешимости 10-й Проблемы Гильберта. Данная работа также основана на представленном в [3] материале. Более того, в [3] подробно обоснованы достоинства свободных метабелевых групп как платформ для построения криптографических схем и протоколов, что также лежит в основе предлагаемой конкретной реализации протокола аутентификации с нулевым разглашением.

## 1. Односторонние функции

Неформально *односторонней* называется функция $f$, значение $y = f(x)$ которой легко вычислимо по аргументу $x$, а обращение, то есть вычисление по данному $y$ хотя бы одного аргумента $x'$, такого, что $y = f(x')$, является трудной задачей.

Обычно первое из этих условий трактуется как существование детерминированного алгоритма, вычисляющего по аргументу (*входу*) $x$ значение $y = f(x)$ за *время* (количество шагов), не превышающее $p(|x|)$, где $p(\cdot)$ — некоторый полином, $|x|$ — размер входа.

Второе условие, то есть трудность обращения, означает, что любой вероятностный алгоритм, полиномиальный по $|y|$, вычисляет $x'$ с условием $y = f(x')$ с пренебрежимо малой вероятностью.

В первом томе монографии О. Голдрейха [4] даны следующие определения.

**Определение 1** (сильно односторонняя функция). Функция $f : \{0,1\}^* \to \{0,1\}^*$ называется *сильно односторонней*, если выполнены следующие требования:

а) существует детерминированный полиномиальный по времени алгоритм $\mathcal{A}$, вычисляющий по аргументу $x$ значение $y = f(x)$;

б) для любого вероятностного полиномиального по времени алгоритма $\mathcal{B}$, любого полинома $p(\cdot)$ и всех достаточно больших натуральных чисел $n$ выполняется неравен-

ство

$$\Pr\left[\mathcal{B}(f(U_n), 1^n) \in f^{-1}(f(U_n)\right] < \frac{1}{p(n)},$$

где $U_n$ означает случайную величину, равномерно распределенную на $\{0,1\}^n$. Заметим, что алгоритму предписан размер его выхода. Это объясняется тем, что размер аргумента $x$ может не вычисляться как полиномиальная функция от $|y|$. В [4] приводится пример, когда в качестве значения функции $f(x)$ берется $|x|$. Заметим, однако, что в данном примере вход $x'$ с условием $y = f(x')$ находится очевидным образом и может быть записан с использованием компрессии полиномиально от $|y|$. Это показывает, что определение 1, возможно, нуждается в коррекции. Алгоритмы с компрессионной записью получили в настоящее время существенное развитие в теории вычислений и в теории групп; см. по этому поводу [20 – 22].

**Определение 2** (слабо односторонняя функция). Функция $f : \{0,1\}^* \to \{0,1\}^*$ называется *слабо односторонней*, если выполнены следующие требования:

а) = а) из определения 1;

б) существует полином $p(\cdot)$, такой, что для любого вероятностного полиномиального по времени алгоритма $\mathcal{B}$ и всех достаточно больших натуральных чисел $n$ выполняется неравенство

$$\Pr\left[\mathcal{B}(f(U_n), 1^n) \notin f^{-1}(f(U_n)\right] > \frac{1}{p(n)}.$$

В [4] доказано, что существование слабо односторонней функции влечет существование сильно односторонней функции.

Приведённые определения связаны с бинарным представлением аргументов и значений функций. Для задания возможно односторонней функции, определённой на группе, можно использовать следующие соображения. Во-первых, на конечно порождённой группе $G$ с фиксированным множеством порождающих элементов можно задать словарную метрику, согласно которой расстояние $|g|$ от элемента $g$ до 1 (его длина) равна наименьшей длине группового слова от фиксированного множества порождающих элементов группы, записывающего элемент $g$. Расстояние между элементами $g$ и $f$ определяется как $|gf^{-1}|$. Метрика позволяет естественным образом определить сферы $\mathbb{S}_r$ и шары $\mathbb{B}_r$ радиуса $r$, которые дадут стратификацию множества элементов группы. Определения 1 и 2 переписываются следующим образом.

**Определение 1′** (сильно односторонняя функция). Функция $\varphi : G \to G$ называется *сильно односторонней*, если выполнены следующие требования:

а) существует детерминированный полиномиальный по времени алгоритм $\mathcal{A}$, вычисляющий по аргументу $g$ значение $f = \varphi(g)$;

б) для любого вероятностного полиномиального по времени алгоритма $\mathcal{B}$, любого полинома $p(\cdot)$ и всех достаточно больших натуральных чисел $n$ выполняется неравенство

$$\Pr\left[\mathcal{B}(\varphi(U_n), \mathbb{B}_n) \in \varphi^{-1}(\varphi(U_n)\right] < \frac{1}{p(n)},$$

где $U_n$ означает случайную величину, равномерно распределенную на $\mathbb{B}_n$. Как и раньше, алгоритму предписан размер его выхода.

**Определение 2′** (слабо односторонняя функция). Функция $\varphi : G \to G$ называется *слабо односторонней*, если выполнены следующие требования:

а) = а) из определения $1'$;

б) существует полином $p(\cdot)$, такой, что для любого вероятностного полиномиального по времени алгоритма $\mathcal{B}$ и всех достаточно больших натуральных чисел $n$ выполняется неравенство

$$\Pr\left[\mathcal{B}(\varphi(U_n), \mathbb{B}_n) \notin \varphi^{-1}(\varphi(U_n)\right] > \frac{1}{p(n)}.$$

Во многих случаях приведённые определения сводятся к определениям 1 и 2 кодированием элементов группы бинарными последовательностями.

Известными кандидатами на роль односторонних являются следующие функции.

**Степенная функция**

$$f : x \longmapsto c = x^e \bmod n, \tag{1}$$

определённая на кольце $\mathbb{Z}_n$ вычетов по модулю $n$, где $n = pq$ — произведение больших различных (секретных) простых чисел. Для её обратимости требуется взаимная простота показателя $e$ со значением функции Эйлера $\varphi(n) = (p-1)(q-1)$.

Задача обращения функции (1) равносильна нахождению решения диофантова уравнения

$$c = \zeta_1^e + n\zeta_2$$

относительно переменных $\zeta_1, \zeta_2$, или, что равносильно, задаче обращения диофантовой функции

$$d(\zeta_1, \zeta_2) = \zeta_1^e + n\zeta_2.$$

Функция (1) используется в качестве функции шифрования в системе RSA и в качестве цифровой подписи на основе RSA. Эта функция фигурирует также в ряде других схем, базирующихся на сложности задачи разложения чисел на множители.

**Показательная функция**

$$g : x \longmapsto g^x, \tag{2}$$

определённая на кольце целых чисел $\mathbb{Z}$ со значениями в конечном поле $\mathbb{F}_q$ порядка $q$, где $g$ — порождающий элемент мультипликативной (циклической) группы $\mathbb{F}_q^*$. Криптостойкость её основана на трудности вычисления дискретного логарифма в конечных полях.

Её обращение равносильно обращению некоторой диофантовой функции $d(\zeta_1, \ldots, \zeta_k)$. Это следует из результатов Ю. В. Матиясевича [12, 13].

В случае простого конечного поля $\mathbb{F}_p = \mathbb{Z}_p$ характеристики $p$ явный вид полинома $d(\zeta_1, \ldots, \zeta_k)$ можно найти в [23, 24].

Функция (2) используется, например, в системе шифрования Эль Гамаля, известных протоколах Диффи — Хеллмана, Масси — Омуры, базовом протоколе Эль Гамаля цифровой подписи и других схемах (см. по этому поводу [25, 26]).

Два приведённых выше примера криптографических функций являются частными случаями диофантовых функций. Диофантова криптография, относительно которой см. [3], позволяет не только использовать универсальный диофантов язык для представления многих известных криптографических функций, но также играть объединяющую роль для этих функций, записывая системы соответствующих им диофантовых уравнений. При этом появляется дополнительная возможность комбинирования и преобразования переменных.

## 2. 10-я Проблема Гильберта и проблема эндоморфной сводимости

### 2.1. 10-я Проблема Гильберта

*Диофантовым* называется полином $d(\zeta_1, \ldots, \zeta_k)$ с целыми коэффициентами от независимых коммутирующих переменных $\zeta_1, \ldots, \zeta_k$.

Уравнение вида

$$d(\zeta_1, \ldots, \zeta_k) = 0 \tag{3}$$

называется *диофантовым уравнением*. Ю. В. Матиясевич в работе [12] (полное доказательство в [13], см. также [14]) установил неразрешимость 10-й Проблемы Гильберта, тем самым завершив усилия ряда математиков, в частности Д. Робинсон, М. Дэвиса и Х. Патнэма. А именно, он доказал, что не существует алгоритма, определяющего по произвольному диофантову уравнению (3), обладает ли оно решением в целых числах. Данный результат часто называют неразрешимостью 10-й Проблемы Гильберта. Более того, Ю. В. Матиясевич заметил, что существует фиксированный диофантов полином $d_0(\zeta_1, \ldots, \zeta_k)$, такой, что алгоритмически неразрешима проблема существования решений в целых числах в классе диофантовых уравнений вида

$$d_0(\zeta_1, \ldots, \zeta_k) = c, \tag{4}$$

где $c$ — произвольное целое число.

В работе Ю. В. Матиясевича и Д. Робинсон [27] неразрешимость 10-й Проблемы Гильберта установлена для класса уравнений от 13 переменных, в докладе Ю. В. Матиясевича [28] было объявлено, что количество переменных может быть уменьшено до 9. Эта оценка изложена со всеми деталями в работе Д. Джонса [29].

Функция $d_0(\zeta_1, \ldots, \zeta_k)$ может рассматриваться в качестве кандидата на роль односторонней функции. Действительно, любое её значение вычислимо за полиномиальное время от $|\zeta_1|+\ldots+|\zeta_k|$. В то же время из-за неразрешимости 10-й Проблемы Гильберта нельзя указать полиномиальный по времени алгоритм, вычисляющий аргумент этой функции с заданным значением.

### 2.2. Проблема эндоморфной сводимости

Будем говорить, что в эффективно заданной группе $G$ разрешима проблема *эндоморфной сводимости*, если существует алгоритм, решающий для любой пары элементов $g$ и $f$ группы $G$, является ли $f$ образом элемента $g$ для какого-либо эндоморфизма $\varphi \in \operatorname{End} G$ или нет.

Эффективность задания группы $G$, способы записи её элементов и способы задания эндоморфизмов могут быть разными. Подробно об этом говорится в работе [3]. В классическом случае группа $G$ предполагается конечно определённой и задаётся своим представлением через конечные множества порождающих элементов и определяющих соотношений вида

$$\mathcal{P}(G) = \langle x_1, \ldots, x_n | r_1 = 1, \ldots, r_m = 1 \rangle,$$

где $r_1, \ldots, r_m$ — групповые слова от $x_1, \ldots, x_n$, называемые *определяющими словами*.

Это означает, что группа $G$ есть фактор-группа $F_n/R$ свободной группы $F_n$ ранга $n$ с базисом $\{x_1, \ldots, x_n\}$ по нормальному замыканию $R = \text{нз}(r_1, \ldots, r_m)$, то есть по наименьшей нормальной подгруппе свободной группы $F_n$, содержащей определяющие слова $r_1, \ldots, r_m$. Группа $G$ является каноническим гомоморфным образом группы $F_n$ относительно гомоморфизма $g \mapsto gR$. Ядром этого гомоморфизма является $R$. Группа $G$ порождается элементами $y_i = x_i R$ для $i = 1, \ldots, n$. Обычно эти элементы обозначают

так же, как их прообразы — через $x_i$ $(i = 1, \ldots, n)$. Элементы группы $G$ записываются, вообще говоря, неоднозначно, как групповые слова от порождающих. Эндоморфизм задается отображением порождающих $x_i \mapsto g_i$ для $i = 1, \ldots, n$. Не всякое такое отображение определяет эндоморфизм. Необходимым и достаточным условием для такого определения является проверка того, что значения определяющих слов после подстановки вместо порождающих элементов их образов равны 1 в группе $G$. Если группа $G$ свободна с базисом $\{x_1, \ldots, x_n\}$, то есть множество определяющих соотношений пусто, или свободна в некотором многообразии $\mathcal{L}$ групп (свободная абелева, нильпотентная, метабелева и т. д.), то любое отображение порождающих элементов однозначно продолжается до эндоморфизма группы. Относительно многообразий групп см. монографию Х. Нейман [30].

Легко показать, что проблема эндоморфной сводимости разрешима для любой конечно порождённой абелевой группы. Из разрешимости уравнений в свободной группе $F_n$ при любом $n$, доказанной Г. С. Маканиным [31], следует разрешимость проблемы эндоморфной сводимости в группе $F_n$. В то же время найдены достаточно просто задаваемые группы, в которых проблема эндоморфной сводимости оказалась неразрешимой. Среди них свободные метабелевы группы $M_n$ достаточно большого ранга $n$. Перейдём к описанию этого результата работы [10] (см. также [3]).

Свободная метабелева группа $M_n$ ранга $n$ определяется как фактор-группа свободной группы $F_n$ ранга $n$ по второму коммутанту $F_n''$. Она является свободной группой многообразия всех метабелевых групп $\mathcal{A}^2$. Для свободных порождающих $x_1, \ldots, x_n$ группы $M_n$ определяются *базисные коммутаторы* — коммутаторы от элементов базиса $x_1, \ldots, x_n$ вида $[\ldots[[x_{i_1}, x_{i_2}], x_{i_3}], \ldots, x_{i_l}]$, где $l \geqslant 2$ называется *весом* коммутатора и выполняются неравенства $i_1 > i_2$; $i_2 \leqslant i_3 \leqslant \ldots \leqslant i_l$. Сами порождающие $x_1, \ldots, x_n$ также считаются базисными коммутаторами веса 1. В базисных коммутаторах скобки стоят слева направо. Коммутаторы с такой расстановкой скобок называются *левонормированными* или *простыми*. Для них внутренние скобки обычно не указываются.

Известно [32, 33], что образы базисных коммутаторов веса $i$ относительно канонического гомоморфизма $M_n \to M_n/\gamma_{i+1}M_n$ образуют базис свободной абелевой группы $\gamma_i M_n/\gamma_{i+1}M_n$. Здесь $\gamma_i M_n$ означает член нижнего центрального ряда группы $M_n$ с номером $i$. Для произвольной группы $G$ по определению $\gamma_1 G = G$, $\gamma_2 G = G'$ (коммутант) и для любого $i \geqslant 3$ полагается $\gamma_i G = [\gamma_{i-1}G, G]$. Напомним, что для произвольных подмножеств $A$ и $B$ группы $G$ выражение $[A, B]$ означает взаимный коммутант этих подмножеств, то есть подгруппу, порождённую в $G$ всеми коммутаторами вида $[a, b]$, где $a \in A$, $b \in B$. Группа $G$ называется *нильпотентной*, если для некоторого $i$ имеем $\gamma_i G = 1$. Наименьшее $i$ с этим свойством называется *ступенью нильпотентности* группы $G$. Тривиальная группа имеет по определению ступень нильпотентности 0, нетривиальная абелева группа — ступень 1 и т. д. Для более детальной информации см. [30, 32 – 35].

Упорядочим все базисные коммутаторы группы $M_n$ по возрастанию весов. Продолжим этот частичный порядок до полного, упорядочив между собой базисные коммутаторы одного веса произвольным образом. Пусть $c_1, \ldots, c_t$ $(t = t(i))$ — полный список всех базисных коммутаторов веса не больше чем $i$ в заданном порядке. Обычно считают, что $c_j = x_j$ для $j = 1, \ldots, n$, то есть что порядок на порождающих элементах как базисных коммутаторах веса 1 соответствует порядку на индексах. Тогда любой

элемент группы $M_n$ при $n \geqslant 2$ для любого $i \geqslant 1$ однозначно записывается в виде

$$g = \prod_{j=1}^{t} c_j^{k_j} (\mathrm{mod} \gamma_{i+1} M_n) \text{ для некоторых } k_j \in \mathbb{Z}.$$

Таким образом по модулю $\gamma_{i+1}M_n$ элементы группы $M_n$ кодируются наборами целых чисел $(k_1, \ldots, k_t)$, где $t = t(i)$. Компоненты набора будем называть *координатами* элемента $g$ по модулю $\gamma_{i+1}M_n$. Легко видеть, что координаты при различных $i$ соответствуют друг другу в очевидном смысле.

Возьмём множество диофантовых уравнений вида (4). Считаем левую часть произвольным полиномом. Построим по такому уравнению пару элементов группы $M_{k+2}$. Элемент $g$ строится следующим образом. Пусть наибольшая степень мономов в записи левой части (4) равна $m$. Полагаем $s = m$, если $m \geqslant 4$, и $s = 6$, если $m \leqslant 3$. Выделим два базисных элемента $x_1$ и $x_2$. Каждой переменной $\zeta_i$ сопоставим элемент базиса $x_{i+2}$. Пусть моном $\zeta_1^{l_1} \ldots \zeta_k^{l_k}$ входит в каноническую запись полинома $d_0(\zeta_1, \ldots, \zeta_k)$ с коэффициентом $b$. Сопоставим ему степень базисного коммутатора веса $s+2$ вида

$$[x_2, x_1, x_2, \ldots, x_2, x_3, \ldots, x_3, \ldots, x_k, \ldots, x_k]^b,$$

где порождающий $x_j$ для $j = 3, \ldots, k+2$ имеет $l_{j-2}$ вхождений, $x_1$ — одно вхождение, $x_2$ имеет $s - l_1 - \ldots - l_k - 1$ вхождение. Обозначим через $V(d_0)$ произведение всех полученных степеней базисных коммутаторов веса $s+2$ (по одному для каждого монома). Определим первый из элементов как

$$g = [x_2, x_1, x_1, x_2] V(d_0).$$

Второй элемент определим как

$$f = f(c) = [x_2, x_1, x_1, x_2][x_2, x_1, x_2, \ldots, x_2]^c, \tag{5}$$

где базисный коммутатор $h = [x_2, x_1, x_2, \ldots, x_2]$ второго множителя имеет вес $s+2$.

В [10] доказано, что если элемент $f$ является эндоморфным образом элемента $g$ в группе $M_{k+2}$ по модулю $\gamma_{s+3}M_{k+2}$ для некоторого эндоморфизма $\mu : M_{k+2} \to M_{k+2}$, то

$$\mu(x_i) = x_i \pmod{\gamma_2 M_{k+2}} \text{ для } i = 1, 2. \tag{6}$$

И если

$$\mu(x_j) = x_1^{\alpha_{j-2}} x_2^{\beta_{j-2}} \pmod{\gamma_2 M_{k+2}} \text{ для } j = 3, \ldots, k+2, \tag{7}$$

то $\zeta_1 = \beta_1, \ldots, \zeta_k = \beta_k$ является решением диофантова уравнения (4). Таким образом, если бы существовал алгоритм, решающий проблему эндоморфной сводимости фиксированного элемента $g$ группы $M_{k+2}$ к элементу $f = f(c)$ для произвольного целого числа $c$ по модулю $\gamma_{s+3}M_{k+2}$, то существовал бы алгоритм, решающий проблему существования решения у уравнений вида (4). Как мы отмечали выше, такого алгоритма нет. Значит, проблема эндоморфной сводимости в группе $M_{k+2}$ по модулю $\gamma_{s+3}M_{k+2}$ алгоритмически неразрешима. Наоборот, по любому решению $\zeta_1 = \beta_1, \ldots, \zeta_k = \beta_k$ уравнения (4) определяется эндоморфизм $\mu$ со свойством (6), соответствующий отображению (7) для $\alpha_j = 0$ $(j = 1, \ldots, k)$. В свою очередь, в [10] (см. также [3]) показано, что проблема эндоморфной сводимости по модулю $\gamma_{s+3}M_{k+2}$ сводится к проблеме эндоморфной сводимости в группе $M_r$ для некоторого $r = k+2+p_{k+2,s+3}$, откуда следует её неразрешимость в группе $M_r'$ при любом $r' \geqslant r$. Объясним это более подробно.

Указанные выше построения возможны для произвольного диофантова уравнения $d(\zeta_1, \ldots, \zeta_k) = c$. Если построить элемент $g = g(d)$, где $d$ — многочлен из левой части уравнения, то элемент $f = f(c)$ является эндоморфным образом элемента $g$ тогда и только тогда, когда данное диофантово уравнение разрешимо в целых числах. Это называется *интерпретацией диофантовых уравнений* в свободных метабелевых группах. Легко видеть, что эндоморфная сводимость элемента $g$ к элементу $f(c)$ в группе $M_{k+2}$ по модулю $\gamma_{s+3}M_{k+2}$ записывается в форме группового уравнения вида

$$g(y_1, y_2, \ldots, y_{k+2}) = f(c) \pmod{\gamma_{s+3}M_{k+2}}, \tag{8}$$

где в левой части порождающие $x_1, x_2, \ldots, x_{k+2}$ заменены на соответствующие неизвестные $y_1, y_2, \ldots, y_{k+2}$. В правой части стоит произведение базисного коммутатора $[x_2, x_1, x_1, x_2]$ на степень $c$ фиксированного базисного коммутатора $h = h(x_1, x_2)$ веса $s + 2$ из (5), то есть элемент фиксированного смежного класса по циклической подгруппе гр$(h)$.

Известно [36], что любая вербальная подгруппа группы $M_n$, в частности любая подгруппа $\gamma_i M_n$, для произвольного $n$ имеет конечную вербальную ширину. Это означает, что для любой пары натуральных чисел $n, i$ существует слово $w_{n,i} = w_{n,i}(z_1, \ldots, z_{p_{n,i}})$, такое, что любой элемент подгруппы $\gamma_i M_n$ является значением этого слова. Точный вид слова $w_{n,i}$ указан, например, в [3]. Поэтому уравнение (8) разрешимо в группе $M_{k+2}$ тогда и только тогда, когда в группе $M_{k+2}$ разрешимо уравнение

$$g(y_1, y_2, \ldots, y_{k+2})w_{k+2,s+3}(z_1, \ldots, z_{p_{k+2,s+3}}) = f(c). \tag{9}$$

В свою очередь, алгоритмическая неразрешимость уравнений вида (9) означает неразрешимость проблемы эндоморфной сводимости в группе $M_r$, где $r = k + 2 + {} + p_{k+2,s+3}$. Очевидно, что проблема эндоморфной сводимости будет неразрешима и для любого большего чем $r$ ранга.

## 3. Общая схема построения односторонних функций на основе неразрешимости проблемы эндоморфной сводимости

Пусть $G$ — эффективно заданная группа, в которой разрешима проблема равенства и неразрешима проблема эндоморфной сводимости. Допустим, что нам необходимо построить на группе $G$ одностороннюю функцию со значением также в $G$. Как правило, эффективно заданные группы конечно порождены. Поэтому предполагаем, что в $G$ существует конечное порождающее множество $X_n = \{x_1, \ldots, x_n\}$. Произвольный элемент $g$ группы $G$ записывается как групповое слово $g = g(x_1, \ldots, x_n)$ от фиксированных порождающих элементов. Каждый эндоморфизм $\varphi : G \to G$ однозначно определяется своими значениями на элементах порождающего множества $X_n$. Если $\varphi(x_i) = h_i$ для $i = 1, \ldots, n$, то значением элемента $g = g(x_1, \ldots, x_n)$ будет элемент $\varphi(g) = g(h_1, \ldots, h_n)$. Однако не любое отображение $\varphi : X_n \to G$ продолжается до эндоморфизма в общем случае. Поэтому представляется перспективным выбирать в качестве группы $G$ свободную группу конечного ранга некоторого многообразия групп $\mathcal{L}$. Тогда любое отображение $\varphi : X_n \to G$, где $X_n$ — базис группы $G$ (т. е. множество свободных порождающих группы $G$ в многообразии $\mathcal{L}$), однозначно определяет эндоморфизм $\varphi : G \to G$. Для его задания достаточно определить образы базисных элементов, записав их в виде групповых слов от этих элементов. Ещё лучше, если в группе $G$ есть нормальная форма записи элементов. Тогда эндоморфизм можно задать, записывая образы базисных элементов в этой нормальной форме. В работе [3] отмечено, что свободные метабелевы группы отвечают описанным требованиям.

Определим функцию $\varphi : G \to G$, сопоставляющую элементам группы, записанным в нормальной форме, нормальные формы их значений относительно эндоморфизма $\varphi$. Для эффективности соответствующих вычислений необходимо, чтобы в группе $G$ существовал эффективный алгоритм записи элемента в нормальной форме по его представлению в виде группового слова от порождающих элементов или в каком-то другом виде, отвечающем эффективности задания группы $G$. Необходима также эффективная процедура вычисления нормальных форм обратного элемента и произведения элементов. Если группа $G$ удовлетворяет этим требованиям, то получаем эффективно вычислимую функцию, определённую на множестве нормальных форм элементов группы $G$ со значениями в этом же множестве. Если определить для элементов группы $G$ некоторую функцию длины, например ввести на ней словарную метрику, то не существует полиномиального алгоритма, ограничивающего длину прообраза по длине образа для данной функции. Более того, никакая рекурсивная функция не даст такого ограничения. Не существует и других эффективных процедур, сводящих решение задачи поиска аргумента по значению функции к ограниченному полному перебору.

При этом, однако, возникает существенный вопрос об обращении функции на генерическом множестве значений. Возможно ли найти генерическое подмножество в множестве всех значений функции относительно естественной меры на этом множестве, на котором проблема поиска решается эффективно? Этот вопрос требует специального исследования в каждом конкретном случае (см. по этому поводу обзор [3] или более детальное изложение в [1, 2, 37]).

Рассмотрим элемент $g = g(x_1, x_2, \ldots, x_{k+2})$, построенный выше. Определим образ элемента $g$ относительно эндоморфизма $\mu$, фиксирующего $x_1$ и $x_2$ и отображающего $x_j$ для $j = 3, \ldots, k+2$ в $x_j^{\beta_{j-2}}$ для некоторых целых чисел $\beta_1, \ldots, \beta_k$. В результате получим по модулю $\gamma_{s+3}M_{k+2}$ элемент $\mu(g) = f(x_1, x_2)^c$. Можно считать, что определена функция $\bar{g} : \mathbb{Z}^k \to \mathbb{Z}$ по правилу $(\beta_1, \ldots, \beta_k) \mapsto c$. Эту функцию можно рассматривать как возможно одностороннюю. Её полиномиальная вычислимость очевидна. В то же время из-за неразрешимости проблемы эндоморфной сводимости элемента $g$ к элементу вида $f(x_1, x_2)^c$ не существует полиномиального по времени алгоритма, вычисляющего по $c$ набор $(\beta_1, \ldots, \beta_k)$.

Отсюда можно вывести следующее общее заключение. Если бы длины значений $\varphi(x_i) = h_i$ для $i = 1, \ldots, n$, соответствующие эндоморфизму $\varphi$, переводящему $g$ в $f = f(c)$, можно было оценить сверху полиномиальными функциями от $c$, то проблема эндоморфной сводимости $g$ к $f$ в группе $M_{k+2}$ была бы разрешимой. Поскольку это не так, подобной оценки не существует. Это приводит к невозможности переборного решения проблемы.

Относительно возможной эффективной обратимости рассматриваемой функции на генерическом множестве значений заметим, что в работе А. Н. Рыбалова [38] доказано, что 10-я Проблема Гильберта остается алгоритмически неразрешимой на любом строго генерическом множестве диофантовых уравнений.

## 4. Протокол аутентификации

Впервые идея использования рассматриваемой схемы для построения протокола аутентификации на платформе группы с разрешимой проблемой равенства и неразрешимой проблемой эндоморфной сводимости была высказана В. А. Романьковым в докладе на семинаре в Graduate Center City University of New York в 2007 г. Впоследствии идея была описана в работе Д. Григорьева и В. Шпильрайна [39]. Авторы опирались на результаты В. А. Романькова из [3, 10].

В общих чертах протокол выглядит следующим образом.

У с т а н о в к а. Выбирается бесконечная эффективно заданная группа $G$ с разрешимой проблемой равенства и неразрешимой проблемой эндоморфной сводимости. Абонент **A** фиксирует публичный элемент $g$ и секретный эндоморфизм $\varphi$, вычисляет и публикует образ $f = \varphi(g)$. Элементы $g$ и $f$ выбираются таким образом, чтобы проблема эндоморфной сводимости для пары $(g, f)$ была трудна. Это означает, что по $f$ трудно вычислить эндоморфизм $\varphi$, переводящий $g$ в $f$.

А л г о р и т м а у т е н т и ф и к а ц и и.

1) В качестве сессионного ключа выбирается эндоморфизм (в работе [39] автоморфизм) $\psi$, вычисляется элемент $v = \psi(f)$ и передается в систему **C**, в которой осуществляется аутентификация пользователя **A**.
2) Система **C** с равной вероятностью выбирает случайный бит и отсылает его **A**.
3) Если **A** получает 0, то он просто публикует $\psi$, а **C** проверяет, что действительно $v$ — образ $f$ относительно $\psi$. Если **A** получает 1, то он вычисляет композицию $\chi = \varphi\psi$, передает её **C**, который проверяет справедливость равенства $v = \chi(g)$.

Схема выглядит как протокол аутентификации с нулевым разглашением, аналогичный известному протоколу Фиата — Шамира (см., например, [25, 26]). Однако для её криптостойкости необходимо выполнение ряда дополнительных условий. Во-первых, необходимо, чтобы существовал фигурирующий в протоколе элемент $g$, проблема вхождения в множество эндоморфных образов которого алгоритмически неразрешима. Как объяснялось выше, это возможно, если в качестве группы $G$ выбрать свободную метабелеву группу $M_n$ достаточно большого ранга $n$. В этом случае элемент $f$ можно брать случайным образом из фиксированного смежного класса по циклической подгруппе $\mathrm{гр}(h)$, полагая $f = f(c)$ в соответствии с (4). Но далее аналогичные условия должны быть выполнены для пары элементов $f$, $v$. Результаты работы [10] уже не позволяют считать, что проблема вхождения в множество эндоморфных образов элемента $f$ алгоритмически неразрешима. Но даже если бы это было так, приведённых условий все равно оказалось бы недостаточно.

Для разрешения этой ситуации можно ввести определение группы с неразрешимой двукратной проблемой эндоморфной сводимости, а именно можно доказать, что в свободной метабелевой группе $M_n$ достаточно большого ранга можно выбрать элемент $g$, элемент $b$, циклическую подгруппу $\mathrm{гр}(h)$, элемент $u$ и конечно порождённую абелеву группу $A$ таким образом, что для любой пары элементов $(g, f)$, где $f \in b\mathrm{гр}(h)$, и любой пары элементов $(f, v)$, где $v \in uA$, одновременно алгоритмически неразрешимы проблемы эндоморфной сводимости. Более того, знание эндоморфизма $\chi$, такого, что $\chi(g) = v$, не позволяет эффективно находить ни эндоморфизм $\varphi$, такой, что $\varphi(g) = f$, ни эндоморфизм $\psi$, такой, что $\psi(f) = v$. Наоборот, знание эндоморфизма $\psi$, такого, что $\psi(f) = v$, не позволяет восстановить ни $\varphi$, ни $\chi$. Легко видеть, что эти условия являются необходимыми в обеспечении криптостойкости приведённого алгоритма.

Соответствующие построения проводятся эффективно. Они также основываются на неразрешимости 10-й Проблемы Гильберта и интерпретации диофантовых уравнений в свободных метабелевых группах.

Заметим также, что близкой по теме проблеме построения двукратной возможно односторонней функции посвящена работа первого автора [40]. Отсюда видно, что создание криптографических приложений, основанных на проблемах теории групп, имеет обратное влияние на развитие самой теории групп. Постановка алгоритмических проблем приобретает новые формы. Отметим в этой связи возросший интерес

к проблемам поиска. Вопросы теории сложности становятся все более актуальными и также приобретают новые формы. Достаточно ещё раз упомянуть понятие генерической сложности проблемы, возникшее главным образом при исследовании практических алгоритмов.

## ЛИТЕРАТУРА


1. *Myasnikov A., Shpilrain V., and Ushakov A.* Group-based cryptography. (Advances courses in Math., CRM, Barselona). Basel, Berlin, New York: Birkhäuser Verlag, 2008. 183 p.
2. *Myasnikov A., Shpilrain V., and Ushakov A.* Non-commutative cryptography and complexity of group-theoretic problems. (Amer. Math. Soc. Surveys and Monographs). Providence, RI: Amer. Math. Soc., 2011. 385 p.
3. *Романьков В. А.* Диофантова криптография на бесконечных группах // Прикладная дискретная математика. 2012. № 2. С. 15–42.
4. *Goldreich O.* Foundations of cryptography. Cambridge: Cambridge Univ. Press, 2001. V. 1. 451 p.; 2004. V. 2. 798 p.
5. *Goldwasser S. and Bellare M.* Lecture Notes on Cryptography. Cambridge: MIT, 2008.
6. *Sipser M.* Introduction to the theory of computation. PWS Publishing, 1997. 416 p.
7. *Levin L. A.* One-way Functions and Pseudorandom Generators // Combinatorica. 1987. V. 7. No. 4. P. 357–363.
8. *Левин Л. А.* Односторонние функции // Проблемы передачи информации. 2003. Т. 39. № 1. С. 103–117.
9. *Кожевников А. А., Николенко С. И.* О полных односторонних функциях // Проблемы передачи информации. 2009. Т. 45. № 2. С. 101–118.
10. *Романьков В. А.* Об уравнениях в свободных метабелевых группах // Сибирский математический журнал. 1979. Т. 20. № 3. С. 671–673.
11. *Романьков В. А.* О неразрешимости проблемы эндоморфной сводимости в свободных нильпотентных группах и в свободных кольцах // Алгебра и логика. 1977. Т. 16. № 4. С. 457–471.
12. *Матиясевич Ю. В.* Диофантовость перечислимых множеств // Докл. АН СССР. 1970. Т. 191. № 2. С. 279–282.
13. *Матиясевич Ю. В.* Диофантово представление перечислимых предикатов // Изв. АН СССР. Сер. матем. 1971. № 35. С. 3–30.
14. *Матиясевич Ю. В.* Десятая проблема Гильберта. М.: Наука, 1993. 223 с.
15. *Shpilrain V. and Zapata G.* Using decision problems in public key cryptography // Groups. Complexity. Cryptology. 2009. V. 1. P. 33–40.
16. *Shpilrain V. and Zapata G.* Using the subgroup membership problem in public key cryptography // Contemp. Math. V. 418. Providence, RI: Amer. Math. Soc., 2006. P. 169–179.
17. *Shpilrain V. and Zapata G.* Combinatorial group theory and public key cryptography // Applicab. Alg. Eng. Comm. Comp. 2006. V. 17. P. 291–302.
18. *Kurt Y.* A new key exchange primitive based on the triple decomposition problem // Preprint: `http://eprint.iacr.org/2006/378`
19. *Birget J.-C., Magliveras S., and Sramka M.* On public-key cryptosystems based on combinatorial group theory // Tatra Mount. Math. Publ. 2006. V. 33. P. 137–148.
20. *Lohrey M.* Word problems on compressed words // Automata, Languages and Programming. LNCS. 2004. V. 3142. P. 906–918.
21. *Lohrey M. and Schleimer S.* Efficient computation in groups via compression // LNCS. 2007. V. 4649. P. 249–258.

22. *Scheimer S.* Polynomial-time word problems // Comment. Math. Helv. 2008. V. 83. No. 4. P. 741–765.
23. *Ерофеев С. Ю.* Диофантовость дискретного логарифма // Прикладная дискретная математика. Приложение. 2011. № 4. С. 31–32.
24. *Ерофеев С. Ю.* Диофантовость дискретного логарифма // Вестник Омского университета. 2010. № 4. С. 13–15.
25. *Menezes A. J., van Oorschot P. C., and Vanstone S. A.* Handbook of Applied Cryptography. CRC Press, 1996. 816 p.
26. *Романьков В. А.* Введение в криптографию. Курс лекций. М.: Форум, 2012. 240 с.
27. *Matijasevich Y. V. and Robinson J.* Reduction of an arbitrary diophantine equation to one in 13 unknowns // Acta Arithmetica. 1975. V. 27. P. 521–553.
28. *Matijasevich Y. V.* Some purely mathematical results inspired by mathematical logic // Proc. Fifth Intern. Congr. Logic, Methodology and Philos. of Sci.(London, Ont.). Dordrecht, Reidel, Holland. 1977. P. 121–127.
29. *Jones J.* Universal diophantine equation // J. Symbolic Logic. 1982. V. 47. No. 3. P. 549–571.
30. *Нейман Х.* Многообразия групп. М.: Мир, 1974. 264 с.
31. *Маканин Г. С.* Уравнения в свободной группе // Изв. АН СССР. Сер. матем. 1982. Т. 46. № 6. С. 1199–1273.
32. *Hall P.* Nilpotent groups // Canad. Math. Cong. Summer Sem. Vankuver: University of Alberta, 1957. P. 12–30.
33. *Холл М.* Теория групп. М.: ИЛ, 1962. 467 с.
34. *Каргаполов М. И., Мерзляков Ю. И.* Основы теории групп. М.: Лань, 2009. 288 с.
35. *Курош А. Г.* Теория групп. М.: Физматгиз, 2008. 808 с.
36. *Segal D.* Words: notes on verbal width in groups // London Math. Soc. Lect. Notes. Cambridge: Cambridge Univ. Press, 2009. V. 361. 215 p.
37. *Kapovich I., Myasnikov A., Shupp P., and Shpilrain V.* Generic-case complexity, decision problems in group theory and random walks // J. Algebra. 2003. V. 264. P. 665–694.
38. *Рыбалов А. Н.* О генерической неразрешимости десятой проблемы Гильберта // Вестник Омского университета. 2011. № 4. С. 19–22.
39. *Grigoriev D. and Shpilrain V.* Zero-knowledge autentification schemes from actions on graphs, groups and rings // Ann. Pure Appl. Logic. 2010. V. 162. P. 194–200.
40. *Ерофеев С. Ю.* Схемы построения двушагово односторонних функций // Вестник Омского университета. 2011. № 4. С. 15–18.

2012 Теоретические основы прикладной дискретной математики №3(17)

DOI 10.17223/20710410/17/3

УДК 519.719.325

# О СОВПАДЕНИИ КЛАССА БЕНТ-ФУНКЦИЙ С КЛАССОМ ФУНКЦИЙ, МИНИМАЛЬНО БЛИЗКИХ К ЛИНЕЙНЫМ[1]

В. И. Солодовников

*Академия криптографии Российской Федерации, г. Москва, Россия*

**E-mail:** vis0707@rambler.ru

Продолжается начатое ранее исследование вопросов близости функций из $(\mathbb{Z}/(p))^n$ в $(\mathbb{Z}/(p))^m$ ($p$ — простое) к линейным функциям. Найдены новые критерии абсолютно минимальной близости функции к линейным. Доказывается, что такая минимальность функции наследуется её гомоморфными образами. Обобщая хорошо известный для булевых функций факт, доказывается, что для $p = 2, 3$ класс всех абсолютно минимально близких к линейным функций совпадает с классом бент-функций.

**Ключевые слова:** *близость функций, абсолютно негомоморфные функции, минимальные функции, бент-функции.*

Нам потребуются следующие обозначения, терминология и результаты работы [1]:

- $X$, $Y$ — нетривиальные аддитивные конечные абелевы группы;
- $Y^X$ — множество всех отображений $f : X \to Y$ из множества $X$ в множество $Y$ (термины «отображение» и «функция» считаем синонимами);
- $\psi\varphi$ — композиция отображений, при которой $\varphi$ действует первым;
- $\mathrm{Hom}(G, H)$ — множество всех гомоморфизмов группы $G$ в группу $H$;
- $\mathbb{Z}$ — кольцо целых чисел, $\mathbb{Z}/(k)$ — кольцо вычетов по модулю $k$;
- $P$ — равномерное вероятностное распределение на $X$, то есть

$$P(A) = |A||X|^{-1} \text{ для любого } A \subseteq X.$$

Функция $f : X \to Y$ называется *сбалансированной*, если мощности полных прообразов всех элементов $Y$ одинаковы.

В работе [1] близость двух произвольных функций $f_1, f_2 \in Y^X$ измеряется корнем из дисперсии

$$(|Y|^{-1} \sum_{y \in Y} (P(f_1 - f_2 = y) - |Y|^{-1})^2)^{\frac{1}{2}}.$$

В работе [2] предложено нормировать эту величину так, чтобы она достигала 1, то есть в качестве меры близости функций рассматривать

$$(|Y|(|Y| - 1)^{-1} \sum_{y \in Y} (P(f_1 - f_2 = y) - |Y|^{-1})^2)^{\frac{1}{2}}.$$

Это позволило упростить многие формулы. С целью дальнейших упрощений здесь мы откажемся и от корня. А именно, *близостью* $\delta$ функций $f_1, f_2 \in Y^X$ определим равенством

$$\delta(f_1, f_2) = |Y|(|Y| - 1)^{-1} \sum_{y \in Y} (P(f_1 - f_2 = y) - |Y|^{-1})^2, \tag{1}$$

[1]Работа выполнена в рамках НИР, проведённой в Академии криптографии Российской Федерации в 2010 г., и является расширенным вариантом статьи (добавлены теорема 5 и второй абзац снизу до списка литературы, изменены некоторые обозначения), опубликованной в журнале «Математические вопросы криптографии», 2011, т. 2, вып. 4, с. 97–108.

а близость классов функций $K_1, K_2 \subseteq Y^X$ — равенством

$$\delta(K_1, K_2) = \max_{\substack{f_1 \in K_1, \\ f_2 \in K_2}} \delta(f_1, f_2).$$

Близость $\delta$ обладает следующими свойствами:

$$0 \leqslant \delta(f_1, f_2) \leqslant 1; \tag{2}$$

$$\delta(f_1, f_2) = 0 \Leftrightarrow f_1 - f_2 \text{— сбалансированная}; \tag{3}$$

$$\delta(f_1, f_2) = 1 \Leftrightarrow f_1 - f_2 = \text{const}; \tag{4}$$

$$\delta(f_1, f_2) = \delta(f_2, f_1); \tag{5}$$

$$\delta(y_1 + g_2 f_1 g_1 + f', y_2 + g_2 f_2 g_1 + f') = \delta(f_1, f_2) \tag{6}$$

*для любых* $f_1, f_2 \in Y^X$, *сбалансированной* $g_1 : X' \to X$, изоморфизма $g_2 : Y \to Y'$, $f' : X' \to Y'$, $y_1, y_2 \in Y'$.

Свойства (2)–(4) означают, что минимально близкие функции — это функции, разность которых сбалансирована, а максимально близкие функции — это функции, разность которых есть константа.

Для любого $a \in X$ подстановку $x \mapsto a + x$ множества $X$ будем обозначать через $a^+$. Все такие подстановки называют *сдвигами* группы $X$. Они образуют изоморфную группе $X$ группу подстановок, называемую группой сдвигов (или группой Кэли) группы $X$ (в теории представлений групп такое представление $X$ называют регулярным).

Гомоморфизмы $h \in \mathrm{Hom}(X, Y)$ по определению обладают следующим характерным свойством: $ha^+ - h = \text{const}$ для всех $a \in X \setminus \{0\}$. Поэтому естественным (в силу свойств (2)–(4)) является следующее определение, обобщающее определение «совершенной нелинейности» в [3].

Функция $f : X \to Y$ называется *абсолютно негомоморфной*, если для любого $a \in X \setminus \{0\}$ функция $fa^+ - f$ является сбалансированной, то есть

$$\delta(fa^+, f) = 0. \tag{7}$$

Множество всех абсолютно негомоморфных функций из $Y^X$ обозначается через $AN(Y^X)$. В частности, $AN(Y^X) = \varnothing$ при $|X| < |Y|$.

К понятию абсолютной негомоморфности можно подойти не только с алгебраической, но и с криптографической стороны. А именно, пусть $G$ — некоторая группа подстановок множества $X$, $fG = \{fg : g \in G\}$ — класс функций, порождённых функцией $f \in Y^X$ и группой $G$. Пусть в некоторой криптосхеме имеется узел, реализующий функции из $fG$, где элементы группы $G$ являются ключами. Тогда величины $\delta(fg_1, fg_2)$, $g_1, g_2 \in G$, характеризуют степень изменения этого узла ключами из $G$: чем ближе эти величины к 0, тем сильнее подстановки из $G$ изменяют функцию $f$. Наоборот, если величина $\delta(fg_1, fg_2)$ близка к 1, то ключи $g_1$ и $g_2$ называют *близкими*. Поскольку, в силу (6), $\delta(fg_1, fg_2) = \delta(fg_1 g_2^{-1}, f)$, то достаточно рассматривать только величины $\delta(fg, f)$, $g \in G$. В частности, когда $G$ — группа сдвигов группы $X$, получаем, что абсолютно негомоморфные функции — это функции, которые *максимально изменяются сдвигами*, то есть соответствующие узлы *не имеют различных близких ключей-сдвигов*.

В работе [1] доказано, что если группа $G$ транзитивна, то для любой $f \in Y^X$

$$\delta(f, 0) = |Y|(|G|(|Y| - 1))^{-1} \sum_{g \in G} (P(fg = f) - |Y|^{-1})$$

и, в частности, для любого $h \in \mathrm{Hom}(X, Y)$

$$\delta(f, h) = |Y|(|X|(|Y| - 1))^{-1} \sum_{a \in X} (P(fa^+ - f = h(a)) - |Y|^{-1}). \tag{8}$$

Приведём некоторые свойства абсолютно негомоморфных функций.

Если $f \in AN(Y^X)$, $h \in \mathrm{Hom}(X, Y)$, $g_1 : X' \to X$ — изоморфизм, $g_2 : Y \to Y'$ — эпиморфизм (сюръективный гомоморфизм групп), $c \in X'$, $b \in Y'$, $h' \in \mathrm{Hom}(X', Y')$, то

$$\delta(f, h) = |X|^{-1}; \tag{9}$$

$$b + h' + g_2 f g_1 c^+ \in AN(Y'^{X'}). \tag{10}$$

Из формулы (9) следует, что абсолютно негомоморфные функции одинаково близки к любому гомоморфизму, а также что они не являются сбалансированными, поскольку сбалансированность равносильна условию $\delta(f, 0) = 0$. В частности, абсолютно негомоморфные функции не могут быть биективными.

В соответствии с теоретико-автоматной терминологией и с учетом групповой структуры алфавитов, пару $(\alpha, \beta)$ гомоморфизмов $\alpha \in \mathrm{Hom}(X, X')$ и $\beta \in \mathrm{Hom}(Y, Y')$ назовем *гомоморфизмом функции* $f : X \to Y$ *в функцию* $f' : X' \to Y'$, если $\beta f = f'\alpha$. Если $\alpha$ и $\beta$ — сюръекции, то гомоморфизм $(\alpha, \beta)$ назовем эпиморфизмом, а функцию $f'$ — гомоморфным образом функции $f$. В этом случае для любых $a \in X$ и $y' \in Y'$

$$P'(f'(\alpha(a))^+ - f' = y') - |Y'|^{-1} = \sum_{y \in \beta^{-1}(y')} (P(fa^+ - f = y) - |Y|^{-1}), \tag{11}$$

где $P'(A') = |A'||X'|^{-1}$ для любого $A' \subseteq X'$, и следовательно, справедлива

**Теорема 1.** Если $(\alpha, \beta)$ — эпиморфизм функции $f \in AN(Y^X)$ в функцию $f' : X' \to Y'$ и $|Y'| > 1$, то $\alpha$ — изоморфизм и $f' \in AN(Y'^{X'})$.

Минимальная близость функций из $Y^X$ к гомоморфизмам обозначается через

$$\delta_0(Y^X) = \min_{f \in Y^X} \delta(f, \mathrm{Hom}(X, Y)).$$

*Минимальными* называются функции, минимально близкие к гомоморфизмам, то есть функции $f \in Y^X$, для которых

$$\delta(f, \mathrm{Hom}(X, Y)) = \delta_0(Y^X).$$

Множество всех минимальных функций из $Y^X$ обозначим через $M(Y^X)$.

Заметим, что минимальные функции существуют всегда, в отличие от абсолютно негомоморфных функций.

Хорошо известно, что любая абелева группа изоморфна прямому произведению аддитивных групп колец вычетов. Поэтому, и в силу формул (6) и (10), без ограничения общности далее считаем, что

$$X = \prod_{i=1}^{n} \mathbb{Z}/(k_i),\ Y = \prod_{j=1}^{m} \mathbb{Z}/(t_j),$$

где $n$, $m$ — произвольные натуральные числа, $k_1, \ldots, k_n$, $t_1, \ldots, t_m$ — произвольные большие 1 натуральные числа.

В работе [1] для функций из $Y^X$ введено понятие *бент-функции* как функции, обладающей следующим свойством: в разложении композиции её и любого неединичного

неприводимого (комплекснозначного) характера группы $Y$ по неприводимым характерам группы $X$ модули всех коэффициентов (они называются коэффициентами Фурье) одинаковы. Множество всех бент-функций из $Y^X$ обозначим через $B(Y^X)$.

Это определение обобщает определения работы [4] (для булевых функций, то есть $m = 1$, $k_1 = \ldots = k_n = t_1 = 2$), работы [5] (для $k_1 = \ldots = k_n = t_1 = \ldots = t_m = p$ — простое число, $m$ делит $n$) и может быть сведено к определению работы [6], где бент-функции определяются как комплекснозначные функции на конечной абелевой группе с единичными модулями всех значений и условием равенства модулей всех коэффициентов Фурье. Для случая $m = 1$, $k_1 = \ldots = k_n = t_1 \geqslant 3$ определение из [1] сужает определение бент-функции в [7] (где участвует только один характер группы $Y$), но в [7] доказано, что при простом $t_1$ эти определения равносильны. В замечательной книге [8] приводится обстоятельный обзор бент-тематики.

В [1] доказано:

$$AN(Y^X) = B(Y^X),$$

что распространяет соответствующие результаты работ [4, 5, 7] на случай произвольных конечных абелевых групп $X$ и $Y$.

Для любого $h \in \mathrm{Hom}(X, Y)$ *матрица гомоморфизма* $h$

$$A_h = (a_{i,j} + (t_j))_{n \times m}$$

определяется равенствами

$$h(e_i) = (a_{i,1} + (t_1), \ldots, a_{i,m} + (t_m)),$$
$$e_i = ((k_1), \ldots, (k_{i-1}), 1 + (k_i), (k_{i+1}), \ldots, (k_n)),\ i = 1, \ldots, n.$$

Она обладает следующими свойствами:

$$\frac{t_j}{(k_i, t_j)} \mid a_{i,j} \tag{12}$$

для любых $i = 1, \ldots, n$, $j = 1, \ldots, m$ (здесь $(k_i, t_j)$ — наибольший общий делитель чисел $k_i$ и $t_j$, а $a \mid b$ обозначает, что $a$ делит $b$),

$$h((x_1 + (k_1), \ldots, x_n + (k_n))) = (x_1, \ldots, x_n)A_h$$

для любых $x_1, \ldots, x_n \in \mathbb{Z}$. Следовательно, матрица $A_h$ однозначно определяет сам гомоморфизм $h$. Наоборот, если $A = (a_{i,j} + (t_j))_{n \times m}$ — произвольная матрица со свойством (12), то соответствие $h_A : X \to Y$, определяемое равенством

$$h_A((x_1 + (k_1), \ldots, x_n + (k_n))) = (x_1, \ldots, x_n)A,$$

является гомоморфизмом групп. Таким образом, между $\mathrm{Hom}(X, Y)$ и всеми матрицами со свойством (12) существует взаимно-однозначное соответствие и, следовательно,

$$|\mathrm{Hom}(X, Y)| = \prod_{\substack{i=1,\ldots,n,\\ j=1,\ldots,m}} (k_i, t_j).$$

Следующая, доказанная в [1], теорема распространяет известный результат работы [9] для булевых функций на случай абелевых групп.

**Теорема 2.** Если $k_i | t_j$ для любых $i = 1, \ldots, n$, $j = 1, \ldots, m$, то для любой функции $f : X \to Y$ набор чисел

$$(P(f = y + h) : y \in Y, h \in \mathrm{Hom}(X, Y))$$

однозначно определяет функцию $f$.

Следующие две теоремы (см. [1, 3]) сводят случай многомерного $Y$ к одномерному и обобщают теорему 3 из [5].

**Теорема 3.** Если $k_i = t_j = t$ для любых $i = 1, \ldots, n$, $j = 1, \ldots, m$, то для любой $f : X \to Y$ следующие свойства равносильны:

1) $f$ — сбалансированная;

2) $hf$ — сбалансированная для любого эпиморфизма $h : Y \to \mathbb{Z}/(t)$.

**Теорема 4.** Если $k_i = t_j = t$ для любых $i = 1, \ldots, n$, $j = 1, \ldots, m$, то для любой $f : X \to Y$ следующие свойства равносильны:

1) $f \in B(Y^X)$;

2) $hf \in B(\mathbb{Z}/(t)^X)$ для любого эпиморфизма $h : Y \to \mathbb{Z}/(t)$.

**Лемма 1.** Следующие свойства равносильны:

а) для любых $x \in X \backslash \{0\}$, $y \in Y$ существует $h \in \mathrm{Hom}(X, Y)$, такой, что $h(x) = y$;

б) $k_1 = \ldots = k_n = t_1 = \ldots t_m = p$ — простое число.

***Доказательство.*** Импликация *б* $\Rightarrow$ *а* очевидна, так как в случае *б* гомоморфизмы являются линейными отображениями (функциями) векторных пространств над полем $\mathbb{Z}/(p)$. Наоборот, пусть выполнено *а*. Тогда, в силу (12), $t_j \mid k_i$ для любых $i = 1, \ldots, n$, $j = 1, \ldots, m$. Для любого $h \in \mathrm{Hom}(X, Y)$ в $h(t_j e_i)$ $j$-я координата равна 0 и, следовательно, $t_j = k_i = p = c_1 c_2$, где $c_1 > 1$. Тогда $c_1 h(c_2 e_1) = 0$ для любого $h \in \mathrm{Hom}(X, Y)$ и, следовательно, $c_2 = 1$. ■

Условие *а* существенно для дальнейших рассуждений. Поэтому далее рассмотрим только случай выполнения условия *б* леммы 1, $p > 1$, который назовем *примарным* (или *$p$-примарным*), при этом функции из $Y^X$ — *примарными* (или *$p$-примарными*). В этом случае гомоморфизмы из $\mathrm{Hom}(X, Y)$ являются линейными функциями (отображениями векторных пространств) и

$$|\mathrm{Hom}(X, Y)| = |Y|^n = p^{nm}.$$

2-Примарный случай называют двоичным. Двоичный случай для $m = 1$ называют булевым. С практической точки зрения оба эти случая наиболее интересны.

**Теорема 5.** Для $p$-примарного случая следующие утверждения равносильны:

1) $B(Y^X) \neq \varnothing$;

2) выполняется одно из условий:

$$p = 2, \;\; 2 \,|\, n \text{ и } n \geqslant 2m,$$
$$p \geqslant 3 \text{ и } n \geqslant m.$$

***Доказательство.*** В [3, 5] указаны следующие два примера абсолютно негомоморфных функций. Если $2 \,|\, n$, то операция умножения в поле $\mathrm{GF}(p^{n/2})$ является абсолютно негомоморфной функцией $\mathrm{GF}(p^{n/2})^2 \to \mathrm{GF}(p^{n/2})$. Если $p \geqslant 3$, то возведение в квадрат в поле $\mathrm{GF}(p^n)$ является абсолютно негомоморфной функцией $\mathrm{GF}(p^n) \to \mathrm{GF}(p^n)$. Отсюда и из свойства (10) следует справедливость импликации $2 \Rightarrow 1$.

Наоборот, пусть выполнено утверждение 1. Тогда, по определению сбалансированности, $n \geqslant m$. Пусть $p = 2$. В [3] доказано, что если $f : X \to Y$ — бент-функция, то мощность полного прообраза любого $y \in Y$ при отображении $f$ равна $2^{n/2-m}k(f, y)$, где $k(f, y)$ — нечётное число (в частности, любая двоичная бент-функция сюръективна). ■

В [1] доказана

**Лемма 2.** Для любой примарной $f : X \to Y$

$$|Y|^{-n} \sum_{h \in \mathrm{Hom}(X,Y)} \delta(f, h) = |X|^{-1}.$$

***Доказательство.*** Приводимое ниже доказательство проще, чем в [1], и не использует коэффициенты Фурье.

Для любого $a \in X\backslash\{0\}$ отображение $\mathrm{Hom}(X, Y) \to Y$, где $h \mapsto h(a)$, является гомоморфизмом групп, который сюръективен в силу леммы 1. Тогда, в силу (8),

$$|Y|^{-n} \sum_{h \in \mathrm{Hom}(X,Y)} \delta(f, h) =$$

$$= |Y|^{-n}|Y|(|X|(|Y| - 1))^{-1}(|Y|^n(1 - |Y|^{-1}) + \sum_{0 \neq a \in X} \sum_{h \in \mathrm{Hom}(X,Y)} (P(fa^+ - f = h(a)) - |Y|^{-1})) =$$

$$= |Y|^{-n}|Y|(|X|(|Y| - 1))^{-1}(|Y|^n(1 - |Y|^{-1}) + (|X| - 1)(|Y|^{n-1}1 - |Y|^n|Y|^{-1})) = |X|^{-1}.$$

Лемма доказана. ■

Из этой леммы получаем оценку минимальной близости функций к линейным функциям.

**Теорема 6.** Для примарного случая

$$\delta_0(Y^X) \geqslant |X|^{-1}.$$

В случае выполнения условия $\delta_0(Y^X) = |X|^{-1}$ минимальные функции назовем *абсолютно минимальными* (в силу теоремы 9 это определение сужает определение «абсолютной минимальности» в работе [1]). Множество всех абсолютно минимальных функций из $Y^X$ обозначим через $AM(Y^X)$, так что

$$M(Y^X) = AM(Y^X), \text{ если } \delta_0(Y^X) = |X|^{-1},$$
$$AM(Y^X) = \varnothing, \text{ если } \delta_0(Y^X) > |X|^{-1}.$$

Из формулы (9) и теоремы 6 следует

**Теорема 7.** В примарном случае $AN(Y^X) \subseteq AM(Y^X)$.

В следующей теореме собраны критерии абсолютной минимальности примарных функций.

**Теорема 8.** Для любой $p$-примарной $f : X \to Y$ следующие утверждения равносильны:

1) $f \in AM(Y^X)$;
2) $\delta(f, h) = |X|^{-1}$ для любого $h \in \mathrm{Hom}(X, Y)$;
3) $\delta(f, h) = \delta(f, h')$ для любых $h, h' \in \mathrm{Hom}(X, Y)$;
4) $\sum\limits_{0 \neq a \in X} (P(fa^+ - f = h(a)) - |Y|^{-1}) = 0$ для любого $h \in \mathrm{Hom}(X, Y)$;
5) $\sum\limits_{0 \neq a \in X} (P(fa^+ - f = h(a)) - |Y|^{-1}) = \sum\limits_{0 \neq a \in X} (P(fa^+ - f = h'(a)) - |Y|^{-1})$ для любых $h, h' \in \mathrm{Hom}(X, Y)$;

6) $\sum\limits_{0\neq c\in Z/(p)}(P(f(ca)^+ - f = cy) - |Y|^{-1}) = 0$ для любых $a \in X\backslash\{0\}$, $y \in Y$;

7) $\sum\limits_{0\neq c\in Z/(p)}(P(f(ca)^+ - f = cy) - |Y|^{-1}) = \sum\limits_{0\neq c\in Z/(p)}(P(f(ca)^+ - f = cy') - |Y|^{-1})$ для любых $a \in X\backslash\{0\}$, $y, y' \in Y$ .

***Доказательство.*** Импликация $1 \Rightarrow 2$ следует из определения минимальности и леммы 2. Очевидно, что $2 \Rightarrow 3$. Импликация $3 \Rightarrow 1$ следует из леммы 2, теоремы 6 и определения минимальности. Из формулы (8) следуют равносильности $2 \Leftrightarrow 4$ и $3 \Leftrightarrow 5$. Импликации $6 \Rightarrow 7$ и $7 \Rightarrow 5$ очевидны.

Осталось доказать $4 \Rightarrow 6$. При $n = 1$ импликация очевидна. Пусть $n \geqslant 2$ и $a_1, \ldots, a_n$ — любой базис пространства $X$. Для любых $y_1, \ldots, y_n \in Y$ определим $h_{y_1,\ldots,y_n} : X \to Y$ равенством $h_{y_1,\ldots,y_n}(a) = \sum\limits_{i=1}^{n} c_i y_i$ для всех $a = \sum\limits_{i=1}^{n} c_i a_i \in X$, $c_1, \ldots, c_n \in \mathbb{Z}/(p)$, так что $h_{y_1,\ldots,y_n} \in \mathrm{Hom}(X, Y)$. Тогда

$$
\begin{aligned}
0 = |Y|^{n-1}\cdot 0 &= \sum_{0\neq a\in X}\ \sum_{y_2,\ldots,y_n\in Y}(P(fa^+ - f = h_{y_1,\ldots,y_n}(a)) - |Y|^{-1}) = \\
&= \sum_{0\neq c_1\in\mathbb{Z}/(p)} |Y|^{n-1}(P(f(c_1a_1)^+ - f = c_1y_1) - |Y|^{-1}) + \\
&+ \sum_{\substack{\bar 0\neq(c_2,\ldots,c_n)\in(\mathbb{Z}/(p))^{n-1},\\ c_1\in\mathbb{Z}/(p)}}\ \sum_{y_2,\ldots,y_n\in Y}(P(fa^+ - f = \sum_{i=1}^{n} c_iy_i)) - |Y|^{-1}) = \\
&= |Y|^{n-1}\sum_{0\neq c_1\in\mathbb{Z}/(p)}(P(f(c_1a_1)^+ - f = c_1y_1) - |Y|^{-1}) + \\
&+ \sum_{\substack{\bar 0\neq(c_2,\ldots,c_n)\in(\mathbb{Z}/(p))^{n-1},\\ c_1\in\mathbb{Z}/(p)}}\ \sum_{y_2,\ldots,y_n\in Y}(P(fa^+ - f = c_1y_1 + \sum_{i=2}^{n} c_iy_i)) - |Y|^{-1}) = \\
&= |Y|^{n-1}\sum_{0\neq c_1\in\mathbb{Z}/(p)}(P(f(c_1a_1)^+ - f = c_1y_1) - |Y|^{-1}) + \\
&+ \sum_{\substack{\bar 0\neq(c_2,\ldots,c_n)\in(\mathbb{Z}/(p))^{n-1},\\ c_1\in\mathbb{Z}/(p)}} |Y|^{n-2}\sum_{y\in Y}(P(fa^+ - f = y) - |Y|^{-1}) = \\
&= |Y|^{n-1}\sum_{0\neq c_1\in\mathbb{Z}/(p)}(P(f(c_1a_1)^+ - f = c_1y_1) - |Y|^{-1}).
\end{aligned}
$$

Теорема доказана. ■

Теорема 1 означает, что свойство абсолютной негомоморфности (т. е. бентовости) наследуется гомоморфными образами, а также то, что бент-функции не имеют нетривиальных гомоморфных образов с $|X'| < |X|$. Покажем, что это справедливо и для свойства абсолютной минимальности.

**Теорема 9.** Если в примарном случае $f \in AM(Y^X)$, $f' : X' \to Y'$ — гомоморфный образ функции $f$ и $|Y'| > 1$, то $|X'| = |X|$ и $f' \in AM(Y'^{X'})$.

***Доказательство.*** Для любых $a \in X\backslash\{0\}$, $y' \in Y'$ по формуле (11)

$$\sum_{0\neq c\in\mathbb{Z}/(p)}(P'(f'(c\alpha(a))^+ - f' = cy') - |Y'|^{-1}) = \sum_{0\neq c\in\mathbb{Z}/(p)}\ \sum_{y\in\beta^{-1}(y')}(P(f(ca)^+ - f = cy) - |Y|^{-1}) = 0$$

в силу утверждения 6 теоремы 8. Тогда $\alpha$ — изоморфизм, утверждение 6 теоремы 8 выполнено для $X'$, $Y'$, $f'$ и теорема доказана. ■

Заметим, что теорема 9 для $Y' = \mathbb{Z}/(p)$ равносильна доказанному В. А. Шишкиным в 2008 г. некоторому свойству коэффициентов Фурье (результат не опубликован).

Следующая теорема обобщает хорошо известный для булевого случая факт (см., например, [4]).

**Теорема 10.** Для $p$-примарного случая, $p \in \{2, 3\}$, следующие утверждения равносильны:

1) $B(Y^X) \neq \varnothing$;
2) $\delta_0(Y^X) = |X|^{-1}$;
3) $B(Y^X) = M(Y^X)$.

***Доказательство.*** Импликация $3 \Rightarrow 1$ очевидна; $1 \Rightarrow 2$ следует из теоремы 7. Докажем импликацию $2 \Rightarrow 3$. При $p = 2$ она следует из утверждения 6 теоремы 8. Пусть $p = 3$. Для любого $p$ выполняются равенства

$$P(f(-ca)^+ - f = -cy) - |Y|^{-1} = P(f - f(ca)^+ = -cy) - |Y|^{-1} = P(f(ca)^+ - f = cy) - |Y|^{-1}.$$

Пусть $f$ — минимальная. Тогда из утверждения 6 теоремы 8 получаем, что для любых $a \in X \backslash \{0\}$, $y \in Y$ имеет место

$$0 = \sum_{c \in \{1,-1\}} (P(f(ca)^+ - f = cy) - |Y|^{-1}) = 2(P(f(a)^+ - f = y) - |Y|^{-1}),$$

и, следовательно, $f$ — абсолютно негомоморфная. ■

Отметим, что все приведённые доказательства не используют аппарат характеров абелевых групп и разложения Фурье.

В заключение введём следующие обозначения и терминологию.

Для любой $f : X \to Y$ обозначим

$$\bar{\delta}(f) = |X|^{-1} \sum_{a \in X} \delta(fa^+, f), \quad \delta_1(Y^X) = \min_{f \in Y^X} \bar{\delta}(f).$$

*Максимально негомоморфными* назовём функции $f \in Y^X$, для которых

$$\bar{\delta}(f) = \delta_1(Y^X).$$

Множество всех максимально негомоморфных функций из $Y^X$ обозначим через $N(Y^X)$, так что $N(Y^X) \neq \varnothing$ и

$$N(Y^X) = AN(Y^X), \text{ если } \delta_1(Y^X) = |X|^{-1},$$
$$AN(Y^X) = \varnothing, \text{ если } \delta_1(Y^X) > |X|^{-1}.$$

Представляется справедливой следующая

**Гипотеза.** Для $p$-примарного случая и любого $p$

$$M(Y^X) = N(Y^X); \tag{13}$$
$$AM(Y^X) = AN(Y^X). \tag{14}$$

Заметим, что равенства (13) и (14) совпадают в том и только в том случае, когда $B(Y^X) \neq \varnothing$ (см. теорему 5).

Теорема 10 равносильна выполнению равенства (14) для $p \in \{2, 3\}$. В силу теорем 7, 9, 4 равенство (14) достаточно доказать для случая $m = 1$.

Заменяя в соответствующих определениях множество всех функций $Y^X$ на произвольный класс функций $K \subseteq Y^X$, приходим к естественному и актуальному для приложений обобщению понятий минимальности и негомоморфности на случай функций из

класса $K$: $\delta_0(K)$, $\delta_1(K)$, $M(K)$, $N(K)$ (например, $K$ — класс всех подстановок, $M(K)$ — минимальные подстановки, $N(K)$ — максимально негомоморфные подстановки).

Наконец, заметим, что, в отличие от приведённого выше общепринятого определения бент-функции (через равномодульность коэффициентов Фурье), в работе [2] для произвольного простого $p$ и в работе [10] для $p = 2$ бент-функциями названы *абсолютно минимальные* функции. При этом, ссылаясь на работу [1], в [2, теорема 3.2] фактически утверждается, что теорема 10 справедлива для любого простого $p$ (чего не удалось доказать автору ни здесь, ни в [1]). Последствием этого явилась необоснованность распространения некоторых свойств бент-функций на абсолютно минимальные функции. Однако теорема 10 теперь обосновывает такое распространение в работе [2] для случая $p = 2, 3$ и в работе [10]. Заметим, что основные результаты работы [2] относятся к случаю $p = 2$.

## ЛИТЕРАТУРА


1. *Солодовников В. И.* Бент-функции из конечной абелевой группы в конечную абелеву группу // Дискретная математика. 2002. Т. 14. № 1. С. 99–113.
2. *Кузьмин А. С., Нечаев А. А., Шишкин В. А.* Бент- и гипербент-функции над конечным полем // Труды по дискретной математике. 2007. Т. 10. С. 97–122.
3. *Nyberg K.* Perfect nonlinear S-boxes // LNCS. 1991. V. 547. P. 378–386.
4. *Rothaus O. S.* On "bent" functions // J. Comb. Theory. Ser. A. 1976. V. 20. No. 3. P. 300–305.
5. *Амбросимов А. С.* Свойства бент-функций $q$-значной логики над конечными полями // Дискретная математика. 1994. Т. 6. № 3. С. 50–60.
6. *Логачёв О. А., Сальников А. А., Ященко В. В.* Бент-функции на конечной абелевой группе // Дискретная математика. 1997. Т. 9. № 4. С. 3–20.
7. *Kumar P. V., Scholts R. A., and Welch L. R.* Generalized bent functions and their properties // J. Comb. Theory. Ser. A. 1985. V. 40. No. 1. P. 90–107.
8. *Токарева Н. Н.* Нелинейные булевы функции: бент-функции и их обобщения. Saarbrucken, Germany: LAP LAMBERT Academic Publishing, 2011.
9. *Golomb S. W.* On the classification of Boolean functions // IRE Trans. Circuit Theory. 1959. V. 1. No. 6. P. 10–27.
10. *Кузьмин А. С., Нечаев А. А., Шишкин В. А.* Параметры (гипер-) бент-функций над полем из $2^l$ элементов // Труды по дискретной математике. 2008. Т. 11/1. С. 47–59.

# МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

DOI 10.17223/20710410/17/4

УДК 519.6

## ОБ ОДНОМ ОБОБЩЕНИИ БЛОЧНЫХ ШИФРОВ ФЕЙСТЕЛЯ

А. М. Коренева*, В. М. Фомичев**

*Национальный исследовательский ядерный университет (*МИФИ*), г. Москва, Россия*

**Финансовый университет при Правительстве Российской Федерации, г. Москва, Россия*

**E-mail:** alisa.koreneva@gmail.com, fomichev@nm.ru

Исследованы блочные шифры на основе регистров сдвига, обобщающие шифры Фейстеля. Доказан критерий инволютивности алгоритмов шифрования из данного класса. С использованием теоретико-графового подхода исследованы перемешивающие свойства раундовой подстановки, даны верхние оценки диаметра и экспонента перемешивающего графа раундовой подстановки.

**Ключевые слова:** *блочный шифр Фейстеля, инволютивность алгоритма шифрования, перемешивающий граф (матрица) преобразования, диаметр графа, экспонент графа.*

### Введение

Многие известные симметричные блочные шифры относятся к шифрам Фейстеля (DES-алгоритм, ГОСТ 28147-89 и др.). Отличительной особенностью этих шифров является то, что раундовая подстановка построена на основе преобразования регистра сдвига длины 2 над множеством двоичных векторов (описание алгоритма можно найти, например, в [1, с. 377]). Эта раундовая подстановка, зависящая от ключа, названа петлёй Фейстеля.

Идеи использования в блочных шифрах петли Фейстеля привлекательны до сих пор. Многочисленные исследования показали, что шифры Фейстеля позволяют обеспечить ряд хороших криптографических свойств, в том числе инволютивность алгоритма шифрования. На рубеже XX и XXI веков Национальный Институт стандартов и технологий США провел конкурс на разработку AES — нового криптографического стандарта, который с 2002 г. стал преемником DES. Среди кандидатов на AES было немало шифров, являющихся обобщениями шифра Фейстеля, например RC6, MARS, TWOFISH, LOKI и другие.

В данной работе разрабатывается идея построения симметричных блочных шифров на основе преобразований регистра сдвига произвольной длины. Увеличение длины базового регистра позволяет рассчитывать на улучшение некоторых характеристик шифра. Например, использование регистра сдвига большей длины дает возможность увеличить размер входного блока данных без существенного усложнения реализации функций, а также предоставляет больший выбор разработчику при построении раундовой функции, ключевого расписания и др. Возможны и другие улучшения криптографических свойств, впрочем, как и некоторые негативные изменения. Криптографические свойства симметричных блочных шифров, построенных на основе регистров сдвига длины больше 2, являются предметом исследований данной работы.

## 1. Инволютивность алгоритмов блочного шифрования регистрового типа

Ряд блочных шифров, в том числе шифры Фейстеля имеют инволютивный алгоритм шифрования [1, теоремы 17.1 и 17.2]. Инволютивность алгоритма шифрования является важным технологичным свойством шифрующих подстановок, так как зашифрование и расшифрование реализуются одним и тем же алгоритмом, изменяются лишь входные данные. Ключи зашифрования и расшифрования в инволютивном шифре Фейстеля обладают симметрией, точнее, они отличаются лишь взаимно обратным порядком использования раундовых ключей.

Рассмотрим обобщение шифров Фейстеля — класс итеративных симметричных блочных шифров на основе регистров сдвига длины $n > 2$ над множеством $r$-мерных двоичных векторов $V_r$, где $n, r$ — натуральные. Определим условия на раундовую функцию, при которых алгоритм шифрования инволютивен.

Введём обозначения. Пусть $q$ — ключ шифрования, $h$ — количество раундов шифрования. Обозначим через $x(j) = [x_1(j), x_2(j), \ldots, x_n(j)]$ блок данных (состояние регистра) после $j$-го раунда шифрования, $j = 1, \ldots, h$, при $j = 0$ — входной блок данных, при $j = h$ — выходной блок данных. Вектор $x_i(j) \in V_r$ назовем $i$-м подблоком блока $x(j)$, $i = 1, \ldots, n$. Обозначим $q_j$ раундовый ключ $j$-го раунда шифрования (двоичный вектор некоторой длины), $j = 1, \ldots, h$.

Раундовая подстановка $\varphi_z$ блочного шифра при раундовом ключе $z$ имеет вид

$$\varphi_z(x(j)) = (x_2(j), \ldots, x_n(j), \psi(x_2(j), \ldots, x_n(j), z) \oplus x_1(j)), j = 1, \ldots, h, \qquad (1)$$

где $\psi(y_2, \ldots, y_n, z)$ для $y_2, \ldots, y_n \in V_r$ есть функция усложнения шифра.

На рис. 1 представлена схема $j$-го раунда шифрования для блочного шифра на базе регистра сдвига длины $n$.

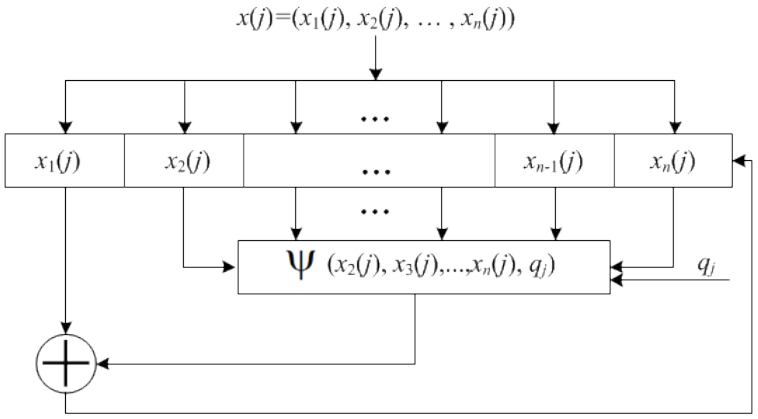


Рис. 1. Схема $j$-го раунда шифрования

Определим условия инволютивности данного блочного шифра. Обозначим через $\tau_n$ инволюцию степени $n > 1$ вида

$$\tau_n(y_1, \ldots, y_n) = (y_n, \ldots, y_1).$$

Для функции усложнения шифра $\psi(y_2, \ldots, y_n, z)$ при $n > 2$ имеем

$$\psi(\tau_{n-1}(y_2, \ldots, y_n), z) = \psi(y_n, \ldots, y_2, z).$$

Функцию усложнения $\psi(y_2, \ldots, y_n, z)$ назовем инвариантной относительно инволюции $\tau_{n-1}$, если $\psi(y_2, \ldots, y_n, z) = \psi(y_n, \ldots, y_2, z)$. Очевидно следующее утверждение.

**Лемма 1.** Если функция $\psi(y_2, \ldots, y_n, z)$ инвариантна относительно инволюции $\tau_{n-1}$, то переменная $y_i$ является существенной тогда и только тогда, когда существенна переменная $y_{n+2-i}$, $i = 2, 3, \ldots, n$.

**Лемма 2.** При любой функции усложнения $\psi(y_2, \ldots, y_n, z)$, инвариантной относительно инволюции $\tau_{n-1}$, для раундовой подстановки выполнено равенство

$$\varphi_z^{-1} = \tau_n \varphi_z \tau_n. \tag{2}$$

***Доказательство.*** Докажем равносильное равенство $\tau_n \varphi_z \tau_n \varphi_z = e$. Для любых $y_1, \ldots, y_n$ и раундовой подстановки $\varphi_z$ в соответствии с (1) выполнено

$$\varphi_z(y_1, \ldots, y_n) = (y_2, \ldots, y_n, y_1 \oplus \psi(y_2, \ldots, y_n, z)).$$

Применим инволюцию $\tau_n$ к $\varphi_z(y_1, \ldots, y_n)$:

$$\tau_n(\varphi_z(y_1, \ldots, y_n)) = (y_1 \oplus \psi(y_2, \ldots, y_n, z), y_n, \ldots, y_2).$$

Отсюда

$$\varphi_z(\tau_n(\varphi_z(y_1, \ldots, y_n))) = (y_n, \ldots, y_2, y_1 \oplus \psi(y_2, \ldots, y_n, z) \oplus \psi(y_n, \ldots, y_2, z)) = (y_n, \ldots, y_2, y_1),$$

так как $\psi(y_2, \ldots, y_n, z) \oplus \psi(y_n, \ldots, y_2, z) \equiv 0$. Тогда

$$\tau_n(\varphi_z(\tau_n(\varphi_z(y_1, \ldots, y_n)))) = \tau_n(y_n, \ldots, y_2, y_1) = (y_1, \ldots, y_n).$$

Лемма доказана. ■

Обратимость рассматриваемых алгоритмов шифрования на основе регистра сдвига длины $n > 2$ обеспечивается при любой функции усложнения, инвариантной относительно инволюции $\tau_{n-1}$, с помощью применения после последнего раунда шифрования инволюции $\tau_n$ к блоку $x(h)$.

**Теорема 1.** Пусть функция усложнения $\psi(y_2, \ldots, y_n, z)$ инвариантна относительно инволюции $\tau_{n-1}$ и алгоритм шифрования состоит из $h$ раундов и инволюции $\tau_n$. Тогда алгоритм шифрования инволютивен и расшифрование отличается от зашифрования использованием раундовых ключей в обратном порядке.

***Доказательство.*** В условиях теоремы подстановки зашифрования и расшифрования $E_q$ и $E_q^{-1}$ определяются формулами

$$E_q(x) = \tau_n \varphi_{q_h} \cdot \ldots \cdot \varphi_{q_1}(x); \tag{3}$$

$$E_q^{-1}(y) = \varphi_{q_1}^{-1} \cdot \ldots \cdot \varphi_{q_h}^{-1} \cdot \tau_n(y). \tag{4}$$

В равенстве (4) последовательно сделаем $h$ раз замену подстановок $\varphi_{q_h}^{-1}, \ldots, \varphi_{q_1}^{-1}$ в соответствии с (2). Получим равенство $E_q^{-1}(y) = \tau_n \varphi_{q_1} \cdot \ldots \cdot \varphi_{q_h}(y)$. Сравнение его с (3) даёт утверждение теоремы. ■

## 2. Перемешивающие свойства алгоритма шифрования

Перемешивающие свойства преобразования $f$ множества $X^n$, заданного системой координатных функций $\{f_1(x_1, \ldots, x_n), \ldots, f_n(x_1, \ldots, x_n)\}$, определяются системой множеств $\{S(f_1), \ldots, S(f_n)\}$, где $S(f_j)$ — множество номеров существенных переменных функции $f_j$, $j = 1, \ldots, n$. Считается, что наилучшее перемешивание достигается тогда,

когда каждая из координатных функций преобразования $f$ зависит от всех переменных, то есть $S(f_j) = \{1, \ldots, n\}$, $j = 1, \ldots, n$. Аппаратная или программная реализация таких преобразований затруднена в связи с необходимостью реализации функций от большого числа переменных. Поэтому хорошее перемешивание часто достигается при итерациях (возведении в степень) преобразования с относительно слабыми перемешивающими свойствами. Показатель степени преобразования, при которой достигается хорошее перемешивание, является важной криптографической характеристикой. В частности, показатель степени раундовой подстановки, при которой достигается хорошее перемешивание, является определяющим для разработчика при выборе числа раундов шифрования блочного шифра.

Рассмотрим раундовую подстановку $\varphi_z$ множества $V_{nr}$, определяемую равенством (1). Функцией обратной связи данного регистра является функция $\psi(x_2(j), \ldots, x_n(j), z) \oplus x_1(j)$. Подстановка $\varphi_z$ задается системой булевых координатных функций $\{\varphi_1(x_1, x_2, \ldots, x_{nr}), \varphi_2(x_1, x_2, \ldots, x_{nr}), \ldots, \varphi_{nr}(x_1, x_2, \ldots, x_{nr})\}$ [1, теорема 5.1].

Для оценки перемешивающих свойств степеней подстановки $\varphi_z$ при фиксированном ключе $z$ используем теоретико-графовый подход. Обозначим $\Gamma(\varphi)$ перемешивающий $nr$-вершинный орграф подстановки $\varphi_z$, в котором пара $(i, j)$ является дугой тогда и только тогда, когда $i \in S(\varphi_j)$, $i, j \in \{1, \ldots, nr\}$. Матрицу $M(\varphi)$ смежности вершин графа $\Gamma(\varphi)$ называют перемешивающей матрицей подстановки $\varphi_z$.

Если орграф $\Gamma(\varphi)$ не является полным (матрица $M(\varphi)$ не положительна, то есть содержит нулевые элементы), то подстановка $\varphi_z$ реализует неполное перемешивание, иначе говоря, $\varphi_z$ не является совершенной. Если матрица $(M(\varphi))^m$ не положительна, то подстановка $(\varphi_z)^m$ реализует неполное перемешивание, $m \geqslant 1$ [1, следствие 2 теоремы 10.3].

Необходимым условием хорошего перемешивания с использованием степеней подстановки $\varphi_z$ является примитивность орграфа $\Gamma(\varphi)$, что равнозначно сильной связности орграфа [1, следствие 1 теоремы 10.3] и наличию в нем простых циклов длины $l_1, \ldots, l_k$, где $k > 1$ и НОД$(l_1, \ldots, l_k) = 1$ [2, с. 393]. Степень преобразования, при которой достигается хорошее перемешивание, оценивается снизу экспонентом орграфа $\Gamma(\varphi)$ (экспонентом матрицы $M(\varphi)$).

Исследуем условия хорошего перемешивания входных блоков с помощью раундовой подстановки $\varphi_z$ множества $V_{nr}$ данных блочных шифров. Построим перемешивающую матрицу $M(\varphi)$ размера $nr \times nr$ в соответствии с равенством (1). Матрица $M(\varphi)$, разбитая на $n^2$ блоков размера $r \times r$, изображена в таблице.

**Перемешивающая матрица $M(\varphi)$ раундовой подстановки $\varphi_z$**

| Переменные | $\varphi_1 \ldots \varphi_r$ | $\varphi_{r+1} \ldots \varphi_{2r}$ | ... | $\varphi_{(n-2)r+1} \ldots \varphi_{(n-1)r}$ | $\varphi_{(n-1)r+1} \ldots \varphi_{nr}$ |
|---|---|---|---|---|---|
| $x_1$ ... $x_r$ | 0 | 0 | ... | 0 | E |
| $x_{r+1}$ ... $x_{2r}$ | E | 0 | ... | 0 | $\psi_1$ |
| $x_{2r+1}$ ... $x_{3r}$ | 0 | E | ... | 0 | $\psi_2$ |
| ... | ... | ... | ... | ... | ... |
| $x_{(n-1)r+1}$ ... $x_{nr}$ | 0 | 0 | ... | E | $\psi_r$ |

Символом 0 обозначены блоки, состоящие из нулей, символом E — единичные матрицы. Блок $\psi_j$ показывает зависимость координатных функций $\varphi_{(n-1)r+1}, \ldots, \varphi_{nr}$ (обратной связи регистра) от переменных $x_{(j-1)r+1}, \ldots, x_{jr}$, $j = 2, \ldots, n$.

Для функции усложнения $\psi(y_2, \ldots, y_n, z)$ определим $r$-вершинный граф $\Gamma_\psi$: пара $(v, w)$ образует дугу тогда и только тогда, когда функция $\varphi_{(n-1)r+w}$ зависит существенно хотя бы от одной из переменных множества $\{x_v, x_{r+v}, \ldots, x_{(n-1)r+v}\}$. То есть

граф $\Gamma_\psi$ получается из графа $\Gamma(\varphi)$ отождествлением $n$ вершин, соответствующих одинаковым координатам $r$-мерных векторов. Обозначим через $\Gamma_0$ граф $\Gamma(\varphi)$ для функции $\psi(y_2,\ldots,y_n,z)\equiv 0$.

**Теорема 2.** Перемешивающий граф $\Gamma(\varphi)$ сильно связен тогда и только тогда, когда сильно связен граф $\Gamma_\psi$.

***Доказательство.*** Граф $\Gamma_0$ состоит из $r$ независимых циклов длины $n$ (рис. 2). Обозначим эти циклы $C_1,\ldots,C_r$.

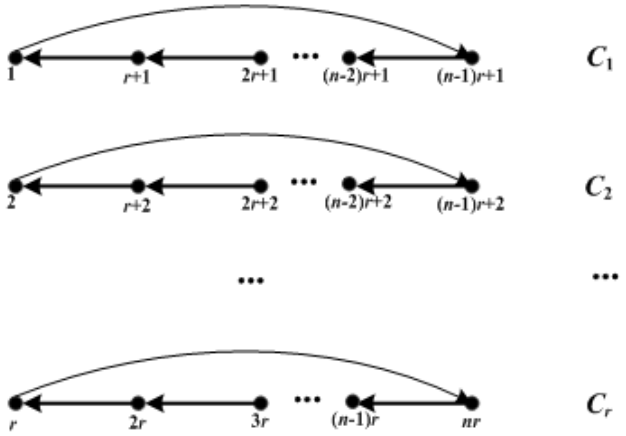


Рис. 2. Граф $\Gamma_0$

Множества вершин графов $\Gamma_0$ и $\Gamma(\varphi)$ совпадают, и $\Gamma_0$ является частью графа $\Gamma(\varphi)$. Значит, если вершины $i,j$ графа $\Gamma(\varphi)$ принадлежат одному циклу, то каждая из них достижима из другой.

Заметим, что при любой функции усложнения $\psi$ граф $\Gamma_\psi$ получается из графа $\Gamma(\varphi)$ отождествлением вершин, принадлежащих одному и тому же циклу. Тогда, учитывая взаимную достижимость вершин одного цикла, в графе $\Gamma(\varphi)$ любая вершина $w$, принадлежащая циклу $C_j$, достижима из любой вершины $v$, принадлежащей циклу $C_i$ при $i\neq j$, где $i,j\in\{1,\ldots,r\}$, тогда и только тогда, когда вершина $j$ достижима из вершины $i$ в графе $\Gamma_\psi$. Следовательно, $\Gamma(\varphi)$ сильно связен тогда и только тогда, когда сильно связен $r$-вершинный граф $\Gamma_\psi$. ■

**Теорема 3.** Если граф $\Gamma_\psi$ сильносвязный и $\operatorname{diam}\Gamma_\psi\leqslant d$, то

1) $\operatorname{diam}\Gamma(\varphi)\leqslant(n-1)\min\{d,r-1\}+n$;
2) если граф $\Gamma(\varphi)$ примитивный и алгоритм блочного шифрования инволютивен, то $\operatorname{diam}\Gamma(\varphi)\leqslant n/2\cdot\min\{d,r-1\}+n$;
3) если граф $\Gamma(\varphi)$ примитивный, то $\exp\Gamma(\varphi)\leqslant n^2r+nr-2n$.

***Доказательство.***

1) Обозначим длину кратчайшего пути в графе $\Gamma(\varphi)$ из $i$ в $j$ через $l_{i,j}$. Диаметр орграфа $\Gamma(\varphi)$ — это максимум $l_{i,j}$ (по всем $i,j\in\{1,\ldots,nr\}$).

Если вершины $i,j$ принадлежат одному циклу графа $\Gamma_0$, то $l_{i,j}\leqslant n$.

Пусть вершины $i,j$ принадлежат циклам $C_v$ и $C_w$ графа $\Gamma_0$, где $v\neq w$. Так как граф $\Gamma_\psi$ сильно связный и $\operatorname{diam}\Gamma_\psi\leqslant d$, то в $\Gamma_\psi$ имеется простой путь $(b_0,b_1,\ldots,b_l)$ из $v$ в $w$ длины $l$, где $v=b_0$, $w=b_l$, $l\leqslant d$ и $l\leqslant r-1$, так как $v\neq w$. Без ущерба для общности положим, что $i,b_0\in C_1$, $b_1\in C_2$, …, $b_l,j\in C_{l+1}$. Тогда из определения графа $\Gamma_\psi$ следует, что в графе $\Gamma(\varphi)$ при некоторых $j_1,j_2,\ldots,j_l\in\{1,\ldots,n-1\}$

имеются дуги $(j_1r+1,(n-1)r+2),(j_2r+2,(n-1)r+3),\ldots,(j_lr+l,(n-1)r+l+1)$. Значит, путь из $i$ в $j$ в графе $\Gamma(\varphi)$ можно составить как последовательность путей $(u_0,\delta_1,u_1,\delta_2,\ldots,u_{l-1},\delta_l,u_l)$, где $\delta_k$ есть дуга $(j_kr+k,(n-1)r+k+1)$ при $k=1,\ldots,l$, путь $u_0$ есть путь длины не более $n-1$ из $i$ в $j_1r+1$ (часть цикла $C_1$), путь $u_{k-1}$ есть путь длины не более $n-2$ из $(n-1)r+k$ в $j_kr+k$ (часть цикла $C_k$) при $k=2,\ldots,l$ и путь $u_l$ есть путь длины не более $n-1$ из $(n-1)r+l+1$ в $j$ (часть цикла $C_{l+1}$). Следовательно, при любых $i,j\in\{1,\ldots,nr\}$ длина данного пути из $i$ в $j$ в графе $\Gamma(\varphi)$ не превышает $l+2(n-1)+(l-1)(n-2)=nl+n-l$. В силу произвольности рассмотренных $i,j$ получим $\operatorname{diam}\Gamma(\varphi)\leqslant nl+n-l$, где $l\leqslant\min\{d,r-1\}$. Следовательно, $\operatorname{diam}\Gamma(\varphi)\leqslant(n-1)\min\{d,r-1\}+n$.

2) Вернёмся к рассмотрению кратчайшего пути $(u_0,\delta_1,u_1,\delta_2,\ldots,u_{l-1},\delta_l,u_l)$ в графе $\Gamma(\varphi)$ из $i$ в $j$ длины $l_{i,j}$. Если алгоритм шифрования инволютивен, то в соответствии с леммой 1 в графе $\Gamma(\varphi)$ при некоторых $j_1,\ldots,j_l\in\{1,\ldots,n-1\}$ имеются пары дуг $(j_1r+1,(n-1)r+2)$ и $((n-j_1)r+1,(n-1)r+2),\ldots,(j_lr+l,(n-1)r+l+1)$ и $((n-j_l)r+l,(n-1)r+l+1)$. Тогда уточним определения элементов пути. Здесь $u_0$ есть путь из $i$ в $a_1$ (часть цикла $C_1$), где $a_1$ — ближайшая из двух вершин $j_1r+1$ и $(n-j_1)r+1$ к вершине $i$. Путь $u_{k-1}$ есть при $k=2,\ldots,l$ путь из $(n-1)r+k$ в $a_k$ (часть цикла $C_k$), где $a_k$ — ближайшая из двух вершин $j_kr+k$ и $(n-j_k)r+k$ к вершине $(n-1)r+k$. И $u_l$ есть путь из $(n-1)r+l+1$ в $j$ (часть цикла $C_{l+1}$). Соответственно $\delta_k$ есть дуга $(a_k,(n-1)r+k+1)$ при $k=1,\ldots,l$.

Следовательно (см. рис. 2), длина пути $u_0$ не превышает $\lfloor n/2\rfloor$, длины путей $u_1,\ldots,u_{l-1}$ не превышают $\lfloor n/2\rfloor-1$ и длина пути $u_l$ не превышает $n-1$. Отсюда

$$l_{i,j}\leqslant l+n/2+(l-1)(n/2-1)+n-1=nl/2+n,$$

где $l\leqslant\min\{d,r-1\}$. Тогда $\operatorname{diam}\Gamma(\varphi)\leqslant nl/2+n$ в силу произвольности рассмотренных $i,j$.

3) Известно [2, с. 227], что если $\lambda$ — длина самого короткого цикла сильно связного примитивного $m$-вершинного графа $\Gamma$, то $\exp\Gamma\leqslant m+\lambda(m-2)$. В $nr$-вершинном графе $\Gamma(\varphi)$ длина самого короткого цикла не превышает $n$ в силу наличия циклов $C_1,\ldots,C_r$. Следовательно, $\exp\Gamma(\varphi)\leqslant nr+n(nr-2)$. ■

**Теорема 4.** Сильносвязный граф $\Gamma(\varphi)$ примитивен, если выполнено любое из следующих условий:

1) координатная функция $\varphi_m$ зависит существенно от переменной $x_m$ при некотором $m\in\{(n-1)r+1,(n-1)r+2,\ldots,nr\}$, в этом случае $\exp\Gamma(\varphi)\leqslant 2nr-2$;

2) алгоритм шифрования инволютивен и при некоторых $m\in\{(n-1)r+1,(n-1)r+2,\ldots,nr\}$ и $\mu\in\{1,\ldots,r\}$ координатная функция $\varphi_m$ зависит существенно от переменной $x_{jr+\mu}$, где $n-2j=1$, в этом случае $\exp\Gamma(\varphi)\leqslant(ln/2)^2+n(r-l)-1$, где $l$ — длина кратчайшего цикла графа $\Gamma_\psi$, проходящего через дугу $(\mu,m)$.

***Доказательство.***

1) Из существенной зависимости функции $\varphi_m$ от переменной $x_m$ следует наличие петли в вершине $m$ графа $\Gamma(\varphi)$. Петля является простым циклом длины 1. Тогда в сильносвязном графе $\Gamma(\varphi)$ имеются простые циклы длины 1 и $l>1$, где НОД $(1,l)=1$. Следовательно, граф $\Gamma(\varphi)$ примитивен и [2, с. 227] $\exp\Gamma(\varphi)\leqslant 2nr-2$.

2) Без ущерба для общности положим $m=(n-1)r+1$, тогда по условию в графе $\Gamma_\psi$ имеется дуга $(\mu,1)$ и в графе $\Gamma(\varphi)$ имеется дуга $(jr+\mu,(n-1)r+1)$ при $j=(n-1)/2$.

Если алгоритм шифрования инволютивен, то в силу леммы 1 функция $\varphi_m$ зависит существенно и от $x_{jr+\mu}$, и от $x_{(n-j)r+\mu}$, то есть в $\Gamma(\varphi)$ имеется дуга $((n-j)r+\mu, (n-1)r+1)$ при $j=(n-1)/2$.

Рассмотрим в сильносвязном графе $\Gamma_\psi$ кратчайший простой цикл $C_\psi$, проходящий через дугу $(\mu, 1)$. Без ущерба для общности можно считать, что $C_\psi = (\mu, 1, \ldots, l-1)$, где $l$ — длина цикла $C_\psi$. Тогда в графе $\Gamma(\varphi)$ имеется цикл $C$, проходящий через дугу $(jr+\mu, (n-1)r+1)$ и составленный при некоторых $j_1, \ldots, j_{l-1} \in \{1, \ldots, n-1\}$ как последовательность путей $(u_1, \delta_1, \ldots, u_l, \delta_l)$. Здесь $u_k$ — путь (часть цикла $C_k$) из $(n-1)r+k$ в $a_k$, где $a_k$ — ближайшая из двух вершин $j_k r+k$ и $(n-j_k)r+k$ к вершине $(n-1)r+k$, $k=1, \ldots, l-1$; $u_l$ есть путь (часть цикла $C_\mu$) из $(n-1)r+\mu$ в $a_\mu$, где $a_\mu$ — ближайшая из двух вершин $jr+\mu$ и $(n-j)r+\mu$ к вершине $(n-1)r+\mu$. Соответственно $\delta_k$ при $k=1, \ldots, l-2$ есть дуга $(a_k, (n-1)r+k+1)$, $\delta_{l-1}$ есть дуга $(a_{l-1}, (n-1)r+\mu)$ и $\delta_l$ есть дуга $(a_\mu, (n-1)r+1)$. Следовательно (см. рис. 2), длина пути $u_k$ не превышает $n/2-1$, $k=1, \ldots, l$, и длина цикла $C$ не превышает $nl/2$.

Вместе с тем в графе $\Gamma(\varphi)$ имеется цикл $C'$, полученный из цикла $C$ заменой пути $u_l$ на путь из вершины $(n-1)r+\mu$ в $b$ и заменой дуги $(a_\mu, (n-1)r+1)$ на дугу $(b, (n-1)r+1)$, где $b$ — более удалённая (от вершины $(n-1)r+\mu$) из двух вершин $jr+\mu$ и $(n-j)r+\mu$. При $n-2j=1$ вершины $a_\mu$ и $b$ являются соседними на цикле $C'$, поэтому длина цикла $C'$ превышает длину цикла $C$ на 1.

Итак, в графе $\Gamma(\varphi)$ имеются простые циклы $C$ и $C'$ взаимно простых длин $\lambda$ и $\lambda+1$, где $\lambda \leqslant nl/2$, и порядок пересечения множеств вершин циклов $C$ и $C'$ равен $\lambda$. Следовательно, сильносвязный граф $\Gamma(\varphi)$ примитивен и верна оценка для экспонента графа $\Gamma(\varphi)$ [3, теорема 1$б$] $\exp\Gamma(\varphi) \leqslant (ln/2)^2 + n(r-l) - 1$. ■

## Выводы

Сформулирован критерий инволютивности алгоритма блочного шифрования, основанного на регистре сдвига длины $n$ над множеством двоичных $r$-мерных векторов (обобщение шифров Фейстеля). Исследованы свойства $nr$-вершинного перемешивающего графа $\Gamma(\varphi)$ раундовой подстановки $\varphi$, а именно:

1) показано, что граф $\Gamma(\varphi)$ сильно связен тогда и только тогда, когда сильно связен $r$-вершинный граф $\Gamma_\psi$, полученный из $\Gamma(\varphi)$ отождествлением вершин, соответствующих одинаковым координатам $r$-мерных векторов;
2) получены достаточные условия примитивности графа $\Gamma(\varphi)$;
3) получены оценки диаметра и экспонента графа $\Gamma(\varphi)$ в различных условиях.

## ЛИТЕРАТУРА

1. *Фомичёв В. М.* Методы дискретной математики в криптологии. М.: ДИАЛОГ-МИФИ, 2010. 424 с.
2. *Сачков В. Н., Тараканов В. Е.* Комбинаторика неотрицательных матриц. М.: ТВП, 2000. 448 с.
3. *Фомичев В. М.* Оценки экспонентов примитивных графов // Прикладная дискретная математика. 2011. № 2(12). С. 101–112.

2012 Математические методы криптографии №3(17)

DOI 10.17223/20710410/17/5

УДК 519.7

# О СОВЕРШЕННЫХ ИМИТОСТОЙКИХ ШИФРАХ

С. М. Рацеев

*Ульяновский государственный университет, г. Ульяновск, Россия*

**E-mail:** RatseevSM@rambler.ru

В работе приводятся конструкции совершенных имитостойких шифров.

**Ключевые слова:** *шифр, совершенный шифр, имитация сообщения.*

Пусть $X$, $K$, $Y$ — конечные множества открытых текстов, ключей и шифрованных текстов соответственно. Обозначим через $\Sigma_B = (X, K, Y, E, D, P(X), P(K))$ вероятностную модель шифра [1], где $E$ и $D$ — множества правил зашифрования и расшифрования соответственно. При этом предполагается, что априорные распределения вероятностей $P(X)$ и $P(K)$ на соответствующих множествах $X$ и $K$ независимы и не содержат нулевых вероятностей. Распределения $P(X)$ и $P(K)$ естественным образом индуцируют распределение вероятностей $P(Y)$ следующим образом:

$$P_Y(y) = \sum_{\substack{(x,k)\in X\times K,\\ E_k(x)=y}} P_X(x)P_K(k).$$

Обозначим через $K(x, y)$ множество таких ключей $k \in K$, для которых $E_k(x) = y$. Условная вероятность $P_{Y|X}(y|x)$ определяется естественным образом:

$$P_{Y|X}(y|x) = \begin{cases} \sum\limits_{k\in K(x,y)} P_K(k), & \text{если } K(x,y) \neq \varnothing, \\ 0, & \text{если } K(x,y) = \varnothing. \end{cases}$$

С помощью теоремы умножения вероятностей можно определить и условную вероятность $P_{Y|X}(y|x)$:

$$P_{X|Y}(x|y) = \frac{P_X(x)P_{Y|X}(y|x)}{P_Y(y)}.$$

Напомним, что шифр $\Sigma_B$ называется совершенным по Шеннону, если для любых $x \in X$ и $y \in Y$ выполняется равенство $P_{X|Y}(x|y) = P_X(x)$. Для совершенного по Шеннону шифра можно дать и эквивалентные определения.

**Утверждение 1.** Для произвольного шифра $\Sigma_B$ следующие условия эквивалентны:

($i$) для любых $x \in X$ и $y \in Y$ выполнено равенство $P_{X|Y}(x|y) = P_X(x)$;

($ii$) для любых $x \in X$ и $y \in Y$ выполнено равенство $P_{Y|X}(y|x) = P_Y(y)$;

($iii$) для любых $x_1, x_2 \in X$ и $y \in Y$ выполнено равенство $P_{Y|X}(y|x_1) = P_{Y|X}(y|x_2)$.

Для удобства читателей сформулируем и докажем следующее несложное, но важное утверждение.

**Теорема 1.** Пусть для шифра $\Sigma_B$ выполнены следующие условия:

($i$) для любой пары $(x, y) \in X \times Y$ существует, и притом единственный, ключ $k \in K$, такой, что $E_k(x) = y$, где $E_k$ — правило зашифрования на ключе $k$;

($ii$) распределение вероятностей $P(K)$ является равномерным.

Тогда шифр $\Sigma_B$ является совершенным по Шеннону, причём распределение вероятностей $P(Y)$ является равномерным и $|K| = |Y|$.

***Доказательство.*** Пусть выполнены условия теоремы. Покажем, что в этом случае выполнен пункт $(ii)$ утверждения 1.

Из условий теоремы и определения вероятности $P_{Y|X}(y|x)$ следует равенство $P_{Y|X}(y|x) = P_K(k) = 1/|K|$. Зафиксируем произвольное значение $y \in Y$. Применяя формулу полной вероятности, получаем

$$P_Y(y) = \sum_{x \in X} P_X(x) P_{Y|X}(y|x) = \sum_{x \in X} P_X(x) \cdot \frac{1}{|K|} = \frac{1}{|K|}.$$

Таким образом, для любых $x \in X$ и $y \in Y$ имеет место равенство $P_{Y|X}(y|x) = P_Y(y)$, распределение вероятностей на множестве $Y$ является равномерным и $|K| = |Y|$. ■

Будем говорить, что матрица $A$ порядка $m \times n$ , $m \geqslant n$, над некоторым $m$-элементным множеством $S$ является латинским прямоугольником относительно столбцов, если транспонированная матрица $A$ является латинским прямоугольником над множеством $S$.

Пусть матрица $A$ порядка $m \times n$, $m \geqslant n$, над множеством шифрованных текстов $Y = \{y_1, \ldots, y_m\}$ является латинским прямоугольником относительно столбцов. Пусть $|K| = m$, $|X| = n$. Занумеруем строки матрицы $A$ элементами множества $K$, а столбцы — элементами множества $X$. Если матрица $A$ является матрицей зашифрования для некоторого шифра $\Sigma_B$ и распределение вероятностей $P(K)$ является равномерным, то из теоремы 1 следует, что шифр $\Sigma_B$ является совершенным по Шеннону.

Рассмотрим вероятностное пространство $\Omega = (K, F_K, P_K)$. Зафиксируем $y \in Y$. Обозначим через $K(y)$ следующее множество: $K(y) = \{k \in K : y \in E_k(X)\}$. Под обозначением $K(y)$ будем также понимать событие $(K(y) \in F_K)$, заключающееся в том, что при случайном выборе ключа $k \in K$ шифртекст $y$ можно расшифровать на ключе $k$, то есть $y \in E_k(X)$. Тогда событию $K(y)$ благоприятствуют все элементы из множества $K(y)$, и только они. Поэтому $P(K(y)) = \sum\limits_{k \in K(y)} P_K(k)$.

Если канал связи готов к работе и на приёме установлены действующие ключи, но в данный момент времени никакого сообщения не передаётся, то противником может быть предпринята попытка имитации сообщения. Вероятность успеха имитации определяется следующим образом:

$$P_{\text{im}} = \max_{y \in Y} P(K(y)).$$

Если же в данный момент передается некоторое сообщение $y \in Y$ (которое получено из открытого текста $x \in X$ на ключе $k \in K$), то противник может заменить его на $\widetilde{y} \in Y$, отличный от $y$. При этом он рассчитывает на то, что на действующем ключе $k$ криптограмма $\widetilde{y}$ будет воспринята как некий осмысленный открытый текст $\widetilde{x}$, отличный от $x$. Пусть $K(\widetilde{y}) \mid K(y)$ — событие, заключающееся в попытке подмены сообщения $y$ сообщением $\widetilde{y}$. Применяя теорему о произведении вероятностей, получаем, что

$$P(K(\widetilde{y}) \mid K(y)) = \frac{P(K(y) \cap K(\widetilde{y}))}{P(K(y))} = \frac{\sum\limits_{k \in K(y, \widetilde{y})} P_K(k)}{\sum\limits_{k \in K(y)} P_K(k)},$$

где $K(y, \widetilde{y}) = K(y) \cap K(\widetilde{y})$. Тогда вероятность успеха подмены сообщения вычисляется по следующей формуле:

$$P_{\text{podm}} = \max_{\substack{y, \widetilde{y} \in Y, \\ y \neq \widetilde{y}}} P(K(\widetilde{y}) \mid K(y)).$$

**Теорема 2** [2]. Для любого шифра $\Sigma_B$ справедливы неравенства

$$P_{\text{im}} \geqslant \frac{|X|}{|Y|}, \quad P_{\text{podm}} \geqslant \frac{|X|-1}{|Y|-1}.$$

При этом $P_{\text{im}} = |X|/|Y|$ тогда и только тогда, когда для любого $y \in Y$ выполнено равенство $P(K(y)) = |X|/|Y|$; $P_{\text{podm}} = (|X|-1)/(|Y|-1)$ тогда и только тогда, когда для любых $y, \widetilde{y} \in Y$, $y \neq \widetilde{y}$, выполнено равенство $P(K(\widetilde{y}) \mid K(y)) = (|X|-1)/(|Y|-1)$.

**Утверждение 2.** Пусть $A$ — некоторая $(m \times n)$-матрица над множеством $Y = \{y_1, \ldots, y_m\}$, $m \geqslant n$, которая является латинским прямоугольником относительно столбцов. Тогда если распределение вероятностей $P(K)$ является равномерным, то для шифра $\Sigma_B$ с матрицей зашифрования $A$ выполнено равенство $P_{\text{im}} = n/m$.

**Утверждение 3.** Для любых натуральных чисел $m$ и $n$, таких, что $m \geqslant n$ и $n \neq m-1$, существует латинский $(m \times n)$-прямоугольник относительно столбцов, в котором имеется ровно $n$ строк, содержащих два различных фиксированных элемента.

***Доказательство.*** Если $m = n$, то все очевидно. Поэтому пусть $m \geqslant n+2$. Построим латинский прямоугольник $A$ над множеством $\{1, 2, \ldots, m\}$ следующим образом. Расположим число 1 в $A$ на позициях с координатами $(1,1)$, $(2,2)$, …, $(n,n)$, а число 2 — на позициях $(1,2)$, $(2,3)$, …, $(n-1,n)$, $(n,1)$.

Обозначим через $T^i$ циклический сдвиг на $i$ позиций влево. В матрице $A$ в $i$-м столбце на свободные позиции поставим элементы $T^{i-1}(3, 4, \ldots, m)$, $i = 1, \ldots, n$. Тогда полученная матрица является латинским прямоугольником относительно столбцов с требуемым свойством. ■

Из данного утверждения следует, что для любых натуральных чисел $m$ и $n$, $m \geqslant n$ и $n \neq m-1$, можно построить такой латинский прямоугольник $A$ размера $m \times n$ над множеством $Y = \{y_1, \ldots, y_m\}$, что при равномерном распределении вероятностей $P(K)$ для шифра $\Sigma_B$ с матрицей зашифрования $A$ будет выполнено равенство $P_{\text{podm}} = 1$. Отдельно рассмотрим случай $n = m-1$.

**Утверждение 4.** Пусть $A = A(n+1, n)$ — некоторая матрица над множеством $Y = \{y_1, \ldots, y_{n+1}\}$, которая является латинским прямоугольником относительно столбцов. Тогда если распределение вероятностей $P(K)$ является равномерным, то для шифра $\Sigma_B$ с матрицей зашифрования $A$ выполнено равенство $P_{\text{podm}} = (n-1)/n$.

***Доказательство.*** Заметим, что из равномерности распределения $P(K)$ следует, что для любых $\widetilde{y}, y \in Y$, $\widetilde{y} \neq y$, выполнено равенство

$$P(K(\widetilde{y} \mid K(y))) = \frac{|K(\widetilde{y}, y)|}{|K(y)|}.$$

Дополним матрицу $A$ до латинского квадрата $B$ размера $(n+1) \times (n+1)$ [3]. Зафиксируем произвольный элемент $y_0 \in Y$. Так как матрица $B$ является латинским квадратом, то элемент $y_0$ присутствует в последнем столбце матрицы $B$. Пусть он находится на позиции $(i_0, n+1)$. Это означает, что в матрице $A$ элемент $y_0$ встречается во всех строках, кроме строки с номером $i_0$, а в $i_0$-й строке матрицы $A$ расположены все элементы множества $Y \setminus \{y_0\}$. Поэтому для любого $y \in Y \setminus \{y_0\}$ выполнено равенство $|K(y, y_0)| = n-1$. Очевидно также, что для любого $y \in Y$ выполнено равенство $|K(y)| = n$. В силу произвольности $y_0$, из теоремы 2 следует, что $P_{\text{podm}} = (n-1)/n$. ■

Несложно проверить также следующее

**Утверждение 5.** Пусть $B$ — квадрат Виженера над множеством $Y = \{y_1, \ldots, y_m\}$. Составим из первых $n$ столбцов матрицы $B$ матрицу $A$, где $1 \leqslant n \leqslant m-1$. Пусть $|K| = m$, $|X| = n$, матрица $A$ является матрицей зашифрования для шифра $\Sigma_B$ и распределение вероятностей $P(K)$ является равномерным. Тогда для шифра $\Sigma_B$ выполнено равенство $P_{\text{podm}} = (n-1)/n$.

Определённая вероятностная модель шифра $\Sigma_B$ позволяет рассматривать в качестве множества открытых текстов $X$ лишь последовательности в некотором конечном алфавите $A$, длины которых ограничены некоторой заранее определённой константой. В работе [2] приводятся модели шифров замены с ограниченным и неограниченным ключом, для которых, в частности, на множество $X$ такое ограничение не накладывается. Поскольку в общем случае шифр замены с ограниченным ключом совершенным не является [2], нас будет интересовать шифр замены с неограниченным ключом. Приведем модель данного шифра.

Пусть $U$ — конечное множество возможных шифрвеличин, а $V$ — конечное множество возможных шифробозначений. Пусть имеются $r > 1$ инъективных отображений (простых замен) из $U$ в $V$. Пронумеруем данные отображения: $E_1$, $E_2$, …, $E_r$. Обозначим $\mathbb{N}_r = \{1, 2, \ldots, r\}$. Опорным шифром шифра замены назовём совокупность $\Sigma = (U, \mathbb{N}_r, V, E, D)$, для которой выполнены следующие свойства:

1) для любых $u \in U$ и $j \in \mathbb{N}_r$ выполнено равенство $D_j(E_j(u)) = u$;

2) $V = \bigcup\limits_{j \in \mathbb{N}_r} E_j(U)$.

При этом $E = \{E_1, \ldots, E_r\}$, $D = \{D_1, \ldots, D_r\}$, $D_j : E_j(U) \to U$, $j \in \mathbb{N}_r$.

Назовём $l$-й степенью опорного шифра $\Sigma$ совокупность $\Sigma^l = \left(U^l, \mathbb{N}_r^l, V^l, E^{(l)}, D^{(l)}\right)$, где $U^l, \mathbb{N}_r^l, V^l$ — декартовы степени соответствующих множеств. Множество $E^{(l)}$ состоит из отображений $E_{\overline{j}} : U^l \to V^l$, $\overline{j} \in \mathbb{N}_r^l$, таких, что для любых $\overline{u} = u_1 \ldots u_l \in U^l$, $\overline{j} = j_1 \ldots j_l \in \mathbb{N}_r^l$ выполнено равенство

$$E_{\overline{j}}(\overline{u}) = E_{j_1}(u_1) \ldots E_{j_l}(u_l) = v_1 \ldots v_l \in V^l,$$

а множество $D^{(l)}$ состоит из отображений $D_{\overline{j}} : E_{\overline{j}}(U^l) \to U^l$, $\overline{j} \in \mathbb{N}_r^l$, таких, что для любых $\overline{v} = v_1 \ldots v_l \in V^l$, $\overline{j} = j_1 \ldots j_l \in \mathbb{N}_r^l$ выполнено равенство

$$D_{\overline{j}}(\overline{v}) = D_{j_1}(v_1) \ldots D_{j_l}(v_l) = u_1 \ldots u_l \in U^l.$$

Пусть $\psi_c$ — случайный генератор ключевого потока, который для любого натурального числа $l$ вырабатывает случайный ключевой поток $j_1 \ldots j_l$, $j_i \in \mathbb{N}_r$.

Обозначим через $\Sigma_H^l$ следующую совокупность величин:

$$\Sigma_H^l = \left(U^l, \mathbb{N}_r^l, V^l, E^{(l)}, D^{(l)}, P(U^l), P(\mathbb{N}_r^l)\right).$$

Шифром замены с неограниченным ключом назовем семейство

$$\Sigma_H = \left(\Sigma_H^l, l \in \mathbb{N}; \psi_c\right).$$

При этом независимые и не содержащие нулевых вероятностей распределения $P(U^l)$ и $P(\mathbb{N}_r^l)$ индуцируют распределения вероятностей на множестве $V^l$:

$$P_{V^l}(\overline{v}) = \sum_{\substack{(\overline{u}, \overline{j}) \in U^l \times \mathbb{N}_r^l, \\ E_{\overline{j}}(\overline{u}) = \overline{v}}} P_{U^l}(\overline{u}) P_{\mathbb{N}_r^l}(\overline{j}).$$

Определим условные вероятности $P_{U^l|V^l}(\overline{u}|\overline{v})$ и $P_{V^l|U^l}(\overline{v}|\overline{u})$:

$$P_{V^l|U^l}(\overline{v}|\overline{u}) = \sum_{\overline{\jmath}\in\mathbb{N}_r^l(\overline{u},\overline{v})} P_{\mathbb{N}_r^l}(\overline{\jmath}), \qquad P_{U^l|V^l}(\overline{u}|\overline{v}) = \frac{P_{U^l}(\overline{u})P_{V^l|U^l}(\overline{v}|\overline{u})}{P_{V^l}(\overline{v})},$$

где $\mathbb{N}_r^l(\overline{u},\overline{v}) = \{\overline{\jmath}\in\mathbb{N}_r^l : E_{\overline{\jmath}}(\overline{u}) = \overline{v}\}$. Говорят, что шифр $\Sigma_H$ является совершенным тогда и только тогда, когда для любого натурального $l$ шифр $\Sigma_H^l$ является совершенным по Шеннону.

**Утверждение 6.** Для шифра $\Sigma_H$ следующие условия эквивалентны:

($i$) для любого $l\in\mathbb{N}$ и любых $\overline{u}\in U^l$, $\overline{v}\in V^l$ выполнено равенство

$$P_{U^l|V^l}(\overline{u}|\overline{v}) = P_{U^l}(\overline{u});$$

($ii$) для любого $l\in\mathbb{N}$ и любых $\overline{u}\in U^l$, $\overline{v}\in V^l$ выполнено равенство

$$P_{V^l|U^l}(\overline{v}|\overline{u}) = P_{V^l}(\overline{v});$$

($iii$) для любого $l\in\mathbb{N}$ и любых $\overline{u}_1,\overline{u}_2\in U^l$, $\overline{v}\in V^l$ выполнено равенство

$$P_{V^l|U^l}(\overline{v}|\overline{u}_1) = P_{V^l|U^l}(\overline{v}|\overline{u}_2).$$

**Теорема 3.** Пусть для шифра замены $\Sigma_H$ выполнены следующие условия:

($i$) простые замены $E_1$, $E_2$,..., $E_r$ шифра $\Sigma_H$ обладают тем свойством, что для любого $u\in U$ и любого $v\in V$ найдется, и при том единственный, элемент $j = j(u,v)\in\mathbb{N}_r$, что $E_j(u) = v$;

($ii$) распределение вероятностей $P(\mathbb{N}_r)$ для случайного генератора $\psi_c$ является равномерным.

Тогда шифр $\Sigma_H$ является совершенным, причём для любого $l\in\mathbb{N}$ выполнено равенство $|V^l| = r^l$ и распределение вероятностей $P(V^l)$ является равномерным.

***Доказательство.*** Зафиксируем произвольное натуральное число $l$. Из условия ($i$) следует, что для любых $\overline{u} = u_1\dots u_l\in U^l$ и $\overline{v} = v_1\dots v_l\in V^l$ найдется, и причём единственный, ключевой поток $\overline{\jmath} = j_1\dots j_l\in\mathbb{N}_r^l$, зависящий от $\overline{u}$ и $\overline{v}$, что

$$E_{\overline{\jmath}}(\overline{u}) = E_{j_1}(u_1)\dots E_{j_l}(u_l) = v_1\dots v_l = \overline{v}.$$

Из данного свойства и того, что $P_{\mathbb{N}_r^l}(\overline{\jmath}) = P_{\mathbb{N}_r}(j_1)\cdot\ldots\cdot P_{\mathbb{N}_r}(j_l) = 1/r^l$ для любого $\overline{\jmath}\in\mathbb{N}_r^l$ (условие ($ii$) теоремы 1), следует справедливость данной теоремы. ■

Обозначим через $P_{\text{im}}^l$ и $P_{\text{podm}}^l$ соответственно вероятности успеха имитации и подмены сообщений для шифра $\Sigma_H^l$. Из теоремы 2 следует, что если для некоторого шифра $\Sigma_H$ выполнено равенство $|U| = |V|$, где $U$, $V$ — множества шифрвеличин и шифробозначений соответственно, то $P_{\text{im}}^l = P_{\text{podm}}^l = 1$ для любого натурального $l$, то есть такие шифры максимально уязвимы к угрозам имитации и подмены сообщения.

**Утверждение 7.** Пусть $A$ — некоторая $(n+1,n)$-матрица над множеством шифробозначений $V = \{v_1,\dots,v_{n+1}\}$, которая является латинским прямоугольником относительно столбцов, и пусть матрица $A$ является матрицей зашифрования для опорного шифра замены с неограниченным ключом $\Sigma_H$. Пусть также случайный генератор

ключевых последовательностей $\varphi_c$ из конструкции шифра $\Sigma_H$ имеет равномерное распределение. Тогда для любого натурального $l$ шифр $\Sigma_H^l$ является совершенным по Шеннону и выполнены следующие равенства:

$$P_{\text{im}}^l = \left(\frac{n}{n+1}\right)^l, \quad P_{\text{podm}}^l = \left(\frac{n-1}{n}\right)^l,$$

то есть $P_{\text{im}}^l, P_{\text{podm}}^l \to 0$ при $l \to +\infty$.

***Доказательство*** следует из теоремы 3 и предложений 2 и 4.

В вероятностной модели шифра $\Sigma_B$ множества $X$ и $Y$ конечны, поэтому вероятности $P_{\text{im}}$ и $P_{\text{podm}}$ имеют достижимые нижние оценки (теорема 2). Так как для любого фиксированного натурального числа $l$ множества $U^l$ и $V^l$ также конечны, то для шифра $\Sigma_H^l$ вероятности $P_{\text{im}}^l$ и $P_{\text{podm}}^l$ также ограничены снизу. Но в модели шифра $\Sigma_H$ ограничения на длины сообщений снимаются, поэтому с ростом числа $l$ (длин сообщений) вероятности $P_{\text{im}}^l$ и $P_{\text{podm}}^l$ стремятся к 0 при выполнении условий предложения 7.

Таким образом, если в качестве матрицы зашифрования $A$ для опорного шифра $\Sigma$ использовать латинский квадрат (например, квадрат Виженера), то при равномерном распределении $P(\mathbb{N}_r)$ шифр $\Sigma_H$ будет совершенным, но максимально уязвимым к угрозам имитации и подмены. Но стоит в данном квадрате Виженера вычеркнуть, например, последний столбец, как шифр $\Sigma_H$ с полученной матрицей зашифрования для опорного шифра $\Sigma$ приобретёт два дополнительных свойства.

## ЛИТЕРАТУРА

1. *Алферов А. П., Зубов А. Ю., Кузьмин А. С., Черемушкин А. В.* Основы криптографии. М.: Гелиос АРВ, 2005. 480 с.
2. *Зубов А. Ю.* Криптографические методы защиты информации. Совершенные шифры. М.: Гелиос АРВ, 2005. 192 с.
3. *Холл М.* Комбинаторика. М.: Мир, 1970. 424 с.

2012 Математические методы криптографии №3(17)

DOI 10.17223/20710410/17/6

УДК 681.03

# ЛАТИНСКИЕ КВАДРАТЫ И ИХ ПРИМЕНЕНИЕ В КРИПТОГРАФИИ

М. Э. Тужилин

*Российский государственный гуманитарный университет, г. Москва, Россия*

**E-mail:** mtmt@rambler.ru

Сделан обзор публикаций, посвящённых применению латинских квадратов в криптографии.

**Ключевые слова:** *латинский квадрат, квазигруппа, криптография.*

## 1. История

**Определение 1.** Латинским квадратом $n$-го порядка $L = (l_{ij})$ называется таблица размера $n \times n$, заполненная $n$ символами упорядоченного множества $M$ таким образом, что в каждой строке и в каждом столбце каждый символ встречается ровно один раз.

Пример латинского квадрата:

$$\begin{array}{ccc} A & B & C \\ C & A & B \\ B & C & A \end{array}$$

Впервые латинские квадраты (4-го порядка) были опубликованы в книге «Шамс аль Маариф» («Книга о Солнце Гнозиса»), написанной Ахмадом аль-Буни в Египте приблизительно в 1200 г.

В настоящее время в качестве множества $M$ обычно берется множество натуральных чисел от 1 до $n$, однако Леонард Эйлер (1707–1783) использовал буквы латинского алфавита, откуда латинские квадраты и получили своё название [1].

Отметим, что многие подходы к изучению и построению латинских квадратов восходят к Эйлеру. Он строил латинские квадраты из латинских прямоугольников. Важной вехой в исследовании латинских квадратов была работа А. Кэли [2], в которой он привёл формулу для числа таких прямоугольников из двух строк. Следующее продвижение — вывод формулы для числа прямоугольников из трёх строк — произошло в 1953 г. [3].

В 20–30-е годы XX века стали интенсивно изучаться неассоциативные алгебраические структуры. Эти исследования, в которых принимали участие и советские математики [4], привели к созданию теории квазигрупп [5].

## 2. Число латинских квадратов

Формула для подсчёта числа $L(n)$ латинских квадратов порядка $n$ не найдена. В таблице приведены известные на настоящее время точные значения $L(n)$ [6].

**Число латинских квадратов**

| $n$ | $L(n)$ | Автор (год) |
|---|---|---|
| 1 | 1 | |
| 2 | 2 | |
| 3 | 12 | |
| 4 | 576 | |
| 5 | 161280 | Euler (1682) |
| 6 | 812851200 | Frolov (1890) |
| 7 | 61479419904000 | Sade (1948) |
| 8 | 108776032459082956800 | Wells (1967) |
| 9 | 5524751496156892842531225600 | Bammel, Rothstein (1975) |
| 10 | 9982437658213039871725064756920320000 | McKay, Rogoyski (1995) |
| 11 | 776966836171770144107444346734230682311065600000 | McKay, Wanless (2005) |

Справедливы оценки для величины $L(n)$ [7]:

$$\prod_{k=1}^{n}(k!)^{n/k} \leqslant L(n) \leqslant \frac{(n!)^{2n}}{n^{n^2}}.$$

## 3. Отношения эквивалентности на множестве латинских квадратов

**Определение 2.** Латинский квадрат называется нормализованным, если первая строка и первый столбец квадрата заполнены в соответствии с порядком, заданным на $M$.

Пример нормализованного латинского квадрата:

$$\begin{array}{ccc} A & B & C \\ B & C & A \\ C & A & B \end{array}$$

Число нормализованных латинских квадратов порядка $n$ в $n!(n-1)!$ меньше, чем $L(n)$.

**Определение 3.** Два латинских квадрата называют изотопными, если один из них может быть получен из другого в результате изотопии — композиции из перестановки строк, перестановки столбцов и замены элементов $M$ по подстановке из симметрической группы $S_n$.

Изотопия является отношением эквивалентности, поэтому все множество латинских квадратов разбивается на непересекающиеся изотопные классы. Из одного латинского квадрата можно получить с помощью изотопии $(n!)^3$ эквивалентных, но некоторые из них при этом могут совпасть, это называется автопаратопией. Доля латинских квадратов с нетривиальной группой автопаратопий стремится к нулю с ростом $n$ [6].

Латинский квадрат можно рассматривать как ортогональный массив. Меняя порядок элементов в каждой упорядоченной тройке этого массива, получаем 6 латинских квадратов, которые называются парастрофами. Это новое отношение эквивалентности, его классы называются главными классами. Каждый из них содержит 1, 2, 3 или 6 изотопных классов. Композиция изотопии и парастрофии называется изострофией. В случае совпадения латинского квадрата и изострофного ему говорят об автострофии. Доказано, что с ростом $n$ почти все латинские квадраты имеют тривиальную группу автострофий [8].

## 4. Ортогональные латинские квадраты

**Определение 4.** Два латинских квадрата $L$ и $K$ называются ортогональными, если все упорядоченные пары $(l_{ij}, k_{ij})$ различны.

Впервые пары ортогональных латинских квадратов были опубликованы в 1725 г. [9] в связи с решением задачи о расположении 16 игральных карт.

Ортогональные квадраты существуют для всех $n$, отличных от 2 и 6. Эйлер не смог построить два ортогональных латинских квадрата порядка 6 и сформулировал задачу их построения в виде «задачи о 36 офицерах» [1]. Он не смог построить и ортогональные латинские квадраты порядка 10 и высказал гипотезу о том, что не существует пар ортогональных квадратов для $n = 4r + 2$. В 1900 г. она была доказана для $n = 6$, а опровергнута в 1959 г. путём построения двух ортогональных квадратов для $n = 22$. В 1960 г. с помощью ЭВМ было построено два ортогональных квадрата для $n = 10$.

Для $n = q$, где $q$ — степень простого числа, существуют так называемые полные системы из $(n-1)$ попарно ортогональных латинских квадратов. Способ их построения в 1938 г. опубликовал Р. Боуз [10] (хотя, как потом выяснилось, этот способ был открыт Муром в 1896 г.): достаточно рассмотреть многочлены $f_a(x, y) = ax+y$ над полем $\mathrm{GF}(q)$ при ненулевых $a$.

Пример полной системы попарно ортогональных латинских квадратов, построенной по методу Боуза, $\alpha$ — примитивный элемент $\mathrm{GF}(4)$

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | $\alpha$ | $\alpha+1$ | 0 | 1 | $\alpha$ | $\alpha+1$ | 0 | 1 | $\alpha$ | $\alpha+1$ |
| 1 | 0 | $\alpha+1$ | $\alpha$ | $\alpha$ | $\alpha+1$ | 0 | 1 | $\alpha+1$ | $\alpha$ | 1 | 0 |
| $\alpha$ | $\alpha+1$ | 0 | 1 | $\alpha+1$ | $\alpha$ | 1 | 0 | 1 | 0 | $\alpha+1$ | $\alpha$ |
| $\alpha+1$ | $\alpha$ | 1 | 0 | 1 | 0 | $\alpha+1$ | $\alpha$ | $\alpha$ | $\alpha+1$ | 0 | 1 |

Для других значений $n$ известны только нижние оценки числа попарно ортогональных квадратов, входящих в систему. Например, для $n = 10$ это 2, для 12 — 5, 14 — 3, 15 — 4, 18 — 3, 20 — 4, 21 — 5, 22 — 3, 24 — 6, 26 — 4, 28 — 5, 30 — 4.

Предложено много способов построения ортогональных латинских квадратов, их можно разбить на две группы. К первой группе относятся способы построения латинского квадрата, для которого с помощью изотопии можно получить ортогональный ему латинский квадрат. Ко второй группе относятся методы конструирования на основе ортогональных квадратов меньших порядков.

## 5. Частичные латинские квадраты

Квадрат, в котором каждый элемент множества $M$ в каждой строке и в каждом столбце встречается не более одного раза, называется частичным. В [11] введено понятие критического множества, соответствующего частичному квадрату, который однозначно может быть дополнен до латинского, причём никакое его подмножество условию однозначности не удовлетворяет. Для небольших значений $n$ известна мощность критического множества, для $n = 1$ это 0, для 2 — 1, 3 — 3, 4 — 7, 5 — 11, 6 — 18. Задача распознавания того, может ли частичный квадрат быть дополнен до латинского, NP-полна.

## 6. Применение латинских квадратов в криптографии

Латинские квадраты находят применение в комбинаторике, алгебре (изучение латинских квадратов тесно связано с изучением квазигрупп), теории кодов, статистике и многих других областях [12].

Впервые в криптографии латинский квадрат был применён в шифре Тритемия [13]. Тритемий построил таблицу (соответствующую таблице Кэли группы $(\mathbb{Z}_{26}, +)$) и использовал её для многоалфавитного шифрования, когда первая буква открытого текста шифруется первым алфавитом (первой строкой таблицы), вторая — вторым и т. д. Впоследствии этот шифр усовершенствовал Дж. Белазо [14], который придумал пароль. В совокупности с идеей Л. Б. Альберти использовать произвольный алфавит это привело к появлению нового шифра на основе квазигруппы, который стал важной вехой на пути развития криптографии.

Значение латинских квадратов для криптографии иллюстрирует теорема Шеннона, в соответствии с которой единственными совершенными шифрами являются шифры гаммирования, наложение гаммы в которых определяется латинским квадратом: «Perfect systems in which the number of cryptograms, the number of messages, and the number of keys are all equal are characterized by the properties that (1) each $M$ is connected to each $E$ by exactly one line, (2) all keys are equally likely. Thus the matrix representation of the system is a Latin square» [15, p. 681].

Попытка обобщить подход Шеннона и ввести понятие «сильно совершенный шифр» предпринята в [16].

В ряду примеров применения латинских квадратов для построения поточных шифров необходимо выделить предложенный в 2005 г. шифр Edon80 [17], который дошёл до третьего тура конкурса ESTREAM. Разработчики шифра из 576 существующих латинских квадратов 4-го порядка тщательно выбрали 4, на основе которых в криптосхеме строится конвейер из 80 латинских квадратов, он используется для выработки гаммы.

При разработке блочного шифра IDEA [18] авторы использовали три квазигруппы, соответствующие операциям сложения по модулю 2, сложения по модулю $2^{16}$ и умножения по модулю $2^{16} + 1$. При этом высокие криптографические свойства шифра они обосновывали тем, что среди соответствующих квазигрупп первая и вторая не изотопны, первая и третья не изотопны, а единственной изотопией между второй и третьей квазигруппами является функция логарифма.

Квазигруппа лежит в основе конструкции, предложенной в 2005 г. в качестве однонаправленной функции [19]. Там введено понятие $e$-преобразования с лидером $l$: вектор $A = (a_0, a_1, \ldots, a_{t-1})$ оно преобразует с помощью квазигруппы $(Q, *)$ в вектор $B = ((l*a_0), ((l*a_0)*a_1), \ldots, ((l*a_0)*a_1)*\ldots*a_{t-1})$. Затем с помощью $e$-преобразования вводятся функции $R_1$ и $R_2$:

$$R_1(A) = e_{a_{t-1}}(\ldots(e_{a_1}(e_{a_0}(A))\ldots),$$
$$R_2(A) = e_{a_{t-1}}(\ldots(e_{a_1}(e_{a_0}(e_{a_{t-1}}(\ldots(e_{a_1}(e_{a_0}(A))\ldots).$$

Доказана теорема о том, что если квазигруппа $(Q, *)$ неассоциативна и некоммутативна, то для вычисления обратной к $R_1$ функции требуется $\mathrm{O}(n^{\lfloor t/3 \rfloor})$ обращений в память (т. е. к латинскому квадрату, соответствующему квазигруппе $Q$), а для вычисления обратной к $R_2$ функции требуется $\mathrm{O}(n^t)$ обращений в память.

В [20] предложена схема аутентификации сообщений. Сообщение в этой схеме делится на блоки по $t$ знаков. Для каждого $j$-го блока вида $a_1a_2\ldots a_t$ вычисляется в квазигруппе $(Q, *)$ значение $b_j = (\ldots((a_1 * a_2) * a3) * \ldots * a_{t-1}) * a_t$, которое является $j$-м элементом подписи. Имеется ряд работ, посвящённых анализу и усовершенствованию данной схемы.

Латинские квадраты являются привлекательным средством для построения схем разделения секрета. Секретом является латинский квадрат, а все участники схемы

получают соответствующие ему частичные квадраты [21]. Предложенная схема может быть усовершенствована [22]. В свою очередь, на основе таких схем разделения секрета можно строить и криптографические хеш-функции [23]. Другой пример построения криптографической хеш-функции на основе случайного латинского квадрата приведён в [24].

Разработанное в 2008 г. для участия в конкурсе SHA-3 на новый американский стандарт хеш-функции семейство Edon-R [25] не прошло во второй тур, но интересно тем, что в основе конструкции лежит построение и использование некоммутативной, неассоциативной, нелинейной квазигруппы. Авторы использовали квазигруппы порядков $2^{256}$ и $2^{512}$, изотопные группам сложения в восьмимерных векторных пространствах над соответствующими полями, и два ортогональных латинских квадрата 8-го порядка.

В [26] предложен протокол с нулевым разглашением. Каждый участник $a$ имеет открытый ключ, которым являются два изотопных латинских квадрата $L_a$ и $L'_a$. Секретным ключом является изотопия между двумя этими латинскими квадратами. Для аутентификации доказывающий участник протокола многократно вырабатывает из $L_a$ случайным образом изотопный ему латинский квадрат $H$, направляет его проверяющему и доказывает ему в зависимости от вопроса, что $H$ изотопен $L_a$ или $H$ изотопен $L'_a$.

В заключение отметим, что о растущем внимании к теме свидетельствует появление обзоров [27 – 29].

## ЛИТЕРАТУРА


1. *Euler L.* Recherches sur une nouvelle espèce de quarrés magiques. Middelburg, 1782.
2. *Cayley A.* On Latin Squares // Messenger of mathematics. 1890. V. XIX. P. 135–137.
3. *Touchard J.* Permutations, discordant with two given permutations // Scripta Math. 1953. No. 19. P. 109–111.
4. *Suschkewitsch A. K.* On the number of Latin Squares // Trans. Amer. Math. Soc. 1929. V. 31. P. 204–214.
5. *Moufang R.* Zur Struktur von Alternativkoerpern // Math. Ann. 1935. V. 110. No. 1. P. 416–430.
6. *McKay B. D. and Wanless I. M.* On the number of Latin Squares // Ann. Combin. 2005. No. 9. P. 335–344.
7. *Van Lint J. H. and Wilson R. M.* A Course in Combinatorics. Cambridge University Press, 1992.
8. *Черемушкин А. В.* Почти все латинские квадраты имеют тривиальную группу автострофий // Прикладная дискретная математика. 2009. № 3(5). С. 29–32.
9. *Ozanam J.* Récréations mathématiques et physiques. Paris, 1725.
10. *Bose R. S.* On the applications of the properties of Galois fields to the problems of construction of Hyper-Graeco-Latin squares // Indian J. Stat. 1938. No. 3. Part 4. P. 323–338.
11. *Nelder J.* Critical sets in latin squares // CSIRO Division Math. Stats, Newsletter. 1977. V. 38. P. 4.
12. *Laywine C. F. and Mullen G. L.* Discrete mathematics using Latin squares. New York: Wiley, 1998.
13. *Trithemius J.* Polygraphiae. Trittelheim, 1518.
14. *Bellaso G. B.* Il vero modo di scrivere in cifra con facilità, prestezza, et securezza di Misser Giovan Battista Bellaso, gentil'huomo bresciano // Iacobo Britannico. Bressa, 1564.

15. *Shannon C.* Communication Theory of Secrecy Systems // Bell System Technical J. 1949. V. 28. P. 656–715.
16. *Massey J.L., Maurer U., and Wang M.* Non-Expanding, Key-Minimal, Robustly-Perfect, Linear and Bilinear Ciphers // Adv. Cryptology — EUROCRYPT'87. Berlin; Heidelberg: Springer Verlag, 1988. P. 237–247.
17. *Gligoroski D., Markovski S., Kocarev L., and Gusev M.* Edon80 // `http://www.ecrypt.eu.org/stream/edon80p3.html`
18. *Lai X. and Massey J.* A Proposal for a New Block Encryption Standard // Adv. Cryptology — EUROCRYPT'90. New York: Springer Verlag, 1991. P. 55–70.
19. *Gligoroski D.* Candidate One-Way Functions and One-Way Permutations Based on Quasigroup String Transformations // `http://eprint.iacr.org/2005/352.pdf`
20. *Dènes J. and Keedwell A. D.* A new Authentication Scheme based in Latin Squares // Discrete Math. 1992. V. 106/107. P. 157–162.
21. *Cooper J., Donovan D., and Seberry J.* Secret Sharing Schemes Arising From Latin Squares // Bulletin of the ICA. 1994. V. 12. P. 33–43.
22. *Chum C.S. and Zhang X.* The Latin squares and the secret sharing schemes // Groups Complex. Cryptol. 2010. V. 2. P. 175–202.
23. *Chum C. S.* Hash functions, Latin squares and secret sharing schemes. New York: ProQuest, 2010.
24. *Pal S. K., Bhardwaj D., Kumar R., and Bhatia V.* A New Cryptographic Hash Function based on Latin Squares and Non-linear Transformations // Adv. Comput. Conf. IACC. Patiala, 2009. P. 862–867.
25. *Gligoroski D., Ødegård R.S., Mihova M., et al.* Cryptographic Hash Function Edon-R // Proc. IWSCN. Trondheim, 2009. P. 1–9.
26. *Dènes J. and Dènes T.* Non-associative algebraic system in cryptology. Protection against "meet in the middle" attack // Quasigroups and Related Systems. 2001. No. 8. P. 7–14.
27. *Глухов М. М.* О применениях квазигрупп в криптографии // Прикладная дискретная математика. 2008. № 2(2). С. 28–32.
28. *Shcherbacov V. A.* Quasigroups in cryptology // Comput. Sci. J. Moldova. 2009. V. 17. No. 2(50). P. 193–228.
29. *Малых А. Е., Данилова В. И.* Об историческом процессе развития теории латинских квадратов и некоторых их приложениях // Вестник Пермского университета. 2010. Вып. 4(4). С. 95–104.

2012 Математические методы криптографии №3(17)
DOI 10.17223/20710410/17/7
УДК 004.34

# О СОДЕРЖАНИИ ПОНЯТИЯ «ЭЛЕКТРОННАЯ ПОДПИСЬ»

А. В. Черемушкин

*Институт криптографии, связи и информатики, г. Москва, Россия*

**E-mail:** avc238@mail.ru

Сравниваются подходы к определению понятия и стандартизации электронной подписи в нашей стране и Евросоюзе.

**Ключевые слова:** *электронная подпись, цифровая подпись.*

8 апреля 2011 г. вступил в силу Федеральный закон Российской Федерации от 6 апреля 2011 г. № 63-ФЗ «Об электронной подписи» [1] (далее ФЗ). В нём введены новые для российского нормативного регулирования технические понятия усиленной и квалифицированной электронной подписи, квалифицированного сертификата и др.

Поскольку данная терминология заимствована из зарубежных технических стандартов, то представляет интерес сравнить содержание данных понятий в западном и российском нормативном и техническом регулировании.

Наиболее проработанными эти вопросы являются в документах Евросоюза, поэтому остановимся на применяющихся там подходах.

## 1. Определение электронной подписи

Основным исходным документом для стран Евросоюза в этой области является Директива 1999/93/ЕС Европейского парламента и совета от 13 декабря 1999 г. о порядке использования электронных подписей в Европейском сообществе [2] (далее Директива).

*Электронная подпись* (ЭП) определяется в Директиве как данные в электронной форме, которые добавлены либо логически ассоциированы с другими данными в электронной форме и служат методом аутентификации этих данных.

Это соответствует расширенному пониманию термина ЭП в международных правовых документах, где в качестве ЭП могут выступать биометрические данные, запись голоса, фото, пароли или другие данные, которые можно использовать для определения лица, подписавшего электронный документ. Сюда же попадают и коды аутентичности сообщений, применяемые в системах с симметричными ключами для защиты от навязывания или подмены передаваемых данных. Использование термина «электронная подпись» не привязано к какой-либо технологии. Могут применяться не только асимметричные, но и симметричные криптографические системы, а также системы, вообще не использующие криптографические алгоритмы.

Сразу следует заметить, что понятие «электронная подпись» предназначено, в основном, для юридического использования, в том смысле, что использование этого термина предполагает фактическое соответствие установленным юридическим нормам. Поэтому Директива, фактически, вводит юридическую категорию, определяющую рамки применения и юридической значимости для различных технологий подписи, верификации, подтверждения и принятия цифровых электронных данных.

ЭП может выступать в качестве основного механизма для реализации различных сервисов безопасности. Приведём технические определения некоторых сервисов безопасности из международных стандартов:

— *Аутентификация*: обеспечение гарантий идентичности предъявленной сущности [ISO/IEC 10181-2].
— *Аутентификация источника данных*: подтверждение подлинности источника полученных данных [ISO 7498-2].
— *Аутентификация сторон*: подтверждение того, что взаимодействующая сторона является той, за которую себя выдаёт [ISO 7498-2].
— *Целостность данных*: свойство, означающее, что данные не были модифицированы или уничтожены неавторизованным образом [ISO 7498-2].
— *Отрицание* (*отречение*) (*Repudiation*): отрицание одним из участвующих в коммуникации субъектов своего участия во всей или части коммуникации [ISO 7498-2].
— *Невозможность отрицания авторства* (*Non-repudiation of origin*): этот сервис предназначен для защиты от отрицания автором факта создания или отправления им сообщения [ISO/IEC 13888-1].
— *Невозможность отрицания* (*Non-repudiation*): этот сервис предназначен для сбора, обработки и обоснования неопровержимой очевидности информации, касающейся предъявленного события или действия, с целью разрешения спора о том, что событие или действие имело место в реальности [ISO/IEC 10181-4].

Заметим, что в определении ЭП не уточняется, о каком типе аутентификации идёт речь. Обычно различают два важных случая: аутентификацию источника данных и аутентификацию сторон. Если первая относится к случаю передачи подписанного сообщения с последующей отложенной проверкой подписи, то вторая выполняется в режиме on-line в рамках протокола идентификации.

Применение подписи для обеспечения целостности гарантирует, что все изменения в передаваемых данных будут обнаружены независимо от того, чем они вызваны — ошибками при передаче или целенаправленным воздействием противника в канале.

Целостность данных, взаимная аутентификация сторон и аутентификация источника представляют примеры сервисов безопасности, которые исчерпывающе описываются техническими определениями в стандартах. Для того чтобы гарантировать выполнение данных свойств, достаточно обеспечить правильную техническую реализацию. При этом соответствие требованиям законов и нормативных документов основано на научной и технологической очевидности проверки:

а) является ли подпись аутентичной, т. е. она соответствует конкретному человеку и не подделана;

б) являются ли подписываемые данные оригинальными, т. е. они соответствуют данным, представленным подписывающему, и они не были изменены.

Стандарты ISO 7498-2 и ISO/IEC 13888-1 определяют несколько типов сервиса «невозможность отрицания». Общее определение сервиса дано в ISO/IEC 10181-4. То, которое обычно ассоциируется с ЭП, называется «невозможность отрицания авторства». Если пользоваться этим определением, то следует помнить, что «отрицание автором факта создания или отправления им сообщения» относится, как правило, только к самой подписи, а не обязательно к исходному подписанному сообщению.

Невозможность отрицания является более сложным сервисом, и поэтому технологические решения должны дополняться нормативными требованиями. Термин «невозможность отрицания» в нормативном применении относится не столько к самой подписи, сколько к словесным декларациям и поведению. Юридически невозможность отрицания факта подписи обеспечивается:

а) соответствующим законом (для открытых систем), и/или

б) соглашением (для специальных систем, которые могут быть открытыми или закрытыми в зависимости от их политики).

Элементы нормативного регулирования невозможности отрицания также различаются в зависимости от функции подписи и от типа подписываемого документа/данных. Они могут быть подразделены на три типа [3]:

а) несемантические (т. е. чисто технические) элементы (например, аутентичность, целостность);

б) контекстуальные или семантические элементы, являющиеся предметом как технического, так и семантического рассмотрения (типа знание, преднамеренность, намерение, понимание, интерпретация, принуждение, ошибка, ложь, невозможность действия и т. п.);

в) чисто нормативные элементы (типа легальная валидность/невалидность, способность/неспособность, полномочие).

Чисто технический подход к определению невозможности отрицания, проведённый без рассмотрения учёта нормативных вопросов, введённых законом или контрактным соглашением, может приводить к заблуждению и неприменим для человеческих поступков. Как показывает пример заключения двустороннего контракта, даже если с технической точки зрения все очевидно, то сторонам может быть недостаточно информации для окончательного заключения контракта.

## 2. Технологии электронной подписи

Рассмотрим основные технологии, позволяющие реализовать ЭП.

**Цифровая подпись.** Стандарт ISO 7498-2:1989 определяет *цифровую подпись* как данные, присоединённые к набору данных либо являющиеся результатом криптографического преобразования исходного набора данных и позволяющие получателю подтвердить подлинность источника и целостность набора данных и обеспечить защиту против подделки, например, со стороны получателя.

Различают цифровые подписи без восстановления сообщения (когда подпись присоединяется с помощью конкатенации к исходному набору данных) и с восстановлением сообщения (когда передается одна подпись, а исходный набор данных получается в результате обратного преобразования).

Термин «цифровая подпись» предполагает наличие двух алгоритмов — для вычисления и проверки цифровой подписи, обычно называемых в математике схемой цифровой подписи [4]. Алгоритм вычисления цифровой подписи должен быть секретным, чтобы исключить возможность вычисления правильного значения подписи (например, определяться закрытым ключом), и зависеть от всех подписываемых данных. Алгоритм проверки — открытым для гарантирования возможности проверки подписи любым из участников без знания какой-либо закрытой информации.

Такое определение позволяет применять его в самых различных контекстах, например для защищённых систем передачи информации и автоматизированных систем, контролирующих работу удалённых объектов, функционирующих полностью в автоматическом режиме.

Цифровая подпись позволяет осуществить аутентификацию источника данных и обеспечить их целостность, а при взаимодействии людей — обеспечивает ещё и невозможность отрицания от авторства. Причём это — единственная технология, которая решает сразу три проблемы.

Использование термина «подпись» в данном контексте оправдано тем, что цифровая подпись имеет много общего с обычной собственноручной подписью на бумажном

документе. Собственноручная подпись также решает перечисленные выше задачи, однако между обычной и цифровой подписями имеются существенные различия. Сведём основные различия между обычной и цифровой подписями в таблицу (табл. 1).

Таблица 1

**Сравнение цифровой и собственноручной подписи**

| *Собственноручная подпись* | *Цифровая подпись* |
|---|---|
| Не зависит от подписываемого текста, всегда одинакова | Зависит от подписываемого текста, разная для различных текстов |
| Неразрывно связана с подписывающим лицом, однозначно определяется его психофизическими свойствами, не может быть утеряна | Определяется секретным ключом, принадлежащим подписывающему лицу, который может быть утерян владельцем |
| Неотделима от носителя (бумаги), поэтому отдельно подписывается каждый экземпляр документа | Легко отделима от документа, поэтому верна для всех его копий |
| Не требует для реализации дополнительных механизмов | Требует дополнительных механизмов, реализующих алгоритмы её вычисления и проверки |
| Не требует создания поддерживающей инфраструктуры | Требует создания доверенной инфраструктуры сертификатов открытых ключей |
| Не имеет срока давности | Имеет ограничения по сроку действия |

Надёжность схемы цифровой подписи оценивается сложностью решения следующих задач:

— *подделка подписи*, то есть нахождение значения подписи под заданным документом лицом, не являющимся владельцем секретного ключа;
— *подделка документа*, то есть модификации подписанного сообщения без знания секретного ключа;
— *подмена сообщения*, то есть подбор двух различных сообщений с одинаковыми значениями подписи без знания секретного ключа;
— *генерация подписанного сообщения*, то есть нахождение хотя бы одного сообщения с правильным значением подписи без знания секретного ключа.

Защита от данных атак обеспечивается выбором схемы цифровой подписи, обладающей соответствующими криптографическими свойствами.

Приведённое выше определение не ограничивает цифровую подпись применением только одной технологии асимметричных криптосистем, хотя эта технология и является сегодня доминирующей.

При использовании асимметричных криптосистем цифровая подпись не позволяет:

— знать владельца открытого ключа;
— знать, что закрытый ключ во время создания подписи находился у владельца закрытого ключа подписи.

Для достижения первого свойства приходится создавать инфраструктуру (сертификатов) открытых ключей. Второе обычно достигается введением информации о статусе сертификата с помощью проставления токенов с временны́ми штампами или временны́ми метками. Токены с временны́ми штампами представляют собой структуру данных, содержащую доверенную временную метку и подписанную специальным центром временных штампов. Временна́я метка — это запись в файле аудита, включающая доверенное значение времени и хешированное представление даты.

Поэтому для реализации схемы цифровой подписи, как правило, используют либо системы с открытыми ключами, что соответственно приводит к необходимости созда-

вать инфраструктуру (сертификатов) открытых ключей и центры временных штампов, либо прибегают к использованию услуг третьей стороны — доверенного центра, выступающего посредником в процедурах подписания и проверки подписи.

При этом должны выполняться два условия — временна́я метка и токен с временны́м штампом должны соответствовать периоду действия, указанному в сертификате, и сам сертификат не должен быть аннулирован.

**Код аутентичности сообщения.** Представляет собой значение некоторой определяемой ключом хеш-функции, вычисленное для данного сообщения [4]. Является разновидностью симметричных криптографических систем, поскольку у подписывающего и проверяющего ключи должны быть одинаковыми.

Позволяет проверить целостность и аутентифицировать источник данных и отправителя, так как получатель может убедиться в том, что это сообщение, помимо него самого, мог создать и отправить только отправитель, обладающий тем же ключом. В то же время код аутентичности не может обеспечить невозможность отрицания авторства отправителем, так как получатель не сможет доказать третьей стороне, что он не сам создал подписанное сообщение.

## 3. Виды подписей

На практике встречаются самые различные системы ЭП. Требования к типу ЭП во многом зависят от условий применения, которые, в свою очередь, влияют на процессы формирования и проверки (верификации) подписи.

Условия проведения верификации подписи и соответственно данных, ассоциированных с подписью, могут значительно различаться в зависимости от назначения и предполагаемого жизненного цикла подписи. Так, в [5] выделены следующие типы подписи в зависимости от особенностей процесса её верификации:

— одноразовые подписи (ephemeral signature): подписи, значения которых не нужно сохранять после проведения верификации;
— краткосрочные подписи (short term signatures): подписи, верификацию которых необходимо проводить только в период срока годности сертификата;
— долговременные подписи (long term signatures): подписи, верификацию которых необходимо проводить не только в период срока годности сертификата владельца ключа подписи, но, возможно, даже и после окончания срока полномочий выдавшего сертификат удостоверяющего центра.

В [3] сформулированы четыре основных функции электронной подписи:

— подпись для идентификации — соответствует аутентификации сторон, не связана ни с каким документом, обеспечивает доказательство владения закрытым ключом в протоколе идентификации;
— подпись для аутентификации — соответствует аутентификации источника данных, в которых данные представляются как объект, а не как документ с определённым семантическим содержанием;
— подпись для декларации знания — предназначена для обеспечения аутентификации источника данных, причём данные имеют семантическое значение, но никак не связаны с намерениями;
— подпись для декларации намерения — обеспечивает невозможность отрицания событий или действий. Относится к специальным наборам данных и требует чёткой интерпретации содержания данных и наличия контекста для процесса создания подписи.

Директива устанавливает, фактически, три вида подписи: усиленная квалифицированная, усиленная неквалифицированная и не являющаяся усиленной (в российских документах называемая «простой»).

### 3.1. Усиленная ЭП

Согласно Директиве, усиленная электронная подпись должна удовлетворять следующим требованиям:

a) она однозначно связана с лицом, подписавшим данные;

b) с её помощью можно подтвердить подлинность лица, подписавшего данные;

c) она создана с использованием средств, которые находятся под единоличным контролем лица, подписавшего данные, и

d) она связана с данными, которым она соответствует, таким способом, что с её помощью можно обнаружить любые последующие изменения подписанных данных.

Таким образом, здесь говорится о выполнении свойств однозначной аутентификации источника и обеспечения целостности. Как известно, эти свойства могут быть обеспечены только применением схемы цифровой подписи. Дело в том, что коды аутентификации сообщений, основанные на симметричных криптографических алгоритмах, создают электронную подпись, которая не является усиленной, поскольку ключом создания подписи обладают две стороны (нарушено условие $a$). Применение некриптографических методов позволяет только указать на лицо, подписавшее данные, не обеспечивая ни аутентификации источника, ни целостности, ни невозможности отрицания.

В то же время, как отмечается в [3], пользоваться таким определением чрезвычайно трудно, так как оно слишком широко и не привязано ни к какой конкретной технологии. Поэтому в случае спора экспертам надо проверить выполнение каждого из четырёх требований.

Условие $a$ можно выполнить, вводя квалифицированные либо неквалифицированные X509 сертификаты, которые должны выпускаться либо доверенной третьей стороной, либо самим подписывающим лицом. Условие $b$ проверяется по сертификату. Условие $c$ реализуется применением специальных средств для вычисления и проверки подписи, представляющих собой независимые защищённые устройства, удовлетворяющие требованиям безопасности. Для выполнения условия $d$ обычно применяют хеш-функции с достаточной длиной ключа.

Усиленная подпись удовлетворяет всем свойствам цифровой подписи из табл. 1, за исключением создания инфраструктуры сертификатов открытых ключей, вместо которой необходимо разработать нормативные документы, регулирующие конфликтные ситуации, связанные с возможностью отрицания факта создания подписи.

В ФЗ определение усиленной неквалифицированной ЭП (п. 3 ст. 5) получено усреднением двух определений — усиленной подписи из Директивы и цифровой подписи из международного стандарта ISO 7498-2: 1989:

1) она получена в результате криптографического преобразования информации с использованием ключа электронной подписи;

2) она позволяет определить лицо, подписавшее электронный документ;

3) она позволяет обнаружить факт внесения изменений в электронный документ после момента его подписания;

4) она создаётся с использованием средств электронной подписи.

Здесь в целом говорится о том же самом, но изменённая формулировка первого пункта приводит к существенному отличию: если в формулировке Директивы усло-

вия *a*, *b* и *d* могут быть выполнены только при использовании цифровой подписи на основе асимметричных криптографических систем, то под определение, приведённое в ФЗ, подпадают ещё и коды аутентификации сообщений. Кроме того, формулировка второго пункта некорректна — подпись не позволяет определить лицо, а подтверждает подлинность определённого лица. Наконец, при таком подходе не делается различия между усиленной и неквалифицированной подписями.

### 3.2. Квалифицированная подпись

Главная цель Директивы — унификация правил использования электронной подписи и наиболее общая формулировка условий, необходимых для признания юридической равнозначности электронной и собственноручной подписей. На основе анализа законов, принятых в различных странах мира, в Директиве закреплены наиболее существенные моменты. Они сформулированы в ст. 5 Директивы: необходимо использовать усиленную электронную подпись, которая должна быть основана на квалифицированном сертификате и сформирована с помощью защищённого устройства создания подписи. По сути, это — определение квалифицированной электронной подписи, хотя в Директиве этот термин и не вводится. Он появляется только в стандарте CWA 14167 со ссылкой на Директиву. В ст. 5 Директивы утверждается, что такая электронная подпись удовлетворяет всем требованиям к подписи под данными в электронной форме, позволяющим считать её равносильной собственноручной подписи на бумаге.

В Приложениях I – IV к Директиве приведены соответственно:

- требования, предъявляемые к квалифицированным сертификатам;
- требования к службе CSP, выпускающей квалифицированные сертификаты;
- требования, предъявляемые к защищённым устройствам создания электронной подписи;
- рекомендации для безопасной верификации подписи.

В Директиве вводятся формальные определения понятий квалифицированного сертификата, провайдера служб сертификатов и защищённого устройства создания подписи:

- *Квалифицированный сертификат* (Qualified Certificate, QC) — сертификат, удовлетворяющий требованиям, сформулированным в Приложении I, созданный провайдером сертификационных услуг, удовлетворяющим требованиям, сформулированным в Приложении II Директивы.
- *Провайдер сертификационных услуг* (Certification-Service-Provider, CSP) — организация (entity), либо юридическое или физическое лицо, которая выпускает сертификаты либо обеспечивает другие сервисы, связанные с электронной подписью.
- *Защищённое устройство создания подписи* (Secure-Signature-Creation Device, SSCD) — сконфигурированное программное или аппаратное устройство создания подписи, которое удовлетворяет требованиям, сформулированным в Приложении III Директивы.

Таким образом, чтобы удостовериться в том, что подпись квалифицированная, получатель должен убедиться, что:

a) сертификат является квалифицированным (удовлетворяет требованиям Приложения I Директивы);

b) провайдер служб сертификатов удовлетворяет требованиям Приложения II Директивы;

c) технология, использованная для формирования подписи, является безопасной (удовлетворяет требованиям Приложения III Директивы);

d) верификация подписи выполнена в соответствии с требованиями Приложения IV Директивы (в частности, проверено, что в сертификате указана политика QC + SSCD и данные для формирования подписи хранятся в устройстве SSCD).

Аналогичный подход принят в п. 1 ст. 6 ФЗ: «Информация в электронной форме, подписанная квалифицированной электронной подписью, признается электронным документом, равнозначным документу на бумажном носителе, подписанному собственноручной подписью, кроме случая, если федеральными законами или принимаемыми в соответствии с ними нормативными правовыми актами установлено требование о необходимости составления документа исключительно на бумажном носителе». В ст. 11 ФЗ приведены условия признания квалифицированной электронной подписи.

Однако можно заметить, что в п. 4 ст. 6 ФЗ вводится немного изменённое определение понятия квалифицированной электронной подписи: «Квалифицированной электронной подписью является электронная подпись, которая соответствует всем признакам неквалифицированной электронной подписи и двум дополнительным признакам:

1) ключ проверки электронной подписи указан в квалифицированном сертификате;

2) для создания и проверки электронной подписи используются средства электронной подписи, получившие подтверждение соответствия требованиям, установленным в соответствии с настоящим Федеральным законом».

Заметим, что в европейских документах неквалифицированная электронная подпись — это усиленная, но не квалифицированная, а согласно приведённому выше определению из ФЗ, усиленная и неквалифицированная электронная подписи совпадают, причём квалифицированная электронная подпись является частным случаем неквалифицированной электронной подписи.

### 3.3. Простая электронная подпись

Согласно п. 2. ст. 5 ФЗ, «простой электронной подписью является электронная подпись, которая посредством использования кодов, паролей или иных средств подтверждает факт создания электронной подписи определённым лицом». Такое определение не вносит ясности в понимание того, чем может являться простая электронная подпись и какие технологии допустимы для её реализации, тем более что все последующие уточнения касаются только квалифицированной подписи.

Различные виды таких подписей обсуждаются в работе [6], где отмечается, что в западном законодательстве введение понятия простой подписи было обусловлено её широким использованием на ранних этапах развития технологий. Поэтому автор [6] выражает недоумение, зачем надо в нашей стране реанимировать устаревшие технологии, которые, к тому же, не обеспечивают выполнения свойств невозможности отрицания, а в отдельных случаях — и обеспечения целостности документов.

Формального определения простой электронной подписи не существует. В [3] эта ситуация обсуждается путем логического перебора возможных ситуаций, приводящих к нарушению имеющегося формального определения квалифицированной подписи. Множество подписей, которые не являются квалифицированными и поэтому не могут быть признаны равнозначными собственноручной подписи, распадается на два класса: усиленные и не являющиеся усиленными (в ФЗ они названы «простыми»). Такой подход к определению (через отрицание) предполагает, что к этим классам можно отнести те электронные подписи, которые хотя и имеют электронную форму, но не удовлетворяют основным условиям квалифицированной подписи, то есть для них не выполнено хотя бы одно из условий $a$, $b$ или $c$ определения квалифицированной подписи.

Рассмотрим два важных случая.

**Подписи, не основанные на квалифицированном сертификате.** Если сертификат не является квалифицированным, например используется PGP, то проверяющий не сможет доказать, что автором подписи является именно данный субъект. Тем не менее, как показывают следующие примеры, такие системы широко применяются.

При аутентификации источника данные для проверки подписи содержатся в сертификате, заверенном некоторым центром сертификации. Поэтому из признания сертификатов этого центра вытекает доверие к результату проверки электронной подписи, хотя она и не может быть признана равносильной собственноручной подписи.

Если подпись получена с помощью защищённого устройства создания подписи, то обеспечено условие целостности, хотя и не ясно, кто подписал документ. В некоторых случаях, когда не важно, кто подписал документ, а имеет значение только его целостность, такая электронная подпись может оказаться достаточной.

Наконец, в системах групповой подписи для проверяющего не важно, кто из участников группы подписал документ. Главное — он может убедиться в том, что подпись сделана кем-то из участников этой группы от её имени.

**Подписи, не созданные защищённым устройством создания подписи.** Если средство создания подписи не является защищённым (не выполнены условия Приложения III Директивы), то подпись может быть как простой, так и усиленной (профиль защиты для таких устройств приведён в [7]). В этом случае можно гарантировать целостность данных и аутентификацию источника, но нельзя обеспечить невозможность отрицания. Иногда этого бывает достаточно. Например, если в системе внутреннего защищённого документооборота, обладающей достаточно высокой защищённостью, для формирования подписи используются идентификационные пластиковые карты, то, хотя сами карты могут и не удовлетворять всем требованиям к защищённым устройствам создания подписи, защита подписи от отрицания будет высокой.

### 3.4. Условия признания простой и неквалифицированной подписи равнозначной собственноручной подписи

В п. 3 ст. 6. ФЗ установлено, что «информация в электронной форме, подписанная простой электронной подписью или неквалифицированной электронной подписью, признаётся электронным документом, равнозначным документу на бумажном носителе, подписанному собственноручной подписью, в случаях, установленных федеральными законами, принимаемыми в соответствии с ними нормативными правовыми актами или соглашением между участниками электронного взаимодействия. Нормативные правовые акты и соглашения между участниками электронного взаимодействия, устанавливающие случаи признания электронных документов, подписанных неквалифицированной электронной подписью, равнозначными документам на бумажных носителях, подписанным собственноручной подписью, должны предусматривать порядок проверки электронной подписи. Нормативные правовые акты и соглашения между участниками электронного взаимодействия, устанавливающие случаи признания электронных документов, подписанных простой электронной подписью, равнозначными документам на бумажных носителях, подписанным собственноручной подписью, должны соответствовать требованиям ст. 9 настоящего Федерального закона».

Далее в ст. 9 установлено, что электронный документ считается подписанным простой электронной подписью при выполнении в том числе одного из следующих условий: либо электронная подпись «содержится в самом электронном документе»; либо она получена в результате криптографического преобразования, причём ключ «применяется в соответствии с правилами, установленными оператором информационной

системы, с использованием которой осуществляются создание и (или) отправка электронного документа, и в созданном и (или) отправленном электронном документе содержится информация, указывающая на лицо, от имени которого был создан и (или) отправлен электронный документ».

При этом как для неквалифицированной, так и для простой электронной подписи должны быть разработаны нормативные правовые акты и (или) соглашения между участниками электронного взаимодействия, в которых должны быть оговорены указанные в ФЗ существенные моменты применения таких электронных подписей.

## 4. Стандартизация технических требований

В основу Директивы положены следующие принципы:

— отказ от дискриминации электронной подписи перед законом;
— признание необходимости информационной защищённости подписи;
— возможность признания квалифицированной электронной подписи как юридически значимой.

Директива определяет следующие правила признания квалифицированной электронной подписи:

— квалифицированная электронная подпись не является синонимом юридически значимой подписи;
— невыполнение требований к квалифицированной электронной подписи не означает отказ от принятия подписи к рассмотрению её юридической значимости;
— выполнение требований к квалифицированной электронной подписи является одним, но не единственными требованием, дающим возможность приравнять электронную подпись к собственноручной подписи.

Технические требования к квалифицированной электронной подписи, сформулированные в Приложениях I – IV, в дальнейшем уточняются и дополняются в многочисленных стандартах серии CWA и ETSI, составляющих нормативную базу разработки, внедрения и использования квалифицированной подписи (см. рис. 1, взятый из работы [8]).

Аббревиатурой CWA (сокр. от CEN Workshop Agreement) обозначается документ, выпущенный Европейским комитетом по стандартизации CEN (Comite Europeende Normalisation (European Committee for Standardization)). Это неофициальный стандарт от независимой организации.

Для электронной подписи основными являются следующие CWA, признаваемые в качестве технических стандартов, соответствующих Директиве, и формулирующие требования по безопасности для доверенных систем управления сертификатами электронных подписей:

— CWA 14167-1: Требования к безопасности системы [9];
— CWA 14167-2: Криптографический модуль CSP для операций подписи с восстановлением — Профиль безопасности [10];
— CWA 14167-3: Криптографический модуль CSP для служб генерации ключей — Профиль безопасности [11];
— CWA 14167-4: Криптографический модуль CSP для операций подписи — Профиль безопасности [12];
— CWA 14169: Защищённое средство создания подписи уровня 'EAL 4+' [13].

В соответствии со стандартом CWA 14167 доверенная система (Trustworthy Systems, TWS) для работы с квалифицированными (QC) и неквалифицированными

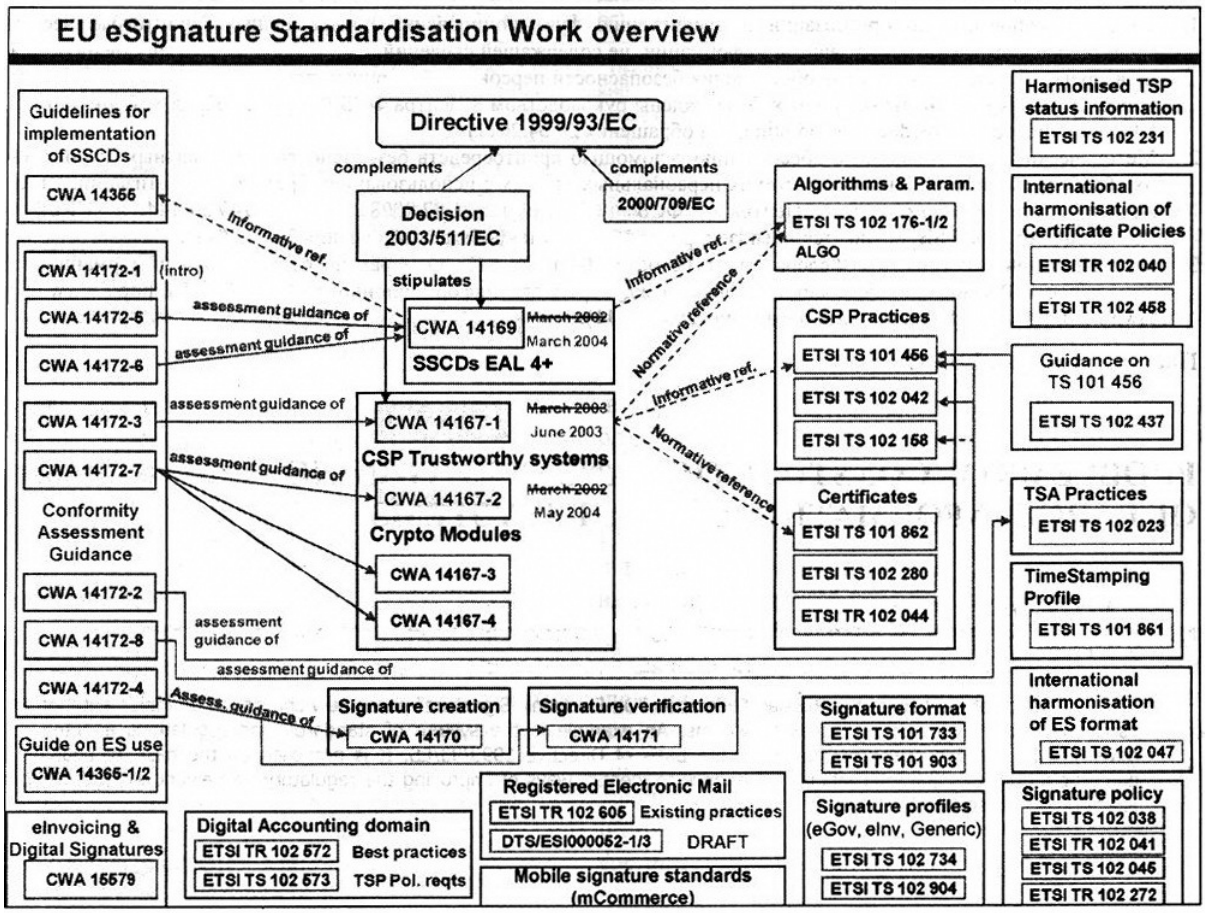


Рис. 1. Стандарты Евросоюза, составляющие нормативную базу разработки, внедрения и использования квалифицированной подписи

(NQC) сертификатами должна включать реализации основных (Core Services) и дополнительных (Supplementary Services) служб.

Основными службами являются:

— служба регистрации (Registration Service) — осуществляет проверку идентичности и, если это необходимо, других специфических атрибутов субъекта;
— служба генерации сертификатов (Certificate Generation Service) — создаёт сертификаты;
— служба распространения (Dissemination Service) — представляет сертификаты и информацию политики безопасности субъектам и доверенным сторонам;
— служба управления отзывом сертификатов (Revocation Management Service) — осуществляет обработку запросов по отзыву сертификатов;
— служба состояния аннулирования (Revocation Status Service) — предоставляет доверенным сторонам информацию о текущем состоянии списка аннулированных сертификатов.

Дополнительными службами являются:

— служба снабжения субъектов устройствами (Subject Device Provision Service) — осуществляет подготовку и снабжение субъектов устройствами создания подписи (SCDev). Включает снабжение защищёнными устройствами создания подписи (SSCD);

— служба штампов времени (Time-stamping Service) — осуществляет простановку меток времени, которые могут понадобиться в процессе проверки подписи.

Стандартом CWA 14169 [13] предусмотрено три варианта построения защищённого устройства создания подписи, соответствующего уровню адекватности EAL 4+:

— т и п 1: специальное устройство для генерации данных для создания и проверки подписи (ключей), которое может управляться только специально выделенным авторизованным лицом (например, системным администратором);
— т и п 2: персональное устройство для создания и проверки подписи, может использоваться только определённым пользователем (например, владельцем смарт-карты). Требует защищённого канала для ввода данных для создания (ключа создания) подписи;
— т и п 3: персональное устройство, сочетающее функции устройств типа 2 и типа 1 (без экспорта данных для создания подписи), например, реализованное в переносном компьютере.

Требования описанных в этом стандарте профилей защиты относятся только к аппаратной платформе, операционной системе и устройствам создания и проверки подписи и генерации и хранения данных для создания и проверки подписи. В них не конкретизируются способы создания защищённых каналов для передачи ключевой информации и защищённых путей и каналов для передачи подписываемой информации между этими устройствами и использующими их приложениями, описанию которых посвящены стандарты серии ETSI TC.

Стандарты CWA 14167 и 14169 были обновлены в 2004 г. после признания их в ЕС в соответствии с решением комиссии (Commission Decision) от 14 июля 2003 г. [14].

В настоящее время выпущено более 40 других CWA, имеющих дело с электронной подписью (см. рис. 1). Среди них:

— CWA 14170: Требования безопасности для приложений, формирующих подпись [15];
— CWA 14171: Общие требования для верификации электронной подписи [5];
— CWA 14172: Согласованные правила оценки в рамках Европейской инициативы по стандартизации электронной подписи EESSI (European Electronic Signature Standardisation Initiative).

Аббревиатурой ETSI TS (сокр. от European Telecommunication Standards Infrastructure Technical Specification) обозначается документ (техническая спецификация), выпущенный независимой профессиональной организацией ETSI, официально признаваемой Евросоюзом наряду с CEN в качестве Европейской организации по стандартизации (European Standards Organization). Стандарты по электронной подписи серии ETSI разрабатываются техническим комитетом по вопросам электронной подписи и инфраструктурам (TC — Technical Committiee / ESI — Electronic Signatures and Infrastructures). Среди документов, выпущенных ETSI TC/ESI, следует прежде всего отметить технические стандарты, содержащие списки рекомендуемых алгоритмов и параметров для безопасных электронных подписей:

— ETSI TS 102 176-1: Хеш-функции и асимметричные алгоритмы [16];
— ETSI TS 102 176-2: Протоколы защищённых каналов и алгоритмы для средств создания подписей [17].

Цель этих стандартов состоит не в перечислении всех приемлемых алгоритмов и протоколов, а в указании проверенных с точки зрения безопасности и наиболее желательных для использования в схемах усиленных электронных подписей с учётом

достижения интероперабельности, то есть возможности совместимости с наиболее широким спектром приложений. Так, в качестве хеш-функций рекомендовано использовать функции, приведённые в табл. 2, а в качестве схем цифровой подписи — схемы, приведённые в табл. 3. Для каждого варианта «хеш-функция — схема подписи — алгоритм генерации ключей» предложены наиболее подходящие значения параметров в зависимости от предполагаемого срока надёжного использования.

Таблица 2

**Рекомендованные хеш-функции**

| Краткое обозначение | Дата принятия | Нормативная ссылка |
|---|---|---|
| sha1 | 01.01.2001 | ISO/IES 10118-3, FIPS Publication 180-2 |
| ripemd160 | 01.01.2001 | ISO/iES 10118-3 |
| sha224 | 2004 | FIPS Publication 180-2 |
| sha256 | 2004 | ISO/IES 10118-3, FIPS Publication 180-2 |
| whirpool | 2004 | ISO/IES 10118-3, |
| sha384 | 31.03.2007 | FIPS Publication 180-2 |
| sha512 | 31.03.2007 | FIPS Publication 180-2 |

Таблица 3

**Рекомендованные схемы цифровой подписи**

| Краткое обозначение | Алгоритмы генерации ключей | Нормативная ссылка |
|---|---|---|
| rsa | rsagen1 | RFC 3447 |
| dsa | dsagen1 | FIPS Publication 180-2, ISO/IES 14888-3:2006 |
| ecdsa-Fp | ecgen1 | ANSI X9.62 |
| ecdsa-F2m | ecgen2 | ANSI X9.62 |
| ecgdsa-Fp | rsagen1 | ISO/IEC 15946-2:2002 |
| ecdsa-F2m | ecgen2 | ISO/IEC 15946-2:2002 |

Вторая часть технического стандарта определяет также список симметричных алгоритмов и протоколов, которые могут использоваться для построения защищённого канала между приложением и защищённым средством создания подписи (SCDev), обеспечивающим как целостность, так и одновременно целостность и конфиденциальность. Такой безопасный канал можно использовать для удалённой загрузки закрытого ключа в средство создания подписей (когда пара ключей не сгенерирована SCDev и необходимо удалённо загрузить в SCDev и личный ключ, и сертификат открытого ключа), удалённого извлечения из него открытого ключа (когда пара ключей сгенерирована средством создания подписи) или/и удалённой загрузки сертификата открытого ключа и связывания его с личным ключом, уже хранимым в средстве создания подписи (когда пара ключей сгенерирована SCDev для удалённой загрузки в SCDev сертификата открытого ключа и необходимо ассоциировать его с ранее сгенерированным личным ключом).

В ФЗ принят другой подход к формированию технических требований, в основе которого лежит передача всех полномочий по формулированию и контролю за выполнением требований по безопасности специальному выделенному федеральному органу исполнительной власти в области обеспечения безопасности, который в соответствии с п. 5 ст. 8:

1) устанавливает требования к форме квалифицированного сертификата;

2) устанавливает требования к средствам электронной подписи и средствам удостоверяющего центра;

3) осуществляет подтверждение соответствия средств электронной подписи и средств удостоверяющего центра требованиям, установленным в соответствии с настоящим Федеральным законом, и публикует перечень таких средств.

Это обусловлено различиями в используемых подходах к стандартизации в Евросоюзе и в нашей стране, связанными с принятием в 2002 г. Федерального закона № 184-ФЗ о техническом регулировании, отменившим с 30.06.2010 г. все государственные стандарты, заменив их на ещё не разработанные специальные технические регламенты (СТР).

Поэтому в соответствии с ФЗ Директором ФСБ России утверждены:

— Требования к средствам электронной подписи и Требования к средствам удостоверяющего центра [18];
— Требования к форме квалифицированного сертификата ключа проверки электронной подписи [19].

В ФЗ и требованиях [18], регламентирующих разработку, производство, реализацию и эксплуатацию средств электронной подписи в Российской Федерации, заложены следующие принципы:

1. Средства электронной подписи (используемые для реализации хотя бы одной из следующих функций: создание электронной подписи, проверка электронной подписи, создание ключа электронной подписи и ключа проверки электронной подписи) отнесены к шифровальным (криптографическим) средствам (ФЗ п. 9 ст. 2); средства удостоверяющего центра — программные и/или аппаратные средства, используемые для реализации функций удостоверяющего центра (ФЗ п. 10 ст. 2).

Поэтому на них распространяются требования, закреплённые Положением о разработке, производстве, реализации и эксплуатации шифровальных (криптографических) средств защиты информации [20] (Положение ПКЗ-2005).

2. В зависимости от способностей противостоять атакам средства электронной подписи подразделяются на шесть классов: КС1, КС2, КС3, КВ1, КВ2 и КА1 (см. табл. 4) в зависимости от уровней криптографической защиты, определенных в порядке возрастания количества и жесткости предъявляемых к криптосредствам требований, которые точно соответствуют принятым в нашей стране шести типам нарушителей Н1, ..., Н6. Более подробно о методологии создания модели нарушителя см., например, в [21].

В соответствии с уровнем криптографической защиты к средствам ЭП и удостоверяющего центра предъявляются определённые функциональные требования по управлению доступом, контролю целостности, очистке оперативной и внешней памяти, идентификации и аутентификации, защите вводимых и экспортируемых данных, регистрации событий, резервному копированию и восстановлению и т. п.

Например, в отношении ключевой информации для средств удостоверяющего центра всех классов, начиная с КС1, не допускается копирование информации ключевых документов (криптографических ключей, в том числе ключей ЭП) на носители (например, жёсткий диск), не являющиеся ключевыми носителями, без её предварительного шифрования (которое должно осуществляться встроенной функцией используемой системы криптографической защиты информации). В то же время для средств удостоверяющего центра классов КВ1, КВ2 и КА1 должны быть приняты организационно-

Таблица 4

**Возможности нарушителя**

| Возможности нарушителя | КС1 | КС2 | КС3 | КВ1 | КВ2 | КА1 |
|---|---|---|---|---|---|---|
| Проведение атак внутри контролируемой зоны (КЗ) | – | + | + | + | + | + |
| Проведение лабораторных исследований вне КЗ | – | + | + | + | + | + |
| Доступ к средствам вычислительной техники, на которых реализованы средства ЭП | – | – | + | + | + | + |
| Наличие аппаратных компонент средства ЭП и среды функционирования | – | – | + | + | + | + |
| Привлечение специалистов по анализу сигналов | – | – | – | + | + | + |
| в т. ч. по недокументированным возможностям программного обеспечения (ПО) | – | – | – | – | + | + |
| Наличие исходных текстов прикладного ПО | – | – | – | – | + | + |
| Наличие всей документации на аппаратуру и ПО | – | – | – | – | – | + |
| Наличие всех аппаратных компонент | – | – | – | – | – | + |

технические меры, исключающие возможность компрометации ключа ЭП, используемого для подписи сертификатов ключей проверки ЭП и списков аннулированных сертификатов, при компрометации ключевой информации, доступной одному, двум и трём лицам соответственно (п. 27 Приложения 2 к [18]).

Построение документов и терминология в западных и отечественных документах различны. Соответствие основных понятий приведено в табл. 5.

Таблица 5

**Сравнение терминологии**

| Директива 1999/93/EC | № 63-ФЗ |
|---|---|
| Провайдер сертификационных услуг (CSP) | Удостоверяющий центр |
| Устройство создания подписи (SCDev) | Средство электронной подписи |
| Защищённое устройство создания подписи (SSCD) | Средство электронной подписи |
| CSP SSCD | Средство удостоверяющего центра |
| Данные для создания подписи (SCD) | Ключ электронной подписи |
| Данные для верификации подписи (SVD) | Ключ проверки электронной подписи |

## Заключение

Приведённый краткий обзор подходов к нормативному и техническому регулированию вопросов использования электронной подписи в нашей стране и за рубежом показывает, что, несмотря на сходство, имеются принципиальные отличия, заключающиеся в несовпадении определений основных понятий и систем стандартизации технических требований. Кроме того, следует обращать внимание на отличия в определениях основных терминов, вводимых в математической литературе и в технических стандартах, обусловленных ориентированностью последних на применения в юридической сфере.

## ЛИТЕРАТУРА

1. Федеральный закон Российской Федерации от 06.04.2011 г. № 63-ФЗ «Об электронной подписи».
2. Directive 1999/93/EC of the European Parliament and of the Council on a Community framework for electronic signatures // Official J. European Communities. OJ L 13. 19.01.2000. P. 12.

3. CEN CWA 14365-1: Guide on the Use of Electronic Signatures. Part 1: Legal and Technical Aspects. March 2004. 28 p. (Ref. No.: CWA 14365-1:2004 E).
4. *Menezes A. J., van Oorschot P. C., and Vanstone S. A.* Handbook of applied cryptography. Boca Raton; New York; London; Tokyo: CRC Press, 1997.
5. CEN CWA 14171: General guidelines for electronic signature verification. May 2004. 46 p. (Ref. No.: CWA 14171:2004 D/E/F).
6. *Пазизин С. В.* От цифровой подписи через электронную цифровую подпись к простой «не подписи» // BIS J. — Информационная безопасность банков. 2012. № 1. С. 54–61.
7. CEN CWA 14365-2: Guide on the Use of Electronic Signatures. Part 2: Protection Profile for Software Signature Creation Devices. March 2004. 63 p. (Ref. No.: CWA 14365-2:2004 E).
8. *Попов В. О.* Проблемы построения и функционирования больших систем защиты информации // Системы высокой доступности. 2011. Т. 7. № 2. С. 6—64.
9. CEN CWA 14167-1: Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures. Part 1: System Security Requirements. June 2003. 47 p. (Ref. No.: CWA 14167-1:2003 D/E/F).
10. CEN CWA 14167-2: Cryptographic module for CSP signing operations with backup — Protection profile — CMCSOB PP. May 2004. 89 p. (Ref. No.: CWA 14167-2:2004 E).
11. CEN CWA 14167-3: Cryptographic module for CSP key generation services — Protection profile (CMCKG-PP). May 2004. 69 p. (Ref. No.: CWA 14167-3:2004 E).
12. CEN CWA 14167-4: Cryptographic module for CSP signing operations — Protection profile (CMCSO PP). May 2004. 83 p. (Ref. No.: CWA 14167-4:2004 E).
13. CEN CWA 14169: Secure signature-creation devices 'EAL 4+'. May 2004. 219 p. (Ref. No.: CWA 14169:2004 E).
14. COMMISSION DECISION of 14 July 2003 on the publication of reference numbers of generally recognised standards for electronic signature products in accordance with Directive 1999/93/EC of the European Parliament and of the Council (notified under document number C(2003) 2439) // Official J. European Union. 15.07.2003. L 175/45–46.
15. CEN CWA 14170: Security requirements for signature creation applications. May 2004. 63 p. (Ref. No.: CWA 14170:2004 E).
16. ETSI TS 102 176-1: Electronic Signatures and Infrastructures (ESI); Algorithms and Parameters for Secure Electronic Signatures; Part 1: Hash functions and asymmetric algorithms. V 2.0.0 (2007-11) (Ref. No.: RTS/ESI-000054-1).
17. ETSI TS 102 176-2: Electronic Signatures and Infrastructures (ESI); Algorithms and Parameters for Secure Electronic Signatures; Part 2: Secure channel protocols and algorithms for signature creation devices. V1.2.1. (2005-07). (Ref. No.: RTS/ESI-000039-2).
18. Приказ ФСБ № 796 от 27 декабря 2011 г. «Об утверждении Требований к средствам электронной подписи и Требований к средствам удостоверяющего центра». (Зарегистрировано в Минюсте РФ 27 января 2012 г., рег. № 23041.)
19. Приказ ФСБ № 795 от 27 декабря 2011 г. «Об утверждении Требований к форме квалифицированного сертификата ключа проверки электронной подписи». (Зарегистрировано в Минюсте РФ 9 февраля 2012 г., рег. № 23191.)
20. Положение о разработке, производстве, реализации и эксплуатации шифровальных (криптографических) средств защиты информации с ограниченным доступом, не содержащей сведений, составляющих государственную тайну. (Положение ПКЗ-2005.) Утверждено приказом ФСБ России от 9 февраля 2005 г. № 66. (Зарегистрировано Минюстом России 3 марта 2005 г., рег. № 6382) (с изменениями, внесёнными приказом ФСБ России от 12 апреля 2010 г. № 173 (зарегистрировано Минюстом России 25 мая 2010 г., рег. № 17350).)

21. Методические рекомендации по обеспечению с помощью криптосредств безопасности персональных данных при их обработке в информационных системах персональных данных с использованием средств автоматизации. Утверждены руководством 8 Центра ФСБ России 21.02.2008 г. № 149/54–144.

# МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

DOI 10.17223/20710410/17/8

УДК 004.94

## ПОСТРОЕНИЕ ИЕРАРХИЧЕСКОГО РОЛЕВОГО УПРАВЛЕНИЯ ДОСТУПОМ

Д. Н. Колегов

*Национальный исследовательский Томский государственный университет, г. Томск, Россия*

**E-mail:** d.n.kolegov@gmail.com

Предлагается подход к построению ролевого управления доступом для компьютерных систем с иерархией сущностей, отражающей установленные организационно-управленческие отношения. Формулируются определения базовых элементов и механизмов иерархической ролевой модели, развивающей семейство моделей ролевого управления доступом *RBAC* и, в отличие от них, позволяющей использовать уровни иерархии сущностей при задании и проверке разрешённых прав доступа субъектов к сущностям на основе модели полурешётки.

**Ключевые слова:** *модели безопасности, ролевое управление доступом, модель RBAC, иерархия сущностей.*

### Введение

Важнейшим механизмом обеспечения безопасности большинства существующих компьютерных систем (КС) является механизм логического управления доступом и информационными потоками, традиционно реализуемый в операционных системах (ОС), системах управления базами данных (СУБД) и автоматизированных системах управления (АСУ). Функционирование такого механизма КС заключается в проверке имеющихся у субъекта прав доступа к сущности и последующем разрешении или запрещении соответствующего доступа. Способ задания разрешённых прав доступа субъектов к сущностям и правил их проверки регламентируется реализуемой в КС политикой управления доступом и информационными потоками, являющейся составной частью политики безопасности КС. Традиционно в КС используются дискреционные, мандатные и ролевые виды политик управления доступом.

В настоящее время наиболее широкое развитие и распространение получил механизм ролевого управления доступом, применяемый в основном в ОС с высоким уровнем безопасности, а также системах защиты СУБД и АСУ.

Как правило, при построении и реализации ролевого механизма управления доступом используется стандартизованная Национальным институтом стандартов и технологий (NIST) США модель *RBAC* [1], разработанная на основе классических моделей ролевого управления доступом [2, 3]. Ролевое управление доступом является достаточно сложным механизмом обеспечения безопасности как с точки зрения научного обоснования свойств его безопасности, так и с учётом особенностей его практической реализации. В реальных КС, в которых одновременно могут работать сотни пользователей, структура ролей может быть очень сложной, а количество различных прав

доступа значительным, проблема реализации и администрирования системы ролевого управления доступом является чрезвычайно важной задачей [4]. Для учёта различных особенностей и условий функционирования современных КС предложено множество модификаций и расширений моделей семейства $RBAC$, как развивающих, так и дополняющих их основные положения элементами и механизмами других моделей [5 – 8].

В то же время в моделях ролевого управления доступом семейства $RBAC$, как и в других известных ролевых моделях и их расширениях, не учитываются следующие особенности функционирования современных КС, затрудняющие и делающие неэффективной реализацию ролевой политики управления доступом:

— иерархичность и распределённость компонент КС;
— наличие идентичной структуры распределённых компонент КС;
— необходимость определения одинаковых правил управления доступом для распределённых компонент КС;
— возможность использования уровней иерархии КС при задании разрешённых прав доступа субъектов к сущностям и правил их проверки;
— существование нескольких иерархий в КС и наличие различных правил управления доступом для одних тех же субъектов и сущностей в зависимости от уровня заданной иерархии;
— необходимость создания большого числа ролей для реализации политики управления доступом;
— необходимость гибкого и масштабируемого задания разрешённых прав доступа субъектов к сущностям при изменении их уровней в иерархии КС, а также при добавлении в КС новых ролей, прав доступа и сущностей.

С целью развития и адаптации моделей логического управления доступом и информационными потоками к свойствам и условиям функционирования реальных КС в работе описываются базовые элементы и механизмы модели иерархического ролевого управления доступом для компьютерных систем с иерархией сущностей, отражающей установленные организационно-управленческие отношения и позволяющей использовать уровни иерархии сущностей при задании и проверке разрешённых прав доступа субъектов к сущностям на основе модели полурешётки.

## 1. Оценка сложности применения моделей ролевого управления доступом семейства $RBAC$ для КС с иерархией сущностей

Пусть имеется организация с иерархической структурой своих подразделений, эксплуатирующая некоторую распределенную КС. В соответствии с установленными организационно-управленческими отношениями и заданной политикой безопасности КС пользователи должны иметь некоторые виды прав доступа к данным своего подразделения, определяемые их функциональными и должностными обязанностями. При этом сотрудники вышестоящего по организационно-штатной структуре подразделения должны иметь права доступа к данным нижестоящих (подчинённых) подразделений, а сотрудники нижестоящих подразделений не должны получать права доступа к данным вышестоящих подразделений. Таким образом, в соответствии с требованиями политики безопасности управления доступом предоставление прав доступа зависит от уровней сущностей в иерархии КС и полномочий пользователя.

Оценим сложность реализации ролевого управления доступом рассматриваемой КС для моделей $RBAC_0$ и $RBAC_1$. Для простоты изложения рассмотрим граничный случай, когда иерархическая структура организации представляется полным

бинарным деревом высоты $H$, а узлами дерева являются классы разбиения множества сущностей.

Пусть $N$ — количество должностей пользователей КС, тогда в рамках модели $RBAC_0$ число ролей, необходимое для реализации политики управления доступом, равно $M = N(2^{H+1} - 1)$. При добавлении нового подразделения в структуру КС необходимо будет добавить $N$ ролей. С учётом того, что в реальных КС число прав доступа и сущностей доступа, определяемых ролью, может исчисляться сотнями, применение модели $RBAC_0$ в таких КС является неэффективным и сложно реализуемым.

Использование модели $RBAC_1$ позволяет более эффективно реализовать политику управления доступом. В этом случае необходимо построить $N$ двоичных деревьев иерархии ролей, аналогичных дереву иерархии КС, с числом вершин $2^{H+1}-1$. Применение модели $RBAC_1$ позволяет более просто организовать назначение ролей и прав доступа пользователям, но требует построения и администрирования нескольких иерархий ролей.

Неэффективным оказывается применение моделей $RBAC$ для реализации ролевого управления доступом в КС с иерархией сущностей в следующих случаях:

— переподчинение подразделений;
— добавление новых сущностей во все подчинённые подразделения;
— добавление нового или удаление существующего права доступа к идентичным сущностям для всех подчинённых подразделений.

Таким образом, подход, основанный на применении моделей семейства $RBAC$ для реализации ролевой политики управления доступом в КС с иерархией сущностей, отражающей установленные организационно-управленческие отношения, не является оптимальным.

## 2. Определение политики иерархического ролевого управления доступом

Пусть имеется некоторая КС с заданной на ней древовидной иерархической структурой и для каждой сущности определён её уровень иерархии. В качестве одного из атрибутов, используемых при управления доступом, рассмотрим уровень иерархии сущности и на его основе определим политику управления доступом в КС. Основная идея такой политики заключается в назначении всем сущностям уровней иерархии в соответствии со структурой КС и предоставлении субъекту права доступа к сущности только в том случае, если уровень иерархии субъекта не меньше уровня иерархии сущности. При этом на множестве иерархии сущностей должна быть задана верхняя полурешётка (далее — полурешётка).

Таким образом, политика иерархического управления доступом — это политика, соответствующая следующим основным требованиям управления доступом в КС:

1) все сущности должны быть идентифицированы;
2) задана полурешётка уровней иерархии КС;
3) для каждой сущности определён уровень её иерархии в КС, задающий ограничения на доступ субъектов к сущностям;
4) субъект обладает правом доступа к сущности КС в том и только в том случае, если уровень иерархии субъекта не меньше уровня иерархии сущности.

Политика иерархического управления доступом, основываясь только на уровнях иерархии сущностей, обладает свойством избыточности разрешённых прав доступа субъектов к сущностям. Для выполнения принципа наименьших привилегий при доступе субъектов с высоким уровнем иерархии к сущностям с низким уровнем иерархии

положения введённой политики управления доступом добавляются к положениям политики ролевого управления доступом. При этом используется механизм задания типа сущности, понимаемого как атрибут сущности, используемый при реализации некоторого метода управления доступом в качестве аналога права доступа [10]. Механизм типов сущностей позволяет сформировать роли, в которых права доступа к сущностям группируются по их типу, что позволяет задавать права доступа субъектов ко всем сущностям одного типа, в том числе к сущностям с разными уровнями иерархии по отношении к субъекту, одновременно.

**Определение 1.** Политика иерархического ролевого управления доступом — это политика, соответствующая следующим основным требованиям управления доступом в КС:

1) все сущности должны быть идентифицированы;
2) задана полурешётка уровней иерархии КС и каждой сущности присвоен уровень иерархии;
3) определено множество типов сущностей и для каждой сущности указан её тип;
4) задано множество ролей, каждая из которых представляет собой некоторое множество прав доступа к сущностям определённого типа;
5) каждый субъект обладает некоторым множеством разрешённых для данного субъекта ролей;
6) субъект обладает правом доступа к сущности КС в том и только в том случае, если субъект обладает ролью, в множестве прав доступа которой имеется данное право доступа к сущности данного типа, и уровень иерархии субъекта не меньше уровня иерархии сущности.

## 3. Определение модели иерархического ролевого управления доступом

**Определение 2.** Модель, описывающую политику иерархического ролевого управления доступом, будем называть иерархической ролевой моделью управления доступом в КС и обозначать *RBAC-H*. Основными элементами данной модели являются:

— $E = O \cup C$ — множество сущностей, где $O$ — множество объектов, $C$ — множество контейнеров и $O \cap C = \varnothing$;
— $U$ — множество пользователей, при этом пользователи по определению не являются сущностями ($U \cap E = \varnothing$);
— $S \subseteq E$ — множество субъект-сессий пользователей;
— $T$ — множество типов сущностей;
— $L$ — множество уровней иерархии сущностей;
— $R_r$ — множество видов прав доступа;
— $R$ — множество ролей;
— $P \subseteq (R_r \times T) \cup (R_r \times E)$ — множество прав доступа ко всем сущностям одного типа и отдельным сущностям;
— $PA : R \to 2^P$ — функция прав доступа ролей, задающая для каждой роли множество прав доступа к сущностям, при этом для каждого права доступа $p \in P$ существует роль $r \in R$, такая, что выполняется условие $p \in PA(r)$;
— $UA : U \to 2^R$ — функция авторизованных ролей пользователей, задающая для каждого пользователя множество ролей, на которые он может быть авторизован;
— $type : E \to T$ — функция типов сущностей;
— $f_e : E \to L$ — функция, задающая уровень иерархии каждой сущности;

— $user : S \to U$ — функция принадлежности субъект-сессии пользователю, задающая для каждой субъект-сессии пользователя, от имени которого она активизирована;
— $roles : S \to 2^R$ — функция, задающая для пользователя множество ролей, на которые он авторизован в текущей сессии, при этом в каждом состоянии КС для каждой субъект-сессии $s \in S$ выполняется условие $roles(s) \subseteq UA(user(s))$.

**Определение 3.** Пусть $X$ — заданное разбиение множества $E$ в соответствии с заданной иерархией сущностей, при этом $|X| = |L|$ . Доменом $d$ сущностей множества $E$ будем называть всякий класс из $X$.

Иерархией доменов назовём заданное на множестве $X$ отношение частичного порядка $\leqslant$, удовлетворяющее следующим условиям:

— если для $d \in X$ существуют $d_1, d_2 \in X$, такие, что $d \leqslant d_1$, $d \leqslant d_2$, то $d_1 \leqslant d_2$ или $d_2 \leqslant d_1$;
— в $X$ существует наибольший элемент.

Описанная иерархия доменов соответствует КС с иерархической древовидной структурой, отражающей организационно-управленческие отношения, и задаёт верхнюю полурешётку $(X, \leqslant)$.

Пусть $L$ — заданное множество уровней иерархии сущностей и существует биективное отображение $X$ на $L$. Определим на множестве $L$ отношение частичного порядка $\leqslant$, где для любых $l_1, l_2 \in L$ верно: $l_1 \leqslant l_2$ тогда и только тогда, когда $d_1 \leqslant d_2$ для соответствующих $d_1, d_2 \in X$, тогда $(L, \leqslant)$ — верхняя полурешётка уровней иерархии сущностей.

Аналогично модели $RBAC$ будем предполагать, что множества $U$, $X$, $T$, $L$, $P$, $R$, $R_r$ и функции $UA$, $PA$, $type$ не изменяются с течением времени.

**Определение 4.** Пусть имеются множества $E$, $S$, $X$, $U$, $T$, $P$, $R$, $R_r$, функции $PA$, $UA$, $type$, $user$, $roles$ и $(L, \leqslant)$ — полурешётка уровней иерархии. Определим предикат $can\_access(s, e, p)$, истинный тогда и только тогда, когда выполняются следующие условия:

— $f_e(e) \leqslant f_e(s)$;
— $(p, type(e)) \in PA(roles(s))$.

**Определение 5.** Будем говорить, что в КС реализовано иерархическое ролевое управление доступом *RBAC-H*, если любая субъект-сессия $s \in S$ пользователя $user(s) \in U$ может обладать правом доступа $p \in R_r$ к сущности $e \in E$ тогда и только тогда, когда истинен предикат $can\_access(s, e, p)$.

Одним из важных механизмов семейства моделей ролевого управления доступом являются ограничения, накладываемые на множества ролей, на которые может быть авторизован пользователь или на которые он авторизуется в течение одной сессии. Основные ограничения, определённые в рамках моделей ролевого управления доступом семейства $RBAC$ [4], могут быть перенесены в предложенную модель. В то же время для большего соответствия иерархического управления доступом процедурам обработки данных, используемым в реальных КС, необходимо определить новые виды ограничений, существенные именно для данного вида управления доступом.

Определим функцию $h : E \to X$, такую, что $h(e) = d$, если $e \in d$, где $e \in E$ и $d \in X$.

**Определение 6.** Будем говорить, что в модели *RBAC-H* заданы ограничения на область доступа пользователя в иерархии сущностей, если выполняются следующие условия:

*Условие* 1. Определена функция $UD : U \to 2^X$.

*Условие* 2. Для любых $s \in S$, $e \in E$, $p \in R_r$ если истинен предикат $can\_access(s, e, p)$ и $user(s) = u$, то $h(e) \in UD(u)$.

**Определение 7.** Будем говорить, что в модели *RBAC-H* заданы ограничения на область авторизации ролей в иерархии сущностей, если выполняются следующие условия:

*Условие* 1. Определена функция $RD : R \to 2^X$.

*Условие* 2. Для любых $s \in S$, $e \in E$, $p \in R_r$ если истинен предикат $can\_access(s, e, p)$ и $r \in roles(s)$, то $h(e) \in RD(r)$.

**Определение 8.** Будем говорить, что в модели *RBAC-H* заданы ограничения на область авторизации ролей пользователя в иерархии сущностей, если выполняются следующие условия:

*Условие* 1. Определена функция $URD : U \times R \to 2^X$.

*Условие* 2. Для любых $s \in S$, $e \in E$, $p \in R_r$ если истинен предикат $can\_access(s, e, p)$, $user(s) = u$ и $r \in roles(s)$, то $h(e) \in URD(u, r)$.

**Определение 9.** Будем говорить, что в модели *RBAC-H* заданы ограничения на область применения прав доступа в иерархии сущностей, если выполняются следующие условия:

*Условие* 1. Определена функция $PD : P \to 2^X$.

*Условие* 2. Для любых $s \in S$, $e \in E$, $p \in R_r$ если истинен предикат $can\_access(s, e, p)$, то $h(e) \in PD(p)$ .

**Определение 10.** Будем говорить, что в модели *RBAC-H* заданы ограничения на область доступа к типу сущностей в иерархии сущностей, если выполняются следующие условия:

*Условие* 1. Определена функция $TD : T \to 2^X$.

*Условие* 2. Для любых $s \in S$, $e \in E$, $p \in R_r$ если истинен предикат $can\_access(s, e, p)$, то $h(e) \in TD(type(e))$.

Общая структура элементов модели иерархического ролевого управления доступом имеет вид, представленный на рис. 1.

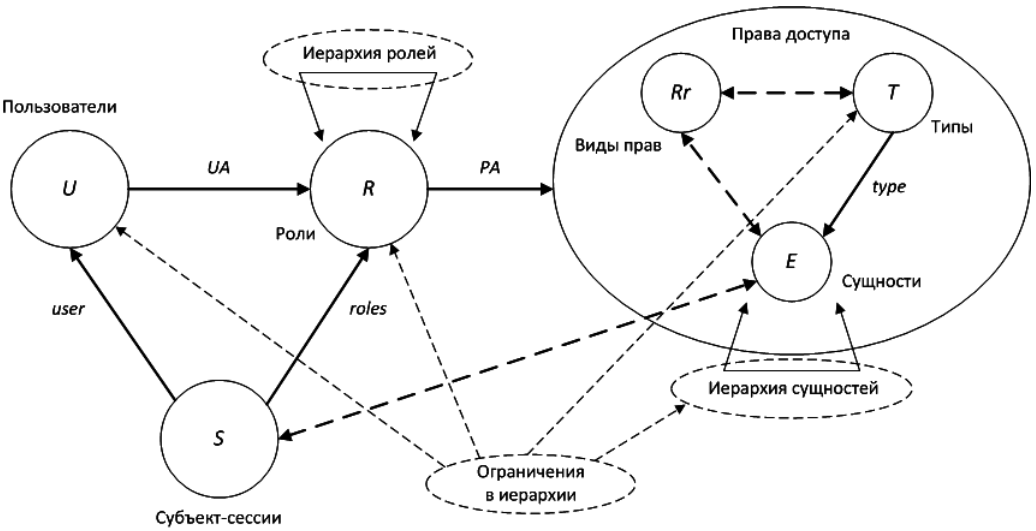


Рис. 1. Структура элементов иерархической ролевой модели *RBAC-H*

## Заключение

Предложено описание базовых элементов иерархической ролевой модели управления доступом *RBAC-H*, ориентированной на КС с иерархией сущностей, отражающей установленные организационно-управленческие отношения. Добавление атрибутов иерархии и типов сущностей к элементам моделей *RBAC* позволяет адаптировать последние к условиям функционирования реальных КС, а также существенно упростить реализацию и администрирование систем ролевого управления доступом.

## ЛИТЕРАТУРА

1. National Institute of Standards and Technology. Role Based Access Control (RBAC) and Role Based Security [Электронный ресурс]. Режим доступа: `http://csrc.nist.gov/groups/SNS/rbac`.
2. *Ferraiolo D. F. and Kuhn D. R.* Role Based Access Controls // Proc. 15th National Computer Security Conference, Baltimore, October 1992. P. 554–563.
3. *Sandhu R. S., Coyne E. J., Feinstein H. L., and Youman C. E.* Role-Based Access Control Models // IEEE Computer. 1996. No. 29(2). P. 38–47.
4. *Девянин П. Н.* Модели безопасности компьютерных систем. Управление доступом и информационными потоками. Учеб. пособие для вузов. М.: Горячая линия-Телеком, 2011. 320 с.
5. *Kuhn D. R., Coyne E. J., and Weil T. R.* Adding attributes to role-based access control // IEEE Computer. 2010. No. 43(6). P. 79–81.
6. *Sandhu R. S. and Al-Kahtani MA.* A Model for Attribute-Based User-Role Assignment // Proc. 18th Annual Computer Security Applications Conference (ACSAC'02), Las Vegas, December 09–13, 2002. P. 353.
7. *Joshi J., Bertino E. A, Latif U., and Ghafoor A.* A Generalized Temporal Role-Based Access Control Model // IEEE Trans. Knowledge and Data Engineering. 2005. No. 17(1). P. 4–23.
8. *Bertion E., Catania B., and Damiani M. L.* GEO-RBAC: A Spatially Aware RBAC // Proc. 10th ACM Symposium on Access Control Models and Technologies (SACMAT'05), Stockholm, Sweeden, June 2005. P. 29–37.
9. *Thomas R. K.* Team-based Access Control (TMAC): A Primitive for Applying Role-based Access Controls in Collaborative Environments // Proc. Second ACM Workshop on Role-based Access Control (RBAC'97), Fairfax, Virginia, USA, November 1997. P. 13–19.
10. *Девянин П. Н.* Формирование словаря терминов теории моделирования безопасности управления доступом и информационными потоками в компьютерных системах // Прикладная дискретная математика. 2011. № 2. С. 17–39.

# МАТЕМАТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ПРОГРАММИРОВАНИЯ

DOI 10.17223/20710410/17/9

УДК 681.3.06

## К ВОЗРОЖДЕНИЮ РУССКОГО ЯЗЫКА ПРОГРАММИРОВАНИЯ

Г. П. Агибалов

*Национальный исследовательский Томский государственный университет, г. Томск, Россия*

**E-mail:** agibalov@isc.tsu.ru

Сообщается о цели, задачах и первых результатах работ по возрождению русского языка программирования, известного в Отечестве как ЛЯПАС — Логический Язык для Представления Алгоритмов Синтеза. Возрождение ориентировано на создание доверенного системного и прикладного программного обеспечения автоматизированного синтеза безопасных компьютерных систем логического управления критически важными объектами (космическими системами, ядерными установками, вооружёнными силами и т. п.). Среди рассматриваемых задач важное место занимает и аппаратная реализация ЛЯПАСа как ЛЯПАС-машины, или компьютера с ЛЯПАСными операциями в качестве машинных команд, выполняемых непосредственно на схемном уровне.

**Ключевые слова:** *ЛЯПАС, доверенное программное обеспечение, безопасные компьютерные системы логического управления.*

### Введение

В условиях засилья компьютерных технологий наиболее серьёзные угрозы безопасности страны (как внутри неё, так и извне) проистекают от использования в компьютерных системах управления критически важными объектами (космическими системами, ядерными установками, вооружёнными силами и т. п.) недоверенного программного обеспечения, заимствованного у своего же потенциального противника. Оно, это программное обеспечение, как правило, несёт в себе недокументированные закладки, через которые возможна утечка одной информации и навязывание другой, в том числе разрушительной, и обнаружить которые, даже в свободном коде, часто бывает невозможно, в особенности, когда он «запутан» мощными средствами обфускации. То, как эффективно такие закладки могут сработать, очень красноречиво показали многие известные военные операции США в Югославии, Ираке, Ливии.

Для предотвращения возможности подобных угроз предлагается к использованию в компьютерных системах управления собственное программно-аппаратное обеспечение на базе русского языка программирования. Да, такой язык есть, он существует с начала 1960-х годов, с той поры, когда в западном мире ещё только вступали в обиход первые языки программирования — Фортран и Алгол. Русским его назвали американцы — Russian programming language, а его оригинальное название звучит как ЛЯПАС (с ударением на второй слог) — Логический Язык для Представления Алгоритмов Синтеза. Он создан в научной школе прикладной дискретной математи-

ки (ПДМ) Томского государственного университета (ТГУ) под руководством Аркадия Дмитриевича Закревского.

Можно ли построить безопасные компьютерные системы управления на базе другого, «не русского» языка программирования (Си, Ада и пр.)? Наверно, можно, если создать при этом ещё и доверенный компилятор с этого языка. Но зачем, когда есть свой язык программирования, не менее, но много более приспособленный для задач логического управления и криптографической защиты информации, реализация которого много дешевле разработки доверенного компилятора для Си или Ада. Ведь речь идёт не о бытовых проблемах населения, которые могут решаться, да и то только до некоторой степени, с помощью «общечеловеческих» компьютерных систем. Речь идёт о безопасности Отечества, которую нельзя обеспечить «оружием» противника.

Реализация предлагаемого проекта возрождения ЛЯПАСа в полном объёме

— вернёт стране престиж компьютерной державы, каковой она была в 50–60-е годы XX столетия (пусть даже, если это будет не столько в сознании внешнего мира, сколько в самосознании собственного народа);

— обезопасит страну от угроз со стороны «кибертеррористов» в лице как отдельных людей, так и отдельных государств — потенциальных противников.

Конкуренции этому проекту не видно, поскольку все озабочены собственной прибылью, но не безопасностью страны и её престижем.

## 1. Безопасные компьютерные системы логического управления

Автоматическое логическое управление современными техническими объектами (летательными аппаратами, ядерными реакторами, безлюдным производством, другими технологическими процессами) требует применения доверенных управляющих компьютерных систем (КС), защищённых от вмешательства со стороны кибертеррористов или злоумышленных хакеров средствами криптографической защиты и разграничения доступа в соответствии с некоторой политикой безопасности. Эти свойства доверенности и защищённости КС определяют её безопасность, а КС с этими свойствами называется безопасной КС. Алгоритмы управления и защиты в управляющих КС могут иметь программную и аппаратную реализацию. Соответственно этому управляющая КС в общем случае состоит из программных и аппаратных модулей, решающих в совокупности все задачи защиты и управления в реальном времени, отсчитываемом моментами поступления управляющих команд, и операционной системы (ОС), организующей в реальном времени вызовы программных модулей и обращения к аппаратным модулям в соответствии с протоколом управления.

Применительно к вычислительному модулю (программному или аппаратному) реальность времени означает, что время его работы (от момента подачи данных на вход до момента выдачи результата на выход) укладывается в промежуток между соседними командами в протоколе управления, а применительно к ОС — что её полная реакция на любой запрос, инициализируемый управляющей командой извне, включая работу всех задействованных ею модулей, должна завершиться до прихода следующего запроса. Это свойство модулей и ОС называется далее реактивностью, а модули и ОС с этим свойством — реактивными. Реактивная ОС в безопасной управляющей КС, в свою очередь, должна обладать свойством безопасности, не допускающим возможности запуска модулей, способных (по ошибке или злому умыслу) обойти защиту компьютерной системы, обеспечиваемую модулями защиты, или существенно понизить её уровень.

Существующие ныне операционные системы общего пользования в семействах Windows, Linux и др., допуская настройку на адекватное использование средств защиты КС на аппаратном уровне, тем не менее по большому счёту не могут быть использованы в безопасной реактивной КС, главным образом, по причине недоверенности, проистекающей из закрытости или недостаточной контролируемости программного кода, а также из-за недостаточной их производительности для управления задачами в реальном времени, на которое они, будучи универсальными, специально не «заточены».

Кроме того, в безопасных КС свойство безопасности должно быть присуще также и программным реализациям прикладных алгоритмов и, в первую очередь, тех, что решают задачи защиты. Оно состоит в отсутствии в программах уязвимостей, через которые возможно проникновение в КС злонамеренного программного кода.

## 2. Математические и программные средства проектирования управляющих КС

В настоящее время на рынке научно-технической продукции нет ни доверенных КС, обладающих указанными свойствами реактивности и защищённости, ни программного обеспечения для их построения. Это вместе с чрезвычайной важностью безопасных КС для управления современными техническими объектами (подчас потенциально опасными, как Чернобыльская АЭС или Саяно-Шушенская ГЭС, например) и огромной трудоёмкостью их создания доказывает актуальность проекта, нацеленного на разработку математических и программных средств, автоматизирующих процесс проектирования таких КС.

Математические и программные средства, подлежащие разработке, должны включать в себя: язык программирования, наиболее подходящий для написания компонент требуемой КС; эффективные средства отладки и трансляции в машинный код программ, написанных на этом языке; расширяемую библиотеку прикладных программ на нём, охватывающую весь спектр задач как логического управления (на комбинационном, конечно-автоматном, микропрограммном и других уровнях представления), так и защиты информации (шифрование, аутентификация, цифровая подпись, контроль целостности и др.) и состоящую из программ двух типов: одни реализуют алгоритмы управления и защиты, а другие — алгоритмы синтеза схем, реализующих алгоритмы управления и защиты; операционную систему для управления задачами в КС.

Язык программирования должен гарантировать безопасность написанных на нём программ первого типа, а его операционные средства должны обеспечивать максимально возможную эффективность машинных программ и в то же время быть удобными для пользователей. Он должен допускать компактное и пригодное к публикации выражение программ и их эффективную трансляцию в машинный язык.

Задачи логического управления и защиты информации являются дискретно-математическими. Они ставятся на целых числах и конечных множествах с отношениями и операциями, могут быть перечислительными, поисковыми, оптимизационными, иметь аналитическое и алгоритмическое решения, допускать аппарат теории чисел, общей алгебры, комбинаторного анализа, дискретных функций, математической логики, кодов, графов, автоматов и т. д. и т. п. Длина значений целочисленных переменных в них может достигать нескольких тысяч бит.

Для того чтобы алгоритмы решения этих задач были эффективными, язык программирования для них должен содержать по существу машинные операции над булевыми векторами и их компонентами и иметь средства для создания эффективных

макроопераций над длинными такими векторами, над большими целыми числами и над большими булевыми и целочисленными матрицами. Заметим, что этих средств языка достаточно и для написания операционной системы.

Библиотека прикладных программ должна удовлетворять следующим требованиям безопасности и реактивности:

1) каждая программа в ней, реализующая алгоритм управления или защиты, не должна иметь уязвимостей, нарушающих её безопасность, и должна быть снабжена индексом реактивности — оценкой времени её выполнения при различных значениях параметров, полученной в результате предварительных исследований в компьютерном эксперименте на представительной выборке примеров;

2) каждая программа, реализующая алгоритм синтеза схем управления, должна быть также снабжена индексом реактивности, который в данном случае является оценкой времени работы доставляемых программой схем, полученной аналогичным образом. Значение индекса реактивности библиотечной программы влияет на широту класса реактивных управляющих КС, в создании которых может быть использована данная программа: чем он меньше, тем шире класс. Поэтому из двух равнофункциональных программ предпочтение при внесении в библиотеку отдаётся программе с меньшим индексом реактивности.

Операционная система также должна удовлетворять требованиям безопасности и реактивности, а именно: она не должна допускать выполнения в КС вредоносных действий и должна поддерживать исполнение прикладных задач (управления и защиты) с соблюдением заданного ограничения на ресурс времени.

## 3. О ЛЯПАСе

Среди всех языков программирования, известных в настоящее время, только один в достаточной степени удовлетворяет всем сформулированным требованиям. Это — ЛЯПАС.

Первые две версии ЛЯПАСа были разработаны соответственно к 1964-му [1 – 3] и к 1970-му [4] годам в ТГУ, а третья и последняя версия, называемая ЛЯПАС-М, — к концу 1974 г. в Институте технической кибернетики (ИТК) АН БССР [5, 6]. В отличие от предыдущих, в ней учтены байтовая структура данных и возможность измерения времени в современных компьютерах, а также система символов на современных отечественных устройствах отображения информации.

В своё время язык ЛЯПАС был реализован на всех отечественных ЭВМ, начиная с одноадресной машины «Урал-1» и кончая БЭСМ-6, а также на ЭВМ семейств ЕС, СМ и VAX (всё это в ТГУ совместно с ИТК) и на персональных компьютерах (ПК) первых поколений (в ИТК); на нём написан ряд крупных систем автоматического синтеза дискретных управляющих систем для многочисленных предприятий МЭП и МРП СССР [7 – 9]; его изучали, реализовывали и применяли также за рубежом — в США [10 – 12], Польше [13], Югославии [14], Чехословакии, ГДР; странами-участницами СЭВ он был принят в качестве международного языка программирования. Позднее, на волне демократии, с ликвидацией военно-промышленного комплекса и производства отечественных ЭВМ интерес к ЛЯПАСу заметно поугас. Сейчас же, когда осознание необходимости для национальной безопасности собственного программного обеспечения компьютерных систем управления возвращается, стал возвращаться интерес и к ЛЯПАСу.

Более подробно с историей создания и развития ЛЯПАСа, с его особенностями и характеристикой систем программирования, созданных на его основе для различных

типов компьютеров, можно познакомиться по обзору [15]. Здесь мы отметим только те его стороны, которые важны в решении дискретно-математических задач логического управления и защиты информации в реальном времени.

Язык ЛЯПАС был задуман и создан для разработки эффективных алгоритмов решения задач прикладной дискретной математики. Его применение не ограничивается написанием и отладкой программ, но предполагает их исследование в эксперименте на компьютере с одновременной доработкой как алгоритма, так и программы, его реализующей, с целью достижения наибольшей эффективности последней, характеризуемой экспериментальной оценкой времени исполнения программы: чем она меньше, тем выше эффективность. Соответственно выбрана и система операций и операндов в нём — с расчётом, с одной стороны, на максимизацию скорости работы получаемых программ, а с другой — на удобство пользователей. Это нашло отражение в двухуровневой архитектуре языка, первый уровень которой максимально приближен к машинному языку, а второй, представляющий собой иерархию библиотечных функций, делает язык сколь угодно выразительным. Наличие в языке «промашинного» первого уровня позволяет создавать на нём эффективные операционные системы, в том числе, что для нас немаловажно, обладающие определёнными выше свойствами безопасности и реактивности. В ЛЯПАСе количество, имена, типы операндов и возможные операции над ними фиксированы, а их размеры задаются и контролируются. Таким образом, ЛЯПАС является одним из тех редких языков программирования, на которых можно писать как безопасные прикладные программы, так и безопасные реактивные ОС. Кроме того, благодаря его двухуровневой архитектуре, для постановки ЛЯПАСа на конкретный класс машин достаточно написать транслятор на машинный язык лишь с первого уровня языка, а для подключения библиотечных функций использовать компилятор (препроцессор), написанный на ЛЯПАСе однажды и независимо от класса машин. Близость первого уровня ЛЯПАСа к машинному языку делает разработку такого транслятора не слишком трудоёмкой, а сам транслятор высокоскоростным.

Есть много других свойств ЛЯПАСа, подтверждённых на практике, которые существенны для систем программирования алгоритмов логического управления, защиты информации и синтеза схем для них. Это и компактность программ на ЛЯПАСе, их обозримость и пригодность к публикации, и реальная возможность написания и контроля их самим разработчиком алгоритма, и их более высокая эффективность (по сравнению с программами на других языках), и т. п.

## 4. СПО на базе ЛЯПАСа для проектирования безопасных управляющих КС

Всё перечисленное выше делает ЛЯПАС едва ли не идеальным средством для разработки системного и прикладного программно-аппаратного обеспечения (СПО), ориентированного на создание безопасных управляющих КС. Существовавшее когда-то СПО на основе ЛЯПАСа-М для ПК функционировало под управлением MS-DOS. Ориентированное на задачи логического проектирования, оно не имело в своей библиотеке программ криптографической защиты информации и разграничения доступа и не отвечает современным требованиям безопасности и реального времени. Предполагается, используя опыт и идеи этой разработки, создать СПО на основе ЛЯПАСа с операционной системой и библиотекой прикладных программ, удовлетворяющими как условиям безопасности и реального времени, так и требованиям к составу библиотечных программ. В нём безопасную реактивную ОС и транслятор с ЛЯПАСа в машинный язык планируется разрабатывать «методом раскрутки», без использования других (проме-

жуточных) языков и преобразований, понижающих доверенность и эффективность системного ПО.

При наличии такого СПО создание конкретной управляющей КС сводится к выбору из него готовой операционной системы, возможно, с адаптацией её ядра под платформу этой КС, а из его библиотеки — нужных прикладных программ с индексом реактивности, не превышающим заданного ресурса времени. В отсутствие таковых может потребоваться пополнение библиотеки недостающими программами. Последнего, впрочем, не понадобится при достаточной развитости библиотеки, достигаемой со временем. Выбранные программы, реализующие алгоритмы управления и защиты, используются в КС непосредственно в качестве её программных модулей, а программы, реализующие алгоритмы синтеза схем, — для синтеза соответствующих аппаратных модулей КС.

Достоинство результата выражается в отличительной совокупности следующих существенных признаков ожидаемого СПО:

— на все 100 % отечественный продукт с открытым и легко читаемым кодом;
— язык программирования, гарантирующий написание программ без уязвимостей;
— безопасный компилятор, исключающий возможность создания программ с уязвимостями;
— безопасная операционная система, исключающая возможность запуска злонамеренного кода;
— библиотека прикладных программ с реальным временем исполнения;
— операционная система, гарантирующая исполнение программ в реальном времени;
— гарантированное отсутствие недокументированных закладок в КС, создаваемых с его помощью.

С его реализацией обеспечиваются:

— улучшение потребительских свойств производимой продукции — управляющих КС;
— повышение производительности труда разработчиков продукции;
— уменьшение стоимости выпускаемой продукции;
— защищённость производимых КС от злоумышленника и ошибок персонала;
— инновационность производства, выход на рынок высокоинтеллектуальной и наукоёмкой продукции.

## 5. ЛЯПАС-машина

Когда-то А. Д. Закревский мечтал о создании L-машины [16] — компьютера для решения логических задач синтеза дискретных автоматов. В настоящее время, когда высокоскоростные логические микросхемы стали общедоступными, эта мечта может стать реальностью путём аппаратной реализации ЛЯПАСа с помощью существующих методов и систем автоматизированного проектирования [17 – 19]. Дело в том, что базовые операции в ЛЯПАСе являются одноадресными, что позволяет создать реально («в железе») ЛЯПАС-машину — недорогой одноадресный компьютер с системой команд, являющихся операциями ЛЯПАСа, в котором (компьютере) программы на языке программирования являются одновременно и исполняемым кодом — исполняются компьютером непосредственно без промежуточной трансляции в машинный язык и без интерпретации операций в них машинными подпрограммами. Это будет компьютер, в котором язык программирования высокого уровня (в данном случае ЛЯПАС) реализован аппаратно, благодаря чему скорость исполнения программ в нём будет на несколько порядков выше скорости их исполнения в существующих компьютерах.

Столь же простая аппаратная реализация других языков программирования вряд ли возможна ввиду их принципиально иного устройства.

Вместе с хранилищем ЛЯПАСных программ ЛЯПАС-машина может стать простейшей и вместе с тем эффективнейшей из мыслимых универсальной безопасной компьютерной системой управления любыми сложными объектами. Для её применения в управлении конкретным объектом достаточно будет загрузить в это хранилище программу на ЛЯПАСе, реализующую алгоритм управления данным объектом.

## 6. Первоочередные задачи возрождения ЛЯПАСа

Возрождение ЛЯПАСа с целью создания на его базе высокоэффективных безопасных компьютерных систем логического управления предполагает решение ряда теоретических и практических задач, направленных в первую очередь на разработку и исследование

- средств трансляции и отладки программ на ЛЯПАСе для их исполнения на современных компьютерах;
- алгоритмов и программ на ЛЯПАСе для логического управления;
- алгоритмов и программ на ЛЯПАСе для логического синтеза управляющих автоматов;
- алгоритмов и программ на ЛЯПАСе для криптографической защиты управляющей информации;
- реактивной ОС на базе ЛЯПАСа для управления в реальном времени;
- архитектуры ЛЯПАС-машины

и на реализацию последней на базе ПЛИС и (или) заказных интегральных схем.

С советских времён в научной школе ПДМ ТГУ сохранилась богатая библиотека программ на ЛЯПАСе для решения задач логического синтеза, которая с успехом может быть использована в синтезе аппаратных модулей будущих безопасных КС логического управления и ЛЯПАС-машины. Реанимация и адаптация этих программ под новые требования и условия также предполагаются.

## 7. Текущее состояние дела

Работы по возрождению ЛЯПАСа ведёт кафедра защиты информации и криптографии (ЗИиК) ТГУ. На начало 2012 г. «оптимизирована» под современные возможности отображения символика ЛЯПАСа и разработаны и запущены в опытную эксплуатацию программные средства, оживившие ЛЯПАС, а именно: интерпретатор ЛЯПАСа, компиляторы с ЛЯПАСа в языки Си и Ассемблер и отладчик ЛЯПАСных программ. С их помощью уже возможна отладка, исполнение, экспериментальное исследование алгоритмов на ЛЯПАСе и их доработка по результатам исследования. В ТГУ язык ЛЯПАС включён в образовательную программу по специальности 090301 — Компьютерная безопасность. Студенты кафедры ЗИиК, обучающиеся по этой специальности, не только осваивают программирование на ЛЯПАСе, но и участвуют в решении перечисленных выше задач по его полноценному возрождению, создавая в рамках курсовых и дипломных работ отдельные средства СПО на базе ЛЯПАСа для проектирования безопасных управляющих КС и для синтеза ЛЯПАС-машины.

## ЛИТЕРАТУРА

1. *Закревский А. Д.* Алгоритмический язык ЛЯПАС и автоматизация синтеза дискретных автоматов. Томск: Изд-во Том. ун-та, 1966. 266 с.

2. Логический язык для представления алгоритмов синтеза релейных устройств / под ред. М. А. Гаврилова. М.: Наука, 1966. 342 с.
3. Труды Сибирского физико-технического института. Вып. 48. Автоматизация синтеза дискретных автоматов. Томск: Изд-во Том. ун-та, 1966.
4. *Закревский А. Д.* Алгоритмы синтеза дискретных автоматов. М.: Наука, 1971. 512 с.
5. *Закревский А. Д.* Язык программирования ЛЯПАС-М // Вычислительная техника в машиностроении. Минск: Ин-т техн. кибернетики АН БССР, 1974. С. 99–111.
6. *Закревский А. Д., Торопов Н. Р.* Система программирования ЛЯПАС-М. Минск: Наука и техника, 1978. 240 с.
7. Синтез асинхронных автоматов на ЭВМ / под ред. А. Д. Закревского. Минск: Наука и техника, 1975. 184 с.
8. Автоматизация проектирования цифровых устройств / под ред. С. С. Бадулина. М.: Радио и связь, 1981. 238 с.
9. *Панкратова И. А., Быкова С. В., Николаева С. В., Оранов А. М.* Система автоматического синтеза комбинационных схем СИНТЕЗ-Ф // Управляющие системы и машины. 1991. № 1. С. 3–9.
10. LYaPAS: A programming language for logic and coding algorithms / ed. M. A. Gavrilov and A. D. Zakrevskii. New York; London: Academic Press, 1969. 475 p.
11. *Charles J. and Albright Jr.* An interpreter for the language LYaPAS. University of North Carolina at Chapel Hill: Department of Computer Science, 1974. 127 p.
12. *Nadler N.* User Group for Russian Programming Language // IEEE, Newsletter for Computer-Aided Design. 1971. Iss. 3.
13. *Michalski A. and Wiewiorowski T.* Odra Ljapas. Warszawa: Computation-Centre Polish Academy of Sciences, 1970. 33 p.
14. *Tratnik I.* Seminar «Analiza in primerjava jezikov za podro'je digitalne tehnike». Lljubljani: Univerzav, 1979. 63 c.
15. *Торопов Н. Р.* Язык программирования ЛЯПАС // Прикладная дискретная математика. 2009. № 2(4). С. 9–25.
16. *Закревский А. Д.* Машина для решения логических задач типа синтеза релейных схем // Синтез релейных устройств. Труды Междунар. симп. по теории релейных устройств и конечных автоматов. М.: Наука, 1965. С. 346–356.
17. *Соловьёв В. В.* Проектирование функциональных узлов цифровых систем на программируемых логических устройствах. Минск: ПК ООО Бестпринт, 1996. 252 с.
18. *Зотов В. Ю.* Проектирование цифровых устройств на основе ПЛИС фирмы XILINX в САПР WtbPACK ISE. М.: Горячая линия-Телеком, 2003.
19. *Wollinger T., Guajardo J., and Paar C.* Cryptography on FPGAs. State of the art implementations and attacks // ACM Trans. on Embedded Computing Systems. 2004. V. 3. No. 3. P. 534–574.

# ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ В ДИСКРЕТНОЙ МАТЕМАТИКЕ

DOI 10.17223/20710410/17/10

УДК 681.324, 519.17

## ИССЛЕДОВАНИЕ ВЛИЯНИЯ СТЕПЕНИ СПЕЦИАЛИЗАЦИИ ШАБЛОНА НА ПРОСТРАНСТВО ПОИСКА ПРИ ЭВОЛЮЦИОННОМ СИНТЕЗЕ МОДЕЛЕЙ[1]

О. Г. Монахов

*Институт вычислительной математики и математической геофизики СО РАН, г. Новосибирск, Россия*

**E-mail:** monakhov@rav.sscc.ru

Описан алгоритм эволюционного синтеза, объединяющий преимущества генетических алгоритмов и генетического программирования, основанный на эволюционных вычислениях, шаблонах (темплейтах, скелетонах) алгоритмов и заданном множестве пар входных — выходных данных. Исследовано влияние степени специализации шаблона на пространство поиска при эволюционном синтезе, получены оценки величины сокращения пространства поиска при введении в шаблон дополнительной информации в виде формул, уточняющих модель и имеющих представление в виде бинарного дерева. Теоретически и экспериментально показано для данного случая экспоненциальное и сверхэкспоненциальное сокращение времени работы алгоритма синтеза при увеличении степени специализации шаблона.

**Ключевые слова:** *генетический алгоритм, генетическое программирование, эволюционный синтез, шаблон, темплейт, скелетон.*

### Введение

В работе рассматривается проблема синтеза алгоритма (модели) $A$ как задача поиска параметров и функций данного шаблона $T$ (темплейта) алгоритма с целью оптимизации заданной функции $F$, характеризующей качество алгоритма $A$. Шаблон (template, skeleton [1 – 3], design pattern [4]) представляет собой управляющую параметризованную структуру алгоритма, которая описывает порядок сканирования структур данных алгоритма и определяет динамику вычислительного процесса в пространственно-временных координатах.

Пусть шаблон $T$ алгоритма $A$ содержит параметры $P = \{p_k : k \geqslant 0\}$, описывающие значения целочисленных и действительных коэффициентов и переменных, значения индексов, параметры структур данных, константы и некоторые примитивные операции алгоритма (величины инкрементов и декрементов, знаки переменных, логические операции и отношения, типы округления переменных). Пусть шаблон $T$ содержит также множество функций (формул) $FM = \{f_n : n \geqslant 0\}$ алгоритма $A$. При задании значений параметров $P$ и определении функций $FM$ в данном шаблоне $T$ получаем (синтезируем) алгоритм $A(T, P, FM)$.

[1]Работа поддержана грантом РФФИ № 08-01-00857.

Если заданы наборы значений входных параметров алгоритма $X_i$, $1 \leqslant i \leqslant N$, и выходных параметров $Y_i$, которые необходимо получить в результате исполнения алгоритма, то качество работы алгоритма характеризует целевая функция $F$, оценивающая величину рассогласования между требуемыми $Y_i$ и полученными выходными значениями алгоритма при данных входных значениях $Y_i' = A(T, P, FM, X_i)$. Кроме того, функция $F$ может оценивать не только точность, но и сложность получаемого алгоритма (например, время выполнения, число итераций, сложность (длину) формул [5]).

Таким образом, проблема синтеза алгоритма состоит в следующем: для данного шаблона $T$ и заданных наборов значений входных — выходных параметров $\{X_i, Y_i : 1 \leqslant i \leqslant N\}$ необходимо найти такие значения параметров $P^*$ и функций $FM^*$ шаблона $T$, определяющие алгоритм $A^*(T, P^*, FM^*)$, что

$$F(A^*(T, P^*, FM^*, X_i)) \leqslant F(A(T, P, FM, X_i))$$

для всех $1 \leqslant i \leqslant N$ при любых других значениях параметров $P \in \mathrm{Dom}(P)$ и функций $FM \in \mathrm{Dom}(FM)$.

Для решения проблемы автоматизированного синтеза (открытия) алгоритмов используется эволюционный алгоритм, основанный на шаблонах [5 – 8], оптимизирующий заданный критерий качества $F$. Этот гибридный алгоритм синтеза использует структуры данных и операторы генетических алгоритмов (ГА) [9] и генетического программирования (ГП) [10] и позволяет получить новые свойства (меньшее время исполнения и синтез алгоритмов с большей циклической сложностью и наличием рекурсивных функций, по сравнению с ГП, и синтез новых функций и предикатов, что достаточно трудно при традиционном ГА).

Эти свойства подхода с использованием шаблонов основаны на предварительном знании прикладной области и выборе обобщённой вычислительной схемы алгоритма, задаваемой в виде шаблона. Традиционное генетическое программирование страдает от слабо ограниченного слепого поиска в огромных пространствах параметров для реальных прикладных проблем и имеет большие временные затраты. В данной работе исследуется использование шаблонов алгоритмов для ограничения и сокращения пространства поиска к допустимым моделям и структурам решений. Этот подход представляет гибкий и прямой путь включения знаний эксперта относительно параметров, функций и структуры решаемой проблемы в разрабатываемую вычислительную модель алгоритма. В предельных случаях, с одной стороны, если мы не знаем шаблон искомого алгоритма, данный подход даёт традиционное генетическое программирование. С другой стороны, если мы знаем полную структуру алгоритма и не знаем только некоторые его параметры, данный подход даёт традиционный генетический алгоритм для поиска оптимальных параметров. Примером такой параметрической оптимизации на основе заданного шаблона с помощью генетического алгоритма служит алгоритм, предназначенный для открытия аналитических описаний новых плотных семейств оптимальных регулярных сетей [6, 7].

В п. 1 представлено краткое описание алгоритма эволюционного синтеза, в п. 2 получены оценки влияния степени специализации шаблона на пространство поиска при эволюционном синтезе.

## 1. Алгоритм эволюционного синтеза

Алгоритм эволюционного синтеза основан на эволюционных вычислениях и моделировании процесса естественного отбора популяции особей, каждая из которых

представлена точкой в пространстве решений задачи оптимизации. Особи представлены структурами данных $Gen$ — хромосомами, включающими недоопредёленные параметры $p_k$ и функции (формулы) $f_n$ шаблона $T$: $Gen = \{P, FM\} = \{p_1, p_2, \ldots, p_k; f_1, f_2, \ldots, f_n\}$, $k, n \geqslant 0$. Эти параметры и функции определяют необходимый алгоритм $A(T, Gen)$ на основе заданного шаблона $T$ и хромосомы $Gen$. Каждая популяция является множеством структур данных $Gen$ и определяет множество алгоритмов $A(T, Gen)$, образующихся из данного шаблона $T$.

Для представления и реализации хромосомы предложен подход, основанный на комбинации линейной структуры хромосомы для параметров $p_k$ (как в ГА [9]) и древовидной структуры хромосомы для каждой из функций (формул) $f_n$ (как в ГП [10]). Линейная структура хромосомы $Gen$ используется для представления таких параметров $p_k$ шаблона $T$, как значения целочисленных и действительных коэффициентов и переменных, значения индексов, величины инкрементов и декрементов, а также знаков переменных, логических операций и отношений, типов округления переменных.

Древовидные структуры в хромосоме $Gen$ используются для представления каждой из функций (формул) $f_n$. Переменные и константы формулы $f_n$ образуют листья (терминальные вершины) дерева — $TS$, а операции и примитивные функции, используемые в формуле $f_n$, соответствуют остальным (нетерминальным) вершинам дерева — $NS$. Каждая операция (примитивная функция) и её операнды (аргументы примитивной функции) представлены соответственно вершиной и её сыновьями в дереве.

Пусть дана формула $f_n = (5x + a)(x - 2)^2$, тогда имеем следующие множества терминальных $TS = \{x, a, 2, 5\}$ и нетерминальных $NS = \{+, -, \cdot, ^2, \}$ вершин. Пример соответствующего дерева представлен на рис. 1.

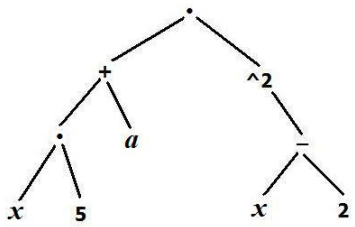


Рис. 1. Представление формулы в виде дерева

Для заданного шаблона $T$ при создании хромосом $Gen$ порождаются аналитические выражения для функций $f_n$ и задаются значения параметров $p_k$, по которым можно производить оценивание и модификации алгоритма $A(T, Gen)$.

Примем, что целевая функция (fitness function, функция качества, функция пригодности) $F$ вычисляет сумму квадратов отклонений выходных данных алгоритма $Y_i' = A(T, Gen, X_i)$ от заданных эталонных значений $Y_i$ для определённых наборов множества входных данных $X_i$, $1 \leqslant i \leqslant N$:

$$F = \sum_{i=1}^{N} (A(T, Gen, X_i) - Y_i)^2.$$

Целью алгоритма является поиск минимума $F$. На практике если получается несколько решений с одинаковым значением целевой функции, то выбирается решение с меньшей оценкой структурной сложности, т. е. с меньшей общей длиной формул решения.

Таким образом, для фиксированных значений функций $f_n$ и параметров $p_k$ можно вычислять выходные значения $Y_i'$ алгоритмов $A(T, Gen, X_i)$, порождённых на основе

заданных шаблонов $T$ и полученных в ходе эволюции хромосом $Gen$, для требуемых наборов множества входных данных $X_i$, $1 \leqslant i \leqslant N$. После выполнения алгоритмов $A$ получаем значения целевой функции $F$ и выбираем лучшие алгоритмы в популяции.

Основная идея алгоритма синтеза состоит в эволюционном преобразовании множества хромосом (параметров и формул шаблона) с помощью генетических операций селекции (отбора), мутации, кроссовера и добавления новых элементов в процессе естественного отбора с целью выживания «сильнейшего», т. е. имеющего наименьшее значение целевой функции, алгоритма.

Оператор *мутации* применяется к особям, случайно выбранным из текущей популяции с вероятностью $p_m \in [0, 1]$. Мутация линейной части хромосомы $Gen$ состоит в изменении значения случайно выбранного параметра $p_k$ на другую случайно выбранную величину из множества допустимых значений. Мутация древовидной части хромосомы $Gen$ состоит в изменении значения случайно выбранной вершины в представлении функции $f_n$ на другую случайно выбранную величину из множества допустимых значений.

Оператор *кроссовера* (скрещивания) применяется к двум особям (родителям), случайно выбранным из текущей популяции с вероятностью $p_c \in [0, 1]$. Кроссовер состоит в порождении двух новых особей путём обмена частями хромосом родителей (подчастями линейных структур и поддеревьями для древовидных структур хромосомы).

Оператор создания *нового элемента* (особи) состоит в генерации случайных значений параметров $p_k$ и функций $f_n$, описывающих алгоритмы. Это позволяет увеличить степень разнообразия особей при создании популяции.

Оператор *селекции* (отбора) реализует принцип выживания наиболее приспособленных особей. Он выбирает наилучших особей с минимальными значениями целевой функции.

Для поиска оптимума заданной целевой функции $F$ итерационный процесс вычислений в алгоритме эволюционного синтеза организован следующим образом.

П е р в а я  и т е р а ц и я (инициализация): порождение начальной популяции. Все особи популяции создаются с помощью оператора *новый элемент* с проверкой и отсеиванием всех непригодных особей. После заполнения массива популяции лучшие особи отбираются и запоминаются в массиве *best*.

П р о м е ж у т о ч н а я  и т е р а ц и я (генерация новой популяции): шаг от текущей к следующей популяции. Основной шаг алгоритма состоит в создании нового поколения особей на основе массива *best* и текущей популяции с использованием операций селекции, мутации, кроссовера и добавления новых элементов.

После оценки целевой функции для каждой особи в поколении проводится сравнение значений этих функций со значениями целевых функций тех особей, которые сохранены в массиве *best*. В том случае, если элемент из нового поколения лучше, чем элемент $best[i]$, для некоторого $i$, помещаем новый элемент на место $i$ в массив *best* и сдвигаем в нём все остальные элементы на единицу вниз. Таким образом, лучшие элементы локализуются в массиве *best* на местах с меньшими значениями индекса.

П о с л е д н я я  и т е р а ц и я (критерий остановки): итерации заканчиваются либо после исполнения заданного числа шагов, либо после нахождения оптимального алгоритма $A(T, Gen)$ (с заданным значением целевой функции $F$). После выполнения данного количества шагов алгоритма синтеза получаем множество (популяцию) алгоритмов $A(T, Gen)$, содержащее в элементе $best[0]$ алгоритм $A^*(T, Gen)$, имеющий минимальное значение целевой функции $F$.

## 2. Оценки влияния степени специализации шаблона на пространство поиска

Алгоритм эволюционного синтеза на основе шаблонов был успешно применён для автоматизированного получения следующих алгоритмов: вычисления степени и факториала натурального числа, определения минимального (максимального) элемента массива, определения суммы квадратов элементов массива, вычисления скалярного произведения векторов, определения формул последовательностей Фибоначчи и Трибоначчи, вычисления суммы двух матриц, сортировки методами пузырька, слияния и Шелла, вычисления корней уравнений, определения минимального покрывающего дерева и кратчайших путей в графе, параллельных алгоритмов балансировки нагрузки в многопроцессорных системах. Данный подход был также применён для открытия аналитических описаний новых плотных семейств оптимальных регулярных сетей [6, 7, 11] и синтеза нового алгоритма определения функции расстояния для циркулянтных графов степени 4.

Эволюционный синтез алгоритмов на основе шаблонов реализован в системе TES (Template-based Evolutionary Synthesis) [8] на языке программирования C, шаблоны также задаются на этом языке.

При реализации алгоритма эволюционного синтеза на основе шаблонов основную трудность представляет поиск неопределённых функций с помощью генетического программирования. Пространство поиска в этом случае представлено множеством деревьев, в котором переменные и константы образуют листья дерева, а операции и примитивные функции — внутренние (нетерминальные) вершины дерева. Обозначим через $S$ число терминальных символов, $M$ — число примитивных функций.

Для оценки влияния степени специализации шаблона на сокращение пространства поиска при эволюционном синтезе с введением в шаблон дополнительной информации в виде формул, уточняющих модель, рассмотрим два простых, но достаточно общих случая: 1) когда в шаблоне известно полное поддерево высотой $h$ и 2) когда в шаблоне известно поддерево с $t$ вершинами. Примем, что множество примитивных функций составляют функции с двумя переменными (операндами), в этом случае достаточно рассматривать бинарные деревья. При поиске полного бинарного дерева высотой $h$ в эволюционном алгоритме перебираются деревья высотой не более $h$. Обозначим $TN(h)$ число различных бинарных деревьев высотой не более $h$. Деревья, состоящие из одной терминальной вершины (корня), имеют высоту 0, число таких деревьев $TN(0) = S$. Число различных деревьев высотой не более $h > 0$ получается из комбинации $M$ корневых вершин (по числу примитивных функций), двух различных поддеревьев высотой не более $h-1$ и дополнительно $S$ деревьев высотой 0. Отсюда число деревьев высотой не более $h > 0$ равно $TN(h) = M \cdot TN^2(h-1) + S$. При $M = 1$ и $S = 1$ получаем известную формулу для числа различных бинарных деревьев высотой не более $h$: $TN(h) = TN^2(h-1) + 1$ (см. [12, 13], последовательность A003095 (M1544)).

Пример числа деревьев высотой не более $h$ для двух различных наборов значений $M$ и $S$ приведён в табл. 1. Как видно из таблицы, с ростом высоты деревьев $h$ пространство поиска увеличивается значительно (сверхэкспоненциально, с порядком роста $O(C^{2^h})$, где $C$ — константа, зависящая от $M$ и $S$, см. рис. 2), поэтому уменьшение путём использования более специализированного шаблона даже на 1 высоты деревьев, представляющих неизвестные функции, приводит к сверхэкспоненциальному сокращению пространства поиска при эволюционном синтезе от нескольких раз до нескольких порядков.

Таблица 1

$TN(h)$ — **число деревьев высотой не более** $h$

| $h$ | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|
| $M = 1,\ S = 1$ | 1 | 2 | 5 | 26 | 677 | 458330 |
| $M = 4,\ S = 2$ | 2 | 18 | 1298 | $6{,}74 \cdot 10^6$ | $1{,}82 \cdot 10^{14}$ | $1{,}32 \cdot 10^{29}$ |

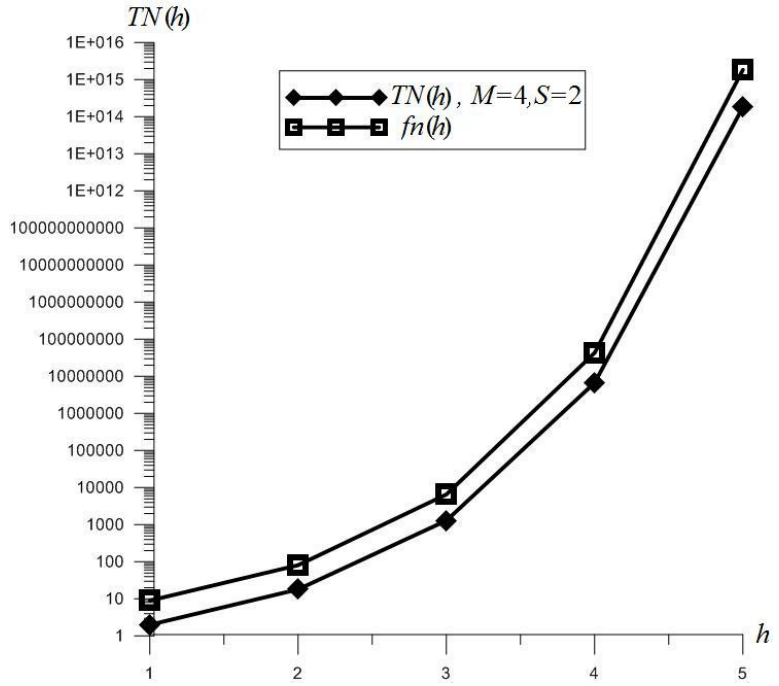


Рис. 2. Графики функций $TN(h)$ (при $M = 4$, $S = 2$) и $fn(h) = 3^{2^h}$

Для примера рассмотрим влияние степени специализации шаблона с различной высотой поддеревьев при поиске полиномиальной функции $F(x) = x^8 + x^4 + x^2 - (x+2)$, представленной на рис. 3.

Число итераций и размер популяции выбраны экспериментально на основе параметров из [9, 10]. Значения основных параметров в экспериментах приведены в табл. 2.

Таблица 2

**Значения параметров алгоритма**

| Параметр | Значение |
|---|---|
| Размер популяции, особей | 100 |
| Число итераций, шт. | 30000 |
| Вероятность кроссовера, % | 70 |
| Вероятность мутации, % | 15 |

Пусть заданы 8 пар вход — выход для $x = 0, 1, \ldots, 7$ и шаблон $T(x) = T(t_i,\ x^4 + x^2 - (x + 2))$, где известный терм $x^4 + x^2 - (x + 2)$ и известное поддерево шаблона $t_i \in \{t_1 = x^2, t_2 = x^4, t_3 = x^8\}$ будут последовательно снижать высоту $h = 3, 2, 1$ неизвестного поддерева функции $F(x)$. Множество терминальных символов в этом случае $TS = \{x, Cr\}$, $S = 2$, где $Cr$ — элемент множества случайных натуральных чисел. Набор операций, используемый для синтеза формул, в шаблонах следующий:

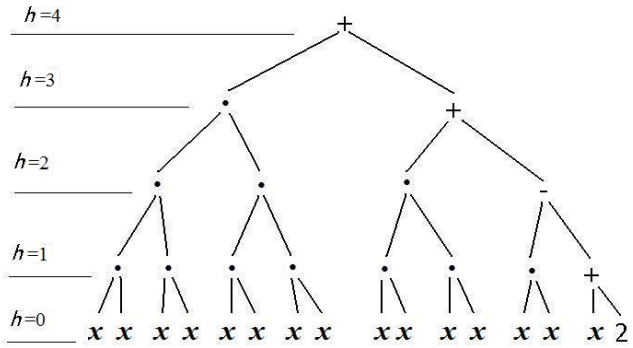


Рис. 3. Представление функции $F(x) = x^8 + x^4 + x^2 - (x + 2)$ в виде дерева

$NS = \{+, -, \cdot, /\}$, $M = 4$. В табл. 3 приведены результаты сравнения характеристик алгоритма эволюционного синтеза — времени поиска решения и числа поколений $G$ — до получения решения (функции $F(x)$) при различных шаблонах (для каждого шаблона приведены средние значения, полученные по результатам 100 экспериментов).

Таблица 3

**Характеристики алгоритма**

| Шаблон $t_i$ | $h$ | Время, с | $G$ |
|---|---|---|---|
| $t_1 = x^2$ | 3 | 43,03 | 19010,39 |
| $t_2 = x^4$ | 2 | 0,076 | 38,53 |
| $t_3 = x^8$ | 1 | 0,0036 | 1,5 |

Из табл. 3 видно, что с уменьшением высоты $h$ неизвестных поддеревьев шаблона пространство поиска значительно сокращается и характеристики алгоритма эволюционного синтеза уменьшаются от нескольких раз до нескольких порядков (сверхэкспоненциально, с оценкой порядка $\mathrm{O}(C^{2^h})$; так, $G(h) \approx 3{,}42^{2^h}$ при $M = 4$, $S = 2$).

Рассмотрим влияние степени специализации шаблона на сокращение пространства поиска при эволюционном синтезе во втором случае — когда в шаблоне известно поддерево с $t$ внутренними вершинами. Обозначим $TN(t)$ число различных бинарных деревьев с $t$ внутренними вершинами. Число таких деревьев, состоящих из одной терминальной вершины (корня), $TN(0) = S$. Число различных деревьев с $t > 0$ внутренними вершинами получается из комбинации $M$ корневых вершин (по числу примитивных функций) и всех комбинаций двух различных поддеревьев, имеющих в сумме $t - 1$ внутренних вершин. Отсюда число различных бинарных деревьев с $t$ внутренними вершинами $TN(t) = \sum\limits_{j=0}^{t-1} M \cdot TN(t-1-j)TN(j)$. При $M = 1$ и $S = 1$ получаем известную формулу для чисел Каталана: $C(t) = \sum\limits_{j=0}^{t-1} C(t-1-j)C(j)$, $C(0) = 1$, $t > 0$. Числа Каталана имеют множество комбинаторных интерпретаций (см. [12, 13], последовательность A000108 (M1459)), одна из которых — число различных бинарных деревьев с $t$ внутренними вершинами [14].

Пример числа деревьев с $t$ внутренними вершинами для двух различных наборов значений $M$ и $S$ приведён в табл. 4. Как видно из таблицы, с ростом числа внутренних

вершин деревьев пространство поиска увеличивается экспоненциально (с порядком роста $O(C^t)$, где $C$ — константа, зависящая от $M$ и $S$, см. рис. 4 для функций $TN(t)$ при $M = 4$, $S = 2$, $f1(t) = 0{,}7e^{1{,}7t}$ и $f2(t) = 6^t$), поэтому уменьшение путём использования более специализированного шаблона числа вершин деревьев, представляющих неизвестные функции, приводит к экспоненциальному сокращению пространства поиска при эволюционном синтезе.

Таблица 4

**$TN(t)$ — число деревьев с $t$ внутренними вершинами**

| $t$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $M = 1$, $S = 1$ | 1 | 1 | 2 | 5 | 14 | 42 | 132 | 429 | 1430 | 4862 | 16796 |
| $M = 4$, $S = 2$ | 2 | 4 | 16 | 80 | 448 | 2688 | 16896 | 109824 | 732160 | 4978688 | 34398208 |

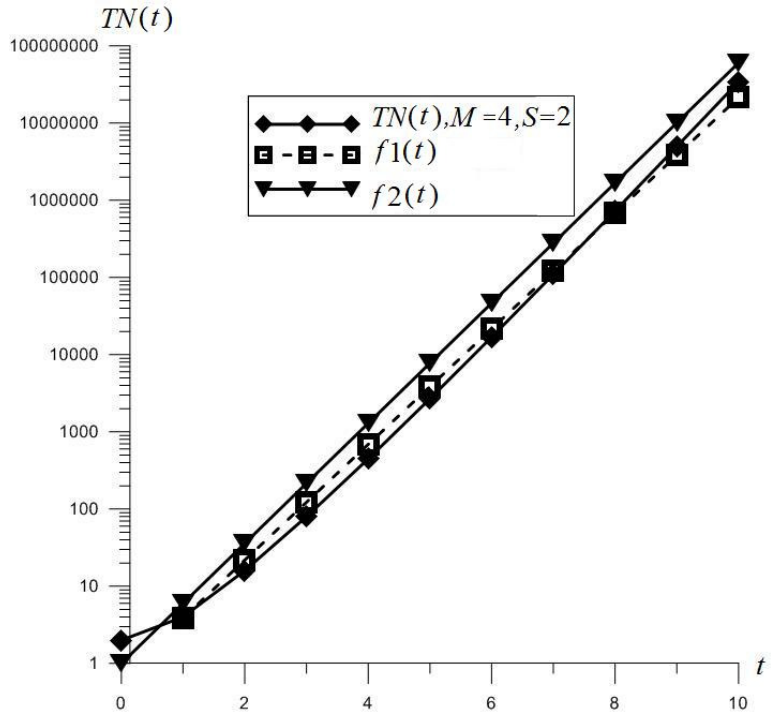


Рис. 4. Графики функций $TN(t)$ при $M = 4$, $S = 2$, $f1(t) = 0{,}7e^{1{,}7t}$ и $f2(t) = 6^t$

Для примера рассмотрим влияние степени специализации шаблона с различным числом вершин поддеревьев при поиске функции $F(x)$=2($x$+2($x$+2($x$+2($x$+2($x$+2))))), $t = 10$, представленной на рис. 5.

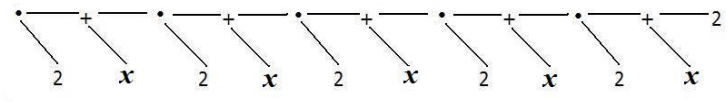


Рис. 5. Представление функции $F(x) = 2(x + 2(x + 2(x + 2(x + 2(x + 2)))))$ в виде дерева

Пусть заданы 8 пар вход — выход для $x = 0, 1, \ldots, 7$ и шаблон $T(x) = T(T_i, F_i)$, $i = 1, \ldots, 5$, где $F_i$ — известная часть функции, $T_i$ — поддерево, которое представляет

неизвестную часть функции и последовательно имеет $t = 1, 2, 3, 4, 5$ внутренних вершин (см. табл. 5). Множество терминальных символов в этом случае $TS = \{x, Cr\}$, $S = 2$, где $Cr$ — элемент множества случайных натуральных чисел. Набор операций, используемый для синтеза формул, следующий: $NS = \{+, -, \cdot, /\}$, $M = 4$. В табл. 5 приведены результаты сравнения характеристик алгоритма эволюционного синтеза — времени поиска решения и числа поколений $G$ до получения решения — при различных шаблонах (для каждого шаблона приведены средние значения, полученные по результатам 100 экспериментов).

Таблица 5

**Сравнение характеристик алгоритма эволюционного синтеза**

| $i$ | $T_i$, $F_i$ | $t$ | Время, с | $G$ |
|---|---|---|---|---|
| 1 | $T_1 = 2F_1$, $F_1 = (x + 2(x + 2(x + 2(x + 2(x + 2)))))$ | 1 | 0,0045 | 1,41 |
| 2 | $T_2 = 2(x + F_2)$, $F_2 = 2(x + 2(x + 2(x + 2(x + 2))))$ | 2 | 0,059 | 25,9 |
| 3 | $T_3 = 2(x + 2F_3)$, $F_3 = (x + 2(x + 2(x + 2(x + 2))))$ | 3 | 0,45 | 231,15 |
| 4 | $T_4 = 2(x + 2(x + F_4))$, $F_4 = 2(x + 2(x + 2(x + 2)))$ | 4 | 18,5 | 9321,4 |
| 5 | $T_5 = 2(x + 2(x + 2F_5))$, $F_5 = (x + 2(x + 2(x + 2)))$ | 5 | 48,4 | 24532,2 |

Из табл. 5 видно, что с уменьшением числа внутренних вершин $t$ неизвестных поддеревьев шаблона пространство поиска сокращается экспоненциально и характеристики алгоритма эволюционного синтеза уменьшаются также экспоненциально (с оценкой порядка $\mathrm{O}(C^t)$; так, $G(t) \approx 0{,}01\mathrm{e}^{2{,}48t}$ при $M = 4$, $S = 2$).

Рассмотрим влияние степени специализации шаблона с различным числом вершин поддеревьев при поиске функции $F(x) = \mathrm{e}^x x^3 \sin(x) \cos(x)$, $t = 8$, включающей функции от одной переменной и имеющей представление в виде дерева с внутренними вершинами со степенями 2 и 3.

Пусть заданы 15 пар вход — выход для $x = 0, 1, \ldots, 14$ и шаблон $T(x) = f(T_i)$, $i = 1, \ldots, 5$, где $T_i$ — поддерево, которое представляет неизвестную часть функции, а известная часть последовательно имеет $t = 1, 2, 3, 4, 5$ внутренних вершин (см. табл. 6). Множество терминальных символов в этом случае $TS = \{x, Cr\}$, $S = 2$, где $Cr$ — элемент множества случайных натуральных чисел. Набор операций, используемый для синтеза формул, следующий: $NS = \{+, -, \cdot, /, \exp, \sin, \cos\}$, $M = 7$. В табл. 6 приведены результаты сравнения характеристик алгоритма эволюционного синтеза — времени поиска решения и числа поколений $G$ до получения решения — при различных шаблонах (для каждого шаблона приведены средние значения, полученные по результатам 100 экспериментов).

Таблица 6

**Сравнение характеристик алгоритма**

| $i$ | $T$, $T_i$ | $t$ | Время, с | $G$ |
|---|---|---|---|---|
| 1 | $T = T_1' \cdot T_1''$, $T_1' = \mathrm{e}^x \cdot \sin(x)$, $T_1'' = x^3 \cdot \cos(x)$ | 1 | 1977,8 | 641,7 |
| 2 | $T = T_2 \cdot \cos(x)$, $T_2 = \mathrm{e}^x \cdot x^3 \cdot \sin(x)$ | 2 | 696,6 | 296,7 |
| 3 | $T = T_3 \cdot \cos(x) \cdot x$, $T_3 = \mathrm{e}^x \cdot x^2 \cdot \sin(x)$ | 3 | 150,5 | 31,9 |
| 4 | $T = T_4 \cdot \cos(x) \cdot x \cdot x$, $T_4 = \mathrm{e}^x \cdot x \cdot \sin(x)$ | 4 | 2,55 | 11,7 |
| 5 | $T = T_5 \cdot \cos(x) \cdot x \cdot \sin(x)$, $F_5 = \mathrm{e}^x \cdot x^2$ | 5 | 0,6 | 2,5 |

Из табл. 6 видно, что с увеличением числа внутренних вершин $t$ известных поддеревьев шаблона пространство поиска и характеристики алгоритма значительно умень-

шаются (с оценкой порядка $O(C^{-t})$; так, $G(t) \approx 0{,}0044e^{-1{,}75t}$ при $M = 7$, $S = 2$), хотя величина показателя меньше, чем в случае бинарного дерева.

Рассмотрим влияние шаблона при поиске описания последовательности Маккея — Гласса (Mackey — Glass), имеющей в дискретной форме следующий вид: $x(n) = 0{,}9x(n-1) + 0{,}2x(n-17)/(1 + x(n-17)^{10})$. Пусть заданы 100 пар вход — выход для $n = 0, 1, \ldots, 99$, множество терминальных символов $TS = \{x, Cr\}$, $S = 2$, множество примитивных функций $NS = \{+, -, \cdot, /, \mathrm{pow}(5, x)\}$, $M = 5$. При поиске описания в форме $x(n) = f(x(n-c), x(n-d))$, где параметры $c \in [0, 21]$, $d \in [0, 21]$ — целые числа, при числе поколений $G = 5000$ и времени поиска решения 4961 с получаем приближённое решение с величиной ошибки $\varepsilon = 0{,}0058$. При использовании шаблона, выражающего зависимость от предыдущего и любого другого члена последовательности в виде $x(n) = a \cdot x(n-1) + b \cdot f(x(n-d))$, где $a \in [0, 2]$, $b \in [0, 2]$ — действительные, $d \in [0, 21]$ — целые числа, получаем точную формулу последовательности при числе поколений $G = 801$ и времени поиска решения 905 с.

## Заключение

Дано описание алгоритма эволюционного синтеза, объединяющего преимущества генетических алгоритмов и генетического программирования, основанного на эволюционных вычислениях, шаблонах (темплейтах, скелетонах) алгоритмов и заданном множестве пар входных — выходных данных. Проведено исследование влияния степени специализации шаблона на пространство поиска при эволюционном синтезе, получены оценки величины сокращения пространства поиска при введении в шаблон дополнительной информации в виде формул, уточняющих модель и имеющих представление в виде бинарного дерева. Для данного случая теоретически и экспериментально показано экспоненциальное и сверхэкспоненциальное сокращение времени работы алгоритма синтеза при увеличении степени специализации шаблона. В планах на будущее стоит исследование влияния на результаты синтеза использования некорректного шаблона и способов его автоматической коррекции алгоритмом эволюционного синтеза.

## ЛИТЕРАТУРА

1. *Cole M.* Algorithmic Skeletons: Structured Management of Parallel Computation. Cambridge: The MIT Press, 1989. 131 p.
2. *Mirenkov N. and Mirenkova T.* Multimedia Skeletons and “Filmification” of Methods // Proc. First Intern. Conf. Visual Information Systems. Melbourne, Australia: Victoria University, 1996. P. 58–67.
3. *Watanobe Y., Mirenkov N., Yoshioka R., and Monakhov O.* Filmification of methods: A visual language for graph algorithms // J. Visual Languages and Computing. 2008. No. 1. P. 123–150.
4. *Gamma E., Helm R., Johnson R., and Vlissides J.* Design Patterns: Elements of Reusable Object-Oriented Software. Reading, MA.: Addison-Wesley, 1994. 416 p.
5. *Монахов О. Г.* Эволюционный синтез алгоритмов на основе шаблонов // Автометрия. 2006. № 1. С. 116–126.
6. *Монахов О. Г., Монахова Э. А.* Синтез новых семейств оптимальных регулярных сетей на основе эволюционных вычислений и шаблонов функций // Автометрия. 2004. № 4. С. 106–116.
7. *Monakhov O. and Monakhova E.* An Algorithm for Discovery of New Families of Optimal Regular Networks // Proc. 6th Inter. Conf. on Discovery Science (DS 2003), Oct. 17–20, 2003, Sapporo, Japan. Lecture Notes in Artificial Intelligence. 2003. V. 2843. P. 244–254.

8. *Монахов О. Г.* Программа эволюционного поиска математических описаний недоопределенных функций на основе шаблонов // Свид. об официальной регистрации программы на ЭВМ № 2011611546. М.: Федеральная служба по интеллектуальной собственности, патентам и товарным знакам, 2011.
9. *Goldberg D. E.* Genetic Algorithms, in Search, Optimization and Machine Learning. Reading, MA.: Addison-Wesley, 1989. 432 p.
10. *Koza J.* Genetic Programming. Cambridge: The MIT Press, 1992. 814 p.
11. *Monakhova E., Monakhov O., and Mukhoed E.* Genetic Construction of Optimal Circulant Network Designs // LNCS. 1999. V. 1596. P. 215–223.
12. *Sloane N. J. A. and Plouffe S.* The Encyclopedia of Integer Sequences. London: Academic Press, 1995. 587 p.
13. `http://oeis.org` — On-Line Encyclopedia of Integer Sequences.
14. *Koshy T.* Catalan numbers with applications. Oxford: Oxford University Press, 2009. 422 p.

2012 Вычислительные методы в дискретной математике №3(17)

DOI 10.17223/20710410/17/11

УДК 510.53+519.61

# О НЕКОТОРЫХ СВОЙСТВАХ ОБРАЗОВ ТРАНСФОРМИРОВАННЫХ ЗАДАЧ

Д. М. Мурин

*Ярославский государственный университет им. П. Г. Демидова, г. Ярославль, Россия*

**E-mail:** nirum87@mail.ru

Одним из способов доказательства NP-полноты задачи является полиномиальное сведение (трансформация) к ней задачи, NP-полнота которой уже установлена. При этом изучению свойств полученного образа, на наш взгляд, уделяется недостаточно внимания. В 1985 г. в работе Дж. Лагариаса и А. М. Одлыжко предложен метод решения NP-полной задачи о рюкзаке, дающий верное решение для «практически всех» рюкзаков, плотность которых не превышает 0,6463... В настоящей работе рассматривается вопрос о том, в какую область задач о рюкзаке (относительно плотности рюкзаков) при доказательстве NP-полноты попадают образы следующих задач: 3-ВЫП, Раскрашиваемость, Точное покрытие.

**Ключевые слова:** *NP-полнота, метод Лагариаса — Одлыжко, задача о рюкзаке.*

## Введение

«Задача» и «алгоритм» являются центральными понятиями теории сложности вычислений. Успехи, достигнутые за последние десятилетия в этой теории, в значительной степени связаны с рассмотрением понятия алгоритма в качестве первостепенного, что отразилось в несколько изменённом наименовании самой теории — теория сложности алгоритмов.

На наш взгляд, в теории сложности вычислений правомерны и должны быть интересны вопросы о «внутренней» структуре задачи и о структуре задачи в соотнесении с другими задачами.

Исходя из понимания задачи (массовой задачи) как множества (как правило, бесконечной серии) индивидуальных задач [1], при изучении структуры задач нам кажется естественным использовать язык теории множеств. Так, вместо полиномиальной сводимости (трансформации) задачи $P_1$ к задаче $P_2$ будем говорить о полиномиальном вложении задачи $P_1$ в задачу $P_2$.

Напомним формулировки рассматриваемых в работе задач.

***3-ВЫП.***

*Условие: Булева формула $F$ в 3-КНФ.*

*Вопрос: Выполнима ли булева формула $F$?*

***Раскрашиваемость.***

*Условие: Неориентированный граф $G = (V, E)$, натуральное число $k$.*

*Вопрос: Является ли граф $G$ $k$-раскрашиваемым?*

***Точное покрытие.***

*Условие: Семейство подмножеств $M = \{M_1, \ldots, M_l\}$ множества $\mathcal{U} = \{u_1, \ldots, u_p\}$.*

*Вопрос: Существует ли для семейства $M$ подсемейство попарно непересекающихся множеств, являющееся покрытием множества $\mathcal{U}$?*

***Задача о рюкзаке.***

*Условие:* $(a_1, a_2, \ldots, a_r, b) \in \mathbb{N}^{r+1}$.

*Вопрос: Имеет ли уравнение* $\sum\limits_{j=1}^{r} a_j x_j = b$, *где* $x_j$, $j = 1, \ldots, r$, — *неизвестные, решение в числах* 0 *и* 1*?*

В работе [2] рассматривается понятие *плотности* рюкзака.

**Определение 1.** *Плотностью* рюкзака $(a_1, a_2, \ldots, a_r, b)$ называется число

$$d = \frac{r}{\log_2(\max\limits_{1 \leqslant i \leqslant r} a_i)}.$$

Рассмотрим задачу о рюкзаке $(a_1, a_2, \ldots, a_r, s) \in \mathbb{N}^{r+1}$, где $s$ — сумма элементов произвольного подмножества множества $\{a_1, a_2, \ldots, a_r\}$; иными словами, $s = \sum\limits_{i=1}^{r} e_i a_i$, где $e_i \in \{0, 1\}, i = 1, \ldots, r$. Только для задач этого типа ответом на вопрос задачи о рюкзаке является «Да».

Пусть $t = \sum\limits_{i=1}^{r} a_i$, тогда можно считать, что $s \geqslant t/n$; иначе, если $s < t/n$, то существует $a_{i_0} \geqslant t/n$ и вопрос «Имеет ли уравнение $\sum\limits_{j=1}^{r} a_j x_j = b$, где $x_j$, $j = 1, \ldots, r$, — неизвестные, решение в числах 0 и 1?» можно заменить вопросом «Имеет ли уравнение $\sum\limits_{j=1, j \neq i_0}^{r} a_j x_j = b$, где $x_j$, $j = 1, \ldots, \hat{i_0}, \ldots, r$, — неизвестные, решение в числах 0 и 1?», уменьшив тем самым размерность задачи о рюкзаке. Аналогично, $s \leqslant (n-1)t/n$.

Определим следующим образом векторы $b_1, \ldots, b_{r+1}$ размерности $r + 1$:

$$\begin{gathered} b_1 = (1, 0, \ldots, 0, N \cdot a_1), \\ b_2 = (0, 1, \ldots, 0, N \cdot a_2), \\ \ldots \\ b_r = (0, 0, \ldots, 1, N \cdot a_r), \\ b_{r+1} = (0, 0, \ldots, 0, N \cdot s), \end{gathered}$$

где $N \geqslant \lceil \sqrt{r/2} \rceil$ — некоторое достаточно большое натуральное число. Очевидно, векторы $b_1, \ldots, b_{r+1}$ линейно независимы. Пусть $\Lambda$ — решётка, образованная этими векторами: $\Lambda = \{z_1 b_1 + \cdots + z_{r+1} b_{r+1} : z_1, \ldots, z_{r+1} \in \mathbb{Z}\}$.

Предположим, что у нас имеется АЛГОРИТМ $\mathfrak{A}$, который за полиномиальное время выдает один из кратчайших ненулевых векторов решётки. Ситуация с данным АЛГОРИТМом $\mathfrak{A}$ следующая: задача поиска кратчайшего ненулевого вектора решётки NP-полна, однако LLL-алгоритм построения приведённого базиса решетки позволяет за полиномиальное время[1] получать «достаточно короткие» векторы решётки [3]. В 1985 г. Дж. Лагариас и А. М. Одлыжко [2] установили, что при плотности рюкзака меньше 0,6463... с ростом размерности рюкзака вероятность того, что произвольный кратчайший вектор решётки $\Lambda$ не даёт решения задачи о рюкзаке, стремится к нулю. Этот результат улучшен в работе [4] до плотности рюкзака меньше 0,9408... благодаря следующей модификации рассматриваемой решётки $\Lambda$: векторы $b_1, \ldots, b_r$ остаются прежними, $b_{r+1} = (1/2, 1/2, \ldots, 1/2, N \cdot s)$, $N \geqslant \lceil \sqrt{r}/2 \rceil$.

Далее в работе излагается доказательство следующей теоремы.

[1] Сложность LLL-алгоритма построения приведённого базиса решётки $\Lambda$ составляет $O((r+1)^6 \log_2^3 B)$ битовых операций, где $B \geqslant |b_i|^2$ для $1 \leqslant i \leqslant r+1$ [3].

**Теорема 1.** Существует последовательность полиномиальных трансформаций (полиномиальных вложений) задачи 3-ВЫП в задачу о рюкзаке (3-ВЫП $\propto$ Раскрашиваемость $\propto$ Точное покрытие $\propto$ Задача о рюкзаке), обладающая тем свойством, что образ задачи 3-ВЫП лежит в области задач о рюкзаке, с высокой вероятностью решаемых методом Лагариаса — Одлыжко.

## 1. Точное покрытие $\propto$ Задача о рюкзаке

В работе Р. М. Карпа [5] задача о точном покрытии следующим образом полиномиально вкладывается в задачу о рюкзаке.

Пусть $w = |M| + 1 = l + 1$, а для всех $1 \leqslant j \leqslant |M|$ и $1 \leqslant i \leqslant p$

$$\varepsilon_{ji} = \begin{cases} 1, & \text{если } u_i \in M_j, \\ 0, & \text{если } u_i \notin M_j. \end{cases}$$

Тогда полагаем $r = |M|$, $a_j = \sum\limits_{i=1}^{p} \varepsilon_{ji} w^{i-1}$ для всех $1 \leqslant j \leqslant |M|$ и $b = (w^p - 1)/(w - 1)$. Таким образом, задача о рюкзаке приобретает следующий вид:

$$\left( \sum_{i=1}^{p} \varepsilon_{1i} w^{i-1}, \ldots, \sum_{i=1}^{p} \varepsilon_{ri} w^{i-1}, \frac{w^p - 1}{w - 1} \right).$$

Отметим, что вопрос о вхождении $u_p$ в одно из множеств семейства $M$ может быть решен за $\mathrm{O}(lp)$ арифметических операций. Если $u_p$ не входит ни в одно множество семейства $M$, то для семейства $M$ подсемейства попарно непересекающихся множеств, являющегося покрытием множества $\mathcal{U}$, не существует.

Оценим плотность получаемых описанным выше образом рюкзаков в предположении, что $u_p$ входит в те множества семейства $M$, чьи индексы содержатся в множестве $I$. Для всех $1 \leqslant j \leqslant |M|$ справедлива цепочка неравенств

$$\sum_{i=1}^{p-1} \varepsilon_{ji} w^{i-1} \leqslant \sum_{i=1}^{p-1} w^{i-1} = (w^{p-1} - 1)/(w - 1) < w^{p-1},$$

поэтому

$$\max_{1 \leqslant j \leqslant |M|} \sum_{i=1}^{p} \varepsilon_{ji} w^{i-1} = \max_{j \in I} \sum_{i=1}^{p} \varepsilon_{ji} w^{i-1} = w^{p-1} + \max_{j \in I} \sum_{i=1}^{p-1} \varepsilon_{ji} w^{i-1},$$

из чего следует оценка

$$d_{\text{ТП}} = \frac{|M|}{\log_2 \left( \max\limits_{1 \leqslant j \leqslant |M|} \sum\limits_{i=1}^{p} \varepsilon_{ji} w^{i-1} \right)} = \frac{|M|}{\log_2 \left( w^{p-1} + \max\limits_{j \in I} \sum\limits_{i=1}^{p-1} \varepsilon_{ji} w^{i-1} \right)} \leqslant$$
$$\leqslant \frac{|M|}{\log_2(w^{p-1})} = \frac{|M|}{(p-1)\log_2 w} = \frac{|M|}{(p-1)\log_2(|M|+1)}.$$

Таким образом, при полиномиальном вложении задачи о точном покрытии в задачу о рюкзаке образ задач о точном покрытии, удовлетворяющих условию $\dfrac{|M|}{(p-1)\log_2(|M|+1)} \leqslant 0{,}9408\ldots$, лежит в области задач о рюкзаке, с высокой вероятностью решаемых методом Лагариаса — Одлыжко.

Проиллюстрируем поведение функции

$$F(p, l) = \frac{l}{(p-1)\log_2(l+1)}.$$

На рис. 1 показаны плоскость 0,9408... и значения функции $F(p,l)$ при $2 \leqslant p \leqslant 100$, $1 \leqslant l \leqslant 300$ ($p$ и $l$ — натуральные числа). Из рисунка видно, что задачи о точном покрытии, удовлетворяющие условию $\dfrac{|M|}{(p-1)\log_2(|M|+1)} \leqslant 0{,}9408...$, составляют значительную долю задач о точном покрытии. Оценим эту долю.

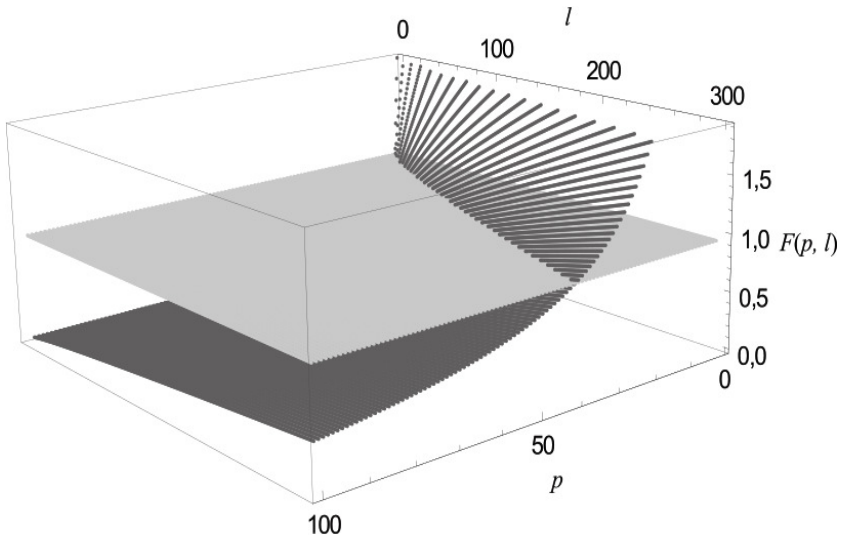


Рис. 1. Поведение функции $F(p,l)$ при $2 \leqslant p \leqslant 100$, $1 \leqslant l \leqslant 300$

Кривая $p = C_1 l/(\log_2(l+1)) + 1$, где $C_1 = (0{,}9408...)^{-1}$, делит положительный квадрант плоскости 0,9408... на две области (рис. 2). Область над кривой соответствует значениям функции $F(p,l)$, меньшим 0,9408... (задачи о точном покрытии с параметрами $p, l$ из данной области с высокой вероятностью решаются методом Лагариаса — Одлыжко), назовём эту область «хорошей». Область под кривой соответствует значениям функции $F(p,l)$, большим 0,9408..., назовем эту область «плохой».

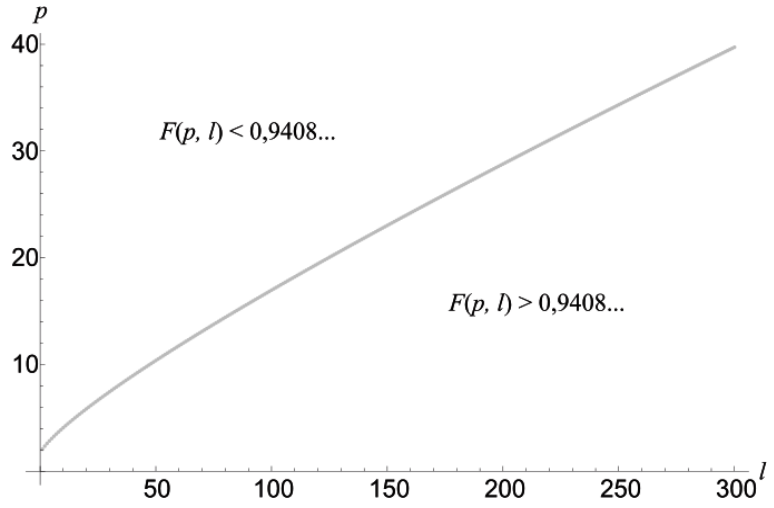


Рис. 2. Разделение положительного квадранта плоскости 0,9408... кривой $p = \frac{C_1 l}{\log_2(l+1)} + 1$

Площадь «плохой» области на интервале $1 \leqslant l \leqslant x-1$ может быть вычислена следующим образом:

$$P_{\text{по}}(x-1) = \int\limits_1^{x-1} \left( \frac{C_1 l}{\log_2(l+1)} + 1 \right) dl = x - 2 + \int\limits_1^{x-1} \frac{C_2(l+1)}{\ln(l+1)} d(l+1) - C_2 \int\limits_1^{x-1} \frac{d(l+1)}{\ln(l+1)} =$$

$$= x - 2 + C_2 \int\limits_1^{x-1} \frac{d(l+1)^2}{\ln(l+1)^2} - C_2 \mathrm{Li}(x) = x - 2 + C_2 \mathrm{Li}(x^2) - C_2 \mathrm{Li}(x),$$

где $C_2 = C_1 / \log_2 e$.

Таким образом, доля «плохой» области в квадрате, ограниченном прямыми $p = x-1$, $l = x-1$ и $p = 1$, $l = 1$, составляет

$$\frac{P_{\text{по}}(x-1)}{(x-2)^2} = \frac{x - 2 + C_2 \mathrm{Li}(x^2) - C_2 \mathrm{Li}(x)}{(x-2)^2}.$$

Воспользуемся тем фактом [6], что при $x \to \infty$

$$\mathrm{Li}(x) = x \left( \sum_{i=1}^{n} \frac{(i-1)!}{(\ln x)^i} + \mathrm{O}\left( \frac{1}{(\ln x)^{n+1}} \right) \right),$$

и перейдём к пределу при $x \to \infty$. Получим

$$\lim_{x\to\infty} \frac{P_{\text{по}}(x-1)}{(x-2)^2} = \lim_{x\to\infty} \frac{x - 2 + C_2 \mathrm{Li}(x^2) - C_2 \mathrm{Li}(x)}{(x-2)^2} =$$

$$= \lim_{x\to\infty} \frac{1}{x-2} + \lim_{x\to\infty} C_2 \left( \sum_{i=1}^{n} \frac{x^2 (i-1)!}{(x-2)^2 (\ln x^2)^i} + \mathrm{O}\left( \frac{x^2}{(x-2)^2 (\ln x)^{n+1}} \right) \right) -$$

$$- \lim_{x\to\infty} C_2 \left( \sum_{i=1}^{n} \frac{x (i-1)!}{(x-2)^2 (\ln x)^i} + \mathrm{O}\left( \frac{x}{(x-2)^2 (\ln x)^{n+1}} \right) \right) = 0.$$

Данный результат подтверждает тот факт, что доля «плохих» параметров $p$ и $l$ мала по сравнению с общим числом параметров задачи о точном покрытии.

Так как у каждого подмножества множества $\mathcal{U}$ есть дополнение до $\mathcal{U}$ и ввиду нецелесообразности включения в семейство $M$ пустого множества и лёгкости задачи о точном покрытии в случае, если $\mathcal{U} \in M$, можно считать, что в семейство $M$ входят не более чем $2^{p-1} - 1$ множеств. Иными словами, $l \leqslant 2^{p-1} - 1$, или $\log_2(l+1) \leqslant p - 1$.

Рассмотрим два интересных, на наш взгляд, случая.

1. Если $l + 1 = 2^{\frac{p-1}{C_3}}$, где $C_3$ — некоторая константа, то число удовлетворяющих критерию $F(p,l) \leqslant 0{,}9408...$ параметров $p$ и $l$ конечно и зависит только от $C_3$. Действительно, в этом случае

$$\frac{l}{(p-1)\log_2(l+1)} = \frac{l}{C_3 \log_2^2(l+1)},$$

и число $l$, таких, что $l \leqslant C_4 \log_2^2(l+1)$, где $C_4 = C_3 \cdot 0{,}9408...$, зависит только от $C_3$.

2. Если $l = C_5(p-1)$, где $C_5$ — некоторая константа, то число удовлетворяющих критерию $F(p,l) \leqslant 0{,}9408...$ параметров $p$ и $l$ бесконечно велико, так как неравенству $C_5 / \log_2(l+1) \leqslant 0{,}9408...$ удовлетворяет бесконечное число параметров $l$.

## 2. Раскрашиваемость $\propto$ Точное покрытие

В работе [7] задача раскрашиваемости следующим образом полиномиально вкладывается в задачу о точном покрытии.

Пусть $G = (V, E)$ — неориентированный граф и $k$ — натуральное число, определяющие задачу раскрашиваемости.

Тогда задача о точном покрытии строится следующим образом:

$$\mathcal{U} = V \cup \{[e, i] : e \in E \text{ и } 1 \leqslant i \leqslant k\} \text{ и } M = \{S_{vi}\} \cup \{T_{ei}\},$$

где $S_{vi} = v \cup \{[e, i] :$ ребро $e$ инцидентно $v\}$ для каждого узла $v \in V$ и $1 \leqslant i \leqslant k$ и $T_{ei} = \{[e, i]\}$ для каждого ребра $e \in E$ и $1 \leqslant i \leqslant k$.

Плотность $d_{\text{P}}$ рюкзаков, получаемых путём последовательных полиномиальных вложений задачи о раскрашиваемости в задачу о точном покрытии, а затем задачи о точном покрытии в задачу о рюкзаке, может быть оценена следующим образом:

$$d_{\text{P}} \leqslant \frac{|M|}{(|\mathcal{U}| - 1)\log_2(|M| + 1)} = \frac{|\{S_{vi}\} \cup \{T_{ei}\}|}{(|V| + |E|k - 1)\log_2(|\{S_{vi}\} \cup \{T_{ei}\}| + 1)} =$$
$$= \frac{|V|k + |E|k}{(|V| + |E|k - 1)\log_2(|V|k + |E|k + 1)} = \frac{(|V| + |E|)k}{(|V| + |E|k - 1)\log_2((|V| + |E|)k + 1)}.$$

Таким образом, при полиномиальном вложении задачи раскрашиваемости в задачу о рюкзаке образ задач раскрашиваемости, удовлетворяющих условию

$$\frac{(|V| + |E|)k}{(|V| + |E|k - 1)\log_2((|V| + |E|)k + 1)} \leqslant 0{,}9408...,$$

лежит в области задач о рюкзаке, с высокой вероятностью решаемых методом Лагариаса — Одлыжко.

## 3. 3-ВЫП $\propto$ Раскрашиваемость

В [7] задача 3-ВЫП следующим образом полиномиально вкладывается в задачу раскрашиваемости.

Пусть дана формула в 3-КНФ с $n$ переменными и $t$ сомножителями, $x_1, x_2, \ldots, x_n$ и $F_1, F_2, \ldots, F_t$ — соответственно переменные и сомножители формулы $F$, $v_1, v_2, \ldots, v_n$ — новые символы. Тогда полагаем $k = n + 1$, узлы графа $G$ таковы:

1) $x_i, \overline{x_i}, v_i$ для $1 \leqslant i \leqslant n$;
2) $F_i$ для $1 \leqslant i \leqslant t$.

Рёбра графа G:

1) все $(v_i, v_j)$, для которых $i \neq j$;
2) все $(v_i, x_j)$ и $(v_i, \overline{x_j})$, для которых $i \neq j$;
3) $(x_i, \overline{x_i})$ для $1 \leqslant i \leqslant n$;
4) $(x_i, F_j)$, если $x_i$ не входит в $F_j$, и $(\overline{x_i}, F_j)$, если $\overline{x_i}$ не входит в $F_j$.

Таким образом, $|V| = 3n + t$, $|E| = n(n-1)/2 + n(n-1) + n + (2n-3)t$, т. е.

$$|E| = 1{,}5n^2 - 0{,}5n + (2n - 3)t,$$
$$|V| + |E| = 1{,}5n^2 + 2{,}5n + 2(n - 1)t,$$
$$[|V| + |E|(n + 1) = 1{,}5n^3 + n^2 + 2{,}5n + ((2n - 1)n - 2)t.$$

Предполагается, что в одном сомножителе дважды одна переменная или её отрицание не встречаются.

Плотность $d_{3\text{-ВЫП}}$ рюкзаков, получаемых путём последовательных трансформаций задачи 3-ВЫП в задачу о рюкзаке, может быть оценена следующим образом:

$$d_{3\text{-ВЫП}} \leqslant \frac{(|V|+|E|)k}{(|V|+|E|k-1)\log_2((|V|+|E|)k+1)} <$$
$$< \frac{(1{,}5n^2+2{,}5n+2(n-1)t)(n+1)}{(1{,}5n^3+n^2+2{,}5n+(2n^2-n-2)t-1)\log_2((1{,}5n^2+2{,}5n+2(n-1)t)(n+1))} =$$
$$= \frac{3n^3+8n^2+5n+4(n^2-1)t}{(3n^3+2n^2+5n+2(2n^2-n-2)t-2)\log_2(1{,}5n^3+4n^2+2{,}5n+2(n^2-1)t)}.$$

Так как $n^3 > n$ при $n > 1$, $n^2-n-1 > 0$ при $n > (1+\sqrt{5})/2$ и $3n^3+2n^2+5n-2 > 2n^3+4n^2+2n$ при $n > 1$, то

$$\frac{3n^3+8n^2+5n+4(n^2-1)t}{(3n^3+2n^2+5n+2(2n^2-n-2)t-2)\log_2(1{,}5n^3+4n^2+2{,}5n+2(n^2-1)t)} <$$
$$< \frac{4n^3+8n^2+4n+4(n^2-1)t}{(2n^3+4n^2+2n+2(n^2-1)t)\log_2(1{,}5n^3+4n^2+2{,}5n+2(n^2-1)t)} =$$
$$= \frac{2}{\log_2(1{,}5n^3+4n^2+2{,}5n+2(n^2-1)t)} < \frac{2}{\log_2(2(n+1)t)}.$$

Неравенство $2/\log_2(2(n+1)t) \leqslant 0{,}9408...$ выполняется при $(n+1)t \geqslant 2^{1{,}1258...}$, что, очевидно, выполнено при $n \geqslant 2$ и $t \geqslant 1$, случай же $n = 1$ выглядит с точки зрения задачи 3-ВЫП очень экзотично.

Таким образом, при полиномиальном вложении задачи 3-ВЫП в задачу о рюкзаке образ всех заслуживающих внимания задач 3-выполнимости лежит в области задач о рюкзаке, с высокой вероятностью решаемых методом Лагариаса — Одлыжко.

Теорема доказана.

На наш взгляд, применимость метода Лагариаса — Одлыжко к решению на практике NP-полных задач является интересной темой для экспериментальных исследований. Разработанный в целях проведения этих исследований программный комплекс проходит процедуру тестирования.

## ЛИТЕРАТУРА


1. *Успенский В. А., Семенов А. Л.* Теория алгоритмов: основные открытия и приложения. М.: Наука, 1987. 288 с.
2. *Odlyzko A. M. and Lagarias J. C.* Solving Low-Density Subset Sum Problems // J. Association Computing Machinery. 1985. V. 32. No. 1. P. 229–246.
3. *Василенко О. Н.* Теоретико-числовые алгоритмы в криптографии. М.: МЦНМО, 2006. 336 с.
4. *Coster M. J., Joux A., LaMacchia B. A., et al.* Improved low-density subset sum algorithms // Computational Complexity. 1992. No. 2. P. 111–128.
5. *Karp R. M.* Reducibility among combinatorial problems // Complexity of Computer Computations: Proc. of a Symp. on the Complexity of Computer Computations, the IBM Research Symposia Series. NY: Plenum Press, 1972. P. 85–103.
6. *Коксон Э. Т.* Асимптотические разложения. М.: Мир, 1966. 159 с.
7. *Ахо А., Хопкрофт Дж., Ульман Дж.* Построение и анализ вычислительных алгоритмов. М.: Мир, 1979. 536 с.

# ПРИКЛАДНАЯ ТЕОРИЯ ГРАФОВ

DOI 10.17223/20710410/17/12

УДК 519.248, 519.176

## СВЯЗНОСТЬ ПЛАНАРНОГО ГРАФА С ВЫСОКОНАДЁЖНЫМИ РЁБРАМИ

Г. Ш. Цициашвили, А. С. Лосев

*Институт прикладной математики ДВО РАН, г. Владивосток, Россия*

**E-mail:** Guram@iam.dvo.ru, A.S.Losev@yandex.ru

Приведены результаты вычислительных экспериментов по определению вероятности несвязности планарных графов с высоконадёжными рёбрами. Полученные результаты подтверждают теоретическую, не более чем кубическую, оценку сложности проводимых вычислений, основанных на рассмотрении двойственных графов и построении асимптотических соотношений. Приведены результаты сравнения используемого методы с методом Монте-Карло, которые свидетельствуют о существенном сокращении числа арифметических операций и времени счета.

**Ключевые слова:** *вероятность связности, двойственный граф, минимальный разрез.*

### Введение

В работе [1] доказаны асимптотические формулы вычисления вероятности несвязности планарного графа с высоконадёжными рёбрами, обобщающие известную формулу Буртина — Питтеля [2]. Параметрами доказанного соотношения являются минимальное число $D$ рёбер в разрезах и число $C$ разрезов с $D$ рёбрами. Рассмотрение двойственных графов [3], в которых разрезы исходного графа порождают циклы, и использование известных алгоритмов перечисления циклов в произвольных графах [4] позволили сократить количество арифметических операций и получить сложность вычисления констант $D$, $C$ не более чем кубическую от числа граней графа, а в частных случаях линейную.

В настоящей работе рассмотрены отдельные случаи планарных графов с минимальным числом рёбер в разрезах $D = 2, 3, 4, 5$, в том числе нанотрубка, имеющая прикладное значение. Для рассмотренных графов построены двойственные графы, точно вычислены константы $D$, $C$ и получены асимптотические соотношения для вероятности несвязности. На основе полученных соотношений проведены вычислительные эксперименты, подтверждающие быстродействие предложенного алгоритма по сравнению с методом Монте-Карло.

### 1. Основные результаты

Рассмотрим неориентированный связный граф $G$ с конечным множеством вершин $U$ и рёбер $W$. Под разрезом графа понимается некоторое подмножество рёбер, после удаления которых граф перестает быть связным. Обозначим через $d(L)$ объём разреза $L$ (число его рёбер), $D$ — минимальный объём разреза, $C$ — число разрезов объёма $D$. Пусть ребра графа $G$ отказывают независимо с вероятностями $h$.

**Теорема 1.** Для вероятности несвязности графа выполняется соотношение

$$\overline{P} \sim Ch^{\,D},\ h \to 0. \tag{1}$$

Данная теорема 1 является обобщением известной асимптотической формулы Буртина — Питтеля [2] и доказана в [1].

Предположим, что граф $G$ является планарным и каждое ребро в нём принадлежит какому-либо простому циклу. Рёбра графа разбивают плоскость на грани. Сопоставим графу $G$ двойственный граф $G^*$. Каждой грани $z$ графа $G$ соответствует вершина $z^*$ графа $G^*$, каждому ребру $w$ графа $G$, принадлежащему граням $z_1$, $z_2$, соответствует ребро $w^*$, соединяющее вершины $z_1^*, z_2^*$. Нас будут интересовать разрезы минимального объёма, а значит, двойственные им циклы минимальной длины.

Обозначим $m$ число рёбер, $n$ — число граней (включая внешнюю) в планарном графе $G$. Пусть элементы $a_{ij}$ матрицы $A$ определяют число рёбер, содержащихся в пересечении граней $z_i \cap z_j$ $(i \neq j)$ графа $G$, $a_{ii} = 0$. Элементы степени $A^l$, $l > 1$, матрицы $A$ обозначим $a_{ij}^{(l)}$. В работе [4] получены формулы вычисления $c_k$ — числа простых циклов длины $k$, $k = 3, 4, 5$, в двойственном графе, а в работе [1] — формула для $c_2$:

$$c_2 = \frac{1}{4}\sum_{1\leqslant i,j\leqslant n} a_{ij}(a_{ij}-1), \quad c_3 = \frac{1}{6}\mathrm{tr}A^3, \quad c_4 = \frac{1}{8}\left(\mathrm{tr}A^4 - 2m - 2\sum_{1\leqslant i\neq j\leqslant n} a_{ij}^{(2)}\right),$$

$$c_5 = \frac{1}{10}\left(\mathrm{tr}A^5 - 5\mathrm{tr}A^3 - 5\sum_{i=1}^{n}\left(\sum_{j=1}^{n} a_{ij} - 2\right)a_{ij}^{(3)}\right).$$

Из формулы Эйлера [5] в работе [1] получены соотношения

$$D \leqslant 5, \quad D = \min(k : 2 \leqslant k \leqslant 5, c_k > 0), \quad C = c_D. \tag{2}$$

Из вышеизложенного следует, что вычисление асимптотических констант $D, C$ произвольных планарных графов путём перехода к двойственным графам и с использованием известных алгоритмов перечисления их циклов имеет сложность не более чем кубическую, что значительно меньше, чем в прямых алгоритмах счёта. В частных случаях, когда любые две внутренние грани имеют не более одного общего ребра, сложность вычислений становится линейной. Последнее часто встречается в приложениях и имеет практическую значимость, в частности при создании и изучении различных наноматериалов, в основе которых лежат гексагональные структуры [3]. Одно из таких соединений рассмотрено далее в примере 2.

## 2. Вычислительный эксперимент

**Пример 1.** Приведём примеры планарных графов, минимальное число рёбер в разрезах которых $D = 2, 3, 4, 5$. На рис. 1–4 построены графы $G_1$, $G_2$, $G_3$, $G_4$ с $D = 2, 3, 4, 5$ соответственно и двойственные к ним графы $G_1^*$, $G_2^*$, $G_3^*$, $G_4^*$.

Из формулы (2) нетрудно определить, что

$$C_{G_1} = 4,\ C_{G_2} = 7,\ C_{G_3} = 8,\ C_{G_4} = 12,$$

где $C_{G_i}$ имеет смысл константы $C$ для графа $G_i$, $i = 1, 2, 3, 4$. В свою очередь, из теоремы 1 следует:

$$\overline{P}_{G_1} \sim 4h^2, \quad \overline{P}_{G_2} \sim 7h^3, \quad \overline{P}_{G_3} \sim 8h^4, \quad \overline{P}_{G_4} \sim 12h^5, \quad h \to 0.$$

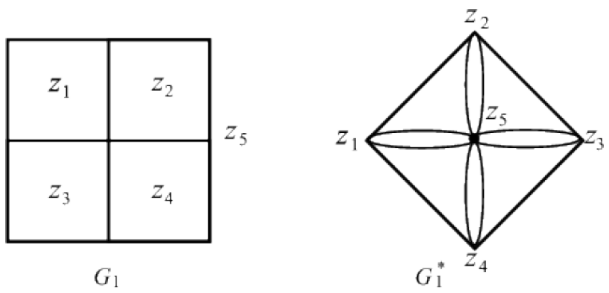


Рис. 1. Пример графа с $D = 2$ и двойственный ему ($z_5$ — внешняя грань)

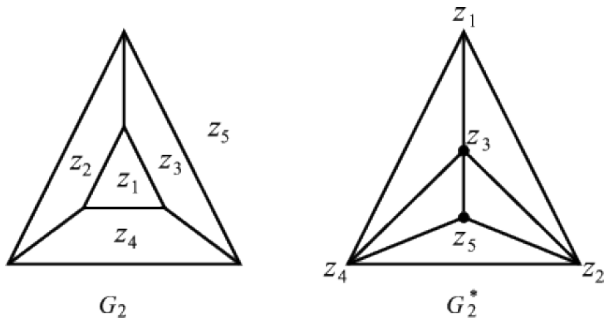


Рис. 2. Пример графа с $D = 3$ и двойственный ему ($z_5$ — внешняя грань)

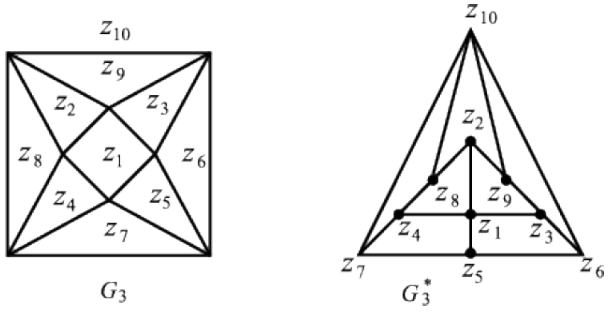


Рис. 3. Пример графа с $D = 4$ и двойственный ему ($z_{10}$ — внешняя грань)

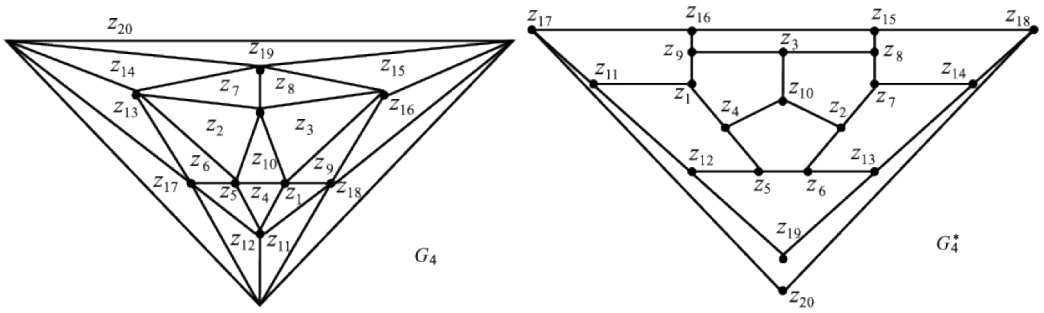


Рис. 4. Пример графа с $D = 5$ и двойственный ему ($z_{20}$ — внешняя грань)

Сравним результаты вычисления вероятности несвязности $\overline{P}$ по асимптотической формуле и методом Монте-Карло $\overline{P}^*$ с числом реализаций $10^6$. Положим $h = 0{,}05$,

тогда

$$\overline{P}_{G_1} \approx 0{,}01, \quad \overline{P}^*_{G_1} \approx 0{,}009982, \quad \left|\frac{\overline{P}^*_{G_1}}{\overline{P}_{G_1}} - 1\right| \approx 0{,}0018,$$

$$\overline{P}_{G_2} \approx 0{,}000875, \quad \overline{P}^*_{G_2} \approx 0{,}000821, \quad \left|\frac{\overline{P}^*_{G_2}}{\overline{P}_{G_2}} - 1\right| \approx 0{,}061714,$$

$$\overline{P}_{G_3} \approx 0{,}00005, \quad \overline{P}^*_{G_3} \approx 0{,}000048, \quad \left|\frac{\overline{P}^*_{G_3}}{\overline{P}_{G_3}} - 1\right| \approx 0{,}041667.$$

В результате проведённых вычислений время счёта методом Монте-Карло составило несколько часов, а по асимптотическому соотношению — не более минуты, что подтверждает полученную теоретическую оценку о том, что вычисление констант имеет не более чем кубическую сложность. Полученные результаты несложно распространить на произвольные планарные графы с соответствующим числом рёбер в разрезах.

**Пример 2.** Аналогичным образом определим вероятность несвязности $\overline{P}$ нанотрубки, которая получается склеиванием гексагональной решётки по противоположным сторонам. Данное соединение является основным элементом в различных наноструктурах и имеет прикладное значение [6].

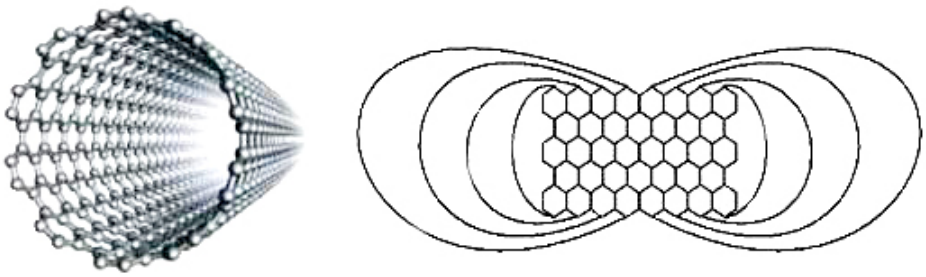

Рис. 5. Нанотрубка и её расположение на плоскости

Из формул (1), (2) следует

$$D = 2, \quad C = 18, \quad \overline{P} \sim 18h^2, \quad h \to 0.$$

Положим $h = 0{,}005$ и проведём вычислительный эксперимент, аналогичный описанному в примере 1. Имеем

$$\overline{P} \approx 0{,}00045, \quad \overline{P}^* \approx 0{,}00044, \quad \left|\frac{\overline{P}^*}{\overline{P}} - 1\right| \approx 0{,}02272.$$

В результате эксперимента подтверждена линейная оценка сложности вычисления констант $D, C$. В итоге время счёта вероятности несвязности нанотрубки методом Монте-Карло составило чуть более суток, что в несколько сотен раз больше, чем по асимптотическому соотношению.

Полученные формулы определения параметров планарного графа на основе построения двойственных графов существенно облегчают задачу в вычислительном плане, что подтверждается расчётами. А построение двойственного графа в обход традиционным подходам существенно уменьшает оценку сложности вычисления вероятности несвязности произвольных планарных графов и имеет не более чем кубическую

сложность. Помимо этого, в большинстве случаев переход к рассмотрению двойственных графов приводит к уменьшению обрабатываемого множества вершин, что также сокращает количество необходимых арифметических операций.

## ЛИТЕРАТУРА

1. *Tsitsiashvili G. Sh.* Complete calculation of disconnection probability in planar graphs // Reliability: Theory and Applications. 2012. V. 1(24). No. 1. P. 154–159.
2. *Буртин Ю., Питтель Б.* Асимптотические оценки надёжности сложных систем // Техническая кибернетика. 1972. Т. 10. № 3. С. 90–96.
3. *Whithney H.* Nonseparable and planar graphs // Trans. American Math. Soc. 1932. V. 34. P. 39–362.
4. *Harary F. and Manvel B.* On the Number of Cycles in a Graph // Matematickycasopis. 1971. V. 21. No. 1. P. 55–63.
5. *Прасолов В. В.* Элементы комбинаторной и дифференциальной топологии. М.: МЦНМО, 2004. 352 с.
6. *Золотухин И. В.* Углеродные нанотрубки // Соросовский образовательный журнал. 1999. № 3. С. 111–115.

# ДИСКРЕТНЫЕ МОДЕЛИ РЕАЛЬНЫХ ПРОЦЕССОВ

DOI 10.17223/20710410/17/13

УДК 621.391.1:004.7

## ИНВАРИАНТЫ КЛЕТОЧНО-АВТОМАТНЫХ МОДЕЛЕЙ РЕАКЦИОННО-ДИФФУЗИОННЫХ ПРОЦЕССОВ[1]

О. Л. Бандман

*Институт вычислительной математики и математической геофизики СО РАН, г. Новосибирск, Россия*

**E-mail:** bandman@ssd.sscc.ru, bandman@academ.org

Вводится понятие инварианта клеточно-автоматной модели (КА-модели). Инвариант равен некоторой безразмерной характеристике моделируемого процесса, которая не зависит от его математического описания и может быть выражена как в модельных, так и в физических величинах. Знание инварианта важно для практического моделирования, поскольку оно позволяет вычислить масштабирующие коэффициенты для перехода от модельных величин к реальным и обратно. Приводятся примеры инвариантов нескольких известных КА-моделей реакционно-диффузионных процессов и предлагается основанный на инвариантах общий подход к решению задачи масштабирования КА-моделей.

**Ключевые слова:** *клеточный автомат, клеточно-автоматное моделирование, нелинейная пространственная динамика, реакционно-диффузионные процессы, масштабирующие инварианты.*

### Введение

Современные компьютерные системы позволяют моделировать сложные физико-химические явления, имитируя превращения и движения реальных или абстрактных частиц в дискретных времени и пространстве. Широкий класс таких явлений составляют реакционно-диффузионные процессы. В них перемещения веществ представляются подчиняющимися законам диффузии, а их превращения (химические, фазовые, биологические) — нелинейными или пороговыми функциями, а также простыми подстановками вида «было — стало». Нелинейность и разрывность реакционной составляющей в такого рода явлениях делают затруднительным или совсем невозможным использование традиционных математических моделей, основанных на дифференциальных уравнениях. Поиски альтернативы привели к развитию дискретного моделирования, в частности к использованию идеи клеточного автомата (КА) фон Неймана [1].

Клеточно-автоматные модели реакционно-диффузионных процессов вызывают большой интерес [2, 3], поскольку позволяют представить сложный нелинейный процесс с помощью простых функций перехода одного конечного автомата. В теории динамических систем такие модели относятся к так называемым *сложным системам* (complex systems). Термин «*сложные системы*» введён в [4]. Его содержательный смысл состоит в том, что много простых вычислителей, функционируя совместно, проявляют себя как единый сложный процесс со свойствами самоорганизации, имитируя

[1]Работа поддержана грантами Президиума РАН (Проект № 15.9, 2011), СО РАН (Интеграционный проект ИП-32б 2009), РФФИ (грант № 11-01-0567а).

разного рода автоволновые процессы, процессы образования устойчивых структур и другие диссипативные явления [5].

В настоящее время клеточно-автоматное моделирование реакционно-диффузионных процессов развивается интенсивно [2, 3] как в плане исследований [5], так и в плане применений. При этом для построения КА-модели требуется найти способ перехода от физико-химического описания моделируемого явления к КА-представлению и после получения результата моделирования необходимо его интерпретировать в привычных для исследователя физических или химических терминах. В ряде случаев это удаётся сделать исходя из опыта, иногда приходится ограничиваться качественным сходством с предполагаемым процессом. Хотя некоторые попытки решить эту задачу уже предприняты [6, 7], систематического метода построения КА-моделей по заданным параметрам процесса пока не существует. Чтобы заполнить этот пробел, в работе предлагается новый подход, основанный на понятии *инварианта КА-модели*.

Инвариантом далее называется безразмерная характеристика процесса, не зависящая от способа его математического представления, которая позволяет определить масштабирующие коэффициенты между реальными и модельными величинами. Поскольку КА моделируют пространственную динамику, то характеризующий КА-модель инвариант связывает величину пространственного (размер клетки в метрах) и временно́го (длительность итерации в секундах) масштабов. Отсюда следует, что в общем случае эти масштабы независимо выбираться не могут. Кроме того, КА-модели реакционно-диффузионных процессов имеют по крайней мере две составляющие: диффузию и реакцию, каждая из которых характеризуется своим инвариантом. Инварианты таких сложных процессов не всегда возможно выразить в виде функции от инвариантов его составляющих, и масштабирующие функции не всегда возможно выразить аналитически через инвариант. В этих сложных случаях единственным путём определения инвариантов и масштабов остается вычислительный эксперимент и построение связывающих их таблиц и кривых.

## 1. Клеточный автомат

КА представляет собой множество одинаковых простых вычислителей, называемых *клетками*, которые задаются парами $(u, \mathbf{x})$, где $u \in A$ — *состояние клетки* из алфавита $A$; $\mathbf{x} \in X$ — имя клетки, равное вектору координат $d$-мерного дискретного пространства $X$ конечных размеров. В пространстве $X$ определены подмножества, называемые *шаблонами*,

$$T_k(\mathbf{x}) = \{\mathbf{x}, \mathbf{x} + \mathbf{a}_1, \ldots, \mathbf{x} + \mathbf{a}_{n-1}\},$$

где $\mathbf{a}_j$, $j = 1, \ldots, n-1$ — векторы смещения координат $x$, $n = |T(\mathbf{x})|$. Клетки с именами из $T_k(\mathbf{x})$ образуют *локальную конфигурацию*

$$S_k(\mathbf{x}) = \{(u_0, \mathbf{x}), (u_1, \mathbf{x} + \mathbf{a}_1), \ldots, (u_{n-1}, \mathbf{x} + \mathbf{a}_{n-1})\}.$$

Множество клеток $\Omega = \{(u_i, \mathbf{x}_i) : \mathbf{x}_i \in X\}$ называется *клеточным массивом*, а перечень состояний клеток $\Omega_A = (u_1, u_2, \ldots, u_{|X|})$ — *глобальным состоянием* КА.

Функционирование КА задается *локальным оператором* и режимом его применения к клеткам из $\Omega$. Локальный оператор $\Theta(\mathbf{x})$ есть композиция более простых локальных операторов: $\Theta(\mathbf{x}) = \Phi(\Theta_1(\mathbf{x}), \ldots, \Theta_l(\mathbf{x}))$. При этом $\Theta_k(\mathbf{x})$ может быть элементарной подстановкой $\theta_k(\mathbf{x})$. Подстановка выражается через локальные конфигурации следующим образом:

$$\theta_k(\mathbf{x}) : S_k(\mathbf{x}) \to S'_k(\mathbf{x}), \tag{1}$$

где $|S_k(\mathbf{x})| \geqslant |S'_k(\mathbf{x})|$, причём $T'_k \subseteq T_k$ и первые $n'$ клеток в локальной конфигурации составляют *базу* подстановки, а остальные $n - n'$ клеток играют роль *контекста*.

Подстановка $\theta_k(\mathbf{x})$ применима к клетке $(u, \mathbf{x}) \in \Omega$, если $S_k(\mathbf{x}) \in \Omega$, причём клетка $(u, \mathbf{x})$ с переменным состоянием считается принадлежащей $\Omega$, если область значений $u$ равна алфавиту $A$. Применение подстановки $\theta_k(\mathbf{x})$ сводится к замене состояний базовых клеток $(u_j, \mathbf{x} + \mathbf{a}_j) \in S'_k(\mathbf{x})$ на значения

$$u'_j = f_j(u_1, \ldots, u_n), \quad j = 0, \ldots, n', \tag{2}$$

где $f_j(u_1, \ldots, u_n)$ — *функция перехода*. Контекстные клетки не меняют своих состояний при применении подстановки (1).

В локальных операторах реакционно-диффузионных КА обычно используются следующие способы композиции подстановок: суперпозиция, случайный выбор одной из подстановок, применение подстановок в случайном порядке [8].

Применение $\Theta(\mathbf{x})$ ко всем $\mathbf{x} \in X$ изменяет глобальное состояние $\Omega_A(t)$ на новое $\Omega_A(t+1)$. Такое изменение составляет *итерацию*. Итерация может выполняться разными способами, которые называются *режимами функционирования КА*. Основными из них являются синхронный и асинхронный.

При *синхронном режиме* на каждой $t$-й итерации выполняется следующее:

1) для всех $(u, \mathbf{x}) \in \Omega(t)$ вычисляются новые состояния $u'(\mathbf{x})$ путём применения к ним функции перехода (2);

2) во всех клетках $(u, \mathbf{x}) \in \Omega(t)$ производится замена состояний $u(\mathbf{x})$ на новые $u'(x)$;

3) $\Omega(t) \to \Omega(t+1)$.

Синхронные КА-модели реакционно-диффузионных процессов более изучены, чем асинхронные. Однако они отображают моделируемое явление либо косвенным образом, либо путём преобразования дифференциального уравнения в частных производных к клеточно-автоматному виду [9, 10]. Локальные операторы синхронных КА ограничены тем, что их подстановки (1) должны иметь одноклеточную базовую локальную конфигурацию, т. е.

$$\forall \theta_i \in \Theta(\mathbf{x}) \quad |S'(\mathbf{x})| = 1, \tag{3}$$

что следует из условий корректности вычислений [11]:

$$\forall \mathbf{x}, \mathbf{y} \in X \quad \forall k, m \in \{1, \ldots, |\Theta(\mathbf{x})|\} \quad T_k(\mathbf{x}) \cap T_m(\mathbf{y}) = \varnothing. \tag{4}$$

Ограничение (3) затрудняет имитацию явления химических или биологических взаимодействий веществ по двум причинам. Во-первых, взаимодействия предполагают участие в них двух или более веществ, что требует изменений состояний двух или более клеток одновременно. Во-вторых, синхронное обновление состояний клеток не естественно, так как в моделируемом микромире нет синхронизации. Поэтому асинхронный режим оказывается часто более подходящим.

*Асинхронный режим* предполагает следующий порядок применений локального оператора:

1) с вероятностью $p = 1/|X|$ выбирается клетка $(u, \mathbf{x}) \in \Omega$;

2) к выбранной клетке применяется локальный оператор $\Theta(\mathbf{x})$, и состояния клеток базовых локальных конфигураций $(u', \mathbf{x}) \in S'(\mathbf{x})$ сразу же меняются на новые значения;

3) условно принимается, что $|X|$ повторений пп. 1 и 2 составляет одну итерацию, такое соглашение удобно для сравнения синхронного и асинхронного режима и соответствует понятию одного шага метода, называемого *кинетическим методом Монте-Карло* [12, 13].

Поскольку в асинхронных КА локальный оператор применяется последовательно к выбранным клеткам, условие (4) всегда выполняется, т. е. вычисление эволюции КА всегда корректно. Это значит, что не может произойти потери данных (в одну клетку одновременно записаться два разных значения). Проблема корректности возникает только тогда, когда алгоритм асинхронного КА реализуется параллельно на нескольких процессорах [8, 14].

Режим функционирования КА является его существенным параметром, т. е. если два КА различаются только режимами функционирования, то они являются представлениями двух разных КА. Чтобы определить однозначно КА, надо задать четыре параметра: $A, X, \Theta, \rho$, где $\rho \in \{\alpha, \sigma\}$, $\alpha$ обозначает асинхронный режим, $\sigma$ — синхронный, а КА $\aleph_\alpha$ и $\aleph_\sigma$ обозначают асинхронный и синхронный КА соответственно.

Задание КА в общем случае не определяет характера его поведения. Эволюции одного и того же КА с разными исходными массивами $\Omega_0$ могут кардинально различаться. Ярким примером является всем известный КА «Игра ‘Жизнь’», который при разных начальных массивах порождает самые разные эволюции. Поэтому для определения КА-модели необходимы пять символов:

$$\mathfrak{A} = \langle A, X, \Theta, \rho, \Omega(0) \rangle.$$

Инвариант является характеристикой КА-модели. В общем случае один и тот же КА при разных начальных условиях имеет разные инварианты.

КА-модели реакционно-диффузионных процессов с явно заданными подстановками диффузии и реакции характеризуются инвариантами, которые зависят от свойств обеих составляющих, и, следовательно, имеет смысл рассмотреть инварианты каждой из них отдельно.

## 2. Инварианты КА-моделей диффузии

Процесс диффузии имеет два математических представления: дискретное (КА) и непрерывное (уравнение Лапласа), которые характеризуются безразмерным коэффициентом $D = d \cdot \tau / h^2$, где $d$ — коэффициент диффузии, выраженный в $\text{м}^2 \cdot \text{с}^{-1}$; $\tau$ — шаг по времени в секундах; $h$ — шаг по пространству в метрах. Поскольку коэффициент $D$ не зависит от начальных условий, то КА диффузии можно отождествлять с КА-моделью, инвариант которой равен соответствующему $D$.

Из содержательного определения коэффициента диффузии как массы вещества, продиффундировавшего за единицу времени через единицу площади при градиенте концентрации равном единице (первый закон Фика для диффузии в изотропной среде), простым рассуждением можно получить значение $D$ следующим образом. В булевом клеточном массиве на границе между областью, в которой все клетки имеют состояние $u = 1$, и областью, в которой состояния клеток $u = 0$, градиент концентрации равен 1. Следовательно, масса вещества (количество частиц), продиффундировавшая за $\Delta t = 1$, равна вероятности того, что смежные пограничные клетки обменяются состояниями. Однако, поскольку такое рассуждение нельзя считать доказательством, ниже приводится его подтверждение результатами вычислительных экспериментов.

Наиболее известны два КА с булевым алфавитом, моделирующих процесс диффузии: 1) синхронный КА, названный в [15] *КА с окрестностью Марголуса*, и 2) асинхронный КА, получивший название *наивной диффузии* [16].

*Синхронный КА диффузии* $\aleph_\sigma = \langle A, X, \Theta \rangle$, $A = \{0, 1\}$, $X = \{\mathbf{x} : \mathbf{x} = (i, j)\}$, функционирует следующим образом. На множестве $X$ выделяются два подмножества: $X_0 = \{(i, j) : i \bmod 2{=}0, j \bmod 2{=}0\}$ — чётное и $X_1 = \{(i, j) : i \bmod 2{=}1, j \bmod 2{=}1\}$ —

нечётное. Вводится шаблон $T_{2\times 2} = \{(i,j),(i,j+1),(i+1,j+1),(i+1,j)\}$, который определяет локальную конфигурацию:

$$S(i,j) = \{(u_0,(i,j)),(u_1,(i,j+1)),(u_2,(i+1,j+1)),(u_3,(i+1,j))\}.$$

Вероятностный локальный оператор $\Theta(i,j)$ перемещает состояния клеток таким образом, что новые состояния $u'_k$, $k = 0,1,2,3$, равны

$$u'_k = \begin{cases} u_{(k+1) \bmod 4}, & \text{если} \quad \text{rand} < p, \\ u_{(k-1) \bmod 4}, & \text{если} \quad \text{rand} \geqslant (1-p), \end{cases}$$

где $p$ — вероятность применения $\Theta(i,j)$. Итерация подразделена на две стадии: на первой стадии $\Theta(i,j)$ применяется синхронно к клеткам $(i,j) \in X_0$, на второй — к клеткам $(i,j) \in X_1$. В [15] сделана попытка доказать, что при $p = 0{,}5$ безразмерный коэффициент диффузии $D_{2\sigma} = 1{,}5$. Эта величина поставлена под сомнение в [17], где утверждается, что в [15] не учтена разница в вероятности перемещения частицы в противоположных направлениях на чётных и нечётных стадиях, и показано, что при $p = 0{,}5$ $D_{2\sigma} = 1$, что согласуется с приведённым выше рассуждением. При $p < 0{,}5$ $D_{2\sigma} = p$.

Одномерный вариант синхронной диффузии функционирует в пространстве $X = \{i = 0,1,2,\ldots,N\}$, подразделённом на три подмножества: $X = X_0 \cup X_1 \cup X_2$, где $X_k = \{i : i \bmod 3 = k\}$, $k = 0,1,2$. Локальный оператор $\Theta(i)$ производит обмен состояниями клетки $(u_0,i)$ со случайно выбранным соседом $(u_1,i-1)$ или $(u_2,i+1)$:

$$\begin{cases} (u'_0 = u_1)\ \&\ \ (u'_1 = u_0), & \text{если} \quad \text{rand} < p, \\ (u'_0 = u_2)\ \&\ \ (u'_2 = u_0), & \text{если} \quad \text{rand} \geqslant (1-p). \end{cases}$$

Итерация подразделена на три стадии: на первой стадии $\Theta(i)$ применяется синхронно к клеткам $i \in X_0$, на второй — к $i \in X_1$, на третьей — к $i \in X_2$, коэффициент диффузии $D_{1\sigma} = p$.

*Асинхронный КА диффузии* $\aleph_\alpha = \langle A, X, \Theta\rangle$, $A = \{0,1\}$, $X = \{\mathbf{x} : \mathbf{x} = (i,j)\}$ (наивная диффузия) предполагает, что выбранная согласно асинхронному режиму равновероятно клетка из $X$ обменивается состоянием с одним из своих соседей, тоже выбранным с равной вероятностью.

В двумерном случае используется 5-точечный шаблон с локальной конфигурацией

$$S(\mathbf{x}) = \{(u_0,\mathbf{x}),(u_1,\mathbf{x}+\mathbf{a}_1)),(u_2,\mathbf{x}+\mathbf{a}_2),(u_3,\mathbf{x}+\mathbf{a}_3),(u_4,\mathbf{x}+\mathbf{a}_4)\}.$$

Локальный оператор $\Theta(\mathbf{x})$ применяется к клеткам $(u_0,\mathbf{x})$ и выполняет обмен состояниями с одной из соседних клеток $(u_k,\mathbf{x}+\mathbf{a}_k) \in S(\mathbf{x})$, выбранных равновероятно:

$$(u'_0 = u_k)\ \&\ (u'_k = u_0),\ \text{если}\ (k-1)/4 < \text{rand} < k/4)\ \&\ \text{rand} < p, \quad k = 1,2,3,4. \qquad (5)$$

Из (5) следует, что средняя вероятность того, что за одну итерацию частица переместится из клетки $(1,\mathbf{x})$ в одну из соседних клеток $(0,\mathbf{x}+\mathbf{a}_k)$, равна 0,5, т. е. при $p = 1$ $D_{2\alpha} = 0{,}5$, при $p < 1$ $D_{2\alpha} = 0{,}5 \cdot p$.

В одномерном варианте $X = \{i : i = 0,1,\ldots,N\}$. Случайно выбранная клетка $(u_0,i)$ производит обмен состояниями со случайно выбранным соседом $(u_1,i-1)$ или $(u_2,i+1)$ с вероятностью $p$. Функция переходов и, следовательно, коэффициент диффузии такие же, как в синхронном случае, $D_{1\sigma} = p$.

*Целочисленные КА-модели диффузии* предложены в [18], где они названы «многочастичными». В [18] даны две версии целочисленной КА-модели диффузии, соответствующие синхронной и асинхронной булевым моделям. Состояние клетки выражается в виде суммы двух частей: $v = \gamma \cdot v + (1-\gamma)v$, $\gamma \in (0,1]$, из которых только одна совершает перемещение в соседнюю клетку. Такой принцип позволяет уменьшить или даже совсем избавиться от автоматного шума и, следовательно, от необходимости осреднения булевого массива.

*Синхронная* целочисленная 2D КА-диффузия $\aleph_\sigma$ использует алфавит $A = \{0, 1, 2, \ldots, M\}$, дискретное пространство $X = \{\mathbf{x} : \mathbf{x} = (i,j)\}$, подразделённое на два подмножества клеток: чётное $X_0$ и нечётное $X_1$, как и в булевом случае. Оператор перехода $\Theta(i,j)$ выполняет следующую функцию переходов:

$$v'_k = \begin{cases} (1-\gamma)v_k + \gamma \cdot v_{(k+1) \bmod 4}, & \text{если} \quad \text{rand} < p, \\ (1-\gamma)v_k + \gamma \cdot v_{(k-1) \bmod 4}, & \text{если} \quad \text{rand} \geqslant (1-p), \end{cases}$$

где $p$ — вероятность применения $\Theta(i,j)$; $\gamma \in (0,1]$ — перемещаемая часть величины $v$, $k = 0, 1, 2, 3$. Как и в булевом варианте, итерация подразделена на две синхронные стадии: на первой стадии $\Theta(\mathbf{x})$ применяется к клеткам $\mathbf{x} \in X_0$, на второй — к клеткам $\mathbf{x} \in X_1$.

Одномерный вариант синхронной целочисленной КА диффузии функционирует в пространстве $X = \{i : i = 0, 1, 2, \ldots, N\}$, подразделённом на три подмножества: $X = X_0 \cup X_1 \cup X_2$, где $X_k = \{i : i \bmod 3 = k\}$, $k = 0, 1, 2$. Локальный оператор $\Theta(i)$ производит обмен частью состояния $\gamma \cdot v_0$ со случайно выбранным соседом из $S(i) = \{(v_0, i), (v_1, i-1), (v_2, i+1)\}$:

$$\begin{cases} (v'_0 = (1-\gamma)v_0 + \gamma \cdot v_1) \;\&\; (v'_1 = (1-\gamma)v_1 + \gamma \cdot v_0), & \text{если} \quad \text{rand} < p, \\ (v'_0 = (1-\gamma)v_0 + \gamma \cdot v_2) \;\&\; (v'_2 = (1-\gamma)v_2 + \gamma \cdot v_0), & \text{если} \quad \text{rand} \geqslant (1-p). \end{cases}$$

Как и в синхронном одномерном КА, итерация подразделена на три стадии: на $k$-й стадии $\Theta(i)$ применяется к $X_k$.

*Асинхронная целочисленная диффузия* работает аналогично наивной диффузии. Отличие как в двумерном, так и в одномерном случаях состоит в том, что межклеточный обмен происходит частью от значения состояния. В двумерном случае функция перехода переводит состояния $(v_0, \mathbf{x})$ и $(v_k, \mathbf{x} + \mathbf{a}_k)$ в $(v'_0, \mathbf{x})$ и $(v'_k, \mathbf{x} + \mathbf{a}_k)$ следующим образом:

$$\begin{cases} v'_0 = & (1-\gamma)v_0 + \gamma \cdot v_k, \\ v'_k = & (1-\gamma)v_k + \gamma \cdot v_0, \end{cases} \qquad (k-1)/4 < \text{rand} < k/4.$$

В одномерном случае применяется та же функция перехода (2), что и в синхронном случае, но без разделения итерации на стадии.

Коэффициенты диффузии целочисленных КА равны соответствующим коэффициентам булевых КА, умноженных на $\gamma$.

Знание инварианта позволяет определить масштабы параметров КА-модели, т. е. длину $h$ стороны клетки в метрах и время итерации $\tau$ в секундах. При этом один из этих масштабов может быть выбран из условий задачи, а второй вычисляется из соотношения

$$D = \frac{d \cdot \tau}{h^2}, \tag{6}$$

где $d$ — коэффициент диффузии среды в $\text{м}^2 \cdot \text{с}^{-1}$. Например, $h$ может быть выбран равным размеру реальных или абстрактных частиц. Тогда $\tau = (D \cdot h^2)/d$ с.

Для подтверждения правильности предположений о значениях коэффициентов диффузии проведены следующие вычислительные эксперименты. Для каждого случая выполнялось моделирование процесса диффузии двумя методами: 1) клеточным автоматом при предполагаемом значении $D_{CA}$ и 2) численным решением уравнения Лапласа при $D_L = D_{CA}$. Одинаковыми были также размеры $I \times J$ клеточного и сеточного пространств, а также начальные условия: $\langle v(0, \mathbf{x})\rangle = u(0, \mathbf{y})$ для всех $\mathbf{x} \in X$ и всех $\mathbf{y} \in Y$, где $Y$ — множество узлов в численном решении уравнения Лапласа. Производилось сравнение значений $\langle v(t, (I/2, j))\rangle$ с $u(t, (I/2, j)$ для $j = 0, \ldots, J$ и $t = 0, \ldots, T$, где $T = 300$ — число итераций, в течение которых процесс практически завершался. Если значения $\langle v(t, (I/2, j))\rangle$ и $u(t, (I/2, j)$ совпадали (с точностью до ошибки осреднения состояний КА), то считалось, что $D_{CA} = D_L$ — инвариант соответствующей КА-модели.

Результаты вычислительного моделирования подтвердили предполагаемые значения инвариантов для всех КА-моделей диффузии, в которых $D = 1$ для одномерной и двумерной синхронной КА-модели, а также для одномерной асинхронной; $D = 0{,}5$ для двумерной наивной КА-модели. Все коэффициенты диффузии целочисленных КА равны соответствующим коэффициентам булевых КА, умноженных на $\gamma$.

## 3. Инварианты КА-моделей реакции

В КА-моделях химических реакций за инвариант следует принять безразмерный коэффициент $R = k \cdot \tau$, где $k$ — константа скорости реакции с размерностью $\text{с}^{-1}$, которая для большинства изученных реакций известна; $\tau$ — длительность итерации клеточного автомата в секундах. Следует отметить, что инвариант КА-модели химической реакции не зависит от размеров клетки и, следовательно, масштабирующие коэффициенты $h$ и $\tau$ не зависят друг от друга, и $h$ может быть выбран исходя из условий задачи. Физический смысл инварианта $R$ — относительное изменение концентрации $(\Delta C/C)$ реактанта за время одной итерации $\tau$. Величина $\tau$ и является тем значением, которое связывает модельные величины с реальными. Если в модели участвует только одна реакция, то $\tau$ естественно выбирать равным длительности этой реакции. Если в моделируемом процессе участвует несколько реакций и, возможно, другие действия (адсорбция, диффузия), то соотношения скоростей регулируются вероятностями применения соответствующих локальных операторов, зависящих от всех инвариантов. Так, например, если в процессе участвует $n$ действий, описываемых $n$ подстановками, т. е. $\Theta(\mathbf{x}) = \{\theta_1(\mathbf{x}), \ldots, \theta_n(\mathbf{x})\}$, то вероятность выполнения каждой $\theta_i(\mathbf{x})$ равна

$$p_i = \frac{R_i}{\sum\limits_{j=1}^{n} R_j} = \frac{k_i}{\sum\limits_{j=1}^{n} k_j}. \tag{7}$$

В этом случае инвариантом процесса следует считать набор $R = \{R_1, \ldots, R_n\}$, в котором $R_i = k_i \cdot \tau$, а время итерации $\tau$ выбирается таким, чтобы самое медленное действие закончилось.

Например, реакция, которая получила название ZGB-реакции по фамилиям авторов [19] и положила начало широкому применению асинхронных вероятностных КА (кинетического метода Монте-Карло), представлена тремя химическими формулами:

$$\begin{aligned} \varnothing &\xrightarrow{k_1} \mathrm{CO_{ads}}, \\ \varnothing + \varnothing &\xrightarrow{k_2} \mathrm{O_{ads}} + \mathrm{O_{ads}}, \\ \mathrm{CO_{ads}} + \mathrm{O_{ads}} &\xrightarrow{k_3} \mathrm{CO_2(gas)}. \end{aligned}$$

На поверхность катализатора из газа адсорбируются монооксись углерода и диссоциированный кислород, которые, реагируя, образуют углекислый газ. Монооксись углерода диффундирует на пустые клетки, углекислый газ освобождает поверхность катализатора. Значения констант адсорбции ($k_1$) и десорбции ($k_2$) зависят от начальных условий процесса, они пропорциональны парциальным давлениям $\mathrm{pH(CO)}$, $\mathrm{pH(O_2)}$ соответствующих газов, значение константы реакции ($k_3$) от начальных условий не зависит. Реакция окисления происходит мгновенно, как только молекулы CO и $O_2$ окажутся в смежных клетках.

КА-модель этого процесса $\mathfrak{A}_\alpha = \langle A, X, \Theta, \Omega(0)\rangle$ определяется значениями $A = \{\varnothing, \mathrm{O}, \mathrm{CO}\}$, $X = \{\mathbf{x} = (i,j)\}$, $\Omega(0) = \{(\varnothing, \mathbf{x}) : \mathbf{x} \in X\}$, $\Theta(\mathbf{x}) = \Phi_s(\Phi_r(\theta_1(\mathbf{x}), \theta_2, (\mathbf{x})), \theta_3(\mathbf{x}))$, где $\Phi_s, \Phi_r$ — локальные операторы случайного выбора и суперпозиции подстановок соответственно:

$$\begin{aligned}
&\theta_1(\mathbf{x}): \quad (\varnothing, \mathbf{x}) \xrightarrow{p_1} (\mathrm{CO}, \mathbf{x}),\\
&\theta_2(\mathbf{x}): \quad (\varnothing, \mathbf{x}), (\varnothing, \mathbf{x} + \mathbf{a}_l\} \xrightarrow{p_2} \{(\mathrm{O}, (i,j)), (\mathrm{O}, (\mathbf{x} + \mathbf{a}_l)\},\\
&\theta_3(\mathbf{x}): \quad (\mathrm{CO}, \mathbf{x}), (\mathrm{O}, (\mathbf{x} + \mathbf{a}_l)\} \longrightarrow \{(\varnothing, \mathbf{x}), (\varnothing, (\mathbf{x} + \mathbf{a}_l)\}, \quad l = 1,2,3,4.
\end{aligned}$$

Вероятности равны

$$p_1 = \frac{k_1}{k_1 + k_2} = \frac{\mathrm{pH(CO)}}{\mathrm{pH(CO)} + \mathrm{pH(O_2)}}, \quad p_2 = \frac{k_2}{k_1 + k_2} = \frac{\mathrm{pH(O_2)}}{\mathrm{pH(CO)} + \mathrm{pH(O_2)}}.$$

Инвариант этой модели равен $R = \langle R_1, R_2\rangle$, где $R_1 = k_1 \cdot \tau$, $R_2 = k_2 \cdot \tau$. Масштаб времени $\tau$ следует выбрать равным времени адсорбции, так как адсорбция наиболее медленное действие. Масштаб длины $h$ не зависит от инварианта, его можно принять равным размеру молекулы CO. Результаты моделирования легко перевести в физические величины: например, количество образующегося углекислого газа за секунду равно

$$M(\mathrm{CO_2}) = \frac{N(\mathrm{CO_2})m(\mathrm{CO_2})}{\tau}\ \text{кг} \cdot \text{с}^{-1},$$

где $N(\mathrm{CO_2})$ — количество применений $\theta_3$ за одну итерацию; $m(\mathrm{CO_2})$ — масса молекулы $\mathrm{CO_2}$.

### 4. Инвариант КА-модели «распространение фронта»

К классу реакционно-диффузионных относятся также явления, в которых реакция выражена нелинейной функцией (чаще всего полиномом). В традиционной математике эти явления описываются дифференциальными уравнениями вида

$$u_t = Du_{xx} + F(u), \tag{8}$$

где правая часть содержит оператор диффузии (Лапласиан) и нелинейную функцию $F(u)$, отображающую реакционную составляющую процесса.

Типичным примером является уравнение «диффузии, соединенной с возрастанием количества вещества» [20, 21] с нелинейной функцией вида

$$F(u) = \alpha u(1 - u)$$

и начальными условиями

$$F(0) = F(1) = 0, \quad 0 < \alpha \leqslant 1, \quad 0 \leqslant u \leqslant 1.$$

В химии это уравнение моделирует реакцию превращения вещества B в вещество A с вероятностью, пропорциональной плотности A, при условии, что A+B=const, т. е.

$$\mathrm{A}+\mathrm{B} \xrightarrow{p} \mathrm{A}+\mathrm{A}.$$

В экологии уравнение (8) моделирует распространение эпидемий, сорняков, животных, в социологии — распространение слухов, привычек и т. д. Во всех случаях уравнение вида (8) моделирует пространственную экспансию, и процесс называется *распространением фронта*. Инвариантом процесса является *скорость распространения фронта*.

Так как в [20] даны аналитические оценки для одномерного случая, то далее исследуется одномерный аналог уравнения (8) $\mathfrak{A}_\sigma = \langle A, X, \Theta, \Omega(0)\rangle$, который построен как композиция КА диффузии и КА реакции [22] при $A = \{0,1\}$, $X = \{i : i = -L, \ldots, 0, \ldots, N\}$, $\Omega(0) = \{(1,i) : i < 0; (0,i) : i \geqslant 0\}$:

$$\Theta(i) = \Phi_{\mathrm{sum}}(\Theta_1(i), \Theta_2(i)). \tag{9}$$

Композиционный локальный оператор $\Phi_{\mathrm{sum}}$ выполняет покомпонентное арифметическое суммирование глобальных состояний $\Omega_A(\Theta_1)$ и $\Omega_A(\Theta_2)$ с преобразованием булева алфавита в вещественный и обратно [23], $\Theta_1(i)$ — локальный оператор КА-модели диффузии, $\Theta_2(i) : (u,i) \to (u',1)$ — одноклеточный бесконтекстный локальный оператор с функцией перехода $u' = F(u)$.

Таким образом, состояния $w(i) \in \Omega_A(t+1)$, полученные в результате применения (9) к $(v,i) \in \Omega(t)$, равны

$$w(i) = \mathrm{Disc}(\langle v'(i)\rangle + F(\langle v'(i)\rangle)), \tag{10}$$

где $v'(i)$ — значение функции перехода оператора $\Theta_1(i)$ к $(v,i) \in \Omega(t)$; $\mathrm{Disc}(x)$ — операция, преобразующая состояния $x \in \{0,1\}$ в $y = (0,1)$ следующим образом [23]:

$$x = \mathrm{Disc}(y) = \begin{cases} 1, & \text{если} \quad \mathrm{rand} < y, \\ 0 & \text{в иных случаях.} \end{cases}$$

В [20] аналитически доказано, что скорость распространения фронта при $t \to \infty$ зависит от коэффициентов диффузии ($D$) и скорости реакции ($\alpha$) следующим образом:

$$V_0 = 2\sqrt{D \cdot \alpha}.$$

Много позднее в [24, 25] было обнаружено, что дискретный характер компьютерного моделирования влияет на скорость распространения фронта в сторону её уменьшения. Отсюда следует, что полученные аналитическим путём асимптотические оценки не всегда соответствуют результатам клеточно-автоматного моделирования, т. е. значение инварианта необходимо устанавливать путём вычислительных экспериментов.

Моделирование было проведено для КА-модели и численного решения уравнения (8). Численное решение $u(t,x)$ сравнивалось с осреднёнными состояниями $\langle w(t,j)\rangle$ результатов КА-моделирования (10). Для более точного осреднения в одномерном случае клеточный массив КА целесообразно взять прямоугольным, т. е. $X = \{(i,j) : i = 0, \ldots, I; j = 0, \ldots, J\}$, оставив локальный оператор (9) моделирующим распространение фронта вдоль одной оси $j$. Такой приём перехода от одномерного случая к квазидвумерному позволяет выбирать достаточно большую окрестность осреднения, захватывающую всю ширину фронта $Av(j) = \{(0,j), \ldots, (I,j)\}$.

В процессе моделирования вычислялась скорость распространения фронта в зависимости от значений коэффициента диффузии $D$ от 0,2 до 3 при $\alpha = 0{,}5$ (рис. 1).

Для преодоления ограничения Куранта при решении уравнения (8) с $D > 0{,}2$ каждая итерация составлялась из $m = D/0{,}2$ временны́х шагов, в которых прибавление реакционной части выполнялось с вероятностью $p = 1/m$. Размер клеточного массива для всех испытаний равен $I \times J = 200 \times 800$. Начальные значения клеточного массива: $v(i,j) = 1$ при $0 < i < 200$, $j < 40$; в остальных случаях $v(i,j) = 0$. Скорость распространения фронта $V$ вычислялась в процессе моделирования на участке между $j_1 = 500$ и $j_2 = 700$ следующим образом: $V = (j_2 - j_1)/(t_2 - t_1)$ при $t_2, t_1$, таких, что $u(j_2, t_2) = u(j_1, t_1) = 0{,}5$. Эксперименты показали, что скорость фронта перестаёт изменяться уже после $t = 200$ при $0{,}1 < \alpha \leqslant 1$. Поэтому измеренные при $j > 500$ значения можно считать установившимися.

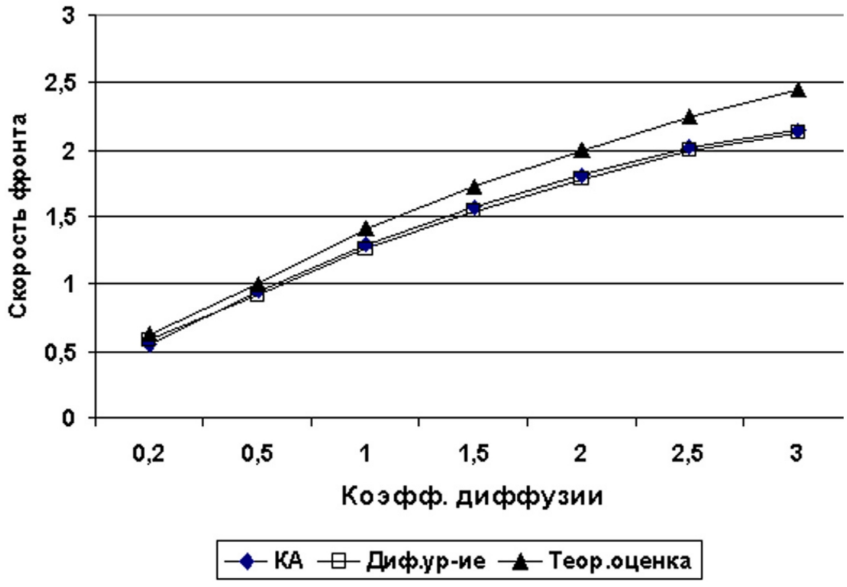


Рис. 1. Полученные путём моделирования зависимости скорости распространения фронта от значений коэффициента диффузии при $\alpha = 0{,}5$

## 5. Инвариант КА-модели диффузионно-ограниченной агрегации

Эта КА-модель (diffusion limited aggregation) [26] отображает целый ряд явлений, связанных с фазовыми переходами. Например, кристаллизация в растворах, образование льда, снежных хлопьев, рост кораллов и др. Наиболее известны асинхронные КА-модели, в которых диффузионная составляющая описывает случайное блуждание частиц в пространстве, а реакционная составляющая преобразует частицу в неподвижную, если она касается другой неподвижной. Процессу соответствует КА-модель $\mathfrak{A}_\alpha = \langle A, X, \Theta, \Omega(0)\rangle$, где $A = \{a, b, c\}$, $X = \{(i,j)\}$, $\Omega(0)$ — исходный клеточный массив, в котором $> 90\,\%$ случайно распределённых клеток имеют состояния $b$, $< 10\,\%$ — состояния $a$ и несколько клеток-зародышей находятся в состоянии $c$. Локальный оператор $\Theta(i,j) = \Phi_r(\theta_1(i,j), \theta_2))$, где $\theta_1(i,j)$ — оператор наивной диффузии (5),

$$\theta_2(i,j) : \{c,(i,j)),(a,(i+k,j+l))\} \xrightarrow{p} \{c,(i,j)),(c,(i+k,j+l))\}, \qquad (11)$$

где $(k,l) \in \{(0,1),(1,0),(-1,0),(0,-1))\}$; $p$ — *коэффициент прилипания*. В процессе эволюции вокруг зародышей образуется растущая структура (рис. 2).

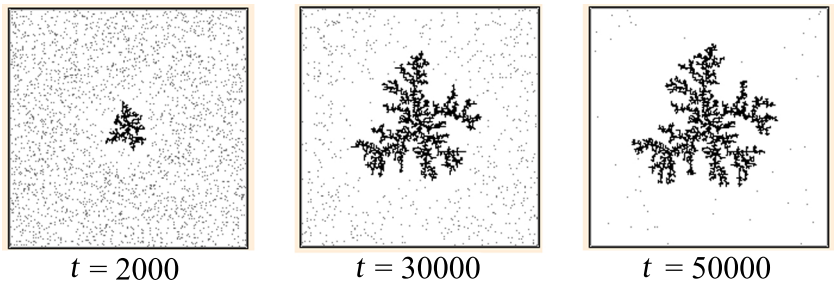


Рис. 2. Эволюция КА, моделирующего процесс диффузионно-ограниченной агрегации. Размер клеточного массива $400 \times 400$. Коэффициент прилипания $p = 1$, $\delta = 1{,}64$

Коэффициент прилипания $p$ определяет однозначно эволюцию КА при заданном $\Omega(0)$, а также её главную характеристику — фрактальную размерность:

$$\delta = \log(N(R))/\log(\pi R^2),$$

где $R$ — радиус круга с центром в зародышевой клетке; $N(R)$ — число клеток в состоянии $c$ в этом круге.

Инвариантом КА-модели следует считать коэффициент прилипания. Он однозначно определяет главную характеристику реального явления диффузионно-ограниченной агрегации. Аналитическое соответствие между $p$ и $\delta$ пока не установлено. Известны только полученные путем моделирования таблицы [26]. Таким образом, чтобы построить КА-модель $\mathfrak{A}_\alpha$ процесса агрегации вещества $a$ из раствора $b$ с заданной фрактальной размерностью $\delta$, необходимо выполнить следующее:

1) выбрать размер клетки $h$ равным размеру частицы (молекулы, гранулы) агрегируемого вещества и определить исходный клеточный массив;
2) по (6) определить шаг по времени $\tau$, исходя из инварианта наивной диффузии $D = 0{,}5$ и физического коэффициента диффузии вещества $d$;
3) по заданной величине $\delta$ определить по таблице из [26] соответствующее значения коэффициента прилипания $p$;
4) записать $\theta_1$ (5) и $\theta_2$ (11) и задать $\Omega(0)$.

## Заключение

Введено понятие инварианта КА-модели, при помощи которого предлагается решить проблему определения масштабирующих коэффициентов, позволяющих построить КА-модель по заданному физическому описанию процесса, а также выразить результаты моделирования в привычных физических понятиях (метрах, секундах, килограммах).

Рассмотрены инварианты для КА-моделей ограниченного класса явлений, называемых реакционно-диффузионными процессами, хотя предлагаемый подход может быть распространён и на КА-модели любого типа. Приведённые примеры показывают, что для КА-моделей, в которых диффузия и реакция заданы явно, инварианты могут быть определены исходя из известных инвариантов диффузии и реакций. В более сложных КА-моделях инварианты должны быть получены путём моделирования и сохраняться в виде таблиц и кривых, подобно тому, как это делалось при определении физических и химических констант в традиционном моделировании.

ЛИТЕРАТУРА


1. *Фон Нейман Дж.* Теория самовоспроизводящихся автоматов. М.: Мир, 1971. 240 с.
2. *Ванаг В. К.* Диссипативные структуры в реакционно-диффузионных системах. Эксперимент и теория. Ижевск: ИКИ, 2008. 300 с.
3. *Boccara N.* Reaction-Diffusion complex systems. Berlin: Springer, 2004. 397 p.
4. *Wolfram S.* Statistical mechanics in cellular automata // Rev. Mod. Phys. 1983. V. 55. P. 601–642.
5. *Ванаг В. К.* Исследование пространственно-распределенных динамических систем методами вероятностного клеточного автомата // УФН. 1999. Т. 169. № 5. С. 481–504.
6. *Bandman O.* Mapping physical phenomena onto CA-models // AUTOMATA-2008. Theory and Applications of Cellular Automata. London: Luniver Press, 2008. P. 381–397.
7. *Бандман О. Л.* Отображение физических процессов на их клеточно-автоматные модели // Вестник Томского госуниверситета. Сер. Математика. Кибернетика. Информатика. 2008. № 3. С. 345–356.
8. *Kalgin K.* Comparative study of Parallel Algorithms for Asynchronous Cellular Automata Simulation on Different computer Architectures // LNSC. 2010. V. 6350. P. 399–408.
9. *Weimar J.* Cellular Automata for reaction-diffusion systems // Parallel Computing. 1997. V. 23. No. 11. P. 1699–1715.
10. *Bandman O.* A Hybrid Approach to Reaction-Diffusion Simulation // LNCS. 2001. V. 2127. P. 1–16.
11. *Ачасова С. М., Бандман О. Л.* Корректность параллельных процессов. Новосибирск: Наука, 1998.
12. *Jansen A. P. J.* An Introduction to Monte Carlo Simulations of Surface Reactions. `arXiv: cond-mat/0303028v1[stst-mech]`. 2003. 51 p.
13. *Metropolis N. and Ulam S.* The Monte Carlo Method // Amer. Stati. Assoc. 1949. V. 44. No. 247. P. 335–341.
14. *Bandman O.* Parallel Simulation of Asynchronous Cellular Automata Evolution // LNCS. 2006. V. 4173. P. 41–47.
15. *Малинецкий Г. Г., Степанцов М. Е.* Моделирование диффузионных процессов с помощью клеточных автоматов с окрестностью Марголуса // Журн. вычисл. математики и мат. физики. 1998. Т. 38. № 6. С. 1017–1031.
16. *Toffolli T. and Margolus N.* Cellular Automata Machines: A New Environment for Modeling. USA: MIT Press, 1987. 260 p.
17. *Калгин К. В.* Клеточно-автоматные модели процесса диффузии // Частное сообщение.
18. *Medvedev Yu. G.* Multiparticle Cellular Automata for Diffusion Simulation // LNCS. 2011. V. 6083. P. 204–211.
19. *Ziff R. M., Gulari E., and Barshad Y.* Kinetic phase transition in an irreversible surface-reaction model // Phys. Rev. Lett. 1986. V. 56(24). P. 2553–2556.
20. *Колмогоров А. Н., Петровский И. Г., Пискунов И. С.* Исследование уравнения диффузии, соединенной с возрастанием количества вещества и его применение к одной биологической проблеме // Бюллетень МГУ, секция А. 1937. Вып. 6. С. 1–25.
21. *Fisher R. A.* The genetical theory of natural selection. Oxford: Univ. Press, 1930.
22. *Бандман О. Л.* Методы композиции клеточных автоматов для моделирования пространственной динамики // Вестник Томского госуниверситета. 2004. № 9(1). С. 183–193.
23. *Bandman O.* Synchronous versus Asynchronous Cellular Automata for Simulating Nano-Kinetics // Bul. Nov. Comp. Center. Iss. 21. Novosibirsk: NCC Publisher, 2004. P. 9–21.

24. *Warren C., Somfai E., and Sander L. M.* Velocity of front propagation in 1-dimensional autocatalytic reactions // Braz. J. Phys. 2000. V. 30. No. 1. P. 157–162.
25. *Brunet E. and Derrida B.* Effect of Microscopic Noise on Front Propagation Velocity // J. Stat. Phy. 2001. V. 103. No. 1–2. P. 269–282.
26. *Ackland G. J. and Tweedie E. S.* Microscopic model of diffusion limited aggregation and electrodeposition in the presence of leveling molecules // Phys. Rev. E. 2006. V. 73. Iss. 1. 011606.

# СВЕДЕНИЯ ОБ АВТОРАХ

**АГИБАЛОВ Геннадий Петрович** — заведующий кафедрой защиты информации и криптографии Национального исследовательского Томского государственного университета, доктор технических наук, профессор. E-mail: **agibalov@isc.tsu.ru**

**АНТОНОВА Ольга Владимировна** — аспирантка кафедры прикладной математики и информатики Новгородского государственного университета, г. Великий Новгород. E-mail: **antonova2906@yandex.ru**

**БАНДМАН Ольга Леонидовна** — профессор, доктор технических наук, главный научный сотрудник Института вычислительной математики и математической геофизики СО РАН, г. Новосибирск. E-mail: **bandman@ssd.sscc.ru**

**ЕДЕМСКИЙ Владимир Анатольевич** — доцент, доктор физико-математических наук, профессор кафедры прикладной математики и информатики Новгородского государственного университета, г. Великий Новгород.
E-mail: **Vladimir.Edemsky@novsu.ru**

**ЕРОФЕЕВ Степан Юрьевич** — аспирант Омского государственного университета им. Ф. М. Достоевского, г. Омск. E-mail: **stepan.erofeev@gmail.com**

**КОЛЕГОВ Денис Николаевич** — кандидат технических наук, доцент кафедры защиты информации и криптографии Национального исследовательского Томского государственного университета, г. Томск. E-mail: **d.n.kolegov@gmail.com**

**КОРЕНЕВА Алиса Михайловна** — студентка факультета кибернетики и информационной безопасности Национального исследовательского ядерного университета (МИФИ), г. Москва. E-mail: **alisa.koreneva@gmail.com**

**ЛОСЕВ Александр Сергеевич** — кандидат физико-математических наук, младший научный сотрудник Института прикладной математики ДВО РАН, г. Владивосток. E-mail: **A.S.Losev@yandex.ru**

**МОНАХОВ Олег Геннадьевич** — старший научный сотрудник, кандидат технических наук, ведущий научный сотрудник Института вычислительной математики и математической геофизики СО РАН, г. Новосибирск. E-mail: **monakhov@rav.sscc.ru**

**МУРИН Дмитрий Михайлович** — аспирант Ярославского государственного университета им. П. Г. Демидова, г. Ярославль. E-mail: **nirum87@mail.ru**

**РАЦЕЕВ Сергей Михайлович** — доцент кафедры информационной безопасности и теории управления Ульяновского государственного университета, г. Ульяновск.
E-mail: **RatseevSM@rambler.ru**

**РОМАНЬКОВ Виталий Анатольевич** — доктор физико-математических наук, профессор Омского государственного университета им. Ф. М. Достоевского, г. Омск. E-mail: **romankov48@mail.ru**

**СОЛОДОВНИКОВ Виктор Игоревич** — кандидат физико-математических наук, Академия криптографии Российской Федерации, г. Москва.
E-mail: **vis0707@rambler.ru**

**ТУЖИЛИН Михаил Эльевич** — старший научный сотрудник, кандидат физико-математических наук, доцент кафедры фундаментальной и прикладной математики Российского государственного гуманитарного университета, г. Москва.
E-mail: **mtmt@rambler.ru**

**ФОМИЧЕВ Владимир Михайлович** — доктор физико-математических наук, профессор кафедры математики Финансового университета при Правительстве Российской Федерации, г. Москва. E-mail: **fomichev@nm.ru**

**ЦИЦИАШВИЛИ Гурами Шалвович** — доктор физико-математических наук, профессор, заместитель директора по науке Института прикладной математики ДВО РАН, г. Владивосток. E-mail: **Guram@iam.dvo.ru**

**ЧЕРЕМУШКИН Александр Васильевич** — доктор физико-математических наук, заведующий кафедрой Института криптографии, связи и информатики, г. Москва.
E-mail: **avc238@mail.ru**

# АННОТАЦИИ СТАТЕЙ НА АНГЛИЙСКОМ ЯЗЫКЕ

*Edemskiy V. A., Antonova O. V.* **LINEAR COMPLEXITY OF GENERALIZED CYCLOTOMIC SEQUENCES WITH PERIOD $2^m p^n$.** A method for analyzing the linear complexity of generalized cyclotomic sequences with period $2^m p^n$ is proposed. It allows to pick out sequences with the high linear complexity. The linear complexity of some sequences is computed on the base of classes of quadratic and biquadratic residues.
**Keywords:** *generalized cyclotomic sequences, linear complexity.*

*Erofeev S. Y., Romankov V. A.* **ON CONSTRUCTING POSSIBLY ONE-WAY FUNCTIONS BASED ON THE NON-DECIDABILITY OF THE ENDOMORPHISM PROBLEM IN GROUPS.** The paper considers a schema for constructing a possibly one-way function on a group with the decidable word problem and undecidable endomorphism problem. Possible prerequisites for reliability of the proposed schema are analyzed. A corresponding authentication protocol with zero knowledge is proposed as an application. It is noted that for its security a more strong assumption on the undecidability of the two-level endomorphism problem is needed.
**Keywords:** *one-way function, endomorphism problem, authentication protocol.*

*Solodovnikov V. I.* **ON THE COINCIDENCE OF THE CLASS OF BENT-FUNCTIONS WITH THE CLASS OF FUNCTIONS WHICH ARE MINIMALLY CLOSE TO LINEAR FUNCTIONS.** For functions from $(\mathbb{Z}/(p))^n$ to $(\mathbb{Z}/(p))^m$ where $p$ is a prime, the property of closeness to linear functions is investigated. It is proved that, for any function, this property is inherited by its homomorphic images. As a generalization of an analogous statement for Boolean functions it is shown that if $p = 2$ or 3 then the class of functions which are absolutely minimally close to linear ones coincides with the class of bent-functions.
**Keywords:** *functions closeness, absolutely non-homomorphic functions, minimal functions, bent-functions.*

*Koreneva A. M., Fomichev V. M.* **ABOUT A FEISTEL BLOCK CIPHER GENERALIZATION.** Generalized Feistel Networks based on shift registers are investigated. The involutivity criterion is proved for the encryption algorithms in this class. The mixing properties of concerned algorithms are researched using a graph-theoretical approach. The upper bounds for the diameter and for the exponent of the mixing graph of the appropriated round function are given.
**Keywords:** *Feistel block cipher, involutivity, mixing graph (matrix) of a transformation, diameter of a graph, exponent of a graph.*

*Ratseev S. M.* **ABOUT PERFECT IMITATION RESISTANT CIPHERS.** Constructions of perfect imitation resistant ciphers are studied.
**Keywords:** *cipher, perfect cipher.*

*Tuzhilin M. E.* **LATIN SQUARES AND THEIR APPLICATIONS IN CRYPTOGRAPHY.** This survey contains examples of Latin squares applications in cryptography.
**Keywords:** *Latin square, quasigroups, cryptography.*

*Cheremushkin A. V.* **ON THE NOTION OF ELECTRONIC SIGNATURE.** An overview of two approaches to standardization and definition of electronic signature contained in Directive 1999/93/EC and in Federal Low of Russian Federation "On electronic signature".
**Keywords:** *electronic signature, digital signature.*

*Kolegov D. N.* **HIERARCHICAL ROLE-BASED ACCESS CONTROL DEVELOPMENT.** A new access control model is proposed. It is obtained by the addition of entity hierarchy and attributes to a RBAC model. The attributes in it are used for specification of the allowed access rights of subjects to entities in a semilattice based access control system.
**Keywords:** *security models, role-based access control, RBAC model, hierarchy of entities.*

*Agibalov G. P.* **TO REANIMATION OF RUSSIAN PROGRAMMING LANGUAGE.** An information is given about reanimation of the Russian programming language LYaPAS aimed at the elaboration of a trusted soft- and hardware for creating the secure computer systems of logic control.
**Keywords:** *LYaPAS, trusted software, trusted hardware, secure computer systems, logic control.*

*Monakhov O. G.* **INFLUENCE OF TEMPLATE SPECIALIZATION DEGREE ON THE SEARCH SPACE IN EVOLUTIONARY SYNTHESIS OF MODELS.** An algorithm for evolutionary synthesis is presented. It combines the advantages of genetic algorithms and genetic programming based on evolutionary computation, template (pattern, skeleton) of algorithm and a given set of pairs of input-output data. The influence of template specialization degree on the search space in the evolutionary synthesis is investigated. Estimates are obtained for the reduction of the search space with the introducing an additional information of template in the form of formulas represented by binary trees and refining the model. The exponential and superexponential reduction in the time of the synthesis algorithm with the increasing the template specialization degree is shown theoretically and experimentally.
**Keywords:** *genetic algorithms, genetic programming, evolutionary synthesis, template, design pattern.*

*Murin D. M.* **ABOUT SOME FEATURES OF THE TRANSFORMED PROBLEMS IMAGES.** One of the way to prove the NP-completeness of a problem is to transform in it polynomially a problem the NP-completeness of which we can prove. Herewith we pay less attention to the research of the received image features. In 1985 Lagarias and Odlyzko offered a method for solving knapsack NP-complete problem that gives a true decision for "near all" knapsacks with the density less than 0.6463... In this paper, we consider the following question: in which knapsack problems area (with regard to the knapsacks density) we can place, while proving the NP-completeness, the images of the problems such as 3-SAT, Colouring, Exact cover.
**Keywords:** *NP-completeness, Lagarias — Odlyzko method, knapsack problem.*

*Tsitsiashvili G. Sh., Losev A. S.* **CONNECTIVITY OF THE PLANAR GRAPH WITH HIGHLY RELIABLE EDGES.** In this paper, an algorithm based on the concept of dual graphs is constructed for calculation of incoherence probability for planar graphs with the high reliable edges. Numerical experiments show that, in a comparison with the Monte-Carlo method, this algorithm decreases calculation complexity significantly.
**Keywords:** *connectivity probability, dual graph, minimal cross section.*

*Bandman O. L.* **INVARIANTS OF REACTION-DIFFUSION CELLULAR AUTOMATA MODELS.** A concept of cellular automata (CA) model invariant is introduced. The invariant is a dimensionless value characterizing the process under simulation which is independent from mathematical description of the process and may be expressed both in model terms and in their physical counterparts. Invariants are important in practical computer simulation as a basis for calculating scaling coefficients needed for transition from CA model values to habitual physical quantities and vice versa. Invariants of some typical CA models of reaction-diffusion processes are presented. Based on the invariant a general approach to solve CA-modelling scaling problem is proposed.
**Keywords:** *cellular automaton, cellular-automata simulation, nonlinear spatial dynamics, reaction-diffusion processes, scaling invariants.*

Журнал «Прикладная дискретная математика» включен в перечень ВАК рецензируемых российских журналов, в которых должны быть опубликованы основные результаты диссертаций, представляемых на соискание учёной степени кандидата и доктора наук, а также в перечень журналов, рекомендованных УМО в области информационной безопасности РФ в качестве учебной литературы по специальности «Компьютерная безопасность».

Журнал «Прикладная дискретная математика» распространяется по подписке; его подписной индекс 38696 в объединённом каталоге «Пресса России». Полнотекстовые электронные версии вышедших номеров журнала доступны на его сайте vestnik.tsu.ru/pdm и на Общероссийском математическом портале www.mathnet.ru. На сайте журнала можно найти также и правила подготовки рукописей статей в журнал.

**Тематика публикаций журнала:**

- *Теоретические основы прикладной дискретной математики*
- *Математические методы криптографии*
- *Математические методы стеганографии*
- *Математические основы компьютерной безопасности*
- *Математические основы надёжности вычислительных и управляющих систем*
- *Математические основы информатики и программирования*
- *Вычислительные методы в дискретной математике*
- *Прикладная теория кодирования*
- *Прикладная теория автоматов*
- *Прикладная теория графов*
- *Логическое проектирование дискретных автоматов*
- *Дискретные модели реальных процессов*
- *Математические основы интеллектуальных систем*
- *Исторические очерки по дискретной математике и её приложениям*