

ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА

Научный журнал

2010

№3(9)

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



ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**РЕДАКЦИОННАЯ КОЛЛЕГИЯ ЖУРНАЛА
«ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА»**

Агибалов Г. П., д-р техн. наук, проф. (председатель); Девянин П. Н., д-р техн. наук, проф. (зам. председателя); Парватов Н. Г., канд. физ.-мат. наук, доц. (зам. председателя); Черемушкин А. В., д-р физ.-мат. наук, чл.-корр. Академии криптографии (зам. председателя); Панкратова И. А., канд. физ.-мат. наук, доц. (отв. секретарь); Алексеев В. Б., д-р физ.-мат. наук, проф.; Бандман О. Л., д-р техн. наук, проф.; Евдокимов А. А., канд. физ.-мат. наук, проф.; Евтушенко Н. В., д-р техн. наук, проф.; Закревский А. Д., д-р техн. наук, проф., чл.-корр. НАН Беларуси; Костюк Ю. Л., д-р техн. наук, проф.; Логачев О. А., канд. физ.-мат. наук, доц.; Матросова А. Ю., д-р техн. наук, проф.; Микони С. В., д-р техн. наук, проф.; Салий В. Н., канд. физ.-мат. наук, проф.; Сафонов К. В., д-р физ.-мат. наук, проф.; Фомичев В. М., д-р физ.-мат. наук, проф.; Чеботарев А. Н., д-р техн. наук, проф.; Шоломов Л. А., д-р физ.-мат. наук, проф.

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.

ООО «Издательство научно-технической литературы»

634050, Томск, пл. Ново-Соборная, 1, тел. (3822) 533-335

Редактор *Н. И. Шидловская*

Верстка *Д. А. Стефанцова*

Изд. лиц. ИД. №04000 от 12.02.2001. Подписано к печати 15.09.2010.
Формат 60 × 84 $\frac{1}{8}$. Бумага офсетная. Печать офсетная. Гарнитура «Таймс».
Усл. п. л. 13,2. Уч.-изд. л. 14,8. Тираж 300 экз. Заказ №17.

Отпечатано в типографии «М-Принт», г. Томск, ул. Пролетарская, 38/1

СОДЕРЖАНИЕ

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

Егоров В. Н. О группах автоморфизмов матриц	5
Логачев О. А. О значениях уровня аффинности для почти всех булевых функций	17
Парватов Н. Г. Точечные и сильно точечные функции на полурешётке	22
Смышляев С. В. Построение классов совершенно уравновешенных булевых функций без барьера	41

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

Агибалов Г. П., Панкратова И. А. Элементы теории статистических аналогов дискретных функций с применением в криптоанализе итеративных блочных шифров	51
Тренькаев В. Н. Реализация шифра Закревского на основе перестраиваемого автомата	69

МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

Качанов М. А. Анализ безопасности информационных потоков в операционных системах семейства <i>GNU/Linux</i>	77
Паутов П. А. Аутентификация в модели доверенной подсистемы на основе коммутативного шифрования	90

ПРИКЛАДНАЯ ТЕОРИЯ ГРАФОВ

Долгов А. А. Семейство точных 2-расширений турниров	96
---	----

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ДИСКРЕТНЫХ АВТОМАТОВ

Чеботарёв А. Н. Решение неравенств над автоматами в проектировании реактивных систем	100
--	-----

ДИСКРЕТНЫЕ МОДЕЛИ РЕАЛЬНЫХ ПРОЦЕССОВ

Бандман О. Л., Громилов С. А., Кинеловский С. А. Кумулятивный синтез: клеточно-автоматная модель процесса образования покрытия, наносимого на мишень с помощью кумулятивного потока частиц	111
СВЕДЕНИЯ ОБ АВТОРАХ	121
АННОТАЦИИ СТАТЕЙ НА АНГЛИЙСКОМ ЯЗЫКЕ	123

CONTENTS

THEORETICAL BACKGROUNDS OF APPLIED DISCRETE MATHEMATICS

Egorov V. N. On automorphism groups of matrices	5
Logachev O. A. On values of affinity level for almost all Boolean functions	17
Parvatov N. G. Point functions on semilattices	22
Smyshlyaev S. V. Construction of Perfectly Balanced Functions without Barriers	41

MATHEMATICAL METHODS OF CRYPTOGRAPHY

Agibalov G. P., Pankratova I. A. Statistical approximation theory for discrete functions with application in cryptanalysis of iterative block ciphers	51
Trenkaev V. N. Zakrevskij's cipher based on reconfigurable FSM	69

MATHEMATICAL BACKGROUNDS OF COMPUTER SECURITY

Kachanov M. A. Security analysis of information flows in GNU/Linux operating systems	77
Pautov P. A. Authentication in trusted subsystem model using commutative encryption	90

APPLIED GRAPH THEORY

Dolgov A. A. A family of exact 2-extensions of tournaments	96
---	----

LOGICAL DESIGN OF DISCRETE AUTOMATA

Chebotarev A. N. Solving inequalities over finite state machines in the reactive systems design	100
--	-----

DISCRETE MODELS FOR REAL PROCESSES

Bandman O. L., Gromilov S. A., Kinelovsky S. A. Cumulative synthesis: a cellular-automata model of target coating formation by means of cumulative flow of particles	111
---	-----

BRIEF INFORMATION ABOUT THE AUTHORS	121
---	-----

PAPER ABSTRACTS	123
-----------------------	-----

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРИКЛАДНОЙ ДИСКРЕТНОЙ МАТЕМАТИКИ

DOI 10.17223/20710410/9/1

УДК 519.142

О ГРУППАХ АВТОМОРФИЗМОВ МАТРИЦ

В. Н. Егоров

*Московский государственный университет им. М. В. Ломоносова,
Институт проблем информационной безопасности, г. Москва, Россия*

E-mail: Egorov49@inbox.ru

В работе рассматриваются группы левых (правых) автоморфизмов матриц, а также группы автоморфизмов. Вид элементов матрицы не играет роли, поэтому рассматриваются квадратные матрицы над кольцом целых чисел. Вводится понятие квазиавтоморфизма матрицы и соответственно понятие группы квазиавтоморфизмов. Дано описание дважды транзитивных групп левых (правых) автоморфизмов в терминах блок-схем. Структурная теория циклических блок-схем использована для вычисления групп левых (правых) автоморфизмов и групп квазиавтоморфизмов циркулянтов. Прикладное значение этой задачи связано с описанием групп автоморфизмов графов и проблемой изоморфизма графов, а также с вопросами групповой эквивалентности дискретных функций.

Ключевые слова: *группы автоморфизмов матриц, группы квазиавтоморфизмов матриц, циркулянты, блок-схемы.*

Введение

При изучении свойств симметрии различных комбинаторных объектов — матриц, графов, блок-схем, дискретных функций и т. д. — естественным образом возникает понятие автоморфизма данного объекта и соответственно группы автоморфизмов. Вместе с тем оказывается, что для всех этих объектов существует удобное обобщение, а именно — группы автоморфизмов матриц.

При этом вводится понятие левых (правых) автоморфизмов. Кроме того, в работе [1] было введено понятие квазиавтоморфизма матрицы и соответственно группы квазиавтоморфизмов. Оно играет важную роль, например, при описании импримитивных групп автоморфизмов графов, в частности циркулянтов.

В данной работе рассматриваются группы левых (правых) автоморфизмов, а также группы автоморфизмов и группы квазиавтоморфизмов матриц. Учитывая, что при этом вид элементов матриц не играет роли, рассматриваются квадратные матрицы над кольцом целых чисел.

В п. 1 приводятся основные определения и обозначения. В п. 2 дано описание матриц, обладающих 2-транзитивными группами левых (правых) автоморфизмов, с помощью матриц инциденций симметричных блок-схем. Получено описание циркулянтов размера $n \times n$ при простом $n \leq 97$ с 2-транзитивной группой левых (правых) автоморфизмов.

В п. 3 изучаются группы квазиавтоморфизмов матриц. Известно [2], что группа автоморфизмов матрицы A является 2-транзитивной тогда и только тогда, когда она

совпадает с симметрической группой S_n . В то же время известны примеры таких матриц A , что группа левых (правых) автоморфизмов является 2-транзитивной, но не совпадает с S_n . В теореме 5 доказано, что если n четно или свободно от квадратов и при этом группа квазиавтоморфизмов циркулянта 2-транзитивна, то она совпадает с S_n , а циркулянт имеет простой вид.

1. Основные определения и обозначения

Обозначим: \mathbb{N} — множество натуральных чисел; \mathbb{Z} — множество целых чисел; (n, m) — наибольший общий делитель чисел n и m ; $M_n(\mathbb{Z})$ — множество целочисленных матриц размера $n \times n$; $(0)_n, I_n, J_n$ — соответственно нулевая, единичная матрица и матрица из единиц; $|A|$ — определитель матрицы A ; A^t — транспонированная матрица; S_n — симметрическая группа подстановок множества вычетов по модулю n ; $g(i)$ и $g(M)$ — соответственно образы элемента $i \in \mathbb{Z}/n\mathbb{Z}$ и подмножества $M \subseteq \mathbb{Z}/n\mathbb{Z}$ под действием подстановки $g \in S_n$; \hat{g} — подстановочная матрица, соответствующая подстановке $g \in S_n$; $\text{Aff}(n)$ — аффинная группа подстановок, т. е. множество таких подстановок g , что $g(i) \equiv (\delta i + \nu) \pmod{n}$, где $i, \delta, \nu \in \mathbb{Z}/n\mathbb{Z}$, причем $(\delta, n) = 1$; G_i — стабилизатор точки i в группе $G \subseteq S_n$.

Определение 1. Пусть $A \in M_n(\mathbb{Z})$, $x, y \in S_n$ и выполняется матричное равенство

$$\hat{x}A = A\hat{y}. \quad (1)$$

Тогда подстановки x и y называются соответственно *левым* и *правым автоморфизмами* матрицы A . Множество $LG(A)$ всех левых автоморфизмов, очевидно, образует группу. Аналогично определяется группа $RG(A)$ всех правых автоморфизмов матрицы A . Если выполняется равенство

$$\hat{x}A = A\hat{x}, \quad (2)$$

то подстановка называется *автоморфизмом* матрицы A . Группу автоморфизмов матрицы A будем обозначать $G(A)$. Матрицы A и B называются эквивалентными, если существуют такие $x, y \in S_n$, что $\hat{x}A = B\hat{y}$.

Все перечисленные группы неоднократно рассматривались ранее (см., например, [1–7]). Заметим, что для произвольной матрицы $A \in M_n(\mathbb{Z})$ всегда существует матрица $A' \in M_n(\mathbb{Z})$ с неотрицательными элементами, такая, что все перечисленные группы идентичны для матриц A и A' . В силу сделанного замечания далее будем рассматривать матрицы с неотрицательными элементами. Отметим некоторые простые, но важные соотношения, непосредственно вытекающие из (1) и (2):

$$G(A) \subseteq LG(A) \cap RG(A); \quad (3.1)$$

$$LG(A) \subseteq G(AA^t); \quad RG(A) \subseteq G(A^tA); \quad (3.2)$$

$$RG(A) = LG(A^t). \quad (3.3)$$

Включения (3.1) и (3.3) очевидны. Для доказательства (3.2) заметим, что если $\hat{x}A = A\hat{y}$, то $\hat{x}^{-1}A\hat{y} = A$. Транспонируя, получим $A^t = \hat{y}^{-1}A^t\hat{x}$, следовательно, $\hat{x}^{-1}AA^t\hat{x} = AA^t$ или $\hat{x}AA^t = AA^t\hat{x}$.

Хорошо известно [2], что если $A \in M_n(\mathbb{Z})$ и $G(A)$ является 2-транзитивной группой, то

$$A = a_1I_n + a_2J_n, \quad a_1, a_2 \in \mathbb{Z},$$

и поэтому $G(A) = S_n$. Задача же описания матриц, обладающих 2-транзитивными группами левых (правых) автоморфизмов, представляется гораздо более сложной и в настоящее время далека от завершения. Интересные и глубокие результаты в этом направлении получены Фейтом [4, 5], который рассматривал блок-схемы с 2-транзитивными группами автоморфизмов.

Приведем формулировку теоремы, принадлежащей Бернсайду [8, с. 29], которая потребуется в дальнейшем.

Теорема 1. Пусть p — простое, а $G \subset S_p$ — транзитивная группа подстановок. Тогда G либо 2-транзитивна, либо изоморфна некоторой собственной подгруппе $\text{Aff}(p)$.

2. 2-транзитивные группы автоморфизмов

Покажем, что любая матрица $A \in M_n(\mathbb{Z})$, обладающая 2-транзитивной группой левых автоморфизмов, может быть представлена через матрицу инциденций некоторой симметричной блок-схемы, за исключением вырожденного случая, когда все строки матрицы равны между собой. Предварительно докажем вспомогательное утверждение.

Лемма 1. Пусть A, B — ненулевые $(0,1)$ -матрицы из $M_n(\mathbb{Z})$, такие, что

$$A^t A = a_1 I_n + a_2 J_n, \quad a_1 \neq 0; \quad (4)$$

$$B A^t = b_1 I_n + b_2 J_n, \quad (5)$$

тогда $A^t A = A A^t$ и $B \in \{A, J_n - A, J_n\}$.

Доказательство. Умножая обе части равенства (5) справа на A , получим

$$B A^t A = b_1 A + b_2 J_n A.$$

Отсюда, учитывая (4), имеем

$$a_1 B + a_2 B J_n = b_1 A + b_2 J_n A. \quad (6)$$

Покажем, что $J_n A = \mu_1 J_n$, а $B J_n = \mu_2 J_n$. Действительно, из (4) вытекает, что число единиц в i -й строке матрицы A^t постоянно и равно $a_1 + a_2$, следовательно, то же самое верно и для столбцов матрицы A , поэтому $J_n A = \mu_1 J_n$, где $\mu_1 = a_1 + a_2$. Далее, умножая обе части равенства (5) справа на J_n , получим

$$B A^t J_n = b_1 J_n + b_2 J_n^2 = (b_1 + n b_2) J_n.$$

Но $A^t J_n = (J_n A)^t = \mu_1 J_n$, поэтому

$$(a_1 + a_2) B J_n = (b_1 + n b_2) J_n,$$

т. е. $B J_n = \mu_2 J_n$, где $\mu_2 = (b_1 + n b_2) / (a_1 + a_2)$.

Теперь равенство (6) можно переписать в виде

$$b_1 A - a_1 B = \mu_3 J_n. \quad (7)$$

Пусть $b_1 = 0$, тогда $a_1 B = -\mu_3 J_n$, а $a_1 \neq 0$, следовательно, $B = (-\mu_3 / a_1) J_n = J_n$, поскольку B — ненулевая $(0,1)$ -матрица.

Пусть $b_1 \neq 0$. Рассмотрим матрицы $A \vee B$, $A \& B$ (дизъюнкция и конъюнкция берутся поэлементно). Из (7) непосредственно следует, что если $\mu_3 = 0$, то $A = B$. Рассмотрим поэтому случай $\mu_3 \neq 0$. Из (7) получаем $A \vee B = J_n$. Если $A \& B = (0)_n$, то, очевидно, $B = J_n - A$. Если же $A \& B \neq (0)_n$ и $A \neq B$, то из (7) можно заключить, что должно выполняться одно из равенств $b_1 = \mu_3$ или $-a_1 = \mu_3$, причем $b_1 - a_1 = \mu_3$. Однако это при $a_1 \neq 0$, $b_1 \neq 0$ невозможно. ■

Теорема 2. Пусть $A \in M_n(\mathbb{Z})$ и $LG(A)$ — 2-транзитивная группа подстановок. Тогда либо

$$A = a_1 V_n + a_2 J_n, \quad a_1 \neq 0,$$

где V_n — матрица инцидентий симметричной блок-схемы и $LG(A) = LG(V_n)$, либо все строки матрицы A равны и $LG(A) = S_n$.

Доказательство. Пусть матрица A содержит равные строки, тогда в силу 2-транзитивности $LG(A)$ пара равных строк w_1 и w_2 может быть с помощью подстановок из $RG(A)$ преобразована в любую пару строк матрицы A . Отсюда непосредственно следует, что все строки матрицы A попарно равны и поэтому $LG(A) = S_n$.

Пусть все строки матрицы A попарно различны. Не теряя общности, можно считать, что все элементы матрицы A являются ненулевыми, в противном случае вместо A можно рассмотреть матрицу $\tilde{A} = A + cJ_n, c \neq 0$, поскольку очевидно, что $LG(A) = LG(\tilde{A})$.

Обозначим через ν число различных элементов матрицы A . Тогда матрицу A можно представить в виде

$$A = a_1 A_1 + a_2 A_2 + \dots + a_\nu A_\nu, \quad (8)$$

где A_i — (0,1) матрица, $a_i \neq a_j \neq 0$, $A_i \& A_j = (0)_n$ при $i \neq j$, $i, j = 1, \dots, \nu$. Заметим сразу, что из (8) следует $LG(A) \subseteq G(A_i A_j^t)$, $i, j = 1, \dots, \nu$. Отсюда, в частности, получаем $A_i J_n = r_i J_n$, поскольку в силу транзитивности группы $LG(A)$, а следовательно и группы $LG(A_i)$, количество единиц в строках матрицы A_i постоянно. Кроме того, $LG(A) \subseteq G(A_i A_j^t)$, $i, j = 1, \dots, \nu$. Доказательство этого факта аналогично доказательству включения (3.2). Таким образом, группы $G(A_i A_j^t)$ являются 2-транзитивными, и поэтому для некоторых $r_i, k_{ij}, \lambda_{ij}$ имеют место равенства

$$A_i A_j^t = (k_{ij} - \lambda_{ij}) I_n + \lambda_{ij} J_n, \quad (9)$$

$$A_i J_n = r_i J_n.$$

При $i = j$ из (9) получаем

$$A_i A_i^t = (r_i - s_i) I_n + s_i J_n,$$

где $s_i = \lambda_{ii}$. Поскольку матрица A не содержит равных строк, существует такое $1 \leq i \leq \nu$, что $r_i - s_i \neq 0$, откуда следует, что

$$|A_i A_i^t| = (r_i + s_i(n-1))(r_i - s_i)^{n-1} \neq 0.$$

Но тогда и $|A_i| \neq 0$, поэтому [9, стр. 146]

$$A_i A_i^t = A_i^t A_i.$$

На основании последнего равенства, пользуясь (9), можно заключить, что при сделанных предположениях в (8) найдутся матрицы A_i и A_j , удовлетворяющие условиям леммы 1, такие, что $A_i \& A_j = (0)_n$. Но тогда в силу леммы 1 $A_j = J_n - A_i$, т.е. $A_i \vee A_j = J_n$ и матрицу A можно представить в виде $A = a_1 V_n + a_2 J_n$, где V_n — (0,1)-матрица, $a_1 \neq 0$. Кроме того, как уже было отмечено выше, матрица A должна удовлетворять соотношениям

$$AA^t = a'_1 I_n + a'_2 J_n, \quad AJ_n = (a'_1 + a'_2) J_n.$$

Отсюда нетрудно показать, что матрица V_n должна удовлетворять соотношениям

$$V_n V_n^t = (k - \lambda)I_n + \lambda J_n, \quad V_n J_n = k J_n.$$

В этом случае [9, теорема 10.2.3] V_n есть матрица инцидентий симметричной блок-схемы с параметрами n, k, λ , удовлетворяющими соотношению

$$k(k - 1) = \lambda(n - 1). \quad (10)$$

Равенство $LG(A) = LG(V_n)$ выполняется в силу того, что $J_n \hat{x} = \hat{y} J_n$ для любых $x, y \in S_n$. ■

Полученное условие является необходимым условием 2-транзитивности группы левых автоморфизмов. Что же касается достаточного условия, то эта задача представляется гораздо более сложной, поскольку в настоящее время полностью не решен вопрос даже о существовании нетривиальных симметричных блок-схем, т. е. блок-схем, для которых $k - \lambda \notin \{0, 1\}$. Наиболее важная теорема существования нетривиальных симметричных блок-схем принадлежит Бруку, Райзеру и Човла. Приведем её формулировку.

Теорема 3 [9, теорема 10.3.1]. Для существования нетривиальной симметричной (n, k, λ) -блок-схемы необходимы условия:

- если n чётно, то $k - \lambda = x^2$, $x \in \mathbb{N}$;
- если n нечётно, то существуют целые x, y, z , не все равные 0, удовлетворяющие уравнению

$$x = (k - \lambda)y + (-1)^{\frac{n-1}{2}} \lambda z^2.$$

Сейчас сформулируем и докажем теорему, анонсированную в [1], которая является дополнением к теореме 3.

Теорема 4. При $n = 2p, 2p^2, 4p, p^m + 1$, где p — простое число, существуют только тривиальные блок-схемы.

Доказательство. Пусть параметры n, k, λ удовлетворяют (10). Не теряя общности, можно положить $k \leq [n/2]$, поскольку всегда можно перейти к блок-схеме — дополнению. Кроме того, будем считать $\lambda \neq 0$, иначе из (10) следует $k - \lambda \in \{0, 1\}$. Разберем случаи.

a) $n = p^m + 1$.

Из (10) следует, что $k(k - 1) = \lambda p$. Поскольку $(k, k - 1) = 1$, то либо $p^m | (k - 1)$, либо $p^m | k$, и $k \leq p^m$ при $\lambda \neq 0$, т. е. $k \geq n - 1$, а это противоречит предположению, что $k \leq [n/2]$.

С помощью (10) нетрудно показать, что при $n = 2, 8$ существуют только тривиальные симметричные блок-схемы, поэтому будем предполагать, что p — нечётное простое число.

b) $n = 2p$.

Перепишем (10) в виде

$$k^2 - (k - \lambda) = \lambda n. \quad (11)$$

По теореме 2 имеем $k - \lambda = x^2$, где $x \in \mathbb{N}$. Из (11) получим $k^2 - x^2 = (k - x)(k + x) = 2\lambda p$. Если $x \neq 0$, то отсюда получаем $k - x < p$ и $k + x < 2p$. Но тогда $k + x = p$, поскольку p — простое, а $(k - x)(k + x)$ делится на p . Отсюда $k - x = 2\lambda$. Однако из равенств $k - x = 2\lambda$, $k + x = p$ следует $2k = 2\lambda + p$, что невозможно, поскольку p — нечётное.

c) $n = 2p^2$.

В этом случае имеем равенства $k - \lambda = x^2$ и $(k - x)(k + x) = 2\lambda p^2$. Как и выше, $k - x < p^2$, $k + x < 2p^2$. Предположим, что $k + x$ делится на p^2 , тогда $k + x = p^2$. Отсюда $k - x = 2\lambda$, $k + x = p^2$, т. е. $2k = 2\lambda + p^2$, что невозможно, поскольку p^2 — нечётное. Остаётся рассмотреть случай $k - x = \mu_1 p$, $k + x = \mu_2 p$. При этом k и x делятся на p , следовательно, λ также делится на p . Однако $2\lambda = \mu_1 \mu_2$, а $(\mu_1 \mu_2, p) = 1$. Противоречие.

d) $n = 4p$.

Аналогично «b» и «c» $k - \lambda = x^2$, $(k - x)(k + x) = 4\lambda p$, $k - x < 2p$. Пусть $k - x$ делится на p , тогда $k - x = p$, $k + x = 4\lambda$. В этом случае $2k = 4\lambda + p$, что невозможно, поскольку p — нечётное. Пусть $k + x$ делится на p . Предположим, что $k + x = sp$, где s — нечётное. Тогда $k - x = r$ — чётное число, следовательно, $2k = r + sp$, что невозможно, так как $r + sp$ — нечётное число. Остаётся рассмотреть случай $k + x = 2p$. При этом $k - x = 2\lambda$, т. е. $k - \lambda = \lambda + x = x^2$. Отсюда $\lambda = x^2 - x$ и $k - x(x - 1) = x^2$ или $k + x = 2x^2$. Последнее равенство, однако, противоречит тому, что $k + x = 2p$, где p — простое. ■

Из теорем 2, 3 и 4 получаем

Следствие 1. Пусть $A \in M_n(\mathbb{Z})$, $n = 2p, 2p^2, 4p, p^m + 1$ или противоречит условиям теоремы 3 и пусть $LG(A)$ — 2-транзитивная группа подстановок. Тогда $A = a_1 \hat{g} + a_2 J_n$, $g \in S_n$ и $LG(A) = S_n$.

Доказательство. Действительно, по теореме 2 $A = a'_1 V_n + a'_2 J_n$, где V_n — матрица симметричной блок-схемы. По теоремам 3 и 4, в свою очередь, $V_n = v_1 \hat{g} + v_2 J_n$, где \hat{g} — подстановочная матрица. Тогда $A = a'_1 v_1 \hat{g} + (a'_2 + v_2) J_n$ и $LG(A) = LG(\hat{g}) = S_n$. ■

Приведём важный результат, принадлежащий Фейту, относительно *циклических блок-схем*. Это симметричные блок-схемы, матрицы инциденций которых являются *циркулянтами*, т. е. имеют вид

$$\begin{bmatrix} c_0 & c_1 & \dots & c_{n-1} \\ c_{n-1} & c_0 & \dots & c_{n-2} \\ \dots & \dots & \dots & \dots \\ c_1 & c_2 & \dots & c_0 \end{bmatrix}. \quad (12)$$

В дальнейшем множество циркулянтов размерности $n \times n$ с элементами из \mathbb{Z} будем обозначать $CM_n(\mathbb{Z})$.

В нашей терминологии изоморфизм блок-схем означает эквивалентность матриц инциденций, а группа автоморфизмов блок-схемы есть группа левых автоморфизмов матрицы инциденций.

Обозначим чрез $D_m(q)$ блок-схему Зингера [9, с. 179] с параметрами

$$n = (q^{m+1} - 1)/(q - 1), \quad k = (q^m - 1)/(q - 1), \quad \lambda = (q^{m-1} - 1)/(q - 1),$$

где $q = p^r$, p — простое, а чрез $H(11)$ — циклическую блок-схему с параметрами $(11, 5, 2)$, матрица инциденций которой имеет вид (12) при

$$n = 11, \quad c_0 = c_1 = c_2 = c_4 = c_7 = 1, \quad c_3 = c_5 = c_6 = c_8 = c_9 = c_{10} = 0.$$

Теорема 5 [5]. Пусть D — циклическая (n, k, λ) блок-схема с 2-транзитивной группой автоморфизмов, $k \leq 50$ и D не изоморфна $H(11)$ или $D_m(q)$ при любых m и q . Тогда (n, k, λ) есть либо $(109, 28, 7)$, либо $(133, 22, 8)$.

Сейчас на основании теоремы 5 получим описание циркулянтов из $CM_n(\mathbb{Z})$ с 2-транзитивной группой левых автоморфизмов при простом $n \leq 97$.

Таблица значений параметра n блок-схемы $D_m(q)$ при различных m и q имеет вид

	2	3	4	5	7	8	9	11
2	7	13	21	31	57	73	91	*
3	15	40	85	*	*	*	*	*
4	31	*	*	*	*	*	*	*
5	63	*	*	*	*	*	*	*
6	*	*	*	*	*	*	*	*

Значком * отмечены значения параметра n , превосходящие 97. Анализируя данную таблицу, видим, что при простом $n < 97$ либо $m = 2$, либо $m = 4$, $q = 2$. При $m = 2$ имеем $\lambda = (q - 1)/(q - 1) = 1$, а при $m = 4$ — $q = 2$ и $(n, k, \lambda) = (31, 15, 7)$, поэтому на основании теорем 2 и 5 имеет место

Следствие 2. Пусть $C \in CM_n(\mathbb{Z})$, n — простое, $n \leq 97$ и $LG(C)$ — 2-транзитивная группа подстановок. Тогда

$$C = c_1 V_n + c_2 J_n,$$

где V_n — матрица инцидентий симметричной блок-схемы с параметрами

$$(n, k, 1), \quad (11, 5, 2), \quad (31, 15, 7).$$

3. Группы квазиавтоморфизмов матриц

В [1] введена некоторая специальная подгруппа группы $LG(A)$ матрицы A , которая называется группой квазиавтоморфизмов матрицы и обозначается $Q(A)$. Группы квазиавтоморфизмов имеют ряд интересных свойств, причём в некоторых случаях строение групп $G(A)$ и $LG(A)$ матрицы A существенно зависит от строения группы квазиавтоморфизмов некоторой ее подматрицы.

Определение 2. Группой квазиавтоморфизмов матрицы $A \in M_n(\mathbb{Z})$ называется подгруппа группы $LG(A)$ вида

$$Q(A) = \{x \in LG(A) \mid \exists y \in LG(A) : \hat{x}^{-1} A \hat{y} = A\}.$$

Нетрудно видеть, что указанное множество является группой. Кроме того, очевидно, что

$$G(A) \subseteq Q(A) \subseteq LG(A).$$

Ниже будет показано, что эти включения могут быть строгими.

Лемма 2. Пусть $A \in M_n(\mathbb{Z})$, тогда $Q(A) \subseteq LG(A^2)$.

Доказательство. Пусть $x \in Q(A)$. Тогда $\hat{x}^{-1} A \hat{y} = A$, $\hat{y}^{-1} A \hat{z} = A$, где $x, y \in LG(A)$, $z \in RG(A)$. Перемножая эти равенства, получаем $\hat{x}^{-1} A^2 \hat{z} = A^2$, т.е. $x \in LG(A^2)$. ■

В п. 2 были рассмотрены условия, при которых группы $G(A)$ и $LG(A)$ являются 2-транзитивными. При этом оказывается, что обязательно $G(A) = S_n$, а относительно группы левых автоморфизмов известны примеры матриц A , таких, что $LG(A)$ 2-транзитивна, но не совпадает с S_n . Таковыми являются, например, матрицы инцидентий зингеровских блок-схем. В связи с этим представляет интерес вопрос о том,

какие 2-транзитивные группы подстановок могут встречаться среди групп $Q(A)$ и какой вид имеют соответствующие матрицы. Рассмотрим этот вопрос для циркулянтов.

Отметим некоторые простые, но важные свойства циркулянтов, которые потребуются в дальнейшем:

- 1) $\langle t_n \rangle \subseteq G(C)$;
- 2) $C^t \in CM_n(\mathbb{Z})$;
- 3) $CM_n(\mathbb{Z})$ является коммутативным кольцом;
- 4) если C — периодический циркулянт периода d , т. е. последовательность c_0, c_1, \dots, c_{n-1} является периодической периода d , то C^t обладает тем же свойством;
- 5) если найдётся такая подстановка $x \in S_n, x \neq e_n$, что $\hat{x}C = C$ или $C\hat{x} = C$, то C — периодический циркулянт;
- 6) если C является матрицей инцидентий симметричной блок-схемы и $\Delta(C) = \{i | c_i = 1\}$, то $\Delta(C)$ — разностное множество.

Для доказательства п. 1–4 представим C в виде многочлена от матрицы \hat{t}_n с коэффициентами из \mathbb{Z} , т. е.

$$C = f(\hat{t}_n), \quad (13)$$

где $f(x) = c_0 + c_1x + \dots + c_{n-1}x^{n-1}$. Тогда, очевидно, $t_n \in G(C)$, а

$$C^t = g(\hat{t}_n), \quad (14)$$

где $g(x) = c_0 + c_{n-1}x + \dots + c_1x^{n-1}$, что доказывает свойства 1 и 2. Свойство 3 также легко следует из (13).

Пусть c_0, c_1, \dots, c_{n-1} — периодическая последовательность периода d , тогда, как нетрудно видеть, последовательность c_0, c_{n-1}, \dots, c_1 также обладает этим свойством. Из равенства (14) теперь следует свойство 4.

Для доказательства свойства 5 достаточно заметить, что равенство $\hat{x}C = C$ выполняется при $x \neq e_n$ в том и только в том случае, если в C имеются равные строки, что равносильно периодичности C . Аналогично при $C\hat{x} = C$ в C есть равные столбцы, т. е. в C^t есть равные строки, поэтому можно воспользоваться свойством 4.

Свойство 6 непосредственно следует из определений циркулянта, блок-схемы и разностного множества.

Теорема 6. Пусть $C \in CM_n(\mathbb{Z})$, а n чётно или свободно от квадратов, и пусть $Q(C)$ — 2-транзитивная группа подстановок. Тогда $C = c_1\hat{t}_n^s + c_2J_n, 0 \leq s \leq n-1$, и $Q(C) = S_n$.

Предварительно докажем вспомогательное утверждение.

Лемма 3. Пусть $C \in CM_n(\mathbb{Z})$ и является $(0,1)$ -матрицей, $|\Delta(C)| = k$ и первая строка матрицы C^2 имеет вид $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$. Тогда

- а) при $n = 2m + 1$ среди элементов $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ ровно k нечётных;
- б) если, кроме того, матрица C есть матрица инцидентий симметричной блок-схемы с параметрами (n, k, λ) , то при $n = 2m$ все элементы $\alpha_0, \alpha_3, \dots, \alpha_{n-1}$ — чётные, а среди элементов $\alpha_0, \alpha_2, \dots, \alpha_{n-2}$ ровно $k - \lambda$ нечётных.

Доказательство. а) По условию леммы

$$C = \hat{t}_n^{s_1} + \hat{t}_n^{s_2} + \dots + \hat{t}_n^{s_k},$$

где $\{s_1, s_2, \dots, s_k\} = \Delta(C)$. Нетрудно видеть, что

$$C^2 = \hat{t}_n^{2s_1} + \hat{t}_n^{2s_2} + \dots + \hat{t}_n^{2s_k} + Q, \quad Q = 2 \cdot \sum_{i \neq j} \hat{t}_n^{s_i + s_j}, \quad 1 \leq i, j \leq k.$$

Матрица Q , очевидно, содержит только чётные элементы, поэтому рассмотрим матрицу

$$C' = \hat{t}_n^{2s_1} + \hat{t}_n^{2s_2} + \dots + \hat{t}_n^{2s_k}. \quad (15)$$

Если $n = 2m + 1$, то $(2, n) = 1$ и из сравнения $2s_i \equiv 2s_j \pmod{n}$ получаем $s_i \equiv s_j \pmod{n}$. Отсюда следует, что C' является $(0, 1)$ -матрицей из $CM_n(\mathbb{Z})$ и $|\Delta(C')| = k$, т. е. утверждение «а» доказано.

б) Пусть $n = 2m$ и C есть матрица инцидентий симметричной блок-схемы, а первая строка матрицы C' имеет вид $\alpha'_0, \alpha'_3, \dots, \alpha'_{n-1}$. Из (15) следует, что все элементы $\alpha'_1, \alpha'_3, \dots, \alpha'_{n-1}$ равны 0, поскольку $2s_i$ есть чётное число по модулю n , следовательно, все элементы $\alpha_1, \alpha_3, \dots, \alpha_{n-1}$ чётны. На основании свойства 6 $\Delta(C)$ является разностным множеством с параметрами (n, k, λ) . Пусть $s_i, s_j \in \Delta(C)$ и $s_i - s_j \equiv m \pmod{n}$, тогда, очевидно, $s_j - s_i \equiv -m \pmod{n} \equiv m \pmod{n}$. Из последнего соотношения следует, что λ чётно и что существует ровно $\lambda/2$ пар $\{s_\xi, s_\eta\}$, таких, что $\xi < \eta$ и $s_\xi - s_\eta \equiv m \pmod{n}$. Но тогда $2s_\xi \equiv 2s_\eta \pmod{n}$ и поэтому $\lambda/2$ элементов среди $\alpha'_0, \alpha'_2, \dots, \alpha'_{n-2}$ равны 2, $k - \lambda$ элементов равны 1, а остальные — нули. ■

Перейдём к доказательству теоремы 6.

Доказательство. По условию $Q(C)$ — 2-транзитивная группа подстановок, следовательно, $LG(C)$ также 2-транзитивна, поскольку $Q(C) \subseteq LG(C)$. Отсюда на основании теоремы 2 получаем $C = c_1 \tilde{C} + c_2 J_n$, где \tilde{C} — матрица инцидентий симметричной блок-схемы. Поскольку $C, J_n \in CM_n(\mathbb{Z})$, то \tilde{C} также лежит в $CM_n(\mathbb{Z})$ (свойство 3). Кроме того, нетрудно видеть, что $Q(C) = Q(\tilde{C})$, и если теорема верна для матрицы \tilde{C} , то она верна и для матрицы C и наоборот. Поэтому далее будем считать, что C — матрица инцидентий симметричной блок-схемы с параметрами (n, k, λ) .

Рассмотрим матрицу C^2 . По лемме 2 $Q(C) \subseteq LG(C^2)$, поэтому $LG(C^2)$ — 2-транзитивная группа подстановок. На основании теоремы 2 получаем $C^2 = c_1 C' + c_2 J_n$, где C' — матрица инцидентий симметричной блок-схемы, причём по свойству 3 $C' \in CM_n(\mathbb{Z})$.

Пусть первые строки матриц C^2 и C' имеют соответственно вид $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ и $\alpha'_0, \alpha'_1, \dots, \alpha'_{n-1}$. Рассмотрим отдельно два случая:

а) $n = 2m$.

Применяя к матрице C лемму 3, видим, что элементы $\alpha_1, \alpha_3, \dots, \alpha_{n-1}$ все чётные, а среди элементов $\alpha_0, \alpha_2, \dots, \alpha_{n-2}$ ровно $k - \lambda$ нечётных. Выбирая подходящие c_1 и c_2 , можно добиться того, что $\alpha'_i = 1$ в том и только в том случае, когда α_i — нечётный элемент. Таким образом, $\Delta(C') = \{s_1, s_2, \dots, s_{k-\lambda}\}$, где все s_i являются чётными вычетами по модулю n . При $k - \lambda > 1$ это множество, как нетрудно видеть, не может быть разностным, поскольку его элементы порождают только чётные разности, однако оно обязано им быть, так как $C' \in CM_n(\mathbb{Z})$ (свойство 6). Отсюда $k - \lambda \in \{0, 1\}$, т. е. C является матрицей инцидентий тривиальной циклической блок-схемы, и поэтому существуют такие c_1 и c_2 , что $C = c_1 \hat{t}_n^s + c_2 J_n$, $0 \leq s \leq n - 1$.

б) Пусть n свободно от квадратов и нечётно. Применяя к матрице C' лемму 3, получаем, что среди элементов $\alpha_0, \alpha_1, \dots, \alpha_{n-1}$ ровно k нечётных. Как и выше, можно считать, что $\alpha'_i = 1$ в том и только в том случае, если α_i — нечётный элемент. Отсюда следует, что $\Delta(C')$ есть разностное множество с параметрами (n, k, λ) . Вернёмся к равенству $C^2 = c_1 C' + c_2 J_n$. Поскольку все элементы матрицы C^2 неотрицательны, то $c_2 > 0$. Сравнивая модули собственных значений матриц C^2 и $c_1 C' + c_2 J_n$ [9, с. 145], получаем

$$k^2 = c_1 k + c_2 n; \quad (16.1)$$

$$k - \lambda = |c_1| \sqrt{k - \lambda}. \quad (16.2)$$

Пусть $c_1 \geq 0$. Если $c_1 = 0$, то из (16.2) следует $k - \lambda = 0$, т. е. параметры k и λ — тривиальные. Если же $c_1 > 0$, то

$$c_1 = \sqrt{k - \lambda}. \quad (17)$$

Вместе с (16.1) это даёт $k^2 = k\sqrt{k - \lambda} + c_2 n$, или

$$k(k - \sqrt{k - \lambda}) = c_2 n. \quad (18)$$

Можно считать, что $k \neq 0, k - \sqrt{k - \lambda} \neq 0$, поскольку в противном случае параметры k и λ тривиальные. Пусть $d = (k, n)$. Положим $k = dx, n = dy$. В этом случае из (10) следует, что λ делится на d . Заметим, что d свободно от квадратов, поскольку n свободно от квадратов, а $k - \lambda$ на основании (16) является полным квадратом, следовательно, $k - \lambda$ делится на d^2 . Отсюда $\sqrt{k - \lambda}$ делится на d и поэтому $k - \sqrt{k - \lambda}$ делится на d . Положим $k - \sqrt{k - \lambda} = dz$. Из (3.6) теперь имеем

$$dxdz = c_2 dy, \quad (19)$$

причём $(y, d) = 1$, поскольку n свободно от квадратов. Но тогда c_2 делится на d . Положим $c_2 = dv$. Подставляя это выражение в (19), получаем $dxdz = dvdy$, или $xz = vy$. Но $(x, y) = 1$, следовательно, v делится на x . Отсюда c_2 делится на k , т. е. $c_2 \geq k$.

Перепишем (11) в виде

$$(k - \sqrt{k - \lambda})(k + \sqrt{k + \lambda}) = \lambda n. \quad (20)$$

Сравнивая (17) и (19), видим, что $\lambda \geq c_2 \geq k$. Но, с другой стороны, из определения блок-схемы $\lambda \leq k$. Поэтому $\lambda = k$, т. е. параметры k и λ тривиальные.

Пусть $c_1 < 0$. Рассуждая аналогичным образом, получим $c_1 = -\sqrt{k - \lambda}$, $k(k + \sqrt{k - \lambda}) = c_2 n$. Применяя те же рассуждения, что и выше, получим, что c_2 делится на k . Но $k + \sqrt{k - \lambda} \leq 2k$, поэтому либо $c_2 = k$, либо $k = n$. Если $k = n$, то $\lambda = k$, т. е. параметры k и λ тривиальные, поэтому рассмотрим случай $c_2 = k$. При этом $k + \sqrt{k + \lambda} = n$, и из (20) следует $k - \sqrt{k - \lambda} = \lambda$. Но, очевидно, $k - (k - \lambda) = \lambda$, поэтому $k - \lambda = \sqrt{k - \lambda}$, а это возможно, только если $k - \lambda \in \{0, 1\}$. Таким образом, при всех случаях параметры k и λ являются тривиальными. Поэтому существуют такие c_1 и c_2 , что

$$C = c_1 \hat{t}_n^s + c_2 J_n, \quad 0 \leq s \leq n - 1,$$

и теорема доказана. ■

При простом n справедливы более общие утверждения.

Теорема 7. Пусть n — простое число, $C \in CM_n(\mathbb{Z})$ и $\text{Aff}(n) \subseteq LG(C)$. Тогда $C = c_1 \hat{t}_n^s + c_2 J_n$.

Доказательство. Пусть i — примитивный элемент поля $\mathbb{Z}/n\mathbb{Z}$, а $g \in \text{Aff}(n)$ — подстановка, соответствующая элементу i , т. е. $g(x) \equiv ix \pmod{n}$. Группа $\text{Aff}(n)$ при простом n 2-транзитивна, поэтому по теореме 2 можно считать, что C — матрица инцидентий симметричной блок-схемы, следовательно, $|C| \neq 0$ [9, с. 144]. По условию $g \in LG(C)$, т. е. $\hat{g}^{-1}C\hat{g}' = C$, причём цикловые структуры подстановок g и g' совпадают [11, с. 22]. Подстановка g в силу примитивности элемента i имеет цикловую структуру $[1^{(1)}, (n-1)^{(1)}]$. Такую же структуру имеет и подстановка g' . Из равенства $\hat{g}^{-1}C\hat{g}' = C$ получаем, что матрица C содержит столбец вида $(01\dots 1)^t$ или $(10\dots 0)^t$. ■

Теорема 8. Пусть n — простое число, $C \in CM_n(\mathbb{Z})$, $C \neq c_1 \hat{t}_n^s + c_2 J_n$. Тогда

$$Q(C) = LG(C) \cap \text{Aff}(n),$$

и либо $Q(C) = G(C)$, либо $G(C) = \langle t_n \rangle$.

Доказательство. Пусть $H = LG(C) \cap \text{Aff}(n)$, $x \in H$ и $\hat{x}^{-1}C\hat{y} = C$. Рассмотрим подстановки $z_1 = x^{-1}t_n x$ и $z_2 = y^{-1}t_n y$. Нетрудно видеть, что $\hat{z}_1^{-1}C\hat{z}_2 = C$. Но $z_1 = t_n^k$, поскольку $x \in \text{Aff}(n)$, а $\text{Aff}(n)$ является нормализатором группы $\langle t_n \rangle$. На основании свойства 5 получаем, что $y^{-1}t_n y = t_n^k$ (в противном случае матрица C имеет равные столбцы, т. е. $C = cJ_n$, так как n простое). Но тогда нетрудно показать, что $y = xt_n^m$, т. е. $y \in H$, поскольку $\langle t_n \rangle \subset H$. Отсюда по определению группы $Q(C)$ получаем $H \subset Q(C)$. С другой стороны, по теореме 6 $Q(C)$ при данных предположениях не является 2-транзитивной, поэтому по теореме 1 $Q(C) \subset \text{Aff}(n)$. Отсюда $Q(C) = H$.

Как отмечено выше, $\text{Aff}_0(n) = \langle g \rangle$, следовательно, $Q_0(C) = \langle g^k \rangle$, $G_0(C) = \langle g^{km} \rangle$ для некоторых натуральных k и m , поскольку

$$G_0(C) \subseteq Q_0(C) \subset \text{Aff}_0(n).$$

Если $m = 1$, то $g^{km} = g$, т. е. $G(C) = Q(C) = \langle t_n, g^k \rangle$. Пусть $\langle g^{km} \rangle \subset \langle g \rangle$. Имеем равенство

$$\hat{g}^{-k}C\hat{g}' = C. \quad (21)$$

При этом, как мы уже показали выше, $g' = g^k t_n^l$. Из (21) получаем $\hat{g}^{-km}C\hat{g}'^m = C$. Но $g^{km} \in G(C)$, поэтому $\hat{g}^{-km}C\hat{g}'^m = C$. Сравнивая последние два равенства, по свойству 5 получаем $g'^m = g^{km}$, или $(g^k t_n^l)^m = g^{km}$. Но тогда подстановки $g^k t_n^l$ и g^{km} перестановочны, следовательно, их коммутатор $[g^k t_n^l, g^{km}]$ равен e_n . Отсюда по свойству сложных коммутаторов

$$[g^k t_n^l, g^{km}] = [g^k, g^{km}][g^k, g^{km}, t_n^l][t_n^l, g^{km}] = [t_n^l, g^{km}] = e_n.$$

Последнее равенство верно, только если $l \equiv 0 \pmod{n}$ либо $g^{km} = e_n$. В первом случае $g' = g^k$, т. е. $G_0(C) = \langle g^k \rangle$, что противоречит нашему предположению. Во втором случае $G_0(C) = \langle e_n \rangle$, т. е. $G(C) = \langle t_n \rangle$. ■

Пример 1. Пусть C — матрица инциденций зингерской блок-схемы с параметрами $(7, 3, 1)$, т. е.

$$C = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

В этом случае $LG(C)$ 2-транзитивна, $Q(C) = \langle t_7, g^2 \rangle$, где $g(x) \equiv 3x \pmod{7}$, $G(C) = \langle t_7 \rangle$. Таким образом, $G(C) \subset Q(C) \subset LG(C)$.

ЛИТЕРАТУРА

1. Егоров В. Н., Марков А. И. О гипотезе Адама для графов с циркулянтными матрицами смежности вершин // ДАН СССР. 1979. Т. 249. № 3. С. 529–532.
2. Давыдов Э. Г. О симметрии графов // Вопросы кибернетики. М., 1973. С. 26–49.

3. *Chao C.* On groups and graphs // TMAS. 1965. V. 118. No. 6. P. 488–497.
4. *Feit W.* Automorphisms of symmetric balanced incomplete block designs // Math. Z. 1970. No. 118. P. 40–49.
5. *Feit W.* On symmetric balanced incomplete block designs with doubly transitive automorphism groups // J. Combin. Theory. 1973. V. 14. No. 2. P. 221–247.
6. *Huang Q., Meng J.* On the isomorphism and automorphism groups of circulants // Grafs Combin. 1996. V. 12. P. 179–187.
7. *Тараканов В. Е.* Группы автоморфизмов циркулянтов и присоединенные матрицы графов // Математические заметки. 1999. Т. 65. Вып. 3. С. 402–411.
8. *Wielandt H.* Finite permutation groups. New York; London: Academic Press, 1964.
9. *Холл М.* Комбинаторика. М.: Мир, 1970.
10. *Adam A.* Research problem 2–10 // J. Combin. Theory. 1967. V. 2. P. 393.
11. *Dembowski P.* Finite geometries. Berlin and New York: Springer Verlag, 1968.

**О ЗНАЧЕНИЯХ УРОВНЯ АФФИННОСТИ
ДЛЯ ПОЧТИ ВСЕХ БУЛЕВЫХ ФУНКЦИЙ¹**

О. А. Логачев

*Институт проблем информационной безопасности,
Московский государственный университет им. М. В. Ломоносова, г. Москва, Россия*

E-mail: logol@iisi.msu.ru

Рассматривается асимптотическое поведение значений параметра булевой функции, называемого уровнем (обобщенным уровнем) аффинности. Показано, что асимптотически при $n \rightarrow \infty$ для почти всех булевых функций от n переменных значения уровня (обобщенного уровня) аффинности принадлежат сегменту $[n - \log_2 n, n - \log_2 n + 1]$.

Ключевые слова: *уровень аффинности, обобщенный уровень аффинности, системы булевых уравнений, криптография.*

Введение

Один из возможных методов линеаризации систем булевых уравнений связан с частичным опробованием некоторого подмножества переменных и сведением исходной системы к линейному следствию. В работах [1, 2] был рассмотрен параметр, называемый уровнем (обобщенным уровнем) аффинности и характеризующий эффективность такой линеаризации. Различные свойства уровня (обобщенного уровня) аффинности изучались в работах [3–6]. Систематическое исследование этого параметра было проведено в [7].

1. Основные понятия и определения

Пусть \mathbb{F}_2 — поле из двух элементов, $V_n = \mathbb{F}_2^n$ — линейное пространство векторов (наборов) длины n над полем \mathbb{F}_2 . Вес Хэмминга вектора $\mathbf{x} = (x_1, \dots, x_n) \in V_n$ определяется как $\text{wt}(\mathbf{x}) = \sum_{i=1}^n x_i$. Пусть L — некоторое подпространство пространства V_n , $\dim L = r$ и $\mathbf{v} \in V_n$. Смежный класс $\pi = L \oplus \mathbf{v}$, где \oplus — сложение по mod 2, будем называть плоскостью размерности r пространства V_n и писать $\dim \pi = r$. Будем считать, что $\dim \pi = -1$, если $\pi = \emptyset$, и $\dim \pi = 0$, если $\pi = \{\mathbf{u}\}$, $\mathbf{u} \in V_n$. Множество всех плоскостей пространства V_n (включая пустую плоскость) обозначим через $\mathcal{P}(V_n)$.

Через \mathcal{F}_n будем обозначать множество всех булевых функций от n переменных. Любая булева функция $f \in \mathcal{F}_n$ может быть представлена в полиномиальной форме

$$f(\mathbf{x}) = f(x_1, \dots, x_n) = \bigoplus_{\mathbf{u} \in V_n} g(\mathbf{u})\mathbf{x}^{\mathbf{u}} = \bigoplus_{(u_1, \dots, u_n) \in V_n} g(u_1, \dots, u_n)x_1^{u_1} \cdot \dots \cdot x_n^{u_n}, \quad (1)$$

называемой алгебраической нормальной формой (АНФ) этой функции, где $g \in \mathcal{F}_n$ и для любого $1 \leq i \leq n$

$$x_i^{u_i} = \begin{cases} 1, & u_i = 0, \\ x_i, & u_i = 1. \end{cases}$$

¹Работа поддержана РФФИ (проекты 09-01-00653-а, 10-01-00475-а).

Алгебраической степени функции f , обозначаемой $\deg f$, является максимальное значение $\text{wt}(\mathbf{u})$ по тем $\mathbf{u} \in V_n$, для которых $g(\mathbf{u}) = 1$. Через $\deg(f, x_i)$ обозначается максимальное значение $\text{wt}(\mathbf{u})$ по тем $\mathbf{u} \in V_n$, для которых $g(\mathbf{u}) = 1$ и $u_i = 1$. Обозначим через \mathcal{A}_n множество аффинных функций, то есть $\mathcal{A}_n = \{f \in \mathcal{F}_n : \deg(f) \leq 1\}$.

Пусть $k \leq n$, $1 \leq i_1 < \dots < i_k \leq n$ и $\mathbf{b} = (b_1, \dots, b_k) \in V_k$. Для булевой функции $f \in \mathcal{F}_n$ обозначим через $f_{i_1, \dots, i_k}^{b_1, \dots, b_k}$ булеву функцию из \mathcal{F}_{n-k} , полученную из f фиксацией переменных $x_{i_1} = b_1, \dots, x_{i_k} = b_k$ и называемую подфункцией функции f .

Булева функция $f \in \mathcal{F}_n$ называется k -аффинной, если существуют наборы $1 \leq i_1 < \dots < i_k \leq n$, $\mathbf{b} = (b_1, \dots, b_k) \in V_k$, такие, что $f_{i_1, \dots, i_k}^{b_1, \dots, b_k} \in \mathcal{A}_{n-k}$.

Определение 1 [2]. Уровнем аффинности $\text{la}(f)$ булевой функции f из \mathcal{F}_n называется минимальное число k , для которого функция f является k -аффинной.

Пусть $f \in \mathcal{F}_n$ и S — произвольное подмножество пространства V_n . Ограничением (сужением) $f|_S$ функции f на множество S будем называть отображение $f' : S \mapsto \mathbb{F}_2$, такое, что $f'(\mathbf{x}) = f|_S(\mathbf{x}) = f(\mathbf{x})$ для всех $\mathbf{x} \in S$.

Пусть $f \in \mathcal{F}_n$. Плоскость $\pi \in \mathcal{P}(V_n) \setminus \{\emptyset\}$ называется локальной аффинностью булевой функции f , если существует аффинная функция $l \in \mathcal{A}_n$, такая, что $f|_\pi = l|_\pi$. Обозначим

$$\tilde{\mathcal{P}}_f(V_n) = \{\pi \in \mathcal{P}(V_n) \setminus \{\emptyset\} : \exists l \in \mathcal{A}_n (f|_\pi = l|_\pi)\}$$

— совокупность локальных аффинностей функции f .

Определение 2 [6]. Обобщенным уровнем аффинности $\text{La}(f)$ функции $f \in \mathcal{F}_n$ называется неотрицательное число

$$\text{La}(f) = n - \max_{\pi \in \tilde{\mathcal{P}}_f(V_n)} \dim \pi.$$

Замечание 1. При всей близости понятий, введенных в определениях 1 и 2, имеется существенное их различие. Обобщенный уровень аффинности, в отличие от уровня аффинности, является аффинным инвариантом, то есть инвариантом относительно действия на функцию полной аффинной группы (см. [8]).

Замечание 2. Очевидно, что

$$\text{La}(f) \leq \text{la}(f) \tag{2}$$

для произвольной булевой функции f из \mathcal{F}_n .

2. Вспомогательные результаты

Асимптотическое поведение уровня (обобщенного уровня) аффинности исследовалось в работах [5–7].

Справедлива следующая асимптотическая нижняя оценка для обобщенного уровня аффинности булевых функций.

Теорема 1 [6]. Пусть $\alpha \in \mathbb{R}$, $\alpha > 1$ — фиксированная константа. Тогда асимптотически при $n \rightarrow \infty$ для почти всех булевых функций f из \mathcal{F}_n справедливо неравенство

$$\text{La}(f) \geq n - \alpha \log_2 n.$$

Следствие 1 [6]. Пусть $\alpha \in \mathbb{R}$, $\alpha > 1$ — фиксированная константа. Тогда асимптотически при $n \rightarrow \infty$ для почти всех булевых функций f из \mathcal{F}_n справедливо неравенство

$$\text{la}(f) \geq n - \alpha \log_2(n).$$

Сформулируем утверждение, непосредственно вытекающее из следствия 1 в силу условий, накладываемых на константу α .

Утверждение 1. Асимптотически при $n \rightarrow \infty$ для почти всех булевых функций f из \mathcal{F}_n справедливо неравенство

$$\text{la}(f) \geq n - \log_2(n). \quad (3)$$

3. Основной результат

Обозначим через $\mathcal{M}_{n,k}$, $1 \leq k \leq n$, множество функций из \mathcal{F}_n , для которых выполняется неравенство $\text{la}(f) \leq k$, и $\overline{\mathcal{M}}_{n,k} = \mathcal{F}_n \setminus \mathcal{M}_{n,k}$. Соответствующую долю множества $\mathcal{M}_{n,k}$ в \mathcal{F}_n обозначим $\delta_{n,k} = \text{card } \mathcal{M}_{n,k} / 2^{2^n}$.

Справедливо следующее утверждение.

Теорема 2. Асимптотически при $n \rightarrow \infty$ для почти всех булевых функций f из \mathcal{F}_n справедливо неравенство

$$\text{la}(f) \leq n - \log_2(n) + 1.$$

Доказательство. Пусть $f \in \mathcal{F}_n$ и $1 \leq k \leq n$. Рассмотрим разложение f в сумму ее подфункций по переменным x_1, \dots, x_k вида

$$f(x_1, \dots, x_n) = \bigoplus (x_1 \oplus u_1 \oplus 1) \dots (x_k \oplus u_k \oplus 1) f_{1, \dots, k}^{u_1, \dots, u_k}(x_{k+1}, \dots, x_n). \quad (4)$$

Если $f \in \overline{\mathcal{M}}_{n,k}$, то необходимо, чтобы все подфункции $f_{1, \dots, k}^{u_1, \dots, u_k}$ из разложения (4) имели алгебраическую степень не менее 2, то есть не являлись бы аффинными функциями из \mathcal{A}_{n-k} . Следовательно,

$$\text{card } \overline{\mathcal{M}}_{n,k} \leq (2^{2^{n-k}} - 2^{n-k+1})^{2^k}$$

и

$$\text{card } \mathcal{M}_{n,k} \geq 2^{2^n} - (2^{2^{n-k}} - 2^{n-k+1})^{2^k}.$$

Тогда

$$\delta_{n,k} \geq 1 - \left(\frac{2^{2^{n-k}} - 2^{n-k+1}}{2^{2^{n-k}}} \right)^{2^k} = 1 - \alpha_{n,k}. \quad (5)$$

Положим $k = n - \log_2 n + 1$ и устремим $n \rightarrow \infty$. Для величины $\alpha_{n, n - \log_2 n + 1}$ справедлива следующая цепочка равенств:

$$\begin{aligned} \alpha_{n, n - \log_2 n + 1} &= \left(1 - \frac{2^{n - (n - \log_2 n + 1) + 1}}{2^{2^{n - (n - \log_2 n + 1)}}} \right)^{2^{n - \log_2 n + 1}} = \\ &= \left(1 - \frac{n}{2^{n/2}} \right)^{\frac{2^{n+1}}{n}} = \left(\left(1 + \frac{-1}{\frac{2^{n/2}}{n}} \right)^{\frac{2^{n/2}}{n}} \right)^{2^{n/2+1}}. \end{aligned} \quad (6)$$

Воспользовавшись известным соотношением

$$\lim_{t \rightarrow \infty} \left(1 + \frac{d}{t} \right)^t = e^d, \quad d \in \mathbb{R}, \quad (7)$$

совместно с (6), получаем

$$\lim_{n \rightarrow \infty} \alpha_{n, n - \log_2 n + 1} = 0. \quad (8)$$

Следовательно, соотношения (5) и (8) дают

$$\lim_{n \rightarrow \infty} \delta_{n, n - \log_2 n + 1} = 1, \quad (9)$$

что и доказывает утверждение теоремы. ■

Следствие 2. Асимптотически при $n \rightarrow \infty$ для почти всех булевых функций f из \mathcal{F}_n справедливо неравенство

$$\text{La}(f) \leq n - \log_2 n + 1.$$

Доказательство. Непосредственно следует из утверждения теоремы 2 и неравенства (2). ■

Поскольку для любой функции f из \mathcal{F}_n значения $\text{la}(f)$ и $\text{La}(f)$ являются неотрицательными целыми числами, то утверждения следствий 1, 2, утверждения 1 и теорем 1, 2 могут быть объединены следующим образом.

Теорема 3. Асимптотически при $n \rightarrow \infty$ для почти всех булевых функций f из \mathcal{F}_n выполнены условия

$$\begin{aligned} 1) \quad n - \lfloor \log_2 n \rfloor &\leq \text{la}(f) \leq n - \lceil \log_2 n \rceil + 1, \\ 2) \quad n - \lfloor \log_2 n \rfloor &\leq \text{La}(f) \leq n - \lceil \log_2 n \rceil + 1. \end{aligned} \quad (10)$$

Легко видеть, что условия (10) выделяют два возможных случая $n = 2^b$ и $n \neq 2^b$. В случае, когда $n = 2^b$, для почти всех булевых функций имеется два возможных значения уровня (обобщенного уровня) аффинности: $n - b$, $n - b + 1$. А в случае, когда n не является степенью 2, для почти всех функций из \mathcal{F}_n имеется одно возможное значение для уровня (обобщенного уровня) аффинности: $n - \lfloor \log_2 n \rfloor = n - \lceil \log_2 n \rceil + 1$.

Замечание 3. Воспользовавшись соотношениями (6) и (7), можно легко показать, что в случае $n = 2^b$

$$\lim_{b \rightarrow \infty} \delta_{n, n-b} \geq 1 - e^{-2}.$$

ЛИТЕРАТУРА

1. Логачев О. А., Сальников А. А., Яценко В. В. Корреляционная иммунность и реальная секретность // Математика и безопасность информационных технологий. М.: МЦНМО, 2004. С. 165–170.
2. Логачев О. А., Сальников А. А., Яценко В. В. Комбинирующие k -аффинные функции // Математика и безопасность информационных технологий. М.: МЦНМО, 2004. С. 176–178.
3. Буряков М. Л., Логачев О. А. О распределении уровня аффинности на множестве булевых функций // Математика и безопасность информационных технологий. М.: МЦНМО, 2005. С. 141–146.
4. Буряков М. Л., Логачев О. А. Об уровне аффинности булевых функций // Дискретная математика. 2005. Т. 17. Вып. 4. С. 98–107.
5. Логачев О. А. Нижняя оценка уровня аффинности для почти всех булевых функций // Там же. 2008. Т. 20. Вып. 4. С. 85–88.
6. Буряков М. Л. Асимптотические оценки уровня аффинности для почти всех булевых функций // Там же. 2008. Т. 20. Вып. 3. С. 73–79.

7. Буряков М. Л. Алгебраические, комбинаторные и криптографические свойства параметров аффинных ограничений булевых функций: дис. ... канд. физ.-мат. наук. М., 2007.
8. Логачев О. А., Сальников А. А., Яценко В. В. Булевы функции в теории кодирования и криптологии. М.: МЦНМО, 2004.

**ТОЧЕЧНЫЕ И СИЛЬНО ТОЧЕЧНЫЕ ФУНКЦИИ
НА ПОЛУРЕШЁТКЕ¹**

Н. Г. Парватов

*Томский государственный университет, г. Томск, Россия***E-mail:** parvatov@mail.tsu.ru

Рассматриваются основные классы квазимонотонных функций на полурешётке, представляющие интерес в связи с проблемами синтеза асинхронных дискретных управляющих систем. Подробно рассматриваются классы точечных и сильно точечных функций на полурешётке.

Ключевые слова: *полурешётка, квазимонотонные функции, слабо существенные квазимонотонные функции, монотонные функции, точечные функции, минимальные точечные функции, сильно точечные функции.*

Введение

Квазимонотонные функции на верхней полурешётке введены Г. П. Агибаловым для описания динамического поведения дискретных управляющих систем, асинхронно изменяющиеся (в разной степени определённые) состояния которых приводят к состязаниям [1]. В его монографии [2] состояния таких систем, упорядоченные по степени неопределённости, рассматриваются как элементы верхней полурешётки. При этом функции состояний и выходов системы оказываются монотонными, поскольку при уточнении входного состояния внутренние и выходные состояния системы могут изменяться лишь в сторону уточнения. Среди монотонных функций выделяются те, которые не допускают дальнейшего монотонного уточнения. Их называют *минимальными точечными*. Также в отдельный класс выделяют функции, не обязательно монотонные, но допускающие монотонное уточнение. Их называют *квазимонотонными* и используют при формулировании задачи синтеза, а также на начальных этапах её решения. Подобная задача может состоять в необходимости создания управляющей системы, у которой функции состояний и выходов уточняют заранее заданные квазимонотонные функции. Поскольку квазимонотонная функция всегда уточняется некоторой минимальной точечной, задачи синтеза асинхронных дискретных управляющих систем сводятся к задачам реализации минимальных точечных функций в том или ином базисе, в зависимости от ситуации квазимонотонном, монотонном или минимальном точечном. В связи со сказанным заслуживают изучения квазимонотонные, монотонные и минимальные точечные функции на полурешётках. Для них актуальны проблемы полноты и выразимости, проблемы формульного представления, проблемы эффективного задания их классов и другие. Проблемы эффективного задания классов квазимонотонных и монотонных функций рассматривались автором в работе [3], проблемы полноты и выразимости — в [4, 5]. В данной работе наряду с квазимонотонными, монотонными и минимальными точечными функциями изучаются столь же важные своими приложениями к синтезу дискретных управляющих систем точечные и сильно

¹Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П11010).

точечные функции, рассматриваются проблемы эффективного задания их классов, а также проблемы формульного представления функций в некоторых из классов.

1. Полурешётки и полурешёточные функции

Верхняя полурешётка. Пусть в конечном множестве L , упорядоченном отношением \leq , для любых элементов a и b имеется точная верхняя грань $a + b$, а точная нижняя грань $a \cdot b$ существует не для любых элементов a и b . Иными словами, множество L вместе с указанным упорядочением является верхней полурешёткой, но не решёткой [6]. В соответствии с [2] полурешётка называется *точечной*, если в ней каждый элемент является суммой некоторых минимальных элементов.

Наибольший элемент верхней полурешётки будем обозначать так: \top . Удобно верхнюю полурешётку L (это касается и любой другой верхней полурешётки) считать вложенной в решётку $L' = L \cup \{\perp\}$ с наименьшим элементом \perp . Это позволяет пользоваться произведениями ab для любых элементов a и b из решётки L' (в частности, для любых элементов a и b из полурешётки L). В этом случае отсутствие произведения в полурешётке L означает, что оно принимает наименьшее значение \perp в решётке L' . Рассмотрим далее некоторые важные конструкции полурешёток, представляющие значительный интерес в связи с приложениями к синтезу дискретных управляющих систем с заданным динамическим поведением.

Полурешётки наборов и функций. Отношение порядка \leq , определённое в полурешётке L , переносится на наборы из множества L^n естественным образом — покомпонентно, так, что выполнение неравенства $a \leq b$ для наборов $a = (a_1, \dots, a_n)$ и $b = (b_1, \dots, b_n)$ означает выполнение всевозможных покомпонентных неравенств $a_i \leq b_i$ при $1 \leq i \leq n$. Таким образом множество L^n становится полурешёткой с покомпонентными сложением и умножением.

Полурешёточное упорядочение \leq множества L переносится и на функции $f : L^n \rightarrow L$, множество которых при всевозможных натуральных n обозначается через P_L . При этом неравенство $f \leq g$ для функций f и g , зависящих от одинакового числа переменных, означает, что для любого набора a значений их переменных выполняется неравенство $f(a) \leq g(a)$. В этом случае функция f называется *минорантой* функции g , а функция g — *мажорантой* функции f . Множество функций из P_L , зависящих от n переменных, становится таким образом полурешёткой, в которой сумма $f + g$ функций f и g определена соотношением

$$(f + g)(x) = f(x) + g(x),$$

а произведение $f \cdot g$ определено, если отличны от \perp (то есть определены в L) все значения

$$(f \cdot g)(x) = f(x) \cdot g(x),$$

где x — произвольный набор из множества L^n .

Полурешётка подмножеств. Для заданного конечного множества E полурешётка всех его непустых подмножеств, упорядоченных включением, обозначается через \tilde{E} . Точные верхние и нижние грани её элементов — подмножеств множества E суть объединения и пересечения этих подмножеств. Интересуясь данной полурешёткой с точностью до изоморфизма (то есть с точностью до обозначений), отождествим одноэлементные подмножества с их элементами, после чего будем использовать (из соображений единообразия обозначений) знаки $\leq, +, \cdot, \top$ вместо \subseteq, \cup, \cap, E для обозначения её упорядочения, операций взятия точной верхней и нижней граней, наибольше-

го элемента. Таким образом, множество E оказывается вложенным в полурешётку \tilde{E} в качестве множества её минимальных элементов. Важной является также

Полурешётка интервалов решётки. Пусть теперь конечное непустое множество E является решёткой с упорядочением \preceq , а также с операциями \vee и \wedge (называемыми дизъюнкцией и конъюнкцией) для взятия точных верхних и нижних граней. Для произвольных элементов a и b этой решётки, таких, что $a \preceq b$, подмножества

$$[a, b] = \{x : a \preceq x \preceq b\}$$

называются её *интервалами*. Их множество, обозначаемое далее $\text{in}(E, \preceq)$, упорядоченное включением, является верхней полурешёткой. Это — *полурешётка интервалов решётки* (E, \preceq) . Придерживаясь введённой традиции, для обозначения упорядочения в этой полурешётке будем использовать знак \leq (вместо включения \subseteq), а операции взятия точных верхних и нижних граней будем обозначать соответственно суммой $+$ и произведением \cdot (хотя произведения совпадают с пересечениями, когда определены). Одноэлементные интервалы будем отождествлять с их элементами (так, что для любого элемента a из E выполняется равенство $[a, a] = a$) и тогда множество E будем считать вложенным в полурешётку $(\text{in}(E, \preceq), \leq)$ интервалов решётки (E, \preceq) в качестве множества минимальных элементов. Таким образом, в этой полурешётке для интервалов $[a, b]$ и $[c, d]$ неравенство $[a, b] \leq [c, d]$ означает совместное выполнение соотношений $c \preceq a$ и $b \preceq d$. Суммой $[a, b] + [c, d]$ является интервал $[a \wedge c, b \vee d]$, а произведение $[a, b] \cdot [c, d]$ определено лишь при выполнении неравенств $a \vee c \preceq b \wedge d$ и совпадает тогда с интервалом $[a \vee c, b \wedge d]$.

Отношение порядка \preceq , а также операции дизъюнкции \vee и конъюнкции \wedge для взятия точных верхних и нижних граней, определённые в решётке (E, \preceq) , переносятся на её интервалы покомпонентно так, что для интервалов $[a, b]$ и $[c, d]$ из множества $\text{in}(E, \preceq)$ неравенство $[a, b] \preceq [c, d]$ означает выполнение соотношений $a \preceq c$ и $b \preceq d$, а конъюнкцией и дизъюнкцией тех же интервалов являются соответствующие интервалы $[a \wedge c, b \wedge d]$ и $[a \vee c, b \vee d]$. Таким путём множество $\text{in}(E, \preceq)$ интервалов решётки (E, \preceq) становится ещё и решёткой с упорядочением \preceq , а также с дизъюнкцией \vee и конъюнкцией \wedge для взятия точных верхних и нижних граней. Это — *решётка интервалов решётки* (E, \preceq) . Очевидно, всякая решётка вложена в качестве подрешётки в свою решётку интервалов, а та вложена изоморфно во вторую декартову степень первой.

Итак, на множестве $\text{in}(E, \preceq)$ интервалов решётки (E, \preceq) оказываются определёнными две алгебраические системы, которые следует тщательно различать друг от друга. Эти системы суть полурешётка интервалов с упорядочением \leq , сложением и умножением для взятия точных верхних и нижних граней, а также решётка интервалов с упорядочением \preceq , дизъюнкцией и конъюнкцией для взятия точных верхних и нижних граней.

В наиболее важном для приложений случае множество E совпадает с множеством E_k чисел $0, \dots, k-1$, упорядоченных отношением \preceq линейно, так: $0 \preceq 1 \preceq \dots \preceq k-1$. Полурешётку интервалов (как и их множество) в этом случае договоримся обозначать через \hat{E}_k . Отметим также, что полурешётка интервалов линейно упорядоченного двухэлементного множества E совпадает с полурешёткой \tilde{E} его непустых подмножеств. В частности, $\hat{E}_2 = \tilde{E}_2 = \{0, 1, \top\}$.

2. Монотонные, квазимонотонные и слабо существенные функции на полурешётке

Наиболее важными для приложений являются классы монотонных и квазимонотонных, а также слабо существенных квазимонотонных и монотонных функций.

Функции из P_L , сохраняющие определённое в верхней полурешётке L отношение порядка \leq , называются *монотонными*. Они составляют клон (замкнутый операциями суперпозиции из [7–9] класс, содержащий все селекторные функции, тождественно равные некоторому своему аргументу), обозначаемый через M_L . Функция из P_L , имеющая монотонную миноранту, называется *квазимонотонной*. Клон квазимонотонных функций на полурешётке L обозначается через Q_L . Функции из P_L , обладающие монотонными минорантами, существенно зависящими не более чем от одной переменной, называются *слабо существенными квазимонотонными* (на полурешётке L). Их клон обозначается через Φ_L . Проблемы эффективного описания клонов M_L, Q_L, Φ_L и $M_L \cap \Phi_L$ посредством предикатов изучались в работе [3], где для каждого из этих клонов найдено так называемое и-описание — множество предикатов, порождающее все инвариантные предикаты клона при помощи конъюнкции, проектирования и подстановок переменных. Одновременно с этим установлены условия, при которых частичная функция имеет продолжение в этих клонах. В том числе в [3] установлена конечная порождаемость этих клонов. Так, конечная порождаемость клонов M_L и Q_L объясняется с использованием результатов работы [10] наличием в них мажоритарной функции

$$m(x_1, \dots, x_{q(L)+1}) = \prod_{i=1}^{q(L)+1} (x_1 + \dots + x_{i-1} + x_{i+1} + \dots + x_{q(L)+1}),$$

зависящей от $(q(L)+1)$ переменных, где $q(L)$ — максимальная мощность подмножества полурешётки L , не ограниченного снизу и минимального по включению с этим свойством. (Пользуясь случаем, упомянем работу [11], ключевую о клонах с мажоритарной функцией.) Конечная порождаемость остальных клонов объясняется с использованием критериев из [12, 13]. В [4] решены проблемы полноты в классах квазимонотонных и монотонных функций на полурешётке \tilde{E}_2 , а также проблема выразимости минимальных точечных функций в классе монотонных функций на той же полурешётке. В [5] решена проблема полноты в классе квазимонотонных функций на произвольной конечной полурешётке при суперпозиции со слабо существенными квазимонотонными функциями.

3. Точечные функции

Важный своими приложениями класс составляют рассматриваемые далее точечные функции на полурешётке.

Точечность. Для любой n -местной монотонной функции f из класса M_L и любого набора a из множества L^n выполняется соотношение

$$f(a) \geq \sum f(a'), \quad (1)$$

где суммирование ведётся по всем минимальным наборам a' полурешётки L^n , таким, что $a' \leq a$. Если для какого-то набора a записанное неравенство выполняется строго, то говорят о *состязании* функции f на этом наборе. В связи с проблемой синтеза дискретных управляющих систем с динамическим поведением представляют интерес функции без состязаний. Такие функции в работах Г. П. Агибалова названы *точечными*, и мы будем придерживаться этой терминологии. Представляют интерес также

точечные функции, сохраняющие множество $E = \min(L, \leq)$ минимальных элементов полурешётки L , то есть принимающие значения из множества E на наборах, составленных из элементов этого множества. Такие функции называются *минимальными точечными*. Они являются минимальными по отношению \leq среди точечных функций, что согласуется с их названием, а также являются минимальными среди монотонных и среди квазимонотонных функций. Договоримся обозначать через T_L класс точечных функций и через $\min T_L$ — класс минимальных точечных функций на полурешётке L .

Точечная функция из T_L однозначно определяется своими значениями на минимальных наборах области определения, то есть на наборах из множества E^n , где n — число аргументов функции. При этом значения n -местной функции f из T_L можно найти, зная значения её ограничения $f' : E^n \rightarrow L$ (такого, что $f'(a) = f(a)$ для любого набора a из множества E^n). Это можно сделать, воспользовавшись методом точечного продолжения из [2], по формуле (1), заменив в ней неравенство равенством. В описанной ситуации функция f из T_L называется *точечным продолжением* функции f' . Эти две функции однозначно определяют друг друга, и для них принято использовать одинаковые обозначения (опуская штрих в записи f'). В частности, одними и теми же буквами обозначаются минимальные точечные функции из множества $\min T_L$ и их ограничения из множества P_E .

Точечные продолжения функций многозначной логики уже возникали в нашем рассмотрении при определении решётки интервалов, когда операции дизъюнкции и конъюнкции, сначала определённые в решётке (E, \preceq) , впоследствии были продолжены на полурешётку $\text{in}(E, \preceq)$, причём точечным образом, что легко проверяется. Эти операции, как и любые другие, определённые на множестве E , продолжают методом точечного расширения на любую полурешётку с множеством минимальных элементов E , например на полурешётку \tilde{E} , а не только на полурешётку интервалов. Докажем следующую теорему.

Теорема 1. Для любого натурального n множество n -местных точечных функций на полурешётке L замкнуто операцией сложения, то есть является верхней полурешёткой с упорядочением \leq . Эта полурешётка точечная, если полурешётка L точечная.

Доказательство. Обозначим через h сумму n -местных точечных функций f и g . Тогда для любого набора a из множества L^n выполняются равенства

$$h(a) = f(a) + g(a) = \sum f(x) + \sum g(x) = \sum (f(x) + g(x)) = \sum h(x),$$

в которых каждое суммирование ведётся по всем наборам x из E^n , таким, что $x \leq a$. При этом первое и последнее равенства имеют место в силу определения функции h , второе — в силу точечности функций f и g , третье — в силу коммутативности сложения в полурешётке. Таким образом, n -местные точечные функции действительно составляют верхнюю полурешётку. Докажем, что эта полурешётка точечная, предположив, что точечной является полурешётка L . Иными словами, следует доказать, что для любой n -местной функции f из рассматриваемой полурешётки имеет место равенство

$$f = \sum g,$$

где суммирование ведётся по всем n -местным минимальным точечным функциям g , таким, что $g \leq f$. Записанное равенство равносильно системе равенств $f(a) = \sum g(a)$ с теми же функциями g , выполняющихся для всех наборов a из множества L^n . В силу точечности функции f и функции $\sum g$ достаточно проверить выполнение этих равенств лишь на наборах a из множества E^n . Но такие равенства действительно имеют

место, поскольку полурешётка L точечная и для любого набора a из множества E^n среди значений $g(a)$ присутствуют всевозможные элементы из множества E , такие, что $g(a) \leq f(a)$. ■

Тесты точечности. Установим теперь тесты точечности, чтобы в дальнейшем с их использованием получить быстрый алгоритм (схему из функциональных элементов) распознавания точечности заданной векторно функции и построить пример замкнутого класса точечных функций.

Отметим, что класс точечных функций инвариантен — замкнут операциями подстановки констант, введения и удаления фиктивных переменных. Несложная проверка этого будет выполнена позднее — при доказательстве леммы 1. Именно это свойство (в несколько усиленном виде) позволяет получить далее тест точечности. Понадобятся следующие обозначения. Для n -местной функции f , натурального числа m , такого, что $1 \leq m \leq n$, и набора $a = (a_1, \dots, a_m)$ из множества L^m обозначим через f_a и f^a соответствующие функции

$$f(a_1, \dots, a_m, x_{m+1}, \dots, x_n) \text{ и } f(x_1, \dots, x_{n-m}, a_1, \dots, a_m),$$

полученные, как видно, последовательной подстановкой констант a_1, \dots, a_m на места первых и соответственно последних m переменных функции f . Имеет место

Лемма 1. Пусть f — n -местная функция из P_L и $1 \leq m < n$. Функция f тогда и только тогда точечная, когда точечными являются функции f_x и f^b для всех наборов x из E^m и b из L^{n-m} .

В этой лемме по-прежнему $E = \min(L)$ — множество минимальных элементов полурешётки L .

Доказательство. Необходимость следует из нижеприведённых соотношений, имеющих место для точечной функции f и для любых наборов x из множества L^m (в частности, из множества E^m) и наборов b из множества L^{n-m} :

$$f_x(b) = f(xb) = \sum_{x'b'} f(x'b') = \sum_{b'} \sum_{x'} f(x'b') = \sum_{b'} f(xb') = \sum_{b'} f_x(y),$$

$$f^b(x) = f(xb) = \sum_{x'b'} f(x'b') = \sum_{x'} \sum_{b'} f(x'b') = \sum_{x'} f(x'b) = \sum_{x'} f^b(x),$$

где суммы вычисляются по наборам x' из E^m и b' из E^{n-m} , таким, что $x' \leq x$ и $b' \leq b$ (и тогда по наборам $x'b'$ из E^n , таким, что $x'b' \leq xb$).

Достаточность следует из следующего соотношения, имеющего место при любых a из L^m и b из L^{n-m} :

$$f(ab) = f^b(a) = \sum_x f^b(x) = \sum_x f_x(b) = \sum_x \sum_y f_x(y) = \sum_x \sum_y f(xy) = \sum_{xy} f(xy),$$

где суммы вычисляются по наборам x из E^m и y из E^{n-m} , таким, что $x \leq a$ и $y \leq b$ (и тогда по наборам xy из E^n , таким, что $xy \leq ab$). При этом первое, третье, пятое и шестое равенства очевидны, а второе и четвёртое выполняются для точечных функций f^b и f_x . Этого достаточно для завершения доказательства. ■

Взяв в лемме $m = n - 1$, получаем следующий тест точечности.

Теорема 2. Функция f из P_L , зависящая от $n > 1$ переменных, тогда и только тогда точечная, когда точечными являются функции f_x и f^b для всевозможных наборов x из множества E^{n-1} и элементов b из полурешётки L .

Отсюда индукцией по n получается

Следствие 1. Функция из P_L тогда и только тогда точечная, когда точечными являются все одноместные функции, полученные из неё подстановками констант на места переменных.

Отметим также

Следствие 2. Замкнутый подстановками констант из множества L класс функций из P_L тогда и только тогда включён в множество T_L , когда все одноместные функции этого класса включены в множество T_L .

Распознавание точечности. Лемма 1 позволяет построить схему из функциональных элементов [14] (иначе — быстрый алгоритм, см. [15]) для распознавания точечности заданной векторно n -местной функции из P_L , такую, что сложность $L(n)$ этой схемы (то есть число её элементов) и её глубина $J(n)$ (максимальная длина пути от входа к выходу) при растущем параметре n ограничены сверху соответствующими величинами $O(|L|^n)$ и $O(n^2)$. При этом предполагается, что, во-первых, схема строится из функциональных элементов некоторого полного базиса (безразлично какого, поскольку сложность и глубина вычисляется с точностью до мультипликативной константы), во-вторых, наборы в множестве L^n линейно упорядочены некоторым образом, в-третьих, функция задаётся вектором своих значений, вычисленных на наборах из L^n в соответствии с их упорядочением, и, наконец, в-четвёртых, её значения из L представляются двоичными векторами фиксированной длины, например, равной $\lceil \log |L| \rceil$.

Действительно, на основании теоремы 2 схема распознавания точечности произвольной функции f из P_L , зависящей от $n > 1$ переменных, при помощи многократной конъюнкции строится из $|E|^{n-1}$ схем распознавания точечности одноместных функций f_x для наборов x из множества E^{n-1} и $|L|$ схем распознавания точечности всевозможных функций f^b для элементов b из множества L , каждая из которых зависит от $n - 1$ переменных. Причём число конъюнкций не превосходит суммы $|E|^{n-1} + |L|$. Это приводит при $n \geq 1$ к рекуррентному неравенству

$$L(n) \leq L(1)|E|^{n-1} + L(n-1)|L| + c_0(|E|^{n-1} + |L|) \leq c_1|E|^{n-1} + L(n-1)|L|$$

для её сложности, где c_0 — константа, определяемая сложностью вычисления конъюнкции в используемом базисе, и c_1 — произвольная константа, удовлетворяющая при любом $n \geq 1$ неравенству

$$c_1 \geq L(1) + c_0(1 + |L|/(|E|^{n-1})),$$

например, равная $L(1) + c_0(1 + |L|)$. Из рекуррентного неравенства получаем

$$\begin{aligned} L(n) &\leq c_1(|E|^{n-1} + |E|^{n-2}|L| + \dots + |E||L|^{n-2} + |L|^{n-1}) = c_1|L|^{n-1} \sum_{i=0}^{n-1} (|E|/|L|)^i = \\ &= c_1|L|^{n-1}(1 - (|E|/|L|)^n)/(1 - |E|/|L|) \leq c_1|L|^{n-1}/(1 - |E|/|L|) = \\ &= c_1|L|^n/(|L| - |E|) = c_2|L|^n = O(|L|^n), \end{aligned}$$

где $c_2 = c_1/(|L| - |E|)$. Ясно, что глубина схемы, вычисляющей конъюнкцию $|E|^{n-1} + |L|$ аргументов, ограничена сверху величиной c_3n , где c_3 — некоторая константа, зависящая от базиса и от значений $|E|$ и $|L|$. Следовательно, для глубины $J(n)$ всей схемы получаем при $n > 1$ следующее рекуррентное неравенство:

$$J(n) \leq c_3n + J(n-1).$$

Всегда можно выбрать константу $c_3 \geq J(1)$. Тогда записанное неравенство выполняется и при $n = 1$, а глубина схемы ограничена сверху арифметической прогрессией:

$$J(n) \leq c_3(n + (n - 1) + \dots + 1) = c_3 n(n + 1)/2 \leq c_3 n^2 = O(n^2).$$

Таким образом, верна

Теорема 3. В любом полном базисе существует последовательность схем распознавания точечности заданной векторно n -местной функции из P_L , сложность и глубина которых при растущем параметре n ограничены сверху соответствующими величинами $O(|L|^n)$ и $O(n^2)$.

Таким образом, распознавание точечности заданной векторно функции осуществляется схемой линейной от числа её входов сложности.

Замкнутые классы точечных функций. При изучении точечных и минимальных точечных функций приходится считаться с незамкнутостью их классов относительно суперпозиции. Точнее, эти классы замкнуты операциями бесповторной суперпозиции и перестановки переменных, но не замкнуты операциями отождествления переменных. Например, одноместная функция $f(x) = x \vee \neg x$ на полурешётке $\tilde{E}_2 = \{0, 1, \top\}$ не является точечной в силу соотношений

$$f(\top) = \top \neq 1 = 1 + 1 = f(0) + f(1),$$

хотя получена отождествлением переменных из двухместной минимальной точечной функции $x \vee \neg y$. Вместе с тем для синтеза дискретных управляющих систем с динамическим поведением представляют интерес замкнутые классы точечных и минимальных точечных функций. Этот интерес вызван тем, что задачи схемного и формульного представления функций в таких классах сводятся к аналогичным задачам для ограничений этих функций на множества минимальных элементов их областей определения и решаются после такого сведения известными методами синтеза функций k -значной логики, возможно незначительно модифицированными. Наряду с этим, задачи синтеза точечных функций в незамкнутых классах сталкиваются со значительными трудностями и требуют разработки новых методов.

Рассмотрим далее одну конструкцию замкнутых классов точечных функций. Для её изложения понадобится следующее определение. Будем говорить, что функция f из P_L , зависящая от n переменных, *сохраняет пару* (A, B) множеств A и B функций из P_L , зависящих от m переменных, если для любых функций $s_1(x), \dots, s_n(x)$ из множества B функция $f(s_1(x), \dots, s_n(x))$ принадлежит множеству A , при этом через x обозначен набор переменных x_1, \dots, x_m . Несложно понять, что множество всех функций из P_L , сохраняющих пару множеств (A, B) , составляет наследственный класс (замкнутый операциями подстановки переменных под знак его функций). Этот класс замкнут суперпозицией, если выполняется включение $A \subseteq B$. Сформулированные свойства проверяются непосредственно. В силу следствия 2 имеет место

Лемма 2. Пусть A и B — множества точечных функций из T_L , зависящих от $m \geq 1$ переменных, причём множество A включено в T_L , а множество B содержит селекторную функцию, тождественно равную некоторой переменной, и содержит все константы из множества L , рассматриваемые как m -местные функции с фиктивными переменными. Тогда функции из P_L , сохраняющие пару множеств (A, B) , составляют наследственный класс, включённый в T_L . Этот класс замкнут суперпозицией, если выполняется включение $A \subseteq B$.

Лемма 2 позволяет получить ряд примеров замкнутых классов точечных функций. Обозначим через \mathcal{M}_{T_L} систему всех максимальных по включению замкнутых классов точечных функций на полурешётке L . Имеет место

Теорема 4. Система \mathcal{M}_{T_L} конечна, и каждый замкнутый класс точечных функций из T_L включён в некоторый её класс.

Доказательство. Рассмотрим произвольный класс N точечных функций из T_L . Расширим его сначала до замкнутого класса $N_1 = [N \cup L]$. Подобное расширение состоит в пополнении класса N константами из L , а также подфункциями его функций, получаемыми подстановками констант на места переменных. В силу следствия 2 класс N_1 содержит только точечные функции из T_L . Далее расширением класса N_1 является класс N_2 функций из P_L , сохраняющих пару (A, B) , где A — множество всех одноместных функций класса N_2 и B — то же множество, но пополненное тождественной функцией. Осталось обратить внимание на то, что классов типа N_2 конечное число (поскольку они однозначно определяются своими одноместными функциями), и максимальные по включению из них составляют систему \mathcal{M}_{T_L} . ■

Обозначив через $\mathcal{M}_{\min T_L}$ систему максимальных по включению замкнутых классов минимальных точечных функций на полурешётке L , тотчас получаем

Следствие 3. Система $\mathcal{M}_{\min T_L}$ конечна и каждый замкнутый класс минимальных точечных функций из $\min T_L$ включён в некоторый её класс.

Доказательство. Несложно понять, что система $\mathcal{M}_{\min T_L}$ состоит из максимальных по включению классов $K \cap \text{pol}_L(E)$ для всевозможных классов K из \mathcal{M}_{T_L} . ■

Из доказательства теоремы 4 и её следствия 3 видно, что максимальные замкнутые классы точечных и минимальных точечных функций из систем \mathcal{M}_{T_L} и $\mathcal{M}_{\min T_L}$ допускают эффективное задание. Представляется важной задача явного описания таких классов, в настоящее время не решённая. В заключение раздела рассмотрим

Пример замкнутого класса точечных функций. Пусть L совпадает с множеством $\text{in}(E, \preceq)$ интервалов решётки (E, \preceq) . В соответствии со сделанными ранее определениями рассматриваем множество L как полурешётку с упорядочением \leq и, одновременно, как решётку с упорядочением \preceq . Покажем, что клон $\text{pol}_L(\leq, \preceq)$ функций из P_L , сохраняющих упорядочения \leq и \preceq множества L (первое — полурешёточное, второе — решёточное), состоит только из точечных функций. В силу следствия 2 достаточно проверить, что точечной является любая одноместная функция s из этого клона. Для этого рассмотрим интервал $[a, b]$ из множества L , где a и b — какие-то элементы решётки E , для которых выполняется неравенство $a \preceq b$. Заметим, что $a \leq [a, b]$, $b \leq [a, b]$, $a \preceq [a, b] \preceq b$, откуда

$$s(a) \leq s([a, b]), s(b) \leq s([a, b]), s(a) \preceq s([a, b]) \preceq s(b).$$

Первое и третье из записанных неравенств влекут совпадение левых границ интервалов $s(a)$ и $s([a, b])$. Аналогично в силу второго и четвёртого неравенств совпадают правые границы интервалов $s(b)$ и $s([a, b])$. Тогда в силу третьего и четвёртого неравенств выполняется равенство $s([a, b]) = s(a) + s(b)$, означающее, что функция s точечная.

Можно показать, что в двоичном случае, когда решётка E состоит из чисел 0 и 1, таких, что $0 \preceq 1$, и имеет три интервала 0, 1, \top , упорядоченных отношениями \leq и \preceq так, что

$$0 \leq \top, 1 \leq \top \text{ и } 0 \preceq \top \preceq 1,$$

функции клона $\text{pol}_L(\leq, \preceq)$ суть суммы его минимальных точечных функций, а последние являются полурешётчными продолжениями монотонных булевых функций. Нечто аналогичное выполняется и в общем случае. Доказывать это не будем.

4. Сильно точечные функции

В связи с задачей выделения замкнутых классов точечных функций введём в рассмотрение сильно точечные функции.

Сильная точечность. Точечную функцию назовём *сильно точечной*, если точечными являются все функции, получаемые из неё отождествлением переменных. Очевидно, что всякий замкнутый класс точечных функций состоит только из сильно точечных функций.

Теорема 5. Множество сильно точечных функций на полурешётке L , зависящих от n переменных, замкнуто сложением, то есть является верхней полурешёткой с упорядочением \leq .

Доказательство. Пусть функция h является суммой функций f и g . Если функция h не является сильно точечной, то отождествлением переменных из неё можно получить неточечную функцию h' . Аналогичное отождествление переменных в функциях f и g приводит к соотношению $h' = f' + g'$, в котором в силу теоремы 1 по крайней мере одна из функций f' или g' неточечная. Но тогда сильно точечной не является по крайней мере одна из функций f или g . Этого достаточно для доказательства. ■

Тесты сильной точечности. В силу сделанного определения *класс сильно точечных функций наследственный*, то есть замкнут операциями подстановки переменных под знак его функций и, более того, *является наибольшим по включению среди наследственных классов точечных функций*. Несколько менее очевидно, что он *инвариантен*, в частности, замкнут подстановками констант на места переменных. Это следует из возможности менять порядок выполнения операций отождествления переменных и подстановки констант без изменения результата этих операций. В соответствии с этим, если в результате подстановки констант в сильно точечную функцию получается функция, не являющаяся сильно точечной, то отождествлением переменных из последней можно получить неточечную функцию. Изменив порядок операций, эту неточечную функцию можно получить из исходной (по предположению сильно точечной) функции, сначала отождествив переменные, а затем подставив константы. Но в силу теста точечности неточечная функция получается уже после выполнения операций отождествления переменных, до подстановки констант. Для сильно точечной исходной функции это невозможно. Полученное противоречие доказывает тезис об инвариантности класса сильно точечных функций на полурешётке.

В действительности, можно сформулировать следующий тест сильной точечности.

Теорема 6. Функция из P_L тогда и только тогда сильно точечная, когда точечными являются все одноместные функции, получаемые из неё при помощи операций отождествления переменных и подстановки констант из множества L на места переменных.

Доказательство. Необходимость уже доказана выше. Достаточность следует из определения сильно точечной функции и теста точечности, в силу которых из функции, не являющейся сильно точечной, отождествлением переменных получается некоторая неточечная функция, из которой подстановками констант получается одноместная неточечная функция. ■

Сформулируем ещё один тест сильной точечности. Понадобится следующее обозначение. Для n -местной функции f и непустого подмножества $A \subseteq \{1, \dots, n\}$ через ${}^A f$ обозначим функцию от $n - |A| + 1$ переменных, получаемую из функции $f(x_1, \dots, x_n)$ отождествлением всех переменных x_i с номерами i из множества A с некоторой одной из этих переменных (всё равно какой именно, например, первой из них). Функции f и ${}^A f$ совпадают при одноэлементном множестве A . Имеет место

Следствие 4. Функция f из P_L , зависящая от n переменных, тогда и только тогда сильно точечная, когда точечными являются функции ${}^A f$ для всевозможных непустых подмножеств $A \subseteq \{1, \dots, n\}$.

Доказательство. Необходимость очевидна из определения сильно точечной функции. Достаточность. В соответствии с тестом сильной точечности функция f является сильно точечной, если точечными являются все одноместные функции, полученные из функций ${}^A f$ подстановками констант на места переменных x_i , где $i \notin A$. Но точечность этих одноместных функций следует из точечности функций ${}^A f$. ■

Распознавание сильной точечности. В силу следствия 4 схема для распознавания сильной точечности произвольной n -местной функции f при помощи многократной (но не более чем 2^n -кратной) конъюнкции строится из схем для распознавания точечности функций ${}^A f$ для всевозможных непустых подмножеств $A \subseteq \{1, \dots, n\}$. В свою очередь для m -элементного множества A проверка точечности функции ${}^A f$, зависящей от $n - m + 1$ переменных, осуществляется схемой из предыдущего раздела, сложность и глубина которой ограничены сверху соответствующими величинами $c_2|L|^{n-m+1}$ и $c_3(n - m + 1)^2$. Суммарная сложность этих схем ограничена сверху величиной

$$\sum_{m=1}^n \binom{n}{m} c_2 |L|^{n-m+1} = c_2 |L| ((1 + |L|)^n - |L|^n) = O((1 + |L|)^n),$$

а максимальная глубина — величиной $c_3 n^2$. Поскольку многократная конъюнкция заведомо реализуется схемой сложности $O((1 + |L|)^n)$ и линейной от n глубины, верна

Теорема 7. В любом полном базисе существует последовательность схем распознавания сильной точечности заданной векторно n -местной функции из P_L , имеющих при растущем параметре n сложность $O((1 + |L|)^n)$ и глубину $O(n^2)$.

Итак, распознавание сильной точечности функции, заданной вектором значений, осуществляется схемой полиномиальной сложности с «почти линейным» полиномом степени $\log_{|L|}(1 + |L|)$.

5. Дизъюнктивные нормальные формы трёхзначных полурешёточных функций

В разделе рассматриваются функции на полурешётке $\tilde{E}_2 = \{0, 1, \top\}$, вычисляемые формулами в базисе $\{0, 1, \vee, \wedge, \neg\}$. Явно описывается класс

$$[0, 1, \vee, \wedge, \neg]$$

всех таких функций.

Поскольку базисные функции являются монотонными продолжениями булевых функций (сохраняют порядок \leq и множество $E_2 = \{0, 1\}$), имеет место включение

$$[0, 1, \vee, \wedge, \neg] \subseteq \text{pol}_{\tilde{E}_2}(\leq, E_2).$$

Будет показано далее, что здесь выполняется равенство.

Днф. Начнём с некоторых замечаний. Заметим, во-первых, что полурешётка \tilde{E}_2 совпадает с полурешёткой $\text{in}(E_2, \preceq)$ интервалов решётки E_2 . Упорядочение \preceq в последней продолжается на множество \tilde{E}_2 так: $0 \preceq \top \preceq 1$, которое становится дистрибутивной решёткой (в соответствии с введённой терминологией — решёткой интервалов решётки \tilde{E}_2) с наименьшим элементом 0, наибольшим элементом 1, с дизъюнкцией \vee и конъюнкцией \wedge для взятия точных верхней и нижней граней. Отрицание \neg является инверсным автоморфизмом этой решетки. В связи с этим, как и одноимённые булевы функции, дизъюнкция \vee и конъюнкция \wedge ассоциативны, коммутативны, идемпотентны, дистрибутивны одна по другой, для них выполняются законы поглощения, а с операцией \neg они удовлетворяют законам де Моргана. Вместо закона исключённого третьего, однако, выполняются лишь неравенства

$$x \vee \neg x \geq 1 \text{ и } x \wedge \neg x \geq 0,$$

являющиеся строгими при $x = \top$.

Сделанные замечания позволяют любую формулу в базисе $\{0, 1, \vee, \wedge, \neg\}$ эквивалентными преобразованиями, не изменяющими вычисляемой формулой функции, привести либо к константам 0 или 1, либо к виду днф $k_1 \vee \dots \vee k_r$, где k_i — конъюнкции, в которые переменные могут входить единожды — под знаком отрицания либо без него, или дважды — с отрицанием и без него.

Отметим, что в отличие от двоичного случая из-за невыполнения закона исключённого третьего не всегда эквивалентные преобразования позволяют избавиться от повторного появления переменных в конъюнкциях. В связи с этим имеет смысл предварительно рассмотреть днф без повторяющихся переменных в конъюнкциях, затем с повторяющимися переменными в каждой конъюнкции и, наконец, с конъюнкциями обоих типов — допускающими либо не допускающими повторы.

Введём некоторые обозначения. Для набора $x = (x_1, \dots, x_n)$ переменных и набора $a = (a_1, \dots, a_n)$ элементов из множества $\{0, 1, \perp, \top\}$ положим

$$x^a = x_1^{a_1} \wedge \dots \wedge x_n^{a_n},$$

понимая под $x_i^{a_i}$ выражения $\neg x_i, x_i, 1$ или $(x_i \wedge \neg x_i)$ при соответствующем значении элемента a_i , равном 0, 1, \top или \perp . Для всякого множества A наборов a^1, \dots, a^m из множества $\{0, 1, \perp, \top\}^n \setminus \{\perp\}^n$ через x^A будем обозначать формулу

$$x^{a^1} \vee \dots \vee x^{a^m},$$

которую и будем называть днф. (Порядок, в котором пронумерованы наборы a^i из множества A , всякий раз предполагается произвольным. Он не важен, поскольку от него не зависит вычисляемая формулой x^A функция.)

В полурешётке \tilde{E}_2^n наборы a и b , не имеющие общей нижней грани, назовём *ортогональными* и будем писать $a \perp b$ в этом случае. Для подмножества $K \subseteq \tilde{E}_2^n$ через $\perp K$ обозначим множество всех наборов в \tilde{E}_2^n , ортогональных всем наборам из K . Заметим, что имеет место включение $\perp \{a\} \supseteq \perp \{b\}$, если для наборов a и b из множества \tilde{E}_2^n выполняется неравенство $a \leq b$. В связи с этим

$$\perp K = \perp \min(K, \leq).$$

Днф «без повторений». Рассмотрим сначала днф, в которых конъюнкции не содержат повторяющихся переменных, то есть днф x^A , где $A \subseteq \tilde{E}_2^n \setminus \{\top\}^n$. Охарактери-

ризуем класс функций, вычисляемых такими днф. С этой целью заметим, что в соответствии со сделанными определениями для любых наборов x из \tilde{E}_2^n и a из $\tilde{E}_2^n \setminus \{\top\}^n$

$$x^a = \begin{cases} 1, & \text{если } x \leq a; \\ 0, & \text{если } x \perp a; \\ \top & \text{в остальных случаях.} \end{cases}$$

Это свойство удобно сначала проверить для $n = 1$, а затем для произвольного целого положительного n . Тогда для любого подмножества $A \subseteq \tilde{E}_2^n \setminus \{\top\}^n$

$$x^A = \begin{cases} 1, & \text{если } x \leq a \text{ для некоторого } a \text{ из } A \text{ (равносильно из } \max(A, \leq)); \\ 0, & \text{если } x \perp a \text{ для любого } a \text{ из } A; \\ \top & \text{в остальных случаях.} \end{cases}$$

Отсюда следует

Теорема 8. Для непустого множества $A \subseteq \tilde{E}_2^n \setminus \{\top\}^n$ и набора переменных $x = (x_1, \dots, x_n)$ формула x^A тогда и только тогда вычисляет функцию $f(x)$ из множества $\text{pol}_{\tilde{E}_2}(\leq, E_2)$, когда выполняются условия

- 1) $f^{-1}(0) = \perp A$;
- 2) $\max(f^{-1}(1)) \subseteq A \subseteq f^{-1}(1)$.

Следствие 5. Функция из множества $\text{pol}_{\tilde{E}_2}(\leq, E_2)$, зависящая от n переменных, тогда и только тогда вычисляется формулой вида x^A для набора переменных $x = (x_1, \dots, x_n)$ и некоторого непустого множества $A \subseteq \tilde{E}_2^n \setminus \{\top\}^n$, когда выполняется условие $f^{-1}(0) = \perp f^{-1}(1)$.

Из теоремы 8 следует, что отличная от констант 0 и 1 минимальная точечная функция f вычисляется сокращённой днф

$$x^K, \text{ где } K = \max(f^{-1}(1), \leq),$$

причём это единственная тупиковая форма, минимальная как по числу конъюнкций, так и по суммарному числу букв в них. В частности, все минимальные точечные функции на полурешётке \tilde{E}_2 вычисляются формулами в базисе $\{0, 1, \vee, \wedge, \neg\}$.

Днф «с повторениями». Теперь рассмотрим днф «с повторениями», в которых любая конъюнкция содержит повторяющиеся переменные. Точнее, рассмотрим частный случай (достаточный для достижения заявленной цели) таких днф, в которых любая конъюнкция содержит все переменные, причём хотя бы одну дважды — с отрицанием и без него. То есть рассмотрим днф x^A , где $A \subseteq \{0, 1, \perp\}^n \setminus \{0, 1\}^n$.

В соответствии со сделанными определениями для любых наборов x из множества \tilde{E}_2^n и b из множества $\{0, 1, \perp\}^n \setminus \{0, 1\}^n$

$$x^b = \begin{cases} \top, & \text{если } x \geq b^*; \\ 0 & \text{в остальных случаях,} \end{cases}$$

где набор b^* получен из b заменой знаков \perp на \top . (Для доказательства нужно заметить, что равенство $x^b = 1$ невозможно в силу существования $b_i = \perp$, а равенство $x^b = \top$ равносильно тому, что всякое выражение $x_i^{b_i}$ принимает значение в множестве $\{1, \top\}$, если $b_i \in \{0, 1\}$, и принимает значение \top , если $b_i = \perp$; каждый из случаев реализуется лишь при $x_i \geq b_i^*$.) Тогда для любого подмножества $B \subseteq \{0, 1, \perp\}^n \setminus \{0, 1\}^n$

$$x^B = \begin{cases} \top, & \text{если } x \geq b \text{ для некоторого } b \text{ из } B^* \text{ (равносильно из } \min(B^*, \leq)); \\ 0 & \text{в остальных случаях,} \end{cases}$$

где $B^* = \{b^* : b \in B\}$. Из сказанного следует

Теорема 9. Для непустого множества $B \subseteq \{0, 1, \perp\}^n \setminus \{0, 1\}^n$ и набора переменных $x = (x_1, \dots, x_n)$ формула x^B тогда и только тогда вычисляет функцию $f : \tilde{E}_2^n \rightarrow \{0, \top\}$ из множества $\text{pol}_{\tilde{E}_2}(\leq, E_2)$, когда выполняется равенство $\min(B^*) = \min(f^{-1}(\top), \leq)$.

Следствие 6. Всякая сюръективная функция $f : \tilde{E}_2^n \rightarrow \{0, \top\}$ из класса $\text{pol}_{\tilde{E}_2}(\leq, E_2)$ вычисляется формулой вида x^B для некоторого непустого множества $B \subseteq \{0, 1, \perp\}^n \setminus \{0, 1\}^n$ и набора переменных $x = (x_1, \dots, x_n)$.

Днф с конъюнкциями двух типов. Совместно используя теоремы 8 и 9, получаем следующую теорему.

Теорема 10. Для непустых подмножеств $A \subseteq \tilde{E}_2^n \setminus \{\top\}^n$ и $B \subseteq \{0, 1, \perp\}^n \setminus \{0, 1\}^n$ с объединением $C = A \cup B$ и набора переменных $x = (x_1, \dots, x_n)$ формула x^C тогда и только тогда вычисляет функцию $f(x)$ из множества $\text{pol}_{\tilde{E}_2}(\leq, E_2)$, когда выполняются условия

- 1) $\max(f^{-1}(1)) \subseteq A \subseteq f^{-1}(1)$;
- 2) $\min(\perp A \cap f^{-1}(\top), \leq) \subseteq B^*$.

Доказательство. Первое условие означает, что множество $f^{-1}(1)$ единиц функции f совпадает с множеством единиц функции, вычисляемой формулой x^A , а тогда и функции, вычисляемой формулой x^C . В частности, первое условие необходимо. Заметим далее, что функция, вычисляемая формулой x^A , имеет наибольшее по включению множество нулей среди n -местных квазимонотонных функций с тем же множеством единиц. Это следует из теста квазимонотонности, в силу которого наборы из множества нулей квазимонотонной функции ортогональны наборам из её множества единиц. В силу сказанного из первого условия следует, что множество нулей функции f включено в множество нулей функции, вычисляемой формулой x^A . Следовательно, при выполнении первого условия для совпадения множества нулей функции f с множеством нулей функции x^C необходимо и достаточно, чтобы функция, вычисляемая формулой x^B , принимала значение \top на тех наборах, на которых функция f принимает значение \top и функция, вычисляемая формулой x^A , принимает значение 0 . При выполнении первого условия множество таких наборов совпадает в силу теоремы 8 с множеством $\perp A \cap f^{-1}(\top)$, а второе условие означает в силу теоремы 9, что функция x^B принимает на наборах из этого множества значение \top . Из сделанных замечаний легко следует как необходимость, так и достаточность доказываемых условий. ■

Следствие 7. Имеет место равенство классов

$$[0, 1, \vee, \wedge, \neg] = [\min T_{\tilde{E}_2}] = \text{pol}_{\tilde{E}_2}(\leq, E_2).$$

Заметим, что выше были рассмотрены далеко не все днф и даже не все днф с повторами в каждой конъюнкции. Однако рассмотренных случаев достаточно для достижения заявленной цели описания класса функций, вычисляемых в базисе $\{0, 1, \vee, \wedge, \neg\}$.

Разложение Шеннона для минимальной точечной функции f от n аргументов на полурешётке \tilde{E}_2 выглядит следующим образом:

$$f(xy) = \neg x f_0(y) \vee x f_1(y) \vee f_0(y) f_1(y),$$

где конъюнкция опущена. Это соотношение выполняется для произвольных значений x из множества \tilde{E}_2 и y из множества \tilde{E}_2^{n-1} , для проверки нужно рассмотреть

различные возможности для x . Последним произведением $f_0(y)f_1(y)$ здесь нельзя пренебречь, в отличие от двоичного случая. Чтобы в этом убедиться, достаточно рассмотреть трёхместную функцию $\neg x_1x_2 \vee x_1x_3 \vee x_2x_3$. Она задана сокращённой днф, а потому минимальная точечная. То же выражение для неё даёт разложение Шеннона. При удалении последнего произведения x_2x_3 днф перестаёт быть сокращённой, а задаваемая ею функция перестаёт быть минимальной точечной.

С использованием разложения Шеннона в записанной выше форме известным методом из [14] можно получить верхнюю оценку $O(2^n/n)$ сложности минимальной точечной функции от n аргументов при реализации её схемами из функциональных элементов в базисе $0, 1, \vee, \wedge, \neg$. Эта оценка только мультипликативной константой отличается от нижней. Хуже обстоит дело с нижними и верхними оценками формульной сложности, различающимися в настоящее время экспоненциальным множителем.

Таким образом, на основании сказанного можно сформулировать важную задачу определения сложности минимальных точечных функций в различных базисах и в различных вычислительных моделях (формулы, схемы из функциональных элементов, переключательные схемы и др.), а также связанную с ней задачу разработки асимптотически наилучших методов синтеза для минимальных точечных функций в различных вычислительных моделях. Сформулированные задачи, актуальные уже в простейшем случае полурешётки \tilde{E}_2 , сохраняют свою значимость и для других полурешёток.

6. Обобщённые днф

Опишем метод формульного представления для квазимонотонных функций в базисах, содержащих некоторые специальные двухместные функции и все слабо существенные квазимонотонные функции, а также для минимальных точечных функций в базисах, содержащих некоторые специальные двухместные функции и все одноместные минимальные точечные функции. Эти результаты будут получены для функций на произвольной конечной дистрибутивной точечной полурешётке.

Дистрибутивная точечная полурешётка. Полурешётка L называется *дистрибутивной*, если таковой является решётка L' , то есть если в последней для любых её элементов a, b и c выполняется соотношение $a(b+c) = ab+ac$ (а с ним и двойственное $a+bc = (a+b)(a+c)$), позволяющее при вычислении в ней раскрывать скобки и выносить общие множители. Основным примером является дистрибутивная полурешётка \tilde{E} непустых подмножеств конечного множества E .

Будем считать заданными дистрибутивную точечную полурешётку L и некоторое её подмножество C , включающее все её минимальные элементы из множества $E = \min(L)$ и называемое далее *специальным*. Имеет место

Лемма 3. Всякий элемент b дистрибутивной точечной полурешётки L со специальным подмножеством C , таким, что $E \subseteq C$, является наибольшим решением системы уравнений $xc = \perp$ для всевозможных элементов c из C , таких, что $bc = \perp$.

Доказательство. В силу дистрибутивности $xc = yc = \perp$ влечёт $(x+y)c = \perp$, откуда сумма любых решений рассматриваемой системы снова является её решением, и система имеет наибольшее решение в решётке L' , равное сумме всех решений. Вместе с тем сам элемент b является решением системы. Следовательно, для наибольшего решения B выполняется неравенство $b \leq B$. В силу точечности полурешётки L элемент B является суммой некоторых её точек. Значит, если записанное неравенство строгое, то для некоторой точки c выполняется неравенство $c \leq B$ и не выполняется неравенство $c \leq b$. Но этого не может быть, так как тогда $bc = \perp$ (из-за невыполнения

второго неравенства $c \leq b$) и в системе присутствует уравнение $xc = \perp$, которому не удовлетворяет элемент B (в силу первого неравенства $c \leq B$). Таким образом, $b = B$ и лемма доказана. ■

Следствие 8. Пусть полурешётка L дистрибутивная и точечная со специальным множеством C , таким, что $E \subseteq C$. Тогда всякая n -местная функция f из P_L однозначно определяется указанием множеств

$$J(c, f) = \{d : d \in L^n \wedge f(d)c = \perp\}$$

для всех элементов c из C .

Доказательство. Для любого набора d из L^n значение $f(d)$ можно найти как наибольшее решение системы уравнений $xc = \perp$ для всевозможных c из C , таких, что $d \in J(c, f)$. ■

Отметим необходимые свойства множеств $J(c, f)$.

Лемма 4. Пусть C — специальное множество дистрибутивной точечной полурешётки L , такое, что $E \subseteq C$. Тогда квазимонотонная функция f слабо существенная, если для всех элементов c из C выполняется неравенство $|J(c, f)| \leq 1$.

Доказательство. Пусть f — n -местная квазимонотонная функция и для всех элементов c из E выполняется неравенство $|J(c, f)| \leq 1$. Рассмотрим произвольное подмножество $U \subseteq L^n$. Заметим, что отсутствие нижней грани у подмножества $f(U) = \{f(d) : d \in U\}$ в полурешётке L означает непустоту всех пересечений $U \cap J(c, f)$ для элементов c из E . В силу неравенств эти пересечения одноэлементные. Это означает, что может существовать не более одного минимального по включению подмножества $U \subseteq L^n$ с неограниченным снизу множеством $f(U)$. По тесту квазимонотонности наборы в таком минимальном множестве U имеют общую компоненту, пусть i -ю, без нижней грани в L . В силу единственности такого минимального множества указанное свойство i -й компоненты выполняется не только для его наборов, но и для наборов всякого такого подмножества $U \subseteq L^n$ с неограниченным снизу множеством $f(U)$. Тогда по тесту слабой существенности из [3] функция f слабо существенная. ■

Специальные функции.

Лемма 5. Для любого элемента c дистрибутивной точечной полурешётки L найдётся двухместная квазимонотонная функция $*_c$ в Q_L , такая, что

- 1) $x *_c x = x$ для всех x из L ;
- 2) $x *_c (x + c) = (x + c) *_c x = x$ для всех x из L , таких, что $xc = \perp$.

Более того, функцию $*_c$, обладающую указанными свойствами, можно выбрать минимальной точечной в классе $\min T_L$.

Доказательство. Указанные условия определяют частичную двухместную функцию из P_L^* , принимающую неопределённое значение на всех наборах, не оговоренных в этих условиях. Можно увидеть, пользуясь дистрибутивностью, что эта частичная функция сохраняет наличие нижних граней у её аргументов (точнее, сохраняет предикаты ε_n из [3]), откуда по тесту квазимонотонности она имеет квазимонотонное продолжение в клоне Q_L . Осталось заметить, что условия 1 и 2 выполняются не только для функции $*_c$, но и для любой её монотонной миноранты. Первое условие — в силу точечности полурешётки L и свойства минорирования, второе — в силу монотонности и минорирования. ■

Квазимонотонную функцию $*_c$ в условиях леммы будем называть *c-специальной*.

Лемма 6. Пусть L — дистрибутивная точечная полурешётка, c — некоторый её элемент и $*_c$ — некоторая c -специальная функция из Q_L . Пусть также для квазимонотонной функции f из Q_L , зависящей от n переменных, множество $J(c, f)$ содержит более одного элемента. Тогда найдутся квазимонотонные функции g и h в Q_L , зависящие от n переменных, такие, что для любого набора d из множества L^n выполняется равенство $f(d) = g(d) *_c h(d)$ и для любого элемента a из полурешётки L выполняются включения

$$J(a, g) \subseteq J(a, f), \quad J(a, h) \subseteq J(a, f),$$

строгие при $a = c$.

Доказательство. Разобьём множество $J(c, f)$ на два класса A и B . Рассмотрим функции g и h из P_L от n переменных, такие, что

$$g(d) = f(d) + c \text{ и } h(d) = f(d) + c \text{ при } d \text{ из } A \text{ и из } B \text{ соответственно,}$$

и принимающие те же значения, что и функция f , в остальных случаях, то есть значения

$$g(d) = f(d) \text{ и } h(d) = f(d) \text{ при } d \text{ из } L^n \setminus A \text{ и из } L^n \setminus B \text{ соответственно.}$$

Функции g и h квазимонотонны вслед за их минорантой f . Остальные условия проверяются непосредственно. ■

Метод разложения. Леммы 4 и 6 позволяют сформулировать метод формульного представления квазимонотонных функций на дистрибутивной точечной полурешётке L со специальным подмножеством $C \subseteq L$ (включающим множество $E = \min(L)$) в базисе, содержащем все слабо существенные функции из Φ_L и некоторый набор c -специальных функций $*_c$ для всех c из множества C . Этот метод для представления функции f формулой в указанном базисе требует:

- 1) найти в множестве C элемент c , такой, что $|J(c, f)| > 1$, и разложить f по функции $*_c$ в соответствии с леммой 6, либо
- 2) (в отсутствие такого c) закончить разложение, при этом в соответствии с леммой 4 функция f слабо существенная;
- 3) выполнять подобные действия рекурсивно для компонент разложения, компонент компонент и т. д., пока возможно.

Если c -специальные функции выбрать минимальными точечными (в соответствии с леммой 5 это возможно) и в формуле, полученной для минимальной точечной функции f по описанным выше правилам, заменить последние полученные слабо существенные компоненты их одноместными минимальными точечными минорантами из класса $\min T_L$ (существующими по определению слабо существенной функции), то в результате получим формулу для функции f в базисе из одноместных минимальных точечных функций и двухместных минимальных c -специальных функций. Тем самым доказана

Теорема 11. Имеют место соотношения

$$Q_L = [\Phi_L \cup (\min T_L)^{(2)}], \quad \min T_L \subseteq [(\min T_L)^{(2)}].$$

Замечание о полурешётке подмножеств. Частным случаем c -специальных функций являются дизъюнкция и конъюнкция на трёхэлементной полурешётке \tilde{E}_2 , первая является 0-специальной, вторая — 1-специальной. Таким образом, полученный

метод в том числе позволяет строить формулы для минимальных точечных функций на полурешётке \tilde{E}_2 в базисе $0, 1, \neg, \vee, \wedge$. В более общей ситуации, когда полурешётка L совпадает с полурешёткой \tilde{E}_k подмножеств множества $E_k = \{0, 1, \dots, k-1\}$, под дизъюнкцией и конъюнкцией обычно понимают минимальные точечные функции, определённые на E_k следующим образом:

$$x_1 \vee x_2 = \max(x_1, x_2), \quad x_1 \wedge x_2 = \min(x_1, x_2)$$

для любых x_1 и x_2 из E_k , а затем точно продолженные на полурешётку $L = \tilde{E}_k$. Как видно, эти функции s -специальные, дизъюнкция — для любого s из $\{0, 0+1, \dots, E_k\}$, а конъюнкция — для любого s из $\{k-1, (k-1) + (k-2), \dots, E_k\}$. Несложно найти и s -специальную функцию для произвольного s из \tilde{E}_k . Таковой является, например, функция \vee_s , определяемая как

$$x_1 \vee_s x_2 = s(s^{-1}(x_1) \vee s^{-1}(x_2)),$$

где s — минимальная точечная подстановка на \tilde{E}_k , отображающая на элемент s элемент вида $0 + \dots + (l-1)$ для некоторого $l, 1 \leq l \leq k$, так, что $s(0 + \dots + (l-1)) = s$. В частности, если s — минимальный элемент полурешётки L , не равный 0, то есть принадлежащий множеству $E_k \setminus \{0\}$, то в качестве такой подстановки s можно выбрать точечное продолжение транспозиции $(0, s)$. Также несложно получить специальные функции композицией конъюнкции с подстановками. Из сказанного следует

Теорема 12. Для полурешётки $L = \tilde{E}$ имеют место соотношения

$$Q_L = [\Phi_L \cup \{\vee\}], \quad \min T_L \subseteq [(\min T_L)^{(1)} \cup \{\vee\}].$$

Более подробный анализ, проведённый в [16], позволяет утверждать, что в условиях этой теоремы все функции в каждом из классов Q_L, M_L, T_L или $\min T_L$ выражаются формулами с использованием одноместных функций класса и дизъюнкции (равно конъюнкции).

ЛИТЕРАТУРА

1. Агibalов Г. П., Оранов А. М. Лекции по теории автоматов. Томск: Изд-во Том. ун-та, 1983. 185 с.
2. Агibalов Г. П. Дискретные автоматы на полурешётках. Томск: Изд-во Том. ун-та, 1993. 227 с.
3. Парватов Н. Г. Об инвариантах некоторых классов квазимонотонных функций на полурешётке // Прикладная дискретная математика. 2009. №4. С. 21–28.
4. Парватов Н. Г. Функциональная полнота в замкнутых классах квазимонотонных и монотонных трёхзначных функций на полурешётке // Дискрет. анализ и исслед. операций. Сер. 1. 2003. Т. 10. № 1. С. 61–78.
5. Парватов Н. Г. Теорема о функциональной полноте в классе квазимонотонных функций на конечной полурешётке // Там же. Сер. 1. 2006. Т. 13. № 3. С. 62–82.
6. Курош А. Г. Лекции по общей алгебре. СПб.: Лань, 2005.
7. Яблонский С. В. Функциональные построения в k -значной логике // Тр. матем. ин-та им. В. А. Стеклова. 1958. Т. 51. С. 5–142.
8. Мальцев А. И. Итеративные алгебры Поста. Новосибирск: Изд-во Новосиб. ун-та, 1976.
9. Мальцев А. И. Итеративные алгебры и многообразия Поста // Алгебра и логика. 1966. Т. 5. № 2. С. 5–24.

10. *Марченко С. С.* К существованию конечных базисов в замкнутых классах булевых функций // Там же. 1984. Т. 23. № 1. С. 88–99.
11. *Baker K. A., Pixly A. F.* Polynomial interpolation and Chinese remainder theorem for algebraic systems // *Math. Zeitschr.* 1975. Bd. 143. N. 2. S. 165–174.
12. *Парватов Н. Г.* Замечания о конечной порождаемости замкнутых классов // *Дискрет. анализ и исслед. операций. Сер. 1.* 2004. Т. 11. № 3. С. 32–47.
13. *Парватов Н. Г.* Клоны с мажоритарной функцией и их обобщения // Там же. Сер. 1. 2010. Т. 17. № 3. С. 46–60.
14. *Wegener I.* The complexity of Boolean functions. Wiley-Teubner, 1987. 458 p.
15. *Алексеев В. Б.* От метода Карацубы для быстрого умножения чисел к быстрым алгоритмам для дискретных функций // *Тр. матем. ин-та им. В. А. Стеклова.* Т. 218. 1997. С. 20–27.
16. *Парватов Н. Г.* К синтезу формул, реализующих и представляющих квазимонотонные и монотонные функции на полурешётках подмножеств конечного множества // *Вестник Томского государственного университета.* 2000. Т. 2711. С. 111–115.

**ПОСТРОЕНИЕ КЛАССОВ СОВЕРШЕННО УРАВНОВЕШЕННЫХ
БУЛЕВЫХ ФУНКЦИЙ БЕЗ БАРЬЕРА¹**

С. В. Смышляев

*Московский государственный университет им. М. В. Ломоносова, г. Москва, Россия***E-mail:** smyshsv@gmail.com

Из результатов предыдущих работ, посвященных классу совершенно уравновешенных булевых функций (булевых функций без запрета), можно сделать вывод, что в данном классе особый интерес представляет подкласс функций без барьера. Ранее было доказано, что он не является пустым, тем не менее никаких оценок его мощности, отличных от тривиальных, предложено не было. В настоящей работе рассматриваются методы построения совершенно уравновешенных булевых функций без барьера, основанные на специального вида операции композиции булевых функций и на важных свойствах данной операции. Как следствие применения одного из методов получена нижняя оценка числа совершенно уравновешенных функций без барьера n переменных: $2^{2^{n-3}-n+2}$.

Ключевые слова: булевы функции без запрета, совершенно уравновешенные функции, барьеры булевых функций, фильтрующий генератор, криптография.

Введение

Понятие функции без запрета (совершенно уравновешенной функции) было рассмотрено в работах [1, 2], в тех же работах был получен ряд важных критериев для данного класса функций. В частности, из результатов [1, 2], а также работы [3] следует отсутствие у определенного класса преобразований двоичных последовательностей, построенных с помощью совершенно уравновешенных функций, некоторых криптографических слабостей.

В работе [4] было введено понятие свойства наличия у булевой функции барьера, достаточного для совершенной уравновешенности; были доказаны некоторые утверждения о данном классе функций. В частности, был получен результат о существовании совершенно уравновешенных функций без барьера, однако каких-либо нетривиальных оценок числа таких функций получено не было.

Изучение свойств функций с барьером было продолжено в [5, 6], и из результатов данных работ (а также работы [7]) следует наличие определенных криптографических слабостей у таких функций. Ввиду этих результатов особый интерес стала представлять задача построения широких классов совершенно уравновешенных функций без барьера.

Благодаря результатам работ [8, 9] некоторые примеры таких классов удалось получить [5, 10, 11], однако общих методов построения предложено не было, так же как и каких-либо оценок мощности таких классов.

В настоящей работе на основе общей схемы построения совершенно уравновешенных функций без барьера по определенным классам функций с барьером (классам функций с левым барьером без правого барьера) приводится важный для теорети-

¹Работа поддержана РФФИ (проект № 09-01-00653-а).

ческих исследований класс таких функций, явно заданных в полиномиальной форме (очень узким подклассом которого является представленный в [5] класс).

В основной части работы вводится ряд понятий, позволяющих формализовать и в полной мере описать метод построения классов совершенно уравновешенных функций без барьера с помощью помеченных графов специального вида. С помощью данного метода строится широкий класс функций с левым барьером без правого барьера.

В заключительной части работы приводится ряд новых результатов о совершенно уравновешенных функциях, в частности доказывается существование при произвольном n не менее $2^{2^{n-3}-n+2}$ совершенно уравновешенных булевых функций n переменных без барьера.

1. Основные определения и обозначения

Для множества двоичных наборов длины n будем использовать обозначение $V_n = \{0, 1\}^n$. Через \mathcal{F}_n будем обозначать множество булевых функций от n переменных.

Пусть $n, m \in \mathbb{N}$, $f \in \mathcal{F}_n$. Рассмотрим систему булевых уравнений:

$$f(x_s, x_{s+1}, \dots, x_{s+n-1}) = y_s, \quad s = 1, 2, \dots, m. \quad (1)$$

Обозначим для $f \in \mathcal{F}_n$ через f_m следующее отображение из V_{m+n-1} в V_m :

$$f_m(x_1, x_2, \dots, x_{m+n-1}) = (f(x_1, \dots, x_n), f(x_2, \dots, x_{n+1}), \dots, f(x_m, \dots, x_{m+n-1})). \quad (2)$$

Отображение f_m можно понимать как порождаемое m тактами работы фильтра — кодирующего устройства, полученного с помощью подключения входов булевой функции f (называемой в таком контексте фильтрующей функцией) к некоторым ячейкам двоичного регистра сдвига.

Определение 1 [2]. Булева функция $f \in \mathcal{F}_n$ называется функцией без запрета (функцией дефекта нуль), если соотношение

$$(f_m)^{-1}(\mathbf{y}) \neq \emptyset$$

выполняется для любого $m \in \mathbb{N}$ и любого $\mathbf{y} \in V_m$.

Определение 2 [2]. Булева функция $f \in \mathcal{F}_n$ называется совершенно уравновешенной, если соотношение

$$\#(f_m)^{-1}(\mathbf{y}) = 2^{n-1}$$

выполняется для любого $m \in \mathbb{N}$ и любого $\mathbf{y} \in V_m$. Множество совершенно уравновешенных функций из \mathcal{F}_n обозначим через \mathcal{PB}_n .

Введем понятие барьера булевой функции, тесно связанное с понятием совершенной уравновешенности.

Определение 3 [4]. Булева функция $f \in \mathcal{F}_n$ называется функцией с правым барьером длины b , если система уравнений

$$\begin{cases} f(x_1, x_2, \dots, x_n) = f(z_1, z_2, \dots, z_n), \\ f(x_2, x_3, \dots, x_{n+1}) = f(z_2, z_3, \dots, z_{n+1}), \\ \dots \\ f(x_{b-1}, x_b, \dots, x_{b+n-2}) = f(z_{b-1}, z_b, \dots, z_{b+n-2}), \\ x_1 = z_1, \dots, x_{n-1} = z_{n-1}, x_n = 0, z_n = 1 \end{cases} \quad (3)$$

имеет решение, а система

$$\begin{cases} f(x_1, x_2, \dots, x_n) = f(z_1, z_2, \dots, z_n), \\ f(x_2, x_3, \dots, x_{n+1}) = f(z_2, z_3, \dots, z_{n+1}), \\ \dots \\ f(x_{b-1}, x_b, \dots, x_{b+n-2}) = f(z_{b-1}, z_b, \dots, z_{b+n-2}), \\ f(x_b, x_{b+1}, \dots, x_{b+n-1}) = f(z_b, z_{b+1}, \dots, z_{b+n-1}), \\ x_1 = z_1, \dots, x_{n-1} = z_{n-1}, x_n = 0, z_n = 1 \end{cases} \quad (4)$$

решений не имеет.

Булева функция $f \in \mathcal{F}_n$ называется функцией с левым барьером длины b , если $f'(x_1, \dots, x_n) \equiv f(x_n, \dots, x_1)$ является функцией с правым барьером длины b .

Булева функция $f \in \mathcal{F}_n$ имеет барьер, если она имеет правый или левый барьер, или оба сразу. При этом длиной барьера функции называется соответственно длина правого барьера, левого барьера или меньшая из длин барьеров.

Замечание 1. Нетрудно заметить, что наличие правого (левого) барьера длины 1 означает линейность функции по последнему (первому) аргументу.

Для длины правого (левого) барьера функции f будем использовать обозначение b_f^R (b_f^L). Случай отсутствия у функции f правого (левого) барьера будем формально обозначать $b_f^R = \infty$ ($b_f^L = \infty$ соответственно).

Отметим, что для всех утверждений, в которых упоминается длина правого барьера некоторых функций, могут быть очевидным образом построены аналоги с использованием понятия левого барьера. Ввиду этого далее будем говорить только о правых барьерах функций.

2. Предварительные результаты

Утверждение 1 [2, 4]. Следующие преобразования множества \mathcal{F}_n оставляют инвариантным множество \mathcal{PB}_n :

- 1°. $\gamma_0: f(x_1, \dots, x_n) \rightarrow f(x_1, \dots, x_n) \oplus 1$;
- 2°. $\gamma_1: f(x_1, \dots, x_n) \rightarrow f(x_1 \oplus 1, \dots, x_n \oplus 1)$;
- 3°. $\gamma_2: f(x_1, \dots, x_n) \rightarrow f(x_n, \dots, x_1)$.

Для исследования свойств совершенно уравновешенных функций важен следующий критерий.

Теорема 1 [2, 12]. Пусть $n \in \mathbb{N}$ и $f \in \mathcal{F}_n$. Тогда следующие утверждения эквивалентны:

- f является совершенно уравновешенной;
- f является функцией без запрета;
- не существует двух различных двоичных последовательностей

$$\mathbf{x} = (x_1, x_2, \dots, x_r), \mathbf{z} = (z_1, z_2, \dots, z_r) \in V_r, r \geq 2n - 1,$$

таких, что

$$x_1 = z_1, x_2 = z_2, \dots, x_{n-1} = z_{n-1}; x_{r-n+2} = z_{r-n+2}, x_{r-n+3} = z_{r-n+3}, \dots, x_r = z_r; \\ f_{r-n+1}(\mathbf{x}) = f_{r-n+1}(\mathbf{z}).$$

Теорема 2 [4]. Наличие барьера у булевой функции является достаточным условием совершенной уравновешенности функции.

Замечание 2. В работе [4], кроме того, было показано, что наличие барьера не является необходимым условием совершенной уравновешенности.

Для построения классов совершенно уравновешенных булевых функций удобно пользоваться следующей конструкцией. Пусть $g \in \mathcal{F}_m$, $h \in \mathcal{F}_n$. Тогда определим функцию $f = g[h] \in \mathcal{F}_{m+n-1}$ следующим образом:

$$\begin{aligned} f(x_1, \dots, x_{m+n-1}) &= g[h](x_1, \dots, x_{m+n-1}) = \\ &= g(h(x_1, \dots, x_n), h(x_{n+1}, \dots, x_{2n}), \dots, h(x_m, \dots, x_{m+n-1})). \end{aligned}$$

Для данной конструкции верны следующие два утверждения.

Теорема 3 [8]. Пусть $g \in \mathcal{F}_m$, $h \in \mathcal{F}_n$. Функция $f = g[h] \in \mathcal{F}_{m+n-1}$ совершенно уравновешена тогда и только тогда, когда функции g и h совершенно уравновешены.

Теорема 4 [9]. Пусть $g \in \mathcal{F}_m$, $h \in \mathcal{F}_n$, $f = g[h]$. Тогда выполнено соотношение $\max\{b_h^R, b_g^R\} \leq b_f^R \leq b_h^R + b_g^R - 1$.

3. Основные результаты

С учетом теорем 3, 4 легко получить следующее утверждение, позволяющее строить классы совершенно уравновешенных булевых функций без барьера с помощью совершенно уравновешенных булевых функций без правого барьера.

Лемма 1. Пусть g, h совершенно уравновешены и принадлежат некоторому классу функций без правого барьера или получены из них с помощью преобразования γ_1 . Тогда функции $g[h^{\gamma_2}]$ и $g^{\gamma_2}[h]$ являются совершенно уравновешенными функциями без барьера.

В следующем утверждении представлен класс совершенно уравновешенных функций без правого барьера, явно заданных в полиномиальной форме. Доказательство не отличается существенно от доказательства аналогичного утверждения для узкого подмножества данного класса, приведенного в работе [5].

Теорема 5. Пусть

$$\begin{aligned} f &= x_1 \oplus x_{m_1} x_{m_1+1} \cdot h^{(1)}(x_{m_1+2}, x_{m_1+3}, \dots, x_n) \oplus \\ &\quad \oplus x_{m_2} x_{m_2+1} \cdot h^{(2)}(x_{m_2+2}, x_{m_2+3}, \dots, x_n) \oplus \dots \oplus \\ &\quad \oplus x_{m_k} x_{m_k+1} \cdot h^{(k)}(x_{m_k+2}, x_{m_k+3}, \dots, x_n), \end{aligned}$$

где $h^{(i)} \in \mathcal{F}_{n-m_i-1}$, $i = 1, 2, \dots, k$, — произвольные булевы функции; $m_{i+1} \geq m_i + 2$, $i = 1, 2, \dots, k-1$; $m_1 \geq 2$; $m_k \leq n-1$. Тогда если среди функций $h^{(i)}$, $i = 1, 2, \dots, k$, нечетное число функций принимает на единичном наборе значение 1, то f является совершенно уравновешенной функцией без правого барьера.

Для изложения метода, позволяющего строить значительно более широкие классы совершенно уравновешенных булевых функций без правого барьера, нам понадобится понятие графа сдвигов булевой функции. Данное понятие было введено в [4]; здесь приведем его в несколько другой форме, более удобной для требуемых построений.

Определение 4. Дополненным графом сдвигов функции $f \in \mathcal{F}_n$ называется ориентированный граф $\Gamma_f = (V, E)$, $\#V = 2^{2n-2}$ (без кратных ребер, с петлями), вершины которого поставлены во взаимно однозначное соответствие упорядоченным парам двоичных наборов длины $n-1$, причем для всяких $x_1, \dots, x_{n-1}, z_1, \dots, z_{n-1}, u_1, \dots, u_{n-1}, v_1, \dots, v_{n-1}$ верно, что дуга

$$\left(\left(\begin{array}{c} x_1, \dots, x_{n-1} \\ z_1, \dots, z_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} u_1, \dots, u_{n-1} \\ v_1, \dots, v_{n-1} \end{array} \right) \right)$$

присутствует в графе тогда и только тогда, когда выполнено следующее условие:

$$\begin{cases} (x_2, \dots, x_{n-1}) = (u_1, \dots, u_{n-2}), \\ (z_2, \dots, z_{n-1}) = (v_1, \dots, v_{n-2}). \end{cases}$$

При этом каждая дуга $\left(\begin{pmatrix} x_1, \dots, x_{n-1} \\ z_1, \dots, z_{n-1} \end{pmatrix} \rightarrow \begin{pmatrix} x_2, \dots, x_n \\ z_2, \dots, z_n \end{pmatrix} \right)$ помечается значением $f(x_1, x_2, \dots, x_{n-1}, x_n) \oplus f(z_1, z_2, \dots, z_{n-1}, z_n)$. Каждому ориентированному пути в дополненном графе сдвигов Γ_f естественным образом соответствует пара двоичных последовательностей, составленных из меток вершин.

Через $I_f \subset \Gamma_f$ обозначим подграф дополненного графа сдвигов, отвечающий множеству пар равных наборов длины $n - 1$; через Γ_f^* — граф, полученный из графа Γ_f удалением всех ребер, лежащих внутри I_f . Γ_f^* называется графом сдвигов функции f .

С использованием теоремы 1 легко доказать следующее утверждение.

Лемма 2. Функция f совершенно уравновешена и не имеет правого барьера в том и только в том случае, когда выполнены следующие условия:

- 1) в Γ_f^* нет пути ненулевой длины по дугам, помеченным нулем, с началом и концом в подграфе I_f ;
- 2) в Γ_f^* существует путь по дугам, помеченным нулем, ведущий из I_f в некоторый ориентированный цикл, проходящий также исключительно через помеченные нулем дуги в графе Γ_f^* .

Учитывая данное утверждение, опишем общую схему метода построения булевых функций из \mathcal{PB}_n без правого барьера. Рассмотрим граф $\Gamma_{(n)}^*$, представляющий собой граф сдвигов произвольной булевой функции из \mathcal{F}_n без пометок дуг; аналогично введем графы $\Gamma_{(n)}$ и $I_{(n)}$. Для построения множества графов сдвигов Γ_f^* некоторого класса совершенно уравновешенных булевых функций без правого барьера производятся следующие действия:

- 1) выделяется некоторый цикл в графе $\Gamma_{(n)}^*$;
- 2) выделяется некоторая вершина в графе $I_{(n)}$ и выбирается некоторый путь из этой вершины в выделенный цикл по дугам графа $\Gamma_{(n)}^*$;
- 3) выделяется некоторое сечение графа $\Gamma_{(n)}^*$, пересекающее все пути ненулевой длины, имеющие начало и конец в подграфе $I_{(n)}$;
- 4) производится частичная разметка дуг $\Gamma_{(n)}^*$ таким образом, что:
 - все дуги выделенного цикла становятся помечены нулями;
 - все дуги выбранного пути от выделенной вершины в цикл становятся помечены нулями;
 - все дуги выбранного сечения становятся помечены единицами;
 - остается возможной корректная разметка оставшихся дуг графа до графа сдвигов некоторой булевой функции.

Выбор и соответствующая разметка цикла в графе $\Gamma_{(n)}^*$ и пути до этого цикла гарантируют отсутствие правого барьера у любой функции f , до графа сдвигов которой разметкой оставшихся дуг можно достроить получившийся граф; разметка сечения гарантирует совершенную уравновешенность любой такой функции.

Таким образом, центральным вопросом становится возможность разметить оставшиеся дуги графа $\Gamma_{(n)}^*$ так, чтобы получить граф сдвигов некоторой булевой функции f .

Определение 5. Пусть $\Gamma_{(n)}^* = (V, E^*)$. Разметка дуг графа $\Gamma_{(n)}^*$ $\varphi: E^* \mapsto \{0, 1\}$ называется корректной, если она удовлетворяет следующим условиям:

1) для любых $x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_n$ верно равенство

$$\varphi \left(\left(\begin{array}{c} x_1, \dots, x_{n-1} \\ z_1, \dots, z_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} x_2, \dots, x_n \\ z_2, \dots, z_n \end{array} \right) \right) = \varphi \left(\left(\begin{array}{c} z_1, \dots, z_{n-1} \\ x_1, \dots, x_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} z_2, \dots, z_n \\ x_2, \dots, x_n \end{array} \right) \right);$$

2) при любых $x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_n, z_1, z_2, \dots, z_n$ верно равенство

$$\begin{aligned} & \varphi \left(\left(\begin{array}{c} x_1, \dots, x_{n-1} \\ z_1, \dots, z_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} x_2, \dots, x_n \\ z_2, \dots, z_n \end{array} \right) \right) = \\ = & \varphi \left(\left(\begin{array}{c} z_1, \dots, z_{n-1} \\ u_1, \dots, u_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} z_2, \dots, z_n \\ u_2, \dots, u_n \end{array} \right) \right) \oplus \varphi \left(\left(\begin{array}{c} x_1, \dots, x_{n-1} \\ u_1, \dots, u_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} x_2, \dots, x_n \\ u_2, \dots, u_n \end{array} \right) \right). \end{aligned}$$

Нетрудно доказать следующее утверждение.

Лемма 3. Граф $\Gamma_{(n)}^*$ с пометками на дугах, соответствующими разметке φ , является графом сдвигов некоторой булевой функции f (а точнее, ровно двух, отличающихся только свободными членами их полиномов) тогда и только тогда, когда φ — корректная разметка.

Таким образом, по корректной разметке φ графа $\Gamma_{(n)}^*$ мы можем однозначно (если, например, дополнительно потребуем $f(0, 0, \dots, 0) = 0$, т. е. $f \in T_0$) восстановить функцию f , такую, что Γ_f^* совпадает с размеченным в соответствии с φ графом $\Gamma_{(n)}^*$.

Чтобы выделять просто устроенные классы корректных разметок, будем использовать следующее понятие.

Определение 6. Пусть $n \in \mathbb{N}$; $G = (V', E')$ — неориентированный граф (без кратных ребер и петель) на 2^n вершинах, вершины которого поставлены во взаимно однозначное соответствие наборам из V_n и ребрам которого приписаны значения 0 и 1 в соответствии с функцией $\psi: E' \mapsto \{0, 1\}$. Через $\tilde{\varphi}_{(G, \psi)}$ будем обозначать частичную разметку графа $\Gamma_{(n)}^*$, полученную в соответствии со следующим правилом: для любых $x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_n$, таких, что в графе G есть ребро e между вершинами (x_1, \dots, x_n) и (z_1, \dots, z_n) , выполнено

$$\tilde{\varphi}_{(G, \psi)} \left(\left(\begin{array}{c} x_1, \dots, x_{n-1} \\ z_1, \dots, z_{n-1} \end{array} \right) \longrightarrow \left(\begin{array}{c} x_2, \dots, x_n \\ z_2, \dots, z_n \end{array} \right) \right) = \psi(e).$$

Лемма 4. Пусть $G = (V', E')$ — неориентированный граф без петель и кратных ребер на 2^n вершинах, вершинам которого поставлены во взаимно однозначное соответствие наборы из V_n . Частичная разметка $\tilde{\varphi}_{(G, \psi)}$ при любом выборе функции $\psi: E' \mapsto \{0, 1\}$ однозначным образом дополнима до корректной разметки графа $\Gamma_{(n)}^*$ (в таком случае будем обозначать ее через $\varphi_{(G, \psi)}$) тогда и только тогда, когда G является деревом.

Следствие 1. С учетом лемм 3 и 4 легко получить, что если граф $G = (V', E')$ удовлетворяет условиям леммы 4, то при любом выборе функции $\psi: E' \mapsto \{0, 1\}$ пара (G, ψ) однозначно определяет пару булевых функций $\{f, f \oplus 1\}$.

Замечание 3. Так как свойства булевых функций f и $f \oplus 1$ с точки зрения совершенной уравновешенности и наличия барьеров идентичны, ниже для определенности будем рассматривать только функции, принимающие значение 0 на нулевом наборе

получим, что мощность класса $S_{(G, \tilde{\psi})}$ в точности равна $2^{2^{n-1}-n-1}$, откуда и следует требуемое утверждение. ■

Замечание 5. Для простоты при доказательстве теоремы 6 для обеспечения отсутствия правого барьера у всех функций из порождаемого класса мы явно задавали в графе сдвигов каждой функции из данного класса цикл $\begin{pmatrix} 0, 1, 0, 1, 0, 1, \dots, 0, 1, \dots \\ 1, 0, 1, 0, 1, 0, \dots, 1, 0, \dots \end{pmatrix}$, проходимый по помеченным нулями дугам. Пользуясь аналогичными приемами, нетрудно доказать, что для произвольной пары периодических последовательностей, таких, что для наименьшего общего кратного T их минимальных периодов выполняются неравенства $2 \leq T \leq n-3-2 \log_2((n-3)(n-5)+1)$, можно построить класс $S_{(G, \tilde{\psi})}$ из $2^{2^{n-1}-n}$ совершенно уравновешенных функций без правого барьера, каждая из которых в графе сдвигов содержит цикл (по помеченным нулями дугам), образованный выбранной парой, и путь к нему из подграфа I_f по помеченным нулями дугам.

Рассмотрим некоторые свойства описанной выше операции композиции специального вида ($f = g[h]$).

Лемма 5. Пусть $n \in \mathbb{N}$, $h \in \mathcal{F}_n$. Тогда $h \in \mathcal{PB}_n$ тогда и только тогда, когда ни при каком $m \in \mathbb{N}$ не существует двух различных функций $g^{(1)}, g^{(2)} \in \mathcal{F}_m$, для которых выполняется $g^{(1)}[h] = g^{(2)}[h]$.

Доказательство. Если $h \notin \mathcal{PB}_n$, то, как следует из теоремы 1, при некотором $m^* \in \mathbb{N}$ существует набор $\mathbf{z} \in V_{m^*}$, не принадлежащий образу отображения h_{m^*} . Положим $m = m^*$ и рассмотрим произвольную пару функций $g^{(1)}, g^{(2)} \in \mathcal{F}_m$, совпадающих на всех наборах, за исключением набора \mathbf{z} . Нетрудно заметить, что $g^{(1)} \neq g^{(2)}$ и $g^{(1)}[h] = g^{(2)}[h]$.

Пусть теперь $h \in \mathcal{PB}_n$, $m \in \mathbb{N}$, $g^{(1)}, g^{(2)}$ — произвольная пара различных функций из \mathcal{F}_m , $f^{(1)} = g^{(1)}[h]$, $f^{(2)} = g^{(2)}[h]$. Зафиксируем набор $\mathbf{z} \in V_m$, такой, что $g^{(1)}(\mathbf{z}) \neq g^{(2)}(\mathbf{z})$. Так как функция h совершенно уравновешена, то найдется $\mathbf{x} \in V_{m+n-1}$, такой, что $h_m(\mathbf{x}) = \mathbf{z}$. Отсюда $f^{(1)}(\mathbf{x}) = g^{(1)}(h_m(\mathbf{x})) = g^{(1)}(\mathbf{z}) \neq g^{(2)}(\mathbf{z}) = g^{(2)}(h_m(\mathbf{x})) = f^{(2)}(\mathbf{x})$ и $f^{(1)} \neq f^{(2)}$. ■

Лемма 6. Пусть $n \in \mathbb{N}$, $h^{(1)}, h^{(2)}, h^{(3)} \in \mathcal{F}_n$, причем $h^{(i)} \neq h^{(j)}$ при $i \neq j$. Пусть $m \in \mathbb{N}$, $g \in \mathcal{PB}_m$, $f^{(i)} = g[h^{(i)}]$, $i = 1, 2, 3$. Тогда по меньшей мере две из функций $f^{(1)}, f^{(2)}, f^{(3)}$ различны.

Доказательство. Очевидно, что среди функций $h^{(1)}, h^{(2)}, h^{(3)}$ найдутся две, $h^{(i)}$ и $h^{(j)}$, $i \neq j$, совпадающие на нулевом наборе. Так как, по условию, $h^{(i)} \neq h^{(j)}$, то найдется набор $\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n) \in V_n$, такой, что $h^{(i)}(\tilde{\mathbf{x}}) \neq h^{(j)}(\tilde{\mathbf{x}})$.

Рассмотрим набор $\mathbf{x} \in V_{2m+3n-4}$, $\mathbf{x} = (\underbrace{0, 0, \dots, 0}_{m+n-2}, \tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n, \underbrace{0, 0, \dots, 0}_{m+n-2})$. Очевидно, что для доказательства утверждения достаточно показать, что $f_{m+2n-2}^{(i)}(\mathbf{x}) \neq f_{m+2n-2}^{(j)}(\mathbf{x})$.

Предположим противное: $f_{m+2n-2}^{(i)}(\mathbf{x}) = f_{m+2n-2}^{(j)}(\mathbf{x})$. Тогда выполнена следующая система (приведем общий вид системы для случая $m \geq n+1$):

$$\left\{ \begin{array}{l}
 g(h^{(i)}(0, 0, \dots, 0), h^{(i)}(0, 0, \dots, 0), \dots, h^{(i)}(0, 0, \dots, 0), h^{(i)}(0, 0, \dots, 0, \tilde{x}_1)) = \\
 = g(h^{(j)}(0, 0, \dots, 0), h^{(j)}(0, 0, \dots, 0), \dots, h^{(j)}(0, 0, \dots, 0), h^{(j)}(0, 0, \dots, 0, \tilde{x}_1)), \\
 g(h^{(i)}(0, 0, \dots, 0), \dots, h^{(i)}(0, 0, \dots, 0, \tilde{x}_1), h^{(i)}(0, 0, \dots, 0, \tilde{x}_1, \tilde{x}_2)) = \\
 = g(h^{(j)}(0, 0, \dots, 0), \dots, h^{(j)}(0, 0, \dots, 0, \tilde{x}_1), h^{(j)}(0, 0, \dots, 0, \tilde{x}_1, \tilde{x}_2)), \\
 \dots \\
 g(h^{(i)}(0, 0, \dots, 0), \dots, h^{(i)}(0, \tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_{n-1}), h^{(i)}(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)) = \\
 = g(h^{(j)}(0, 0, \dots, 0), \dots, h^{(j)}(0, \tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_{n-1}), h^{(j)}(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)), \\
 \dots \\
 g(h^{(i)}(\tilde{x}_n, 0, \dots, 0), h^{(i)}(0, 0, \dots, 0), \dots, h^{(i)}(0, 0, \dots, 0)) = \\
 = g(h^{(j)}(\tilde{x}_n, 0, \dots, 0), h^{(j)}(0, 0, \dots, 0), \dots, h^{(j)}(0, 0, \dots, 0)); \\
 h^{(i)}(0, 0, \dots, 0) = h^{(j)}(0, 0, \dots, 0), \\
 h^{(i)}(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n) \neq h^{(j)}(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n).
 \end{array} \right. \quad (5)$$

Нетрудно видеть, что система (5) по теореме 1 не может быть выполнена в случае совершенно уравновешенной g . Полученное противоречие с условием завершает доказательство утверждения. ■

Замечание 6. Заметим, что более сильное утверждение о том, что в случае совершенно уравновешенной g из неравенства $h^{(1)} \neq h^{(2)}$ следует $g[h^{(1)}] \neq g[h^{(2)}]$, вообще говоря, верным не является. Для построения контрпримера достаточно рассмотреть функции $g(x_1, x_2) = x_1 \oplus x_2 \in \mathcal{PB}_2$ и $h^{(2)} = h^{(1)} \oplus 1$, где $h^{(1)}$ — произвольная булева функция.

Теорема 7. Пусть $n \in \mathbb{N}$; $b \in \mathbb{N}$ или $b = \infty$. Мощность множества функций из \mathcal{PB}_{n+2} с левым барьером длины b без правого барьера не меньше мощности множества функций из \mathcal{PB}_n с левым барьером длины b .

Доказательство. Нетрудно заметить, что для доказательства утверждения достаточно построить для всякого $n \in \mathbb{N}$ отображение $\Phi_n: \mathcal{PB}_n \mapsto \mathcal{PB}_{n+2}$, удовлетворяющее следующим условиям:

- 1) для всякой $f \in \mathcal{PB}_n$ верно $b_{\Phi_n(f)}^R = \infty$;
- 2) для всякой $f \in \mathcal{PB}_n$ верно $b_{\Phi_n(f)}^L = b_f^L$;
- 3) Φ_n инъективно.

Для всякой $f \in \mathcal{PB}_n$ положим $\Phi_n(f) = f[h]$, где $h(x_1, x_2, x_3) = x_1 \oplus x_2 x_3$. По теореме 3 если $f \in \mathcal{PB}_n$, то $\Phi_n(f) \in \mathcal{PB}_{n+2}$. Как следует из теоремы 5, $b_h^R = \infty$, поэтому, как следует из теоремы 4, $b_{\Phi_n(f)}^R = \infty$ для любой $f \in \mathcal{PB}_n$, и условие 1 выполнено. Функция h линейна по первой переменной, то есть $b_h^L = 1$, поэтому, по теореме 4, для всякой $f \in \mathcal{PB}_n$ выполняется соотношение $\max\{b_f^L, 1\} \leq b_{\Phi_n(f)}^L \leq b_f^L + 1 - 1$, $b_{\Phi_n(f)}^L = b_f^L$, и условие 2 выполнено. Чтобы доказать, что определенное таким образом Φ_n является инъективным, достаточно заметить, что $h \in \mathcal{PB}_3$, и применить лемму 5. ■

Следствие 2. При любом $n \in \mathbb{N}$ число функций из \mathcal{PB}_{n+2} без правого барьера больше числа функций из \mathcal{PB}_n с правым барьером.

Доказательство. При $n = 1, 2$ данный результат можно получить непосредственно из полученной в работе [4] классификации. Пусть теперь $n \geq 3$. Булевых функций с правым барьером в множестве \mathcal{PB}_n ровно столько же, сколько функций

с левым барьером, каждой из которых, как показано в теореме 7, можно поставить в соответствие с помощью отображения Φ_n свою функцию с левым барьером без правого барьера из множества \mathcal{PB}_{n+2} . Учитывая при всяком $n \geq 3$ существование в множестве \mathcal{PB}_{n+2} функций без барьера (которые, в соответствии с теоремой 4, не могут быть получены отображением Φ_n ни из каких функций с левым барьером), получим требуемое утверждение. ■

Непосредственно из теоремы 7 легко получить следующее утверждение.

Следствие 3. При любом $n \geq 5$ существует не менее $2^{2^{n-3}-n+2}$ совершенно уравновешенных булевых функций без барьера.

ЛИТЕРАТУРА

1. Hedlund G. A. Endomorphisms and automorphisms of the shift dynamical system // Math. Sys. Theory. 1969. No. 3. P. 320–375.
2. Сумароков С. Н. Запреты двоичных функций и обратимость для одного класса кодирующих устройств // Обзорение прикладной и промышленной математики. 1994. Т. 1. Вып. 1. С. 33–55.
3. Anderson R. J. Searching for the Optimum Correlation Attack // LNCS. 1995. V. 1008. P. 137–143.
4. Логачев О. А., Смышляев С. В., Яценко В. В. Новые методы изучения совершенно уравновешенных булевых функций // Дискретная математика. 2009. Т. 21. Вып. 2. С. 51–74.
5. Смышляев С. В. О некоторых свойствах совершенно уравновешенных булевых функций // Материалы Четвертой Междунар. научн. конф. по проблемам безопасности и противодействия терроризму (МГУ им. М. В. Ломоносова, Москва, 30–31 октября 2008). М.: МЦНМО, 2009. С. 57–64.
6. Смышляев С. В. О криптографических слабостях некоторых классов преобразований двоичных последовательностей // Прикладная дискретная математика. 2010. № 1(7). С. 5–15.
7. Golic Dj. J. On the Security of Nonlinear Filter Generators // LNCS. 1996. V. 1039. P. 173–188.
8. Логачев О. А. Об одном классе совершенно уравновешенных булевых функций // Материалы Третьей Междунар. научн. конф. по проблемам безопасности и противодействия терроризму (МГУ им. М. В. Ломоносова, Москва, 25–27 октября 2007). М.: МЦНМО, 2008. С. 137–141.
9. Смышляев С. В. Барьеры совершенно уравновешенных булевых функций // Дискретная математика. 2010. Т. 22. Вып. 2. С. 66–79.
10. Смышляев С. В. О совершенно уравновешенных булевых функциях без барьера // Материалы Восьмой Междунар. научн. конф. «Дискретные модели в теории управляющих систем» (МГУ им. М. В. Ломоносова, Москва, 6–9 апреля 2009). М.: МАКС Пресс, 2009. С. 278–284.
11. Смышляев С. В. О преобразовании двоичных последовательностей с помощью совершенно уравновешенных булевых функций // Материалы Пятой Междунар. научн. конференции по проблемам безопасности и противодействия терроризму (МГУ им. М. В. Ломоносова, Москва, 29–30 октября 2009). М.: МЦНМО, 2010. С. 31–41.
12. Логачев О. А., Сальников А. А., Яценко В. В. Булевы функции в теории кодирования и криптологии. М.: МЦНМО, 2004.

МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

DOI 10.17223/20710410/9/5

УДК 519.7

ЭЛЕМЕНТЫ ТЕОРИИ СТАТИСТИЧЕСКИХ АНАЛОГОВ
ДИСКРЕТНЫХ ФУНКЦИЙ С ПРИМЕНЕНИЕМ
В КРИПТОАНАЛИЗЕ ИТЕРАТИВНЫХ БЛОЧНЫХ ШИФРОВ¹

Г. П. Агибалов, И. А. Панкратова

*Томский государственный университет, г. Томск, Россия***E-mail:** agibalov@isc.tsu.ru, pank@isc.tsu.ru

Вводится понятие статистической независимости булевой функции от подмножества аргументов. На его основе определяется понятие статистического аналога дискретной функции как булева уравнения, выполняемого с некоторой вероятностью, и изучаются его свойства. Формулируются конструктивные тесты статистической независимости. Излагаются методы построения линейных статистических аналогов функций итеративного блочного шифрования с аддитивным раундовым ключом и некоторые алгоритмы криптоанализа симметричных шифров, основанные на решении систем линейных и нелинейных статистических аналогов методом максимального правдоподобия. Приводимые определения, методы и алгоритмы иллюстрируются на примере DES. В частности, показано, что одним из алгоритмов криптоанализа, предложенных в статье, можно найти 34 бита ключа 16-раундового DES, используя пару известных статистических аналогов, на базе которых алгоритм М. Matsui доставляет только 26 из этих бит. Статья может служить учебно-методическим пособием по теме в заголовке, в том числе по линейному криптоанализу.

Ключевые слова: *статистическая независимость, статистические аналоги функций, итеративные блочные шифры, криптоанализ, линейный криптоанализ, нелинейный криптоанализ, DES.*

Введение

Известно, что в криптоанализе двоичных симметричных шифров значительную роль играют системы булевых уравнений и методы их решения [1]. Системой уравнений переменные биты открытого текста и соответствующего шифртекста связываются с неизвестными битами ключа в соответствии с алгоритмом шифрования. Путём решения этой системы с известными битами открытого текста и шифртекста как раз и достигается цель криптоанализа — находятся (все или некоторые) биты ключа. Известно, однако, что решение произвольной системы нелинейных уравнений (пусть даже степени 2) имеет экспоненциальную сложность. Есть много приёмов «упрощения» систем уравнений, благодаря которым система, не поддающаяся никакому методу, после упрощения нередко становится решаемой некоторым методом за приемлемое время. Один из таких приёмов, восходящий к [2, 3], заключается в замене заданной системы уравнений E системой её так называемых приближённых соотношений (approximate

¹Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П11010).

expressions), где каждое соотношение e является булевым уравнением, связывающим переменные системы E и выполняющимся с некоторой вероятностью $p \neq 1/2$. Чем более простыми (в некотором смысле) выбраны приближённые соотношения для системы E (например, линейными), тем легче решается система из них с подставленными открытыми и шифртекстами, а чем больше количество соотношений и выше эффективность каждого соотношения в ней, выражающаяся для e разностью $|p - 1/2|$, тем выше результативность полученного решения, т. е. вероятность того, что корень системы приближённых соотношений будет корнем системы E . Именно так устроен, например, линейный криптоанализ двоичных симметричных шифров, в котором вместо системы уравнений шифрования применяется система её линейных приближённых соотношений с ненулевыми эффективностями, решаемая при известных значениях бит открытого текста и шифртекста алгоритмом полиномиальной сложности. Для достижения достаточно высокой результативности такого решения обычно требуется иметь много различных открытых текстов и соответствующих шифртекстов, чтобы получить достаточно большое число независимых уравнений из системы приближённых соотношений.

В данной работе, написанной, главным образом, с целью уточнения, формализации и дальнейшего развития понятийного аппарата линейного криптоанализа Mitsuru Matsui [2], излагаются элементы теории статистических аналогов, выступающих в криптоанализе в роли приближённых соотношений М. Matsui, не обязательно линейных. Они вводятся для функций шифрования как булевы уравнения, которые выполняются с некоторой вероятностью, связывают переменные символов открытого текста, его шифртекста и ключа и обладают свойством статистической независимости ассоциированных с ними булевых функций от переменных символов открытого текста. Последнее свойство существенным образом отличает статистический аналог от приближённого соотношения и гарантирует сохранение вероятности аналога после подстановки в него открытого текста и соответствующего шифртекста, чего может не быть для приближённого соотношения. Формулируются конструктивные тесты статистической независимости булевой функции от подмножества её аргументов. Доказывается сохраняемость вероятности статистического аналога при любом фиксировании в нём символов соответствующих открытого и шифрованного текстов. Для суперпозиции двух дискретных функций определяется суперпозиция одной из них (внутренней) и статистического аналога другой (внешней) и показывается, что в случае аддитивности внутренней функции полученная суперпозиция является функцией статистического аналога для первой суперпозиции с вероятностью статистического аналога её внешней функции. Излагаются методы построения линейных статистических аналогов для функций блоков замены, раундовых функций и многораундовых шифров с аддитивным раундовым ключом и алгоритмы криптоанализа итеративных блочных шифров путём решения систем линейных и нелинейных статистических аналогов функций шифрования методом максимального правдоподобия. Изложение иллюстрируется примерами из криптоанализа DES. Показано, в частности, что один из предложенных здесь алгоритмов криптоанализа позволяет на основе пары нелинейных статистических аналогов 16-раундового DES, построенных М. Matsui, найти 34 бита ключа DES, в то время как алгоритм самого М. Matsui [3] на основе тех же двух приближённых соотношений получает только 26 из этих бит. Работа может быть рекомендована в качестве учебно-методического пособия по рассматриваемой теме.

1. Статистическая независимость

Для любой булевой функции f и для любого подмножества U её аргументов будем говорить, что f *статистически не зависит* от переменных множества U , если для любой её подфункции f' , полученной фиксированием значений всех переменных в U , имеет место $\Pr[f' = 0] = \Pr[f = 0]$, где для булевой функции g от s переменных, имеющей в своём векторе значений ровно $w_0(g)$ символов 0, $\Pr[g = 0] = w_0(g)2^{-s}$.

Пусть далее \oplus есть сложение в \mathbb{Z}_2 , т.е. по mod 2. Считаем, что в применении к булевым векторам эта операция выполняется покомпонентно.

Утверждение 1. Функция $f(x, k) = g(x \oplus k)$, где x, k — переменные со значениями в $(\mathbb{Z}_2)^n$, статистически не зависит от переменных в x .

Доказательство. В самом деле, $w_0(f) = 2^n w_0(g)$ и $g(x \oplus k)$ при любом фиксированном $x = a$ пробегает всё множество значений функции g . Таким образом, $\Pr[f(x, k) = 0] = w_0(f)/2^{2n} = w_0(g)/2^n$ и $\Pr[f(a, k) = 0] = w_0(g(a \oplus k))/2^n = w_0(g)/2^n$. ■

Через (a, b) будем обозначать скалярное произведение булевых векторов a и b .

Утверждение 2. Функция $f(x, y)$, где x, y — переменные со значениями в $(\mathbb{Z}_2)^n$ и $(\mathbb{Z}_2)^m$ соответственно, статистически не зависит от переменных в x , если и только если функция $f(x, y) \oplus (u, x)$ уравновешена для любого ненулевого вектора $u \in (\mathbb{Z}_2)^n$.

Доказательство. Обозначим через $w_1(f)$ вес функции f (количество единиц в её векторе значений). Непосредственно проверяется, что f статистически не зависит от переменных в x , если и только если $w_1(f(a, y)) = w_1(f)/2^n$ для любого вектора $a \in (\mathbb{Z}_2)^n$.

Необходимость. Разложим функцию $f(x, y) \oplus (u, x)$ по всем переменным в x ; коэффициенты этого разложения имеют вид $f_a(y) = f(a, y) \oplus (u, a)$ для всевозможных $a \in (\mathbb{Z}_2)^n$. Если $(u, a) = 0$ (а это условие при фиксированном ненулевом u выполняется ровно для половины всех a), то $w_1(f_a) = w_1(f(a, y)) = w_1(f)/2^n$. Если же $(u, a) = 1$, то $w_1(f_a) = 2^m - w_1(f(a, y)) = 2^m - w_1(f)/2^n$. Известно, что вес функции равен сумме весов коэффициентов её разложения; запишем:

$$w_1(f(x, y) \oplus (u, x)) = 2^{n-1} w_1(f)/2^n + 2^{n-1} (2^m - w_1(f)/2^n) = 2^{n+m-1},$$

что и доказывает уравновешенность функции $f(x, y) \oplus (u, x)$.

Достаточность. Докажем сначала, что $w_1(f(a, y)) = w_1(f)/2^n$ для нулевого вектора a . Снова запишем вес функции $f(x, y) \oplus (u, x)$ как сумму весов коэффициентов разложения и учтём уравновешенность этой функции:

$$\begin{aligned} w_1(f(x, y) \oplus (u, x)) &= 2^{n+m-1} = \sum_{a \in (\mathbb{Z}_2)^n} w_1(f(a, y) \oplus (u, a)) = \\ &= \sum_{a, (u, a)=0} w_1(f(a, y)) + \sum_{a, (u, a)=1} (2^m - w_1(f(a, y))), \end{aligned}$$

откуда $\sum_{a, (u, a)=0} w_1(f(a, y)) = \sum_{a, (u, a)=1} w_1(f(a, y))$. Просуммируем обе части последнего равенства по всем $u \neq 0$:

$$\sum_{u \neq 0} \sum_{\substack{a, \\ (u, a)=0}} w_1(f(a, y)) = \sum_{u \neq 0} \sum_{\substack{a, \\ (u, a)=1}} w_1(f(a, y)).$$

Заметим, что при любом фиксированном $a \neq 0$ и всевозможных $u \neq 0$ равенство $(u, a) = 1$ выполняется 2^{n-1} раз, а равенство $(u, a) = 0$ верно в остальных $(2^{n-1} - 1)$

случаях. При $a = 0$ всегда $(u, a) = 0$. Поэтому получим

$$(2^n - 1)w_1(f(0, y)) + (2^{n-1} - 1)\sum_{a \neq 0} w_1(f(a, y)) = 2^{n-1}\sum_{a \neq 0} w_1(f(a, y)),$$

откуда

$$\sum_{a \in (\mathbb{Z}_2)^n} w_1(f(a, y)) = 2^n w_1(f(0, y))$$

и $w_1(f(0, y)) = w_1(f)/2^n$.

Для случая $a \neq 0$ рассмотрим функцию $g(x, y) = f(x \oplus a, y)$. Ясно, что $f(a, y) = g(0, y)$; кроме того, функция $g(x, y) \oplus (u, x)$ уравновешена в случае уравновешенности $f(x, y) \oplus (u, x)$, так как

$$\begin{aligned} w_1(f(x, y) \oplus (u, x)) &= \sum_{x, y} (f(x, y) \oplus (u, x)) = \\ &= \sum_{x, y} (f(x \oplus a, y) \oplus (u, x \oplus a)) = \sum_{x, y} (g(x, y) \oplus (u, x) \oplus (u, a)). \end{aligned}$$

Последняя сумма здесь (в зависимости от значения (u, a)) есть вес функции $g(x, y) \oplus (u, x)$ или её отрицания, что для уравновешенной функции одно и то же. По доказанному выше $w_1(g(0, y)) = w_1(g)/2^n$, т. е. $w_1(f(a, y)) = w_1(f)/2^n$. ■

С использованием преобразования Уолша — Адамара (см., например, [4]) тест может быть переформулирован следующим (более конструктивным) образом: функция $f(x, y)$ статистически не зависит от переменных в x , если и только если для любого ненулевого вектора $u \in (\mathbb{Z}_2)^n$ имеет место равенство $\hat{f}(u, 0) = 0$, где \hat{f} — преобразование Уолша — Адамара функции f .

2. Понятие статистического аналога

2.1. Основные определения

Рассмотрим произвольную функцию $F : X \times K \rightarrow Y$, где $X = (\mathbb{Z}_2)^n$, $K = (\mathbb{Z}_2)^m$, $Y = (\mathbb{Z}_2)^r$ для некоторых натуральных n , r и целого $m \geq 0$. В частности, F может быть функцией одного раунда итеративного блочного шифра, и тогда X и Y суть множества блоков соответственно на входе и выходе раунда, а K — множество раундовых ключей. Ею может быть и функция симметричного шифрования открытых текстов из X в шифртексты из Y на ключах из K . При $m = 0$ функция F рассматривается как отображение $F : X \rightarrow Y$. В этом случае она может быть функцией, например, бесключевого блока замены. Следующие определения предполагают $m \geq 1$.

Статистическим аналогом (СА) функции F называется всякое уравнение $\varphi(x, y, k) = 0$, в котором $x = x_1 x_2 \dots x_n$, $y = y_1 y_2 \dots y_r$, $k = k_1 k_2 \dots k_m$ — переменные (булевы векторы) со значениями в X , Y , K соответственно, связанные соотношением $y = F(x, k)$, и $\varphi : X \times Y \times K \rightarrow \mathbb{Z}_2$ — булева функция от $n + m + r$ переменных, существенно зависящая хотя бы от одной переменной в каждом из наборов x , y и k , такая, что функция $\varphi_F(x, k) = \varphi(x, F(x, k), k)$, называемая *ассоциированной* с этим СА, статистически не зависит от переменных в x . Число $p = \text{Pr}[\varphi_F = 0]$ называется *вероятностью* данного СА. Говорят также, что он *выполняется с вероятностью p* и имеет *эффективность* $\varepsilon = |p - 1/2|$. СА называют *эффективным*, если $p \neq 1/2$, или, что то же самое, $\varepsilon > 0$. Функция φ в нём называется *функцией* самого аналога, который, в свою очередь, именуется как СА, *заданный* этой функцией.

Эти определения легко переписываются на случай $m = 0$, а именно: опускаются все вхождения символа k и требование статистической независимости φ_F от x . Таким

образом, в этом случае фактически имеем дело с функцией $F(x)$, с её СА $\varphi(x, y) = 0$, где $\varphi(x, y)$ — любая булева функция от $n + r$ переменных, и с его вероятностью $p = \Pr[\varphi_F(x) = 0]$, где $\varphi_F(x) = \varphi(x, F(x))$.

В случае $m > 0$ статистическая независимость функции $\varphi_F(x, k)$ от x придаёт заданному функцией φ СА функции F следующее важное свойство: фиксирование в уравнении СА для F любого значения x и того значения y , в которое F преобразует это x при равновероятно выбранном k , не изменяет вероятности выполнения этого уравнения. Строго говоря, верно следующее

Утверждение 3. Пусть СА $\varphi(x, y, k) = 0$ функции $F(x, k)$ имеет вероятность p . Пусть также $x^{(i)}$ — произвольное значение переменной x и $y^{(i)} = F(x^{(i)}, k)$ для некоторого k , выбранного в K случайно с вероятностью 2^{-m} . Тогда $\Pr[\varphi(x^{(i)}, y^{(i)}, k) = 0] = p$.

Доказательство. В самом деле, $\Pr[\varphi(x^{(i)}, y^{(i)}, k) = 0] = \Pr[\varphi_F(x^{(i)}, k) = 0] = \Pr[\varphi'_F(k) = 0] = \Pr[\varphi_F(x, k) = 0] = p$. ■

Заметим, что свойство статистической независимости ассоциированной функции $\varphi_F(x, k)$ от x , обуславливающее наше понятие статистического аналога функции F , существенно отличает его от других понятий того же предназначения, известных под названиями approximate expression, statistical relation и т. п. и не предполагающих данного свойства. В его же отсутствие может непредсказуемо измениться вероятность используемого в криптоанализе шифра approximate expression (statistical relation и т. п.) после подстановки в него открытого текста и соответствующего шифртекста, что делает практически неэффективным алгоритм криптоанализа, основываемый обычно на решении системы вероятностных уравнений методом максимального правдоподобия.

Класс функции φ некоторого СА (в некоторой классификации булевых функций) называется также *классом* этого СА. В частности, статистический аналог называется *линейным (ЛСА)*, если его функция φ линейная, т. е. если $\varphi(x, y, k) = (a, x) \oplus (b, y) \oplus (c, k)$ для некоторых констант $a \in X \setminus \{0\}$, $b \in Y \setminus \{0\}$, $c \in K \setminus \{0\}$. Нередко ЛСА $(a, x) \oplus (b, y) \oplus (c, k) = 0$ записывается как $(a, x) \oplus (b, y) = (c, k)$. В случае $m = 0$ он имеет вид $(a, x) \oplus (b, y) = 0$.

СА функции F , принадлежащий некоторому классу C , называется *оптимальным* (в классе C), если его эффективность наибольшая среди эффективностей всех СА этой функции, входящих в C . Таким образом, можно говорить, например, об оптимальных ЛСА для данной функции F .

СА с нелинейной функцией φ называется *нелинейным*, или *НСА*.

Пример 1. Пусть $n = m = r$ и $y = F(x, k) = x \oplus k$. Тогда для любого ЛСА $(a, x) \oplus (b, y) = (c, k)$ для F с некоторой вероятностью p выполняется уравнение $(a, x) \oplus (b, x \oplus k) = (c, k)$, или, что то же самое, $(a \oplus b, x) = (b \oplus c, k)$. Возможны два случая.

1) $a = b$. В этом случае имеем $(b \oplus c, k) = 0$, $p = \Pr[(b \oplus c, k) = 0]$, и следовательно, если $b = c$, то $p = 1$ и $\varepsilon = 1/2$, а если $b \neq c$, то $p = 1/2$ и $\varepsilon = 0$.

2) $a \neq b$. В этом случае получаем уравнение $(a \oplus b, x) = (b \oplus c, k)$, которое при каждом k выполняется для половины возможных значений x , поэтому $p = 1/2$ и $\varepsilon = 0$.

Таким образом, СА вида $(a, x \oplus y \oplus k) = 0$, и только они являются эффективными ЛСА для функции $x \oplus k$.

Пример 2. Пусть $n = 3$, $m = r = 1$, $y = F(x, k) = x_1 k_1 \oplus x_1 x_3 \oplus x_2 x_3$, $a = 100$, $b = c = 1$. Тогда ЛСА $(a, x) \oplus (b, y) \oplus (c, k) = 0$ для F имеет вероятность p , с которой выполняется уравнение $x_1 \oplus x_1 k_1 \oplus x_1 x_3 \oplus x_2 x_3 \oplus k_1 = 0$. Левая часть последнего обращается в 0 на шести из шестнадцати возможных наборов значений переменных в ней, поэтому $p = 3/8$ и $\varepsilon = 1/8$.

2.2. Статистический аналог суперпозиции функций

Многие функции шифрования строятся как суперпозиции других, более простых функций, в связи с чем возникает задача построения функции статистического аналога суперпозиции из её компонент и их статистических аналогов. Здесь мы рассмотрим эту задачу в ситуации, когда функция F представлена суперпозицией других функций как $F(x, k) = G(H(x, k))$, где $H : X \times K \rightarrow Z$, $G : Z \rightarrow Y$ и $Z = (\mathbb{Z}_2)^l$ для некоторого $l \geq 1$. Примером такого представления F может служить суперпозиция $S(x \oplus k)$ функции замены (S) и суммы (\oplus) заменяемого информационного блока (x) и раундового ключа (k) в итеративных блочных шифрах с аддитивным раундовым ключом [5], в частности в DES.

Пусть уравнение $\psi(z, y) = 0$ является СА функции $G(z)$ и $p = \Pr[\psi(z, G(z)) = 0]$ — его вероятность. Построим функцию $\varphi(x, y, k) = \psi(H(x, k), y)$. Будем иметь $\varphi_F(x, k) = \varphi(x, F(x, k), k) = \psi(H(x, k), F(x, k)) = \psi(H(x, k), G(H(x, k)))$. Спрашивается, является ли уравнение $\varphi(x, y, k) = 0$ статистическим аналогом для F , или, равносильно, зависит ли статистически функция $\varphi_F(x, k)$ от переменных в наборе x . В каждом конкретном случае ответ на этот вопрос можно получить с помощью теста статистической независимости либо проверив непосредственно выполнение равенства $\Pr[\varphi'_F(k) = 0] = \Pr[\varphi_F(x, k) = 0]$, где $\varphi'_F(k) = \psi(H'(k), G(H'(k)))$ и $H'(k)$ — произвольная подфункция функции $H(x, k)$, полученная фиксированием под знаком последней значений всех переменных в x .

В случае $X = K = Z$ и $F(x, k) = G(x \oplus k)$, т. е. когда $H(x, k) = x \oplus k$, функцию $F(x, k)$ называют *функцией с аддитивным параметром* — k . В этом случае $\varphi(x, y, k) = \psi(x \oplus k, y)$.

Утверждение 4. Для функции F с аддитивным параметром функция $\varphi_F(x, k)$ статистически не зависит от x .

Доказательство. Утверждение справедливо в силу утверждения 1 при $f = \varphi_F$ и $g(x \oplus k) = \psi(x \oplus k, G(x \oplus k))$. ■

Следствие 1. Для функции $F(x, k)$ с аддитивным параметром уравнение $\psi(x \oplus k, y) = 0$ является статистическим аналогом.

Утверждение 5. Вероятность статистического аналога $\psi(x \oplus k, y) = 0$ функции $F(x, k)$ с аддитивным параметром равна p .

Доказательство. В самом деле, $\Pr[\psi(x \oplus k, G(x \oplus k)) = 0] = 2^{-n} |\{x \oplus k \in Z : \psi(x \oplus k, G(x \oplus k)) = 0\}| = 2^{-n} |\{z \in Z : \psi(z, G(z)) = 0\}| = \Pr[\psi(z, G(z)) = 0] = p$. ■

2.3. Сложение статистических аналогов

Покажем, что множество всех статистических аналогов одной и той же функции замкнуто относительно сложения (по частям) в поле \mathbb{Z}_2 различных и независимых СА, и приведём формулу для вероятности суммы таких СА. В этой связи индукцией по натуральному s докажем следующую лемму, известную по [2] как Piling-up Lemma.

Лемма 1. Для s независимых случайных переменных X_i с $\Pr[X_i = 0] = p_i$ и $\Pr[X_i = 1] = 1 - p_i$ для $i = 1, 2, \dots, s$ вероятность $\Pr[X_1 \oplus X_2 \oplus \dots \oplus X_s = 0]$ вычисляется как $q_s = 1/2 + 2^{s-1} \prod_{i=1}^s (p_i - 1/2)$.

Доказательство. В самом деле, при $s = 1$ это очевидно. Предположим, что это верно при некотором $s \geq 1$, т. е. $\Pr[X_1 \oplus X_2 \oplus \dots \oplus X_s = 0] = q_s$, и докажем, что $\Pr[X_1 \oplus \dots \oplus X_s \oplus X_{s+1} = 0] = q_{s+1}$. Сумма по mod 2 равна 0 тогда и только тогда, когда оба её слагаемых равны одновременно 0 или 1, поэтому $\Pr[X_1 \oplus \dots \oplus X_s \oplus X_{s+1} = 0] =$

$= q_s p_{s+1} + (1 - q_s)(1 - p_{s+1})$. Положим здесь $q = q_s - 1/2$ и $p = p_{s+1} - 1/2$. Тогда $\Pr[X_1 \oplus \dots \oplus X_s \oplus X_{s+1} = 0] = (q + 1/2)(p + 1/2) + (1/2 - q)(1/2 - p) = 1/2 + 2pq = 1/2 + 2(p_{s+1} - 1/2)(q_s - 1/2) = 1/2 + 2 \cdot 2^{s-1} \prod_{i=1}^s (p_i - 1/2)(p_{s+1} - 1/2) = q_{s+1}$. ■

Утверждение 6. Пусть $\varphi_1(x, y, k) = 0$ и $\varphi_2(x, y, k) = 0$ — различные и независимые СА функции F и $\varphi = \varphi_1 \oplus \varphi_2$. Тогда $\varphi(x, y, k) = 0$ есть также СА функции F .

Доказательство. В случае $m = |k| = 0$ утверждение очевидно. Пусть $m > 0$. Требуется доказать статистическую независимость φ_F от x . Пусть p_1 и p_2 — вероятности заданных в условии СА соответственно. Тогда $\Pr[\varphi'_{1F} = 0] = \Pr[\varphi_{1F} = 0] = p_1$, $\Pr[\varphi'_{2F} = 0] = \Pr[\varphi_{2F} = 0] = p_2$ и в силу леммы 1 $\Pr[\varphi'_F = 0] = \Pr[\varphi'_{1F} \oplus \varphi'_{2F} = 0] = 1/2 + 2(p_1 - 1/2)(p_2 - 1/2) = \Pr[\varphi_{1F} \oplus \varphi_{2F} = 0] = \Pr[\varphi_F = 0]$. ■

Таким образом, доказано, что сумма любых s различных и независимых статистических аналогов некоторой функции с вероятностями p_1, p_2, \dots, p_s соответственно действительно является СА этой функции, и его вероятность вычисляется по формуле для q_s в лемме 1. Его эффективность ε , как видно, не превосходит эффективности любого из слагаемых. В частности, если $p_i = 1/2$ хотя бы для одного $i \in \{1, \dots, s\}$, то $\varepsilon = 0$.

Все приводимые далее статистические аналоги, как линейные, так и нелинейные, для функций в DES заимствованы из литературы по *линейному* криптоанализу, где они подаются под названием *linear approximate equations (relations, expressions)*, см., например, [2, 3].

3. Линейные статистические аналоги для DES

На примере DES рассмотрим методы построения эффективных ЛСА для блоков замены, раундовых функций и функций шифрования многораундовых итеративных блочных симметричных шифров.

3.1. ЛСА блоков замены DES

Блоки замены в DES по традиции будем называть S-блоками. Рассмотрим сначала функцию любого S-блока $S_i : (\mathbb{Z}_2)^6 \rightarrow (\mathbb{Z}_2)^4, i \in \{1, 2, \dots, 8\}$, и произвольный её ЛСА $(a, x) \oplus (b, y) = 0$. Здесь $a \in (\mathbb{Z}_2)^6, b \in (\mathbb{Z}_2)^4, x$ и y — переменные со значениями в $(\mathbb{Z}_2)^6$ и $(\mathbb{Z}_2)^4$ соответственно и $y = S_i(x)$. Определим $N_i(a, b) = |\{x \in (\mathbb{Z}_2)^6 : (a, x) = (b, S_i(x))\}|$. Например, $N_1(011011, 0100) = 22, N_5(010000, 1111) = 12$. По определению, $2^{-6}N_i(a, b)$ есть вероятность, с которой выполняется равенство $(a, x) \oplus (b, S_i(x)) = 0$ при равновероятном выборе $x \in (\mathbb{Z}_2)^6$, поэтому $\Pr[(a, x) \oplus (b, y) = 0] = 2^{-6}N_i(a, b)$. Так, $\Pr[(011011, x) \oplus (0100, S_1(x)) = 0] = 22/64 = 11/32, \Pr[(010000, x) \oplus (1111, S_5(x)) = 0] = 12/64 = 3/16$. Вычислив $2^{-6}N_5(a, b)$ для всех пар ab в $(\mathbb{Z}_2)^6 \times (\mathbb{Z}_2)^4$, т. е. вероятности всевозможных ЛСА функции S_5 , можно убедиться, что ЛСА $(010000, x) \oplus (1111, y) = 0$ является оптимальным (в классе линейных СА) для S_5 (с вероятностью $p = 3/16$ и эффективностью $\varepsilon = 5/16$). В таблице приведены найденные таким образом оптимальные линейные статистические аналоги для всех восьми S-блоков DES, их вероятности p и эффективности ε . Из неё видно, что ЛСА, указанный для S_5 , имеет наибольшую эффективность среди эффективностей ЛСА всех S-блоков DES.

Оптимальные ЛСА для S-блоков DES

Номер S-блока	a	b	p	ε
1	010000	1111	7/32	9/32
2	100010	1011	1/4	1/4
3	100010	1111	1/4	1/4
4	100010	1111	1/4	1/4
	101000	1111	1/4	1/4
	101011	0110	1/4	1/4
	101011	1001	1/4	1/4
5	010000	1111	3/16	5/16
6	010000	0111	9/32	7/32
	100010	1011	9/32	7/32
7	111011	0100	7/32	9/32
8	010000	1111	1/4	1/4
	100010	1110	1/4	1/4

3.2. ЛСА раундовой функции DES

Пусть для произвольного булева вектора $v = v_{t-1}v_{t-2}\dots v_0$ и для любых различных i_1, i_2, \dots, i_s в $\{0, 1, \dots, t-1\}$ символ $v(i_1, i_2, \dots, i_s)$ обозначает (i_1, i_2, \dots, i_s) -проекцию вектора v , т.е. $v(i_1, i_2, \dots, i_s) = v_{i_1}v_{i_2}\dots v_{i_s}$, а символ $v[i_1, i_2, \dots, i_s]$ — сумму по mod 2 всех компонент этой проекции, т.е. $v[i_1, i_2, \dots, i_s] = v_{i_1} \oplus v_{i_2} \oplus \dots \oplus v_{i_s}$.

Функция одного раунда в DES является отображением $F : X \times K \rightarrow Y$, где $X = (\mathbb{Z}_2)^n$, $K = (\mathbb{Z}_2)^m$, $Y = (\mathbb{Z}_2)^r$ для $n = r = 64$, $m = 48$, и для любых $k \in K$ и $x = x_L x_R \in X$, где $|x_L| = |x_R|$, определяется равенством $F(x, k) = y$, в котором $y = y_L y_R \in Y$, $y_L = x_R$, $y_R = x_L \oplus f(x_R, k)$ для некоторой функции $f : (\mathbb{Z}_2)^{32} \times (\mathbb{Z}_2)^{48} \rightarrow (\mathbb{Z}_2)^{32}$. Последняя является суперпозицией элементарных операций над булевыми векторами (расширение, перестановка, сложение по mod 2) и функций S-блоков, такой, что для любого номера S-блока $i = 1, 2, \dots, 8$ существуют i_1, i_2, \dots, i_6 и l_1, l_2, l_3, l_4 в $\{0, 1, \dots, 31\}$, а также j_1, j_2, \dots, j_6 в $\{0, 1, \dots, 47\}$, для которых

$$f(x_R, k)(l_1, l_2, l_3, l_4) = S_i(x_R(i_1, i_2, \dots, i_6) \oplus k(j_1, j_2, \dots, j_6)). \quad (1)$$

Здесь и далее в изложении, относящемся к DES, предполагается, что компоненты векторов $x \in X$, $y \in Y$ и $k \in K$ занумерованы справа налево целыми числами, начиная с 0. В этом предположении верно, в частности, $x_R(t) = x(t)$ для $0 \leq t \leq 31$.

Равенство (1) означает, что функция $F(x, k) = f(x_R, k)(l_1, l_2, l_3, l_4)$ получена суперпозицией функции $G = S_i$ и функции $x \oplus k = x_R(i_1, i_2, \dots, i_6) \oplus k(j_1, j_2, \dots, j_6)$ и, таким образом, является функцией с аддитивным параметром.

Возьмём любой ЛСА i -го S-блока $(a, u) \oplus (b, v) = 0$ с некоторыми вероятностью p_i и эффективностью ε_i . В нём $a \in (\mathbb{Z}_2)^6$, $b \in (\mathbb{Z}_2)^4$, u и v — переменные со значениями в $(\mathbb{Z}_2)^6$ и $(\mathbb{Z}_2)^4$ соответственно и $v = S_i(u)$. Подставив в него сначала $S_i(u)$ вместо v , а затем $x_R(i_1, i_2, \dots, i_6) \oplus k(j_1, j_2, \dots, j_6)$ вместо u и формулу $f(x_R, k)(l_1, l_2, l_3, l_4)$ вместо равной ей по (1) подформулы $S_i(x_R(i_1, i_2, \dots, i_6) \oplus k(j_1, j_2, \dots, j_6))$, получим уравнение $(a, x_R(i_1, i_2, \dots, i_6) \oplus k(j_1, j_2, \dots, j_6)) = (b, f(x_R, k)(l_1, l_2, l_3, l_4))$. Обозначив в последнем $f(x_R, k)$ как y_f (переменный булев вектор длиной 32), придём к следующему линейному уравнению:

$$(a, x_R(i_1, i_2, \dots, i_6)) \oplus (b, y_f(l_1, l_2, l_3, l_4)) = (a, k(j_1, j_2, \dots, j_6)). \quad (2)$$

В нём функция, равная сумме его левой и правой частей, построена как суперпозиция функции $\psi (= (a, u) \oplus (b, v))$ статистического аналога для $G (= S_i)$ и функции $x \oplus k (= x_R(i_1, i_2, \dots, i_6) \oplus k(j_1, j_2, \dots, j_6))$. Следовательно, по следствию 1, уравнение (2) является статистическим аналогом и тем самым ЛСА функции f . По утверждению 5 его вероятность равна p_i , а эффективность — ε_i .

Например, если взяты 5-й S-блок и его ЛСА из таблицы, т. е. если $i = 5$ и $a = 010000$, $b = 1111$, то $(i_1, i_2, \dots, i_6) = (16, 15, \dots, 11)$, $(j_1, j_2, \dots, j_6) = (23, 22, \dots, 18)$, $(l_1, l_2, l_3, l_4) = (24, 18, 7, 29)$ и ЛСА (2) для f имеет вид

$$x_R[15] \oplus y_f[7, 18, 24, 29] = k[22]. \quad (3)$$

Его вероятность и эффективность равны соответственно $3/16$ и $5/16$ (те же, что у взятого ЛСА 5-го S-блока).

Аналогичным образом строятся ЛСА для f по ЛСА остальных S-блоков, в том числе и по не приведённым в таблице. Так, ЛСА для f , построенный по ЛСА $(011011, u) \oplus (0100, v) = 0$ для S-блока S_1 , есть

$$x_R[27, 28, 30, 31] \oplus y_f[15] = k[42, 43, 45, 46] \quad (4)$$

и выполняется с вероятностью $11/32$. Следующие три ЛСА функции f построены этим же методом по другим ЛСА блоков S_1 и S_5 :

$$x_R[29] \oplus y_f[15] = k[44]; \quad (5)$$

$$x_R[15] \oplus y_f[7, 18, 24] = k[22]; \quad (6)$$

$$x_R[12, 16] \oplus y_f[7, 18, 24] = k[19, 23]. \quad (7)$$

Их вероятности равны $15/32$, $21/32$, $1/4$ соответственно.

Кроме того, новые ЛСА функции f можно строить как суммы уже построенных для неё ЛСА (утверждение 6). Так, сумма (3) и (4) является ЛСА $x_R[15, 27, 28, 30, 31] \oplus y_f[7, 15, 18, 24, 29] = k[22, 42, 43, 45, 46]$ для f с вероятностью $1/2 + 2(3/16 - 1/2)(11/32 - 1/2) \approx 0,6$. Поскольку эффективность суммы статистических аналогов не выше эффективности слагаемых, а эффективность ЛСА (3) наибольшая среди эффективностей ЛСА всех S-блоков, то построенные так новые ЛСА будут иметь эффективность не выше $5/16$ — эффективности ЛСА (3).

Как следует из приведённых построений, ЛСА для f в общем виде представляется уравнением

$$x_R[i_1, i_2, \dots, i_{s_1}] \oplus y_f[l_1, l_2, \dots, l_{s_2}] = k[j_1, j_2, \dots, i_{s_3}], \quad (8)$$

выполняемым с некоторой вероятностью p . В нём s_1, s_2, s_3 — некоторые натуральные числа, x_R, y_f, k — переменные со значениями в $(\mathbb{Z}_2)^{32}$, $(\mathbb{Z}_2)^{32}$, $(\mathbb{Z}_2)^{48}$ соответственно, $0 \leq i_t \leq 31$, $0 \leq l_t \leq 31$ и $0 \leq j_t \leq 47$ для всех подходящих t .

На i -м раунде DES, $i \geq 1$, пара 32-битных векторов $L_{i-1}R_{i-1}$ преобразуется по раундовому ключу K_i в пару 32-битных векторов L_iR_i по правилам

$$L_i = R_{i-1}, R_i = L_{i-1} \oplus f(R_{i-1}, K_i). \quad (9)$$

Положив в (8) $x_R = R_{i-1}$, $k = K_i$ и ввиду (9) $y_f = f(x_R, k) = f(R_{i-1}, K_i) = R_i \oplus L_{i-1}$, получим ЛСА для i -го раунда DES

$$R_{i-1}[i_1, i_2, \dots, i_{s_1}] \oplus L_{i-1}[l_1, l_2, \dots, l_{s_2}] \oplus R_i[l_1, l_2, \dots, l_{s_2}] = K_i[j_1, j_2, \dots, i_{s_3}]$$

с вероятностью p . Полезно помнить, что в нём ввиду (9) $R_{i-1} = L_i$.

Применяя данный метод, построим, в качестве примера, некоторые ЛСА для первых пяти раундов DES. Потом они пригодятся нам в построении ЛСА для многораундовых DES.

Так, положив в (3) $x_R = R_0$, $k = K_1$ и $y_f = f(x_R, k) = f(R_0, K_1) = L_0 \oplus R_1$, получим следующий ЛСА 1-го раунда DES, выполненный, как и (3), с вероятностью $3/16$:

$$R_0[15] \oplus L_0[7, 18, 24, 29] \oplus R_1[7, 18, 24, 29] = K_1[22]. \quad (10)$$

Положив в (3) $x_R = R_1 = L_2$, $k = K_2$ и $y_f = f(x_R, k) = f(R_1, K_2) = L_1 \oplus R_2$, получим ЛСА 2-го раунда DES также с вероятностью $3/16$:

$$L_1[7, 18, 24, 29] \oplus L_2[15] \oplus R_2[7, 18, 24, 29] = K_2[22]. \quad (11)$$

Положив же в (3) $x_R = R_2 = L_3$, $k = K_3$ и $y_f = f(x_R, k) = f(R_2, K_3) = L_2 \oplus R_3$, получим ЛСА 3-го раунда DES, выполненный опять же с вероятностью $3/16$:

$$L_3[15] \oplus L_2[7, 18, 24, 29] \oplus R_3[7, 18, 24, 29] = K_3[22]. \quad (12)$$

При $x_R = R_3$, $k = K_4$ и $y_f = f(x_R, k) = f(R_3, K_4) = L_3 \oplus R_4$ в (3) имеем ЛСА 4-го раунда DES с вероятностью $3/16$:

$$R_3[15] \oplus L_3[7, 18, 24, 29] \oplus R_4[7, 18, 24, 29] = K_4[22]. \quad (13)$$

Аналогичными заменами из (4) получаются ещё один ЛСА для 1-го раунда DES, но уже с вероятностью $11/32$

$$R_0[27, 28, 30, 31] \oplus L_0[15] \oplus R_1[15] = K_1[42, 43, 45, 46] \quad (14)$$

и ЛСА для 5-го раунда DES с вероятностью $11/32$

$$L_5[27, 28, 30, 31] \oplus L_4[15] \oplus R_5[15] = K_5[42, 43, 45, 46]. \quad (15)$$

3.3. ЛСА многораундовых DES

Линейные статистические аналоги для многораундовых DES строятся путём суммирования нескольких ЛСА одиночных раундов DES. Так, сумма (10) и (12), равная

$$R_0[15] \oplus L_0[7, 18, 24, 29] \oplus L_3[15] \oplus R_3[7, 18, 24, 29] = K_1[22] \oplus K_3[22],$$

является ЛСА 3-раундового DES с вероятностью $1/2 + 2(3/16 - 1/2)(3/16 - 1/2) = 0,70$, а сумма (11), (13), (14) и (15), равная

$$\begin{aligned} & L_0[15] \oplus R_0[7, 18, 24, 27, 28, 29, 30, 31] \oplus L_5[7, 18, 24, 27, 28, 29, 30, 31] \oplus R_5[15] = \\ & = K_1[42, 43, 45, 46] \oplus K_2[22] \oplus K_4[22] \oplus K_5[42, 43, 45, 46], \end{aligned} \quad (16)$$

— ЛСА 5-раундового DES с вероятностью $1/2 + 2^3(3/16 - 1/2)^2(11/32 - 1/2)^2 = 0,519$.

Пусть далее $A_{i,j}$ обозначает ЛСА i -го раунда DES, полученный подстановкой R_{i-1} и K_i вместо x_R и k соответственно и $L_{i-1} \oplus R_i$ вместо y_f в ЛСА функции f , заданный уравнением $(j+2)$ для $j+2 = 3, 4, \dots, 7$. Непосредственно проверяется, что сумма

$$A = A_{1,5} \oplus A_{3,4} \oplus A_{4,3} \oplus A_{5,1} \oplus A_{7,1} \oplus A_{8,3} \oplus A_{9,4} \oplus A_{11,4} \oplus A_{12,3} \oplus A_{13,1} \oplus A_{15,1}$$

образует ЛСА для 15-раундового DES

$$\begin{aligned} & L_0[7, 18, 24] \oplus R_0[12, 16] \oplus L_{15}[15] \oplus R_{15}[7, 18, 24, 29] = \\ & = K_1[19, 23] \oplus K_3[22] \oplus K_4[44] \oplus K_5[22] \oplus K_7[22] \oplus \\ & \oplus K_8[44] \oplus K_9[22] \oplus K_{11}[22] \oplus K_{12}[44] \oplus K_{13}[22] \oplus K_{15}[22], \end{aligned} \quad (17)$$

выполняемый с вероятностью $1/2 + 2^{10}(3/16 - 1/2)^4(15/32 - 1/2)^3(21/32 - 1/2)^3(1/4 - 1/2) = 1/2 + 1,19 \cdot 2^{-22}$, а сумма $A \oplus A_{16,2}$ есть ЛСА для 16-раундового DES

$$\begin{aligned} & L_0[7, 18, 24] \oplus R_0[12, 16] \oplus L_{16}[7, 18, 24, 27, 28, 29, 30, 31] \oplus R_{16}[15] = \\ & = K_1[19, 23] \oplus K_3[22] \oplus K_4[44] \oplus K_5[22] \oplus K_7[22] \oplus K_8[44] \oplus \\ & \oplus K_9[22] \oplus K_{11}[22] \oplus K_{12}[44] \oplus K_{13}[22] \oplus K_{15}[22] \oplus K_{16}[42, 43, 45, 46], \end{aligned} \quad (18)$$

выполняемый с вероятностью $1/2 + 2(1,19 \cdot 2^{-22})(11/32 - 1/2) = 1/2 - 1,49 \cdot 2^{-24}$. Кроме того, сумма

$$A_{15,5} \oplus A_{13,4} \oplus A_{12,3} \oplus A_{11,1} \oplus A_{9,1} \oplus A_{8,3} \oplus A_{7,4} \oplus A_{5,4} \oplus A_{4,3} \oplus A_{3,1} \oplus A_{1,1}$$

есть ещё один ЛСА для 15-раундового DES с вероятностью $1/2 + 1,19 \cdot 2^{-22}$

$$\begin{aligned} & R_{15}[7, 18, 24] \oplus L_{15}[12, 16] \oplus R_0[15] \oplus L_0[7, 18, 24, 29] = \\ & = K_{15}[19, 23] \oplus K_{13}[22] \oplus K_{12}[44] \oplus K_{11}[22] \oplus \\ & \oplus K_9[22] \oplus K_8[44] \oplus K_7[22] \oplus K_5[22] \oplus K_4[44] \oplus K_3[22] \oplus K_1[22], \end{aligned} \quad (19)$$

а сумма

$$A' = A_{16,5} \oplus A_{14,4} \oplus A_{13,3} \oplus A_{12,1} \oplus A_{10,1} \oplus A_{9,3} \oplus A_{8,4} \oplus A_{6,4} \oplus A_{5,3} \oplus A_{4,1} \oplus A_{2,1} \oplus A_{1,2}$$

— ещё один ЛСА 16-раундового DES с вероятностью $1/2 - 1,49 \cdot 2^{-24}$

$$\begin{aligned} & R_{16}[7, 18, 24] \oplus L_{16}[12, 16] \oplus R_0[7, 18, 24, 27, 28, 29, 30, 31] \oplus L_0[15] = \\ & = K_{16}[19, 23] \oplus K_{14}[22] \oplus K_{13}[44] \oplus K_{12}[22] \oplus K_{10}[22] \oplus K_9[44] \oplus \\ & \oplus K_8[22] \oplus K_6[22] \oplus K_5[44] \oplus K_4[22] \oplus K_2[22] \oplus K_1[42, 43, 45, 46]. \end{aligned} \quad (20)$$

Заметим, что последние два ЛСА могут быть получены из (17) и (18) соответственно по следующему правилу, справедливому благодаря симметрии раундов DES: если в ЛСА t -раундового DES произвести взаимную замену L_0, R_0, K_i на R_t, L_t, K_{t+1-i} соответственно для $i = 1, 2, \dots, t$, то получится снова ЛСА t -раундового DES с той же вероятностью.

Наконец можно убедиться, что $A - A_{1,5}$ (т.е. A за исключением слагаемого $A_{1,5}$) есть уравнение

$$\begin{aligned} & R_1[7, 18, 24] \oplus L_{15}[15] \oplus R_{15}[7, 18, 24, 29] = K_3[22] \oplus K_4[44] \oplus K_5[22] \oplus \\ & \oplus K_7[22] \oplus K_8[44] \oplus K_9[22] \oplus K_{11}[22] \oplus K_{12}[44] \oplus K_{13}[22] \oplus K_{15}[22], \end{aligned} \quad (21)$$

а $A' - A_{16,5} - A_{1,2}$ есть уравнение

$$\begin{aligned} & R_1[15] \oplus L_1[7, 18, 24, 29] \oplus L_{15}[7, 18, 24] = K_{14}[22] \oplus K_{13}[44] \oplus K_{12}[22] \oplus \\ & \oplus K_{10}[22] \oplus K_9[44] \oplus K_8[22] \oplus K_6[22] \oplus K_5[44] \oplus K_4[22] \oplus K_2[22], \end{aligned} \quad (22)$$

и оба они выполняются с вероятностью $1/2 + 2^9(3/16 - 1/2)^4(15/32 - 1/2)^3(21/32 - 1/2)^3 = 1/2 + 1,19 \cdot 2^{-21}$, представляя собой два различных ЛСА для одной и той же функции — 14-раундового DES со 2-го по 15-й раунды. Взаимная замена L_1, R_1, K_i на R_{15}, L_{15}, K_{17-i} соответственно превращает один из них в другой по свойству симметрии раундов DES.

4. Криптоанализ на основе ЛСА

В криптографии этот метод известен как линейный криптоанализ. Применительно к DES его впервые описал японец Mitsuru Matsui [2]. Линейный криптоанализ является атакой с известным открытым текстом, направленной на частичное раскрытие ключа шифра и осуществляемой на основе некоторой системы эффективных линейных статистических аналогов функции шифрования

$$(a^{(i)}, x) \oplus (b^{(i)}, y) = (c^{(i)}, k), i = 1, 2, \dots, s, \quad (23)$$

выполнимых с (отличными от $1/2$) вероятностями p_1, p_2, \dots, p_s соответственно.

Известные открытые тексты $x^{(j)} \in (\mathbb{Z}_2)^n$ и их криптограммы $y^{(j)} \in (\mathbb{Z}_2)^r$, $j = 1, 2, \dots, N$, подставляются в уравнения данной системы, и получается система линейных булевых уравнений для некоторых компонент неизвестного ключа k , а именно:

$$(c^{(i)}, k) = d_{ij}, \quad i = 1, 2, \dots, s; \quad j = 1, 2, \dots, N, \quad (24)$$

где $d_{ij} = (a^{(i)}, x^{(j)}) \oplus (b^{(i)}, y^{(j)}) \in \mathbb{Z}_2$ для всех $i = 1, 2, \dots, s$ и $j = 1, 2, \dots, N$. Каждое уравнение в ней вероятностное — по утверждению 3 выполняется с той же вероятностью, что и аналог в (23), из которого оно получено.

Система уравнений (24) относится к классу так называемых случайных систем уравнений с искажённой правой частью [6], ставших в последнее время предметом многочисленных исследований (см., например, «Труды по дискретной математике», издаваемые с 1997 г. совместно Российской академией наук и Академией криптографии РФ как приложение к журналу «Дискретная математика», где можно найти и разные методы решения таких систем). Для решения системы (24) воспользуемся методом максимального правдоподобия (МП).

Пусть $t_i = N - \sum_{j=1}^N d_{ij}$, $i = 1, 2, \dots, s$. Это есть количество тех известных пар x/y (открытый текст/криптограмма), для которых левая часть i -го уравнения в (23) обращается в 0. Для каждого $i = 1, 2, \dots, s$ определим $d_i \in \mathbb{Z}_2$ по следующим правилам:

- 1) $d_i = 0$, если $t_i > N/2$ и $p_i > 1/2$ или $t_i \leq N/2$ и $p_i < 1/2$;
- 2) $d_i = 1$, если $t_i \leq N/2$ и $p_i > 1/2$ или $t_i > N/2$ и $p_i < 1/2$.

Следуя методу МП, систему (24) заменим детерминированной системой булевых уравнений

$$(c^{(i)}, k) = d_i, \quad i = 1, 2, \dots, s, \quad (25)$$

которую можно решить методом Гаусса.

Любое решение любой совместной подсистемы последней системы относительно компонент вектора k , явно входящих в уравнения подсистемы, представляется как результат криптоанализа.

При $m = |k| \leq s$ совместная система линейно независимых уравнений (25) имеет 2^{m-s} решений: в них значения некоторых $m - s$ неизвестных выбираются произвольно, а остальные s неизвестных вычисляются по ним однозначно. Это значит, что методом линейного криптоанализа на основе s статистических линейных аналогов шифра в действительности можно определить самое большее s бит ключа. В частности, по одному ЛСА с k_j в правой части находится ровно один ключевой бит — k_j .

Ввиду вероятностного характера уравнений в (23) и (24) результат линейного криптоанализа оказывается также вероятностным: найденные значения компонент ключа являются истинными лишь с некоторой вероятностью. Эта *вероятность успеха* тем

выше, чем выше эффективности использованных статистических линейных аналогов и чем больше открытых текстов занято в атаке. Так, в случае одного ЛСА в системе (23) с вероятностью p и эффективностью $\varepsilon = |p - 1/2| > 0$ для вероятности успеха, близкой к 0,98, требуется около ε^{-2} известных открытых текстов. Например, для нахождения данным методом одного бита ключа в 5-раундовом DES, равного правой части уравнения (16), необходимо иметь $|0,519 - 1/2|^{-2} \approx 2800$ открытых текстов. Оба ЛСА (18) и (20) для 16-раундового DES выполняются одновременно с вероятностью $1/2 - 1,49 \cdot 2^{-24}$, поэтому линейный криптоанализ на их основе позволяет с большой вероятностью успеха определить сразу два бита ключа 16-раундового DES, используя $(1,49 \cdot 2^{-24})^{-2} \approx 2^{47}$ известных открытых текстов.

Основная трудность, с которой сталкивается разработчик метода линейного криптоанализа для конкретного шифра, заключается в построении достаточно большого числа линейных статистических аналогов его функции шифрования с не слишком малой их эффективностью. К сожалению, не много найдётся реальных шифров, для которых такое построение действительно возможно. Более перспективным видится применение в криптоанализе вместо ЛСА нелинейных статистических аналогов функций шифров.

5. Нелинейные статистические аналоги DES

Имея для $(n - 1)$ -раундового DES линейный статистический аналог $(a, L_0R_0) \oplus \oplus(b', L_{n-1}) \oplus \oplus(b'', R_{n-1}) = (c, K)$, выполняемый с некоторой вероятностью p , и равенство $(b', L_{n-1}) = (b', f(L_n, K_n)) \oplus (b', R_n)$, справедливое ввиду (9) при верном раундовом ключе K_n , получаем нелинейный статистический аналог для n -раундового DES

$$(a, L_0R_0) \oplus (b'', L_n) \oplus (b', R_n) \oplus (b', f(L_n, K_n)) = (c, K),$$

выполняемый с той же вероятностью p . Вводя обозначения $b = b''b'$ и $d = b'$, можно переписать последний как

$$(a, L_0R_0) \oplus (b, L_nR_n) \oplus (d, f(L_n, K_n)) = (c, K).$$

Таким способом, например, из ЛСА (17) и (19) для 15-раундового DES ввиду равенств

$$L_{15}[15] = f(L_{16}, K_{16})[15] \oplus R_{16}[15]$$

и

$$L_{15}[7, 18, 24] = f(L_{16}, K_{16})[7, 18, 24] \oplus R_{16}[7, 18, 24]$$

получаются следующие НСА для 16-раундового DES:

$$\begin{aligned} L_0[7, 18, 24] \oplus R_0[12, 16] \oplus L_{16}[7, 18, 24, 29] \oplus R_{16}[15] \oplus f(L_{16}, K_{16})[15] = \\ = K_1[19, 23] \oplus K_3[22] \oplus K_4[44] \oplus K_5[22] \oplus K_7[22] \oplus \\ \oplus K_8[44] \oplus K_9[22] \oplus K_{11}[22] \oplus K_{12}[44] \oplus K_{13}[22] \oplus K_{15}[22] \end{aligned} \quad (26)$$

и

$$\begin{aligned} L_0[15] \oplus R_0[7, 18, 24, 29] \oplus L_{16}[12, 16] \oplus R_{16}[7, 18, 24] \oplus f(L_{16}, K_{16})[7, 18, 24] = \\ = K_{15}[19, 23] \oplus K_{13}[22] \oplus K_{12}[44] \oplus K_{11}[22] \oplus K_9[22] \oplus K_8[44] \oplus \\ \oplus K_7[22] \oplus K_5[22] \oplus K_4[44] \oplus K_3[22] \oplus K_1[22] \end{aligned} \quad (27)$$

соответственно. Каждый из них выполняется с вероятностью $1/2 + 1,19 \cdot 2^{-22}$.

Кроме того, имея для $(n - 2)$ -раундового DES линейный статистический аналог $(a', L_1) \oplus (a'', R_1) \oplus (b', L_{n-1}) \oplus (b'', R_{n-1}) = (c, K)$, выполняемый с некоторой вероятностью p , и равенства $(a'', R_1) = (a'', f(R_0, K_1)) \oplus (a'', L_0)$ и $(b', L_{n-1}) = (b', f(L_n, K_n)) \oplus (b', R_n)$, справедливые ввиду (9) при верных раундовых ключах K_1 и K_n , получаем нелинейный статистический аналог для n -раундового DES

$$(a'', L_0) \oplus (a', R_0) \oplus (a'', f(R_0, K_1)) \oplus (b'', L_n) \oplus (b', R_n) \oplus (b', f(L_n, K_n)) = (c, K),$$

выполняемый с той же вероятностью p .

Так, из уравнения (21) и равенств

$$R_1[7, 18, 24] = f(R_0, K_1)[7, 18, 24] \oplus L_0[7, 18, 24], \quad L_{15}[15] = f(L_{16}, K_{16})[15] \oplus R_{16}[15]$$

и из уравнения (22) и равенств

$$R_1[15] = f(R_0, K_1)[15] \oplus L_0[15], \quad L_{15}[7, 18, 24] = f(L_{16}, K_{16})[7, 18, 24] \oplus R_{16}[7, 18, 24]$$

получаются следующие НСА для 16-раундового DES, выполняемые с вероятностью $1/2 + 1,19 \cdot 2^{-21}$:

$$\begin{aligned} L_0[7, 18, 24] \oplus f(R_0, K_1)[7, 18, 24] \oplus L_{16}[7, 18, 24, 29] \oplus R_{16}[15] \oplus f(L_{16}, K_{16})[15] = \\ = K_3[22] \oplus K_4[44] \oplus K_5[22] \oplus K_7[22] \oplus K_8[44] \oplus \\ \oplus K_9[22] \oplus K_{11}[22] \oplus K_{12}[44] \oplus K_{13}[22] \oplus K_{15}[22] \end{aligned} \quad (28)$$

и

$$\begin{aligned} L_0[15] \oplus f(R_0, K_1)[15] \oplus R_0[7, 18, 24, 29] \oplus R_{16}[7, 18, 24] \oplus f(L_{16}, K_{16})[7, 18, 24] = \\ = K_{14}[22] \oplus K_{13}[44] \oplus K_{12}[22] \oplus K_{10}[22] \oplus K_9[44] \oplus \\ \oplus K_8[22] \oplus K_6[22] \oplus K_5[44] \oplus K_4[22] \oplus K_2[22] \end{aligned} \quad (29)$$

соответственно.

По определению раундовой функции f функция $f(X, K)[15]$ реализуется на выходе S-блока S_1 , а функция $f(X, K)[7, 18, 24]$ — на выходах S-блока S_5 . Тем самым каждая из этих двух функций существенно зависит только от шести бит раундового ключа K : первая — от $K[42], K[43], \dots, K[47]$, вторая — от $K[18], K[19], \dots, K[23]$. Это значит, что в приведенных выше уравнениях (26), (27) и (28), (29) нелинейные слагаемые зависят на самом деле соответственно от 6 и от 12 неизвестных ключевых бит, а именно: в (26) — от $K_{16}[42], K_{16}[43], \dots, K_{16}[47]$; в (27) — от $K_{16}[18], K_{16}[19], \dots, K_{16}[23]$; в (28) — от $K_1[18], K_1[19], \dots, K_1[23], K_{16}[42], K_{16}[43], \dots, K_{16}[47]$; в (29) — от $K_1[42], K_1[43], \dots, K_1[47], K_{16}[18], K_{16}[19], \dots, K_{16}[23]$.

6. Криптоанализ на основе НСА

Чтобы подчеркнуть единородство этого метода и линейного криптоанализа, будем называть его *нелинейным криптоанализом*, видя в словах «линейный» и «нелинейный» не противоположность, привносимую частицей «не», но, прежде всего, единство их корня. Нелинейный криптоанализ, как и линейный, направлен на частичное раскрытие ключа шифра, однако в отличие от линейного он может быть атакой как с известным открытым текстом, так и с выбором одного. Но в любом случае для реализации нелинейного криптоанализа предполагается наличие некоторого эффективного нелинейного статистического аналога функции шифрования. Возможны разные алгоритмы

нелинейного криптоанализа, основанные на методе МП и использующие разные свойства заданного НСА. Здесь мы представим три таких алгоритма: один предполагает в НСА свойство условной разделимости, два других — свойство малости линейаризационного множества. Первые два алгоритма являются атаками с известным открытым текстом, третий — атакой с выбором открытого текста.

Пусть для рассматриваемого симметричного шифра имеется нелинейный статистический аналог $\varphi(x, y, k) = 0$ функции шифрования $F(x, k)$, выполняемый с некоторой вероятностью p , и $0 < p < 1$.

6.1. Криптоанализ на основе условно разделимого НСА

Представим заданный НСА как $\varphi'(x, y, k') = (1, k'')$, где k' и k'' — наборы некоторых переменных в k , не имеющие общих переменных, $(1, k'')$ — сумма всех тех переменных в k , если таковые есть, которые входят только в линейные слагаемые полинома Жегалкина (АНФ) функции φ (по ним φ линейная), и $\varphi'(x, y, k')$ есть сумма остальных слагаемых в полиноме. В отсутствие переменных в k , по которым функция $\varphi(x, y, k)$ линейная, считаем $(1, k'') = 0$ и $\varphi'(x, y, k') = \varphi(x, y, k)$. Пусть также x' есть набор всех тех переменных в x , которые являются существенными аргументами функции φ (входят в её АНФ явно).

Будем называть НСА $\varphi = 0$ *разделимым* (по переменным), если ассоциированная с ним функция $\varphi_F(x, k) = \varphi(x, F(x, k), k)$ статистически не зависит от переменных в наборе $x'k'$ и $|k''| > 1$. Если, кроме того, число переменных в k' сравнительно мало (в пределах трёх-четырёх десятков — с позиции производительности современных компьютеров), то данный НСА называется *условно разделимым*. Важность этого понятия очевидна: в случае статистической независимости φ_F от $x'k'$ любое фиксирование открытого текста x , соответствующего шифртекста y и значений переменных в k' в уравнении $\varphi(x, y, k) = 0$ приводит его к линейному уравнению с неизвестными в k'' , выполнимому с той же вероятностью, что и $\varphi = 0$.

Теперь неизвестные значения переменных в k' и сумма значений переменных в k'' могут быть найдены следующим алгоритмом, где N — количество известных открытых текстов.

1. Для каждого из возможных значений $k'^{(j)}$ набора k' ($j = 1, 2, \dots, 2^{|k'|}$) и для каждой пары известных открытого текста $x^{(i)}$ и его шифртекста $y^{(i)}$ ($i = 1, 2, \dots, N$) определяется $d_{ij} = \varphi'(x^{(i)}, y^{(i)}, k'^{(j)})$, после чего для каждого $k'^{(j)}$ подсчитывается количество $t_j = N - \sum_{i=1}^N d_{ij}$ всех таких пар $(x^{(i)}, y^{(i)})$, для которых $d_{ij} = 0$, и определяются m и l из условий: $t_m = \max t_j$ и $t_l = \min t_j$ по всем j от 1 до N .

2. Если $|t_m - N/2| > |t_l - N/2|$, то полагаем $k' = k'^{(m)}$ и, кроме того, $d = 0$ в случае $p > 1/2$ и $d = 1$ в случае $p < 1/2$. Если же $|t_m - N/2| < |t_l - N/2|$, то полагаем $k' = k'^{(l)}$ и, кроме того, $d = 1$ для $p > 1/2$ и $d = 0$ для $p < 1/2$.

(Иначе говоря, за значение k' берём то $k'^{(j)}$, при котором $\varphi'(x^{(i)}, y^{(i)}, k'^{(j)})$ со всевозможными парами $(x^{(i)}, y^{(i)})$ принимает некоторое значение $d \in \{0, 1\}$ чаще (другого) при $p > 1/2$ и реже при $p < 1/2$.)

3. Полагаем $(1, k'') = d$, тем самым находим ещё один бит информации о ключе k .

К сожалению, мы не знаем, зависят ли статистически от переменных в $x'k'$ функции, ассоциированные с приведёнными выше НСА (26) — (29) для DES, и, следовательно, не знаем, являются ли последние разделимыми по переменным. В соответствии с нашей теорией это значит, что мы не вправе использовать эти НСА в данном алгоритме, поскольку нет гарантии того, что вероятность выполнения уравнения, по-

лученного подстановкой символов открытого и соответствующего зашифрованного текстов в любой из них, не будет сильно отличаться от его вероятности p и не совпадёт с $1/2$. Сам М. Matsui признаёт в [3], что эта вероятность «is expected to be closer to $1/2$ (not necessarily $1/2$)», и тем не менее с верой в не только русское «наука полагает, а Бог располагает» применяет алгоритм с каждым из linear approximate equations (26) — (29) в криптоанализе DES.

Согласно [2], количество N известных открытых текстов, при котором вероятность успеха данного алгоритма в применении к DES близка к 0,97, оценивается величиной $8\varepsilon^{-2}$. Так, применив его с $N = 8(1,19 \cdot 2^{-22})^{-2} \approx 2^{47}$ известными открытыми текстами дважды: сначала — к НСА (26), затем — к НСА (27), М. Matsui находит 14 бит ключа DES, а именно: $K_{16}[42], K_{16}[43], \dots, K_{16}[47]$, один бит из правой части в (26), $K_{16}[18], K_{16}[19], \dots, K_{16}[23]$ и один бит из правой части в (27). Аналогичным образом с использованием $N = 8(1,19 \cdot 2^{-21})^{-2} \approx 2^{45}$ известных открытых текстов по (28) и (29) находятся 26 бит ключа DES, а именно: $K_1[18], K_1[19], \dots, K_1[23], K_{16}[42], K_{16}[43], \dots, K_{16}[47]$, один бит из правой части в (28), $K_1[42], K_1[43], \dots, K_1[47], K_{16}[18], K_{16}[19], \dots, K_{16}[23]$ и один бит из правой части в (29).

6.2. Криптоанализ на основе НСА с малым линеаризационным множеством

Атака с известным открытым текстом

Подставив в заданный НСА $\varphi(x, y, k) = 0$ вместо x и y соответственно известные открытые тексты $x^{(i)} \in X$ и их криптограммы $y^{(i)} \in Y$ для $i = 1, 2, \dots, N$, получим систему булевых уравнений для компонент неизвестного ключа k , а именно:

$$\varphi_i(k) = 0, \quad i = 1, 2, \dots, N, \quad (30)$$

где $\varphi_i(k) = \varphi(x^{(i)}, y^{(i)}, k)$ для всех $i \in \{1, 2, \dots, N\}$. Каждое уравнение в системе (30) вероятностное и выполняется с той же вероятностью p , что и НСА, из которого оно получено.

Следуя [7], назовём подмножество переменных в k *линеаризационным*, если при фиксации любых их значений каждое уравнение в системе (30) превращается в линейное (линеаризуется). Зафиксируем некоторое (лучше — наименьшей мощности, или кратчайшее) линеаризационное множество L переменных в системе (30). Для каждого набора $L^{(j)}$ значений переменных в L возьмём подфункцию $\varphi_i^{(j)}(k')$ функции $\varphi_i(k)$, полученную подстановкой вместо переменных в L их значений в наборе $L^{(j)}$. Здесь $j = 1, 2, \dots, s = 2^{|L|}$. Ввиду свойства линеаризационного множества функция $\varphi_i^{(j)}(k')$ является аффинной. Пусть $\varphi_i^{(j)}(k') = (c_i^{(j)}, k') \oplus d_i^{(j)}$. Таким образом, получаем систему линейных уравнений

$$(c_i^{(j)}, k') = d_i^{(j)}, \quad i = 1, 2, \dots, N; j = 1, 2, \dots, s, \quad (31)$$

где каждое уравнение выполняется с вероятностью $q = s^{-1}p$. Эта система представляет собой объединение s подсистем E_1, \dots, E_s , где E_j для любого $j \in \{1, 2, \dots, s\}$ состоит из уравнений в (31) для $i = 1, 2, \dots, N$. Каждая подсистема E_j решается подобно системе (24), а именно: полагаем $t_j = N - \sum_{i=1}^N d_i^{(j)}$, определяем $d^{(j)}$ по следующим правилам:

- 1) $d^{(j)} = 0$, если $t_j > N/2$ и $q > 1/2$ или $t_j \leq N/2$ и $q < 1/2$,
- 2) $d^{(j)} = 1$, если $t_j \leq N/2$ и $q > 1/2$ или $t_j > N/2$ и $q < 1/2$,

и записываем детерминированную систему уравнений

$$(c_i^{(j)}, k') = d^{(j)}, \quad i = 1, 2, \dots, N.$$

Если эта система совместна, то её решение относительно k' вместе с набором $L^{(j)}$ является результатом криптоанализа — предполагаемым ключом шифра. Это надо понимать так, что если последняя система совместна при нескольких значениях $j \in \{1, 2, \dots, s\}$, то результат криптоанализа будет неоднозначным, что, естественно, возможно при недостаточном количестве N использованных пар (открытый текст, шифртекст).

Ясно, что данный алгоритм реально выполним лишь тогда, когда линеаризационное множество L достаточно мало. Легко видеть, что каждый из приведённых выше НСА (26), (27), (28) и (29) для 16-раундового DES этим свойством обладает, ибо множества $L_1 = \{K_{16}[42], K_{16}[43], \dots, K_{16}[47]\}$, $L_2 = \{K_{16}[18], K_{16}[19], \dots, K_{16}[23]\}$, $L_3 = \{K_1[18], K_1[19], \dots, K_1[23], K_{16}[42], K_{16}[43], \dots, K_{16}[47]\}$ и $L_4 = \{K_1[42], K_1[43], \dots, K_1[47], K_{16}[18], K_{16}[19], \dots, K_{16}[23]\}$ являются линеаризационными в системе (30) для этих НСА соответственно. Таким образом, применив данный алгоритм с НСА (26) или с НСА (27), можно получить 18 бит ключа DES: 6 бит в L_1 и 12 бит из правой части в (26) или 6 бит в L_2 и 12 бит из правой части в (27) соответственно. Применив же его с НСА (28) или с НСА (29), можно получить 22 бита ключа DES, а именно: 12 бит в L_3 и 10 бит из правой части в (28) или 12 бит в L_4 и 10 бит из правой части в (29). Если же применить алгоритм сначала, скажем, с НСА (28), а затем с НСА (29), то можно получить 44 бита раундовых ключей DES, или, с учётом расписания ключей, 34 бита исходного ключа, в то время как М. Matsui находит от тех же самых двух НСА только 26 из этих бит.

Заметим, что если функция φ является сильно t -аффинной [8], то система (30) имеет линеаризационное множество мощности t , поэтому для противостояния этой атаке необходимо, чтобы функция шифрования не допускала статистического аналога с функцией, имеющей малый уровень сильной аффинности.

Атака с выбором открытого текста

Мощность линеаризационного множества переменных в системе (30) зависит как от вида φ , так и от того, какие именно пары открытых текстов $x^{(i)} \in X$ и их криптограмм $y^{(i)} \in Y$ для каждого $i = 1, 2, \dots, N$ подставлены в НСА $\varphi(x, y, k) = 0$ с целью получения этой системы. Если выбрать открытые тексты такими, при которых система (30), полученная подстановкой их и соответствующих шифртекстов в уравнение $\varphi(x, y, k) = 0$, будет иметь линеаризационное множество L наименьшей мощности (или близкой к нему), и использовать это L в последнем алгоритме криптоанализа, то можно достичь максимальной (или близкой к ней) скорости выполнения данного алгоритма (при заданной функции φ). Это и будет атака с выбором открытого текста. Дальнейшее её ускорение возможно на пути выбора более подходящего НСА. Впрочем, последнее замечание относится и к атаке с известным открытым текстом.

ЛИТЕРАТУРА

1. Агибалов Г. П. Методы решения систем уравнений над конечным полем // Вестник Томского государственного университета. Приложение. 2006. № 17. С. 4–9.
2. Matsui M. Linear Cryptanalysis Method for DES Cipher // LNCS. 1993. V. 765. P. 386–397.
3. Matsui M. The First Experimental Cryptanalysis of the Data Encryption Standard // LNCS. 1994. V. 839. P. 1–11.
4. Логачев О. А., Сальников А. А., Яценко В. В. Булевы функции в теории кодирования и криптографии. М.: МЦНМО, 2004.

5. Агibalов Г. П. Элементы теории дифференциального криптоанализа итеративных блочных шифров с аддитивным раундовым ключом // Прикладная дискретная математика. 2008. № 1. С. 34–43.
6. Балакин Г. В. Введение в теорию случайных систем уравнений // Труды по дискретной математике. Т. 1. М.: ТВП, 1997. С. 1–18.
7. Агibalов Г. П. Логические уравнения в криптоанализе генераторов ключевого потока // Вестник Томского государственного университета. Приложение. 2003. № 6. С. 31–41.
8. Буряков М. Л., Логачев О. А. Об уровне аффинности булевых функций // Дискретная математика. 2005. Т. 17. Вып. 4. С. 98–107.

**РЕАЛИЗАЦИЯ ШИФРА ЗАКРЕВСКОГО
НА ОСНОВЕ ПЕРЕСТРАИВАЕМОГО АВТОМАТА¹**

В. Н. Тренькаев

*Томский государственный университет, г. Томск, Россия***E-mail:** tvnik@sibmail.com

Предлагается реализация шифра Закревского на основе перестраиваемого автомата, настройка которого вместе с начальным состоянием является ключом шифра. Показано, что множество шифрующих автоматов, получаемых в результате всех возможных настроек перестраиваемого автомата, обладает достаточной мощностью, чтобы противостоять атаке грубой силы. Вместе с тем предложенная реализация имеет практически приемлемую длину ключа. Также показано, что данная реализация не стойка к атаке на основе выбранного открытого текста, когда криптоаналитик знает начальное состояние и имеет несколько экземпляров шифратора.

Ключевые слова: шифр Закревского, обратимый автомат, автомат с биективной функцией выходов, перестраиваемый автомат, кратные безусловные эксперименты по идентификации автомата.

Введение

Известно [1, 2], что модель конечного автомата активно применяется в криптографии. Однако существует не так много используемых на практике шифров, в которых алгоритм шифрования (расшифрования) задается конечным автоматом. К числу прочих академических автоматных шифров можно отнести и шифр Закревского [3], в котором не указана процедура порождения шифрующих автоматов с требуемыми свойствами. При этом шифр Закревского можно также отнести к классу так называемых недетерминированных шифров (по терминологии [4]), т. е. шифров, у которых алгоритм шифрования формируется (выбирается) на этапе предвычислений в зависимости от секретного ключа.

Шифр с выбираемым криптоалгоритмом можно рассматривать как перестраиваемый автомат [5–7], т. е. автомат с возможностью настройки на требуемый алгоритм функционирования. Обычно перестраиваемый автомат имеет зафиксированную структуру логической схемы, его реализующей, и настройка автомата заключается в изменении связей между функциональными элементами схемы (структурная настройка) или их функциональности (функциональная настройка). При этом результатом каждой настройки перестраиваемого автомата является некоторый автомат из заданного класса.

В данной работе предлагается реализация шифра Закревского с использованием перестраиваемого автомата с функциональной настройкой. Каждая настройка перестраиваемого автомата порождает приведенный сильносвязный автомат с биективной функцией выходов, который задает алгоритм шифрования. Логическая сеть перестраиваемого автомата такова, что при любой настройке реализуется некоторая фиксиро-

¹Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П1010).

ванная функция выходов, в то время как функция переходов «составляется» из функций переходов двух известных базовых автоматов. Таким образом, при аппаратной реализации перестраиваемого автомата мы имеем избыточность на уровне дублирования функциональных узлов, отвечающих за реализацию функции переходов.

1. Основные определения и обозначения

Определение 1. Конечным автоматом A называется пятерка (X, S, Y, ψ, φ) , где S — конечное непустое множество состояний; X и Y — конечные входной и выходной алфавиты соответственно; $\psi : X \times S \rightarrow S$ и $\varphi : X \times S \rightarrow Y$ — функции переходов и выходов соответственно.

Четверку $s - x/y \rightarrow s'$, где $s' = \psi(x, s)$ и $y = \varphi(x, s)$, называют *переходом* автомата A и говорят, что автомат A из состояния s (обозначается A/s) под действием входного символа x переходит в состояние s' с выдачей выходного символа y .

Говорят, что входное слово $x_1x_2 \dots x_l \in X^*$ переводит автомат A/s в состояние s' с выдачей выходного слова (*реакции*) $y_1y_2 \dots y_l \in Y^*$, если существует (или говорят, что *реализуется* под действием $x_1x_2 \dots x_l$) последовательность переходов $s = s_1 - x_1/y_1 \rightarrow s_2, s_2 - x_2/y_2 \rightarrow s_3, \dots, s_l - x_l/y_l \rightarrow s_{l+1} = s'$.

Автомат A при фиксированном состоянии s реализует отображение (далее *словарный оператор*) $f_s : X^* \rightarrow Y^*$, для которого $f_s(x_1x_2 \dots x_l) = y_1y_2 \dots y_l$.

Определение 2. Автомат A называется *сильносвязным*, если для любых состояний s и s' существует входное слово, которое переводит автомат из состояния s в состояние s' .

Определение 3. Автомат A называется *приведенным*, если для любого состояния s не существует состояния s' , такого, что $s \neq s'$ и $f_s = f_{s'}$.

Определение 4. Автомат A *обратим*, если при любом состоянии s для отображения f_s существует обратное отображение f_s^{-1} .

Определение 5. Автомат $A^{-1} = (Y, S, X, \psi', \varphi')$ называется *обратным* к автомату $A = (X, S, Y, \psi, \varphi)$, реализующему $\{f_s : s \in S\}$, если A^{-1} реализует $\{f_s^{-1} : s \in S\}$.

Несложно показать, что при $|X| = |Y|$ автомат A обратим, если и только если для любого $s \in S$ функция $\varphi_s(x) = \varphi(x, s)$ является биекцией из X в Y .

Определение 6. Автомат A называется *автоматом с биективной функцией выходов*, если для любого $s \in S$ функция $\varphi_s(x)$ является биекцией.

Таким образом, автомат с биективной функцией выходов (и только он при $|X| = |Y|$) является обратимым и для него существует обратный автомат. В этом случае A^{-1} может быть получен по A следующим образом: для каждого перехода $s - x/y \rightarrow s'$ автомата A строится соответствующий переход $s - y/x \rightarrow s'$ автомата A^{-1} .

2. Шифр Закревского

Шифр Закревского является симметричным шифром, в котором множества открытых и шифрованных сообщений являются множествами слов в некоторых алфавитах, алгоритмы шифрования и расшифрования задаются взаимно обратными сильносвязными автоматами с биективными функциями выходов, и ключом шифра являются начальное состояние и функции переходов и выходов обоих автоматов.

Пусть X и Y — алфавиты соответственно открытых и шифрованных сообщений, причем далее везде $|X| = |Y|$. Тогда шифрование по Закревскому заключается в преобразовании открытого сообщения $\alpha \in X^*$ в шифрованное сообщение $\beta \in Y^*$ с помощью автомата $A = (X, S, Y, \psi, \varphi)$ (с необходимыми свойствами) при фиксированном

начальном состоянии s , т.е. мы имеем $f_s(\alpha) = \beta$. Чтобы расшифровать β , требуется построить обратный автомат $A^{-1} = (Y, S, X, \psi', \varphi')$ и подать на него β , поскольку $f_s^{-1}(\beta) = \alpha$.

Пусть мы имеем два разных ключа k_A и k_B , т.е. автомат $A = (X, S, Y, \psi_A, \varphi_A)$ с начальным состоянием q и автомат $B = (X, S, Y, \psi_B, \varphi_B)$ с начальным состоянием p . Ключи k_A и k_B называются *эквивалентными*, если автоматы A/q и B/p реализуют один и тот же словарный оператор.

Показано [1], что число всех попарно неэквивалентных ключей шифра Закревского не меньше m^n , где $|X| = |Y| = m$, $|S| = n$, то есть атака на шифр, основанная на методе полного (тотального) опробования ключей, практически не осуществима при $m, n > 20$. Кроме того, в рамках автоматной модели криптоанализ шифра Закревского сводится к решению задачи восстановления (идентификации) автомата с помощью проведения эксперимента с ним, которая в общем случае считается труднорешаемой. Однако к недостаткам шифра Закревского можно отнести большой размер ключа.

Действительно, пусть $r = \lceil \log_2 |S| \rceil$, т.е. r есть наименьшее целое, такое, что $2^r \geq |S|$, и $v = \lceil \log_2 |Y| \rceil$. Тогда для задания ключа потребуется не менее $mn(r + v) + r$ бит. При $r = v = 5$ и $m = n = 20$ мы имеем 4005 бит, что на порядок больше используемых на практике размеров в 128/256 бит.

Кроме того, существует проблема генерирования ключей, так как ключевое множество шифра Закревского задано описанием свойств его элементов, но для практического использования требуется задаться порождающей процедурой, допускающей простую программную и/или аппаратную реализацию. Иными словами, необходим алгоритм генерирования сильносвязанных автоматов с биективной функцией выходов и с низкой вероятностью повтора, чтобы ключ выбирался случайно и равновероятно. При этом порождающая процедура может зависеть от некоторого параметра. Тогда автомат с заданными свойствами будет строиться под управлением некоторого ключа инициализации приемлемого размера, например пароля пользователя. Для решения данной задачи предлагается использовать перестраиваемый автомат.

3. Перестраиваемый автомат

С любым автоматом можно связать логическую сеть, моделирующую его поведение. Будем считать, что логические сети (совокупности элементов, связанных между собой путем отождествления некоторых их полюсов) могут включать в себя многофункциональные настраиваемые элементы, т.е. элементы, поведение которых зависит от $k \in K$, где K — конечное множество настроек.

Автомат, реализуемый такой логической сетью, будем называть *перестраиваемым*, полагая, что его функции переходов и выходов зависят не только от $(x, s) \in X \times S$, но и от $k \in K$, т.е. перестраиваемый автомат — это шестерка $(X, S, Y, K, \psi, \varphi)$, где $\psi : X \times S \times K \rightarrow S$ и $\varphi : X \times S \times K \rightarrow Y$. Будем говорить, что, фиксируя некоторое k из K , мы *настраиваем* автомат.

Таким образом, перестраиваемый автомат задает множество J автоматов $A_k = (X, S, Y, \psi_k, \varphi_k)$, где $\psi_k(x, s) = \psi(x, s, k)$ и $\varphi_k(x, s) = \varphi(x, s, k)$ для $k \in K$.

Пусть I, O и K — конечные множества, $C = \{0, 1\}$ и заданы функции $\delta_0 : I \rightarrow O$, $\delta_1 : I \rightarrow O$, $\pi : I \times K \rightarrow C$, а также функция $\rho : O \times O \times C \rightarrow O$, такая, что $\rho(d_0, d_1, c) = d_c$ для всех d_0, d_1 в O и $c \in C$. Таким образом, ρ работает как мультиплексор, который в зависимости от управляющего символа $c \in C$ «пропускает со входов на выход» либо d_0 , либо d_1 .

Определим функцию $\lambda : I \times K \rightarrow O$ так, что $\lambda(i, k) = \lambda_k(i) = \rho(\delta_0(i), \delta_1(i), \pi_k(i))$ для любых $i \in I$ и $k \in K$, и будем называть её *настраиваемой композицией* (с управлением π , зависящим от входа в I и настройки в K). При фиксированной настройке k она показана схематически на рис. 1.

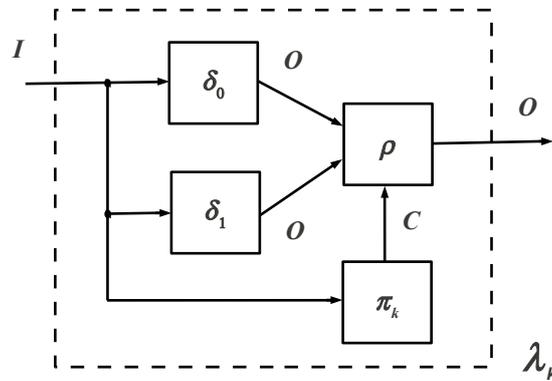


Рис. 1. Настраиваемая композиция λ с настройкой k

Построим перестраиваемый автомат $R = (X, S, Y, K, \psi, \varphi)$ следующим образом. Функция выходов φ зависит от $k \in K$ фиктивно, т. е. является фактически отображением $\varphi : X \times S \rightarrow Y$, и $\{\varphi_s(x) : s \in S\}$ есть множество различных биекций. Функция переходов ψ является настраиваемой композицией λ , в которой $I = X \times S$ и $O = S$, т. е. для любой пары $(x, s) \in X \times S$ верно $\psi_k(x, s) = \lambda_k(x, s)$. Логическая сеть, реализующая автомат R , изображена на рис. 2. Она состоит из компонент *State*, *Out* и *Reg*. Компонента *State* реализует настраиваемую функцию переходов $\psi_k(x, s)$, компонента *Out* — фиксированную функцию выходов $\varphi(x, s)$, компонента *Reg* — память автомата. Последняя компонента состояние, поступающее ей на вход, выдает на выход в следующий такт работы. Данная сеть является каноническим представлением автомата схемой, состоящей из комбинационной части (компоненты *State* и *Out*) и элементов памяти (компонента *Reg*).

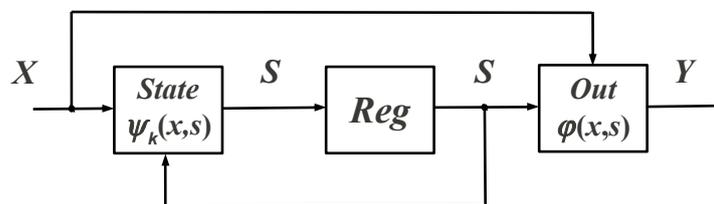


Рис. 2. Перестраиваемый автомат R

Нетрудно заметить, что поведение перестраиваемого автомата R формируется в результате совместной работы двух автоматов $B_0 = (X, S, Y, \delta_0, \varphi)$ и $B_1 = (X, S, Y, \delta_1, \varphi)$. Причем состояние, в которое переходит автомат R из текущего, формирует как δ_0 ,

так и δ_1 , но в память «закладывается» только одно из двух возможных значений, т. е. память является общей для автоматов B_0 и B_1 .

Далее предполагается, что функции δ_0 и δ_1 в настраиваемой композиции λ автомата R таковы, что существует входное слово α длины n , которое переводит автомат B_0 и автомат B_1 из некоторого состояния t в то же состояние t , посещая при этом (проводя автомат через) все другие состояния из S , и, кроме того, в автоматах B_0/t и B_1/t под действием α реализуется одинаковая последовательность переходов. Тогда при любой настройке k автомата R для любой пары состояний в полученном автомате $A_k \in J$ существует входное слово, которое переводит A_k из первого из этих состояний во второе. Таким образом, по построению автомата R справедливо следующее утверждение.

Утверждение 1. Перестраиваемый автомат $R = (X, S, Y, K, \psi, \varphi)$ при каждой настройке $k \in K$ является приведенным сильносвязным автоматом $A_k = (X, S, Y, \psi_k, \varphi)$ с биективной функцией выходов, причем для любой пары (x, s) из $X \times S$ верно: если $\pi_k(x, s) = 0$, то $\psi_k(x, s) = \delta_0(x, s)$, иначе $\psi_k(x, s) = \delta_1(x, s)$.

4. Реализация шифра Закревского на основе перестраиваемого автомата

Ввиду утверждения 1 любая настройка автомата R задает (порождает) шифрующий автомат для шифра Закревского. При этом (в случае $X = Y$) для шифрования и расшифрования используется логическая сеть (рис. 2), где при расшифровании компонента Out реализует функцию $\eta(x, s)$, такую, что $\eta_s(x) = \varphi_s^{-1}(x)$ при любом $s \in S$. Именно эта сеть и предлагается в качестве реализации шифра Закревского. В ней ключом выступает настройка автомата R вместе с некоторым начальным состоянием. Таким образом, ключевое множество Ω в реализации шифра Закревского на основе перестраиваемого автомата является множеством $\{A_k/s : A_k \in J, s \in S\}$. Оно, естественно, много меньше множества всех ключей в шифре Закревского с теми же параметрами автомата. Вместе с тем, при надлежащем выборе последних, его можно сделать достаточно большим, чтобы противостоять атаке грубой силы.

Теорема 1. Ключевое множество Ω не содержит попарно эквивалентных ключей, и его мощность равна $n2^{n(m-1)}$.

Доказательство. По построению существует входное слово α длины n , под действием которого в автоматах B_0 и B_1 , а значит, и в любом $A_k \in J$ реализуется одинаковая последовательность переходов. Следовательно, для n пар (x, s) значения функции $\pi_k(x, s)$ не зависят от k , и число всех таких функций, сопоставляемых разным значениям k , равно 2^{nm-n} . Таким образом, $|\Omega| \leq n2^{n(m-1)}$.

Покажем, что ключевое множество Ω не содержит попарно эквивалентных ключей и, следовательно, $|\Omega| = n2^{n(m-1)}$. Пусть мы имеем два произвольных разных ключа, т. е. автомат $A = (X, S, Y, \psi_A, \varphi) \in J$ с начальным состоянием q и автомат $B = (X, S, Y, \psi_B, \varphi) \in J$ с начальным состоянием p . Если q и p — разные состояния, то поскольку по построению $\{\varphi_s(x) : s \in S\}$ — множество различных биекций, существует хотя бы один входной символ x , такой, что $\varphi_q(x) \neq \varphi_p(x)$. Следовательно, автоматы A/q и B/p реализуют разные словарные операторы.

Рассмотрим случай, когда q и p — одинаковые состояния. Так как A/q и B/p — разные ключи, то A и B — разные автоматы и, следовательно, существует хотя бы одна пара $(x, s) \in X \times S$, такая, что $\psi_A(x, s) \neq \psi_B(x, s)$. Также по построению при любом $k \in K$ в автомате $A_k \in J$ существует последовательность переходов $t = s_{i_1} - x_1/y_1 \rightarrow s_{i_2} - x_2/y_2 \rightarrow s_{i_3} - x_3/y_3 \rightarrow \dots \rightarrow s_{i_n} - x_n/y_n \rightarrow s_{i_{n+1}} = t$. Причем $x_1x_2 \dots x_n$ переводит автомат A_k из состояния t в состояние t , посещая при этом все другие

состояния из S . Следовательно, используя данную последовательность переходов, всегда можно построить входное слово β , которое переводит A/q и B/p (при условии, что q и p — одинаковые состояния, а это так по предположению) в состояние s . Пусть $\psi_A(x, s) = s'$ и $\psi_B(x, s) = s''$. Так как s' и s'' — разные состояния, то существует хотя бы один входной символ z , такой, что $\varphi_{s'}(z) \neq \varphi_{s''}(z)$. Таким образом, существует входное слово $\beta x z$, в ответ на которое автоматы A/q и B/p выдают разные выходные слова, т. е. A/q и B/p реализуют разные словарные операторы. ■

Каждой настройке $k \in K$ автомата R во взаимно-однозначное соответствие ставится вектор значений функции $\pi_k(x, s)$, в котором n компонент предопределены заранее. Он является некоторым булевым вектором h длины $|X \times S|$, поэтому длина ключа предложенной реализации шифра Закревского не превышает числа $mn + r$. Например, при $m = n = 20$ и $r = 5$ это число равно 405 (а не 4005 — длине ключа в шифре Закревского при тех же значениях m , n и r). Однако, ввиду теоремы 1, достаточно взять $m = n = 10$ и $r = 4$, чтобы достичь в реализации приемлемого числа ($2^{90} \cdot 10$) всех возможных ключей и приемлемой длины ключа (104 бита). Также можно отметить, что булев вектор h может быть получен на основе некоторого генератора псевдослучайных последовательностей, который, в свою очередь, может инициализироваться булевым вектором меньшей длины, чем h , но достаточной для обеспечения его (булева вектора h) случайности. Таким образом, предложенная реализация шифра Закревского имеет малую длину ключа при достаточно большой мощности ключевого множества.

5. Криптоанализ реализации шифра Закревского на основе перестраиваемого автомата

Рассмотрим способность реализации шифра Закревского противостоять криптоаналитической атаке с использованием выбранного открытого текста. В рамках автоматной модели имеем задачу восстановления (идентификации) автомата с помощью проведения с ним эксперимента [8]. Здесь ограничимся применением кратных безусловных экспериментов, т. е. будем предполагать, что у криптоаналитика имеется в наличии несколько экземпляров (копий) неизвестного автомата, находящихся перед экспериментом в одном и том же начальном состоянии (их число называется кратностью эксперимента), и прикладываемые к ним входные слова определяются заранее, а не по ходу эксперимента. Задача заключается в том, чтобы по реакциям экземпляров автомата определить сам автомат.

В нашем случае для эксперимента предъявлены экземпляры некоторого автомата E из множества J , которое задается перестраиваемым автоматом R , построенным вышеописанным способом. Будем предполагать, что начальное состояние автомата E , одно то же во всех экземплярах, известно. Требуется по наблюдаемым реакциям этих экземпляров на входные слова определить функцию переходов автомата E .

Под *длиной эксперимента* понимают сумму длин всех применённых в нём входных слов, а под его кратностью — количество использованных копий автомата. Покажем, что любой автомат $E \in J$ при известном его начальном состоянии может быть восстановлен кратным безусловным экспериментом, длина и кратность которого не превышают $(n + 2)mn$ и mn соответственно.

По построению автомата R существует входное слово длины n , которое при любом $k \in K$ переводит автомат $A_k \in J$ из некоторого состояния в то же самое состояние, посещая при этом все другие состояния из S . Следовательно, любой автомат $E \in J$,

предъявленный для эксперимента, с известным начальным состоянием можно перевести входным словом α длины не более n в любое заданное состояние s .

Тогда задача восстановления автомата E сводится к задаче восстановления в нем произвольного перехода $s - x/y \rightarrow s'$, у которого состояние s является известным экспериментатору, x — выбираемый входной символ, y — наблюдаемый выходной символ. По утверждению 1 неизвестное состояние s' принадлежит $\{\delta_0(x, s), \delta_1(x, s)\}$. Если $\delta_0(x, s) = \delta_1(x, s) = p$, то $s' = p$. Если $\delta_0(x, s)$ и $\delta_1(x, s)$ — разные состояния, то по свойству функции выходов φ автомата R существует хотя бы один входной символ z , такой, что $\varphi(z, \delta_0(x, s)) \neq \varphi(z, \delta_1(x, s))$. Тогда по реакции автомата E из начального состояния на входное слово $\alpha x z$ длины не более $n + 2$ можно однозначно идентифицировать состояние s' .

Так как для восстановления переходов автомата E достаточно перебрать все пары (x, s) из $X \times S$, используя для каждой пары свою копию автомата E , то длина эксперимента будет не более $(n + 2)mn$, а кратность — mn . Тем самым доказана следующая теорема.

Теорема 2. Существует кратный безусловный эксперимент с длиной не более $(n + 2)mn$ и кратностью не более mn , посредством которого однозначно восстанавливается любой автомат из класса J с известным начальным состоянием.

В переводе на язык криптографии это значит, что для реализации шифра Закревского на основе перестраиваемого автомата R имеет место следующее свойство: если часть ключа, представленная начальным состоянием автомата шифрования, известна, то остальная его часть полностью раскрывается простой атакой с выбором не более mn открытых текстов с общей длиной не более $(n + 2)mn$ символов. О её стойкости к другим атакам с той же или иными угрозами ничего пока неизвестно.

Заключение

В данной работе предложена ориентированная на практику реализация шифра Закревского на основе автомата, перестраиваемого на разные шифрующие автоматы по параметру настройки. Показано, что: 1) каждая настройка перестраиваемого автомата порождает приведенный сильносвязный автомат с биективной функцией выходов; 2) количество настроек достаточно велико, чтобы противостоять атаке грубой силы; 3) настройка задается булевым вектором длины, приемлемой для практического использования в криптографии; 4) аппаратная реализация перестраиваемого автомата имеет избыточность на уровне дублирования функциональных узлов, отвечающих за реализацию функции переходов; 5) при известном начальном состоянии шифрующего автомата, полученного настройкой перестраиваемого автомата, задача его идентификации с помощью кратного безусловного эксперимента имеет полиномиальную сложность (от размеров автомата); 6) вопрос о стойкости данной реализации к атакам других типов требует дополнительных исследований.

ЛИТЕРАТУРА

1. Агибалов Г. П. Конечные автоматы в криптографии // Прикладная дискретная математика. Приложение. 2009. № 2. С. 43–73.
2. Бабаш А. В., Шанкин Г. Н. Криптография. М.: СОЛОН-Р, 2002. 512 с.
3. Закревский А. Д. Метод автоматической шифрации сообщений // Прикладная дискретная математика. 2009. № 2. С. 127–137.
4. Молдовян А. А., Молдовян Н. А., Советов Б. Я. Криптография. СПб.: Изд-во «Лань», 2001. 224 с.

5. *Шидловский С. В.* Автоматическое управление. Перестраиваемые структуры. Томск: Томский государственный университет, 2006. 288 с.
6. *Glaser J., Damm M., Haase J., Grimm Ch.* A dedicated reconfigurable architecture for finite state machines // LNCS. 2010. No. 5992. P. 122–133.
7. *Sklyarov V.* Reconfigurable models of finite state machines and their implementation in FPGAs // J. Systems Architecture. 2002. No. 47. P. 1047–1064.
8. *Гилл А.* Введение в теорию конечных автоматов. М.: Наука, 1966. 272 с.

МАТЕМАТИЧЕСКИЕ ОСНОВЫ КОМПЬЮТЕРНОЙ БЕЗОПАСНОСТИ

DOI 10.17223/20710410/9/7

УДК 004.94

АНАЛИЗ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ПОТОКОВ В ОПЕРАЦИОННЫХ СИСТЕМАХ СЕМЕЙСТВА *GNU/LINUX*¹

М. А. Качанов

*Томский государственный университет, г. Томск, Россия***E-mail:** m.a.kachanov@gmail.com

В данной работе анализируется безопасность информационных потоков в операционных системах семейства *GNU/Linux*. Рассматриваются информационные потоки по времени с участием доверенных субъектов, приводятся примеры. Предлагается метод проверки возможности реализации информационного потока по памяти между сущностями компьютерной системы, защищенной с помощью средства *SELinux*.

Ключевые слова: компьютерная безопасность, информационный поток, *Linux*.

Введение

В настоящее время одной из актуальных задач теории компьютерной безопасности является анализ безопасности управления доступом и информационными потоками в компьютерных системах (КС) [1]. При её решении, в частности, необходимо идентифицировать информационные потоки разных типов, которые в реальных КС могут быть реализованы многими способами.

Основные виды информационных потоков по памяти и по времени были описаны и исследованы в работе [1] в рамках семейства ДП-моделей. В этих моделях предполагается, что доверенные субъекты не участвуют в реализации информационных потоков по времени, однако в реальных КС существуют процессы, в которых данные предположения частично нарушаются, то есть доверенные субъекты могут участвовать в реализации некоторых информационных потоков по времени, но это не означает, что всякий субъект, реализовавший поток от себя к доверенному субъекту или наоборот, сможет прочесть данные из любой сущности системы, к которой доверенный субъект имеет права доступа. Под участием в данном случае понимается возможность реализации потока от доверенного субъекта к некоторым заданным сущностям компьютерной системы при выполнении другими субъектами определенных действий. Подобное участие доверенных субъектов в реализации потоков может и не привести к утечке конфиденциальных данных, но тем не менее, как будет показано ниже, позволяет реализовать передачу данных между недоверенными субъектами. Данные потоки возникают из-за особенностей реализации доверенных субъектов в реальных компьютерных системах,

¹Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П1010).

закрывающихся в том, что в процессе функционирования доверенные субъекты заносят данные о действиях субъектов системы в доступные на чтение другим субъектам сущности, к которым первые могут и не получать доступ. Поэтому представляется целесообразным построение модели безопасности, адекватной некоторым новым видам информационных потоков, с последующим описанием условий нарушения безопасности последних. Для этого необходимо описать такие информационные потоки в КС и исследовать механизмы их возникновения.

Кроме того, в рамках решения задачи анализа безопасности управления доступом и информационными потоками часто возникает задача анализа конфигурации средств, реализующих данное управление. Одним из таких средств является *SELinux*, применяемое в операционных системах (ОС) семейства *GNU/Linux* и реализующее различные виды управления доступом. Это средство является сложно конфигурируемым, поэтому в процессе его эксплуатации возникает задача построения формальной модели анализа реализуемых им политик безопасности в КС.

В данной работе проводится анализ безопасности информационных потоков в ОС семейства *GNU/Linux* и решаются две задачи. Первой из них является идентификация информационных потоков по времени в ОС семейства *GNU/Linux*, в том числе и потоков нового типа, а именно информационных потоков по времени с участием доверенных субъектов. Второй задачей является анализ возможности реализации информационного потока по памяти между сущностями компьютерной системы, защищенной с помощью средства *SELinux*, которое де-факто является наиболее распространенным при реализации современных политик безопасности в ОС семейства *GNU/Linux*.

Работа состоит из двух разделов. Первый раздел посвящен информационным потокам по времени в КС. Рассматривается новый вид информационного потока по времени — с участием доверенных субъектов, на примере таких компьютерных систем, как ОС *GNU/Linux* и система управления базами данных (СУБД) *MySQL*. Приводятся конкретные примеры возможности реализации информационных потоков по времени в ОС *GNU/Linux* с использованием виртуальной файловой системы, сокетов и времени последнего доступа к файлу.

Во втором разделе проводится анализ безопасности информационных потоков по памяти в *SELinux*. Приводится краткое описание данного средства, предлагается метод проверки возможности реализации информационного потока по памяти между сущностями КС и предлагается программное средство автоматизированного анализа политики безопасности *SELinux*. В заключении подводятся итоги работы.

1. Примеры информационных потоков по времени в *GNU/Linux*

1.1. Информационные потоки по времени без участия доверенных субъектов

Приводятся примеры информационных потоков по времени в *GNU/Linux* без участия доверенных субъектов. Хотя данный вид информационных потоков и был рассмотрен в рамках семейства ДП-моделей [1], в литературе редко приводятся примеры возможности их реализации на практике в реальных компьютерных системах, поэтому представляется целесообразным подробно описать способы реализации данных информационных потоков в *GNU/Linux*.

Далее приводятся примеры в следующем виде. В *описании* кратко излагается основная идея реализации информационного потока. В *цели* указывается желаемый результат использования потока. Далее уточняются *необходимые условия* для его реали-

зации. После этого описывается *метод реализации потока* с конкретными *примерами и результатами* его использования в *GNU/Linux*. Затем следуют некоторые *наблюдения*, облегчающие реализацию потока на практике. В конце предлагаются *механизмы защиты* ОС от реализации рассматриваемого информационного потока по времени.

Пример 1. Информационный поток по времени с использованием сокетов.

Описание. Используется тот факт, что если слушающий сокет уже создан одним процессом, то другой процесс сокет на том же порту создать не сможет, и ядро вернет ему код ошибки. По факту возможности/невозможности создания сокета передаются данные.

Цель. Передача данных между процессами, запущенными от имени разных пользователей.

Необходимые условия. Два процесса, имеющие право создать слушающий сокет на некотором порту.

Метод реализации потока. Если процесс P_1 хочет передать 1 процессу P_2 , то он создает слушающий сокет на оговоренном порту, иначе не создает. P_2 пытается создать сокет на том же порту. Если это удастся сделать, то была передана 1, иначе 0.

Примеры/Результаты. Для достижения высокой скорости передачи данных при создании сокета следует ядру указать, чтобы порт не блокировался на некоторое время после разрыва соединения.

Заключения/Наблюдения. Для успешной реализации потока желательно, чтобы порт был из высокого диапазона (непривилегированный).

Механизмы защиты. На уровне ядра запретить процессам создавать слушающие сокеты, кроме тех, что им действительно необходимы. Тем не менее, если два процесса в соответствии с политикой безопасности могут создать сокет на одном и том же порту, то такие меры не защитят от потока по времени. Можно также контролировать частоту создания сокета, но данная защита не является надежной. Другой вариант — изменить ядро так, чтобы по коду возврата системного вызова нельзя было определить, успешно ли завершилась операция, но на практике это может означать потерю работоспособности системы.

Пример 2. Информационный поток по времени с использованием времени последнего доступа к файлу.

Описание. Используется особенность файловой системы, заключающаяся в том, что в метаданных, относящихся к файлу, сохраняется время последних доступа и модификации.

Цель. Передача данных между процессами, запущенными от имени разных пользователей.

Необходимые условия. Файл *file*, доступный на открытие процессу P_1 и «видимый» процессом P_2 . Под видимостью понимается возможность выполнить системный вызов *stat* с файлом *file*. Фактически P_2 в списке файлов директории видит файл *file*.

Метод реализации потока. Если процесс P_1 хочет передать 1 процессу P_2 , то он осуществляет доступ к файлу *file*, в противном случае — не осуществляет. Процесс P_2 выполняет системный вызов *stat* с файлом *file* и узнает время последнего доступа. Если оно изменилось с момента предыдущего выполнения вызова *stat*, то считаем, что P_1 передал 1, иначе — 0.

Примеры/Результаты. Пусть пользователь U_1 имеет доступ только к своей домашней директории, а пользователь U_2 «видит» (в оговоренном выше смысле) корневой каталог домашней директории. Тогда пользователь U_1 может передать данные

пользователю U_2 , открывая/не открывая свой домашний каталог, а U_2 будет делать `stat` на домашнем каталоге U_1 .

Заключения/Наблюдения. Поток имеет место, если к файлу не происходит обращений со стороны других процессов. В *GNU/Linux* отсутствует иерархичность во времени доступа к файлу. То есть, если получим доступ к корневой директории, то время доступа к файлу в ней не изменится.

Механизмы защиты. Можно монтировать файловую систему без учета времени доступа:

```
mount -noatime -nodiratime
```

1.2. Информационные потоки по времени с участием доверенных субъектов

Случаи возникновения информационных потоков по времени, описанные в работе [1], охватывают множество возможностей их реализации в реальных КС, но не всегда полно отражают действительность. Так, новый вид информационных потоков по времени, а именно с участием доверенных субъектов, был обнаружен в таких КС, как ОС *GNU/Linux*, а также СУБД *MySQL*.

Для реализации информационного потока по времени в ОС *GNU/Linux* используется виртуальная файловая система *proc*. Особенностью *proc* является то, что информация о действиях одного процесса может отображаться в файлах, доступных для чтения процессам, запущенным от имени других пользователей. Например, пусть имеются два процесса (P_1 и P_2) с идентификаторами *pid1* и *pid2* соответственно и пусть процесс P_2 имеет право чтения файла */proc/pid1/status*. В этом файле, в частности, отображается количество нитей (*threads*), которыми оперирует процесс P_1 . При создании либо удалении процессом P_1 нити информация об этом будет заноситься ядром ОС *GNU/Linux* в файл */proc/pid1/status*. Читая данный файл, процесс P_2 может получить данные от процесса P_1 . В данном файле, в частности, отображается количество нитей, которыми оперирует процесс P_1 . Между процессами существует договоренность, что если P_1 хочет передать 0, то он создает одну нить, а если хочет передать 1, то создает две нити. Создание нити выполняется с помощью вызова функции `pthread_create` библиотеки `pthread`. P_2 читает информацию о количестве нитей из файла */proc/pid1/status* и, таким образом, получает данные от P_1 . На первый взгляд может показаться, что данный способ реализации информационного потока по времени подпадает под уже описанные в рамках ДП-моделей случаи, но это не так. Существенной особенностью приведенного выше примера является то, что при создании нити P_1 не осуществляет никаких обращений к файловой системе, а данные в файл */proc/pid1/status* записывает ядро ОС, которое является доверенным субъектом. Таким образом, P_1 может вообще не иметь никаких прав доступа в файловой системе, но тем не менее информационный поток по времени может быть реализован. Стоит уточнить, что реализация *proc* в ОС *GNU/Linux* такова, что пользователь, от имени которого запущен P_1 , хоть и является владельцем файла */proc/pid1/status*, но тем не менее не может менять права доступа к нему и не может открыть этот файл на запись. Для предотвращения возможности реализации подобного информационного потока по времени может быть использовано средство *SELinux*, позволяющее наложить дополнительные ограничения на стандартную политику безопасности *GNU/Linux* и запретить чтение файла */proc/pid1/status* всем процессам, кроме P_1 .

В случае СУБД *MySQL* аналогичная ситуация возникает, когда некоторый пользователь осуществляет запросы к базе данных (БД). Ядро СУБД, являясь доверенным

субъектом, ведет статистику о количестве и типах запросов, об объеме принятых и переданных данных и некоторых других параметрах. Например, при всяком запросе пользователя `show databases` ядро СУБД будет увеличивать текущее значение счетчика подобных запросов на единицу. Стоит отметить, что даже пользователь с минимальными правами к БД может тем или иным образом влиять на параметры, статистику о которых ведет ядро СУБД. С помощью запроса `show status` пользователи системы могут получить полный отчет о накопленной статистике и увидеть текущие значения параметров, в том числе количество определенных запросов всех пользователей системы. Таким образом, один пользователь БД может передать данные другому пользователю, лишь совершая запросы к БД, разрешенные ему политикой безопасности, причем второй пользователь может не иметь никаких прав доступа к таблицам БД, с которыми работает первый пользователь.

Оба приведенных примера объединяет то, что информационные потоки по времени, возникающие в результате осуществления описанных действий, реализуются за счет отображения ядром системы, которое является доверенным субъектом, информации о её функционировании в сущностях, к которым субъекты системы непосредственно не получали доступа. Ядро системы само заносит данные о действиях субъектов системы в доступные на чтение другим субъектам сущности, причем первые могут и не иметь никаких прав доступа к данным сущностям.

В связи с обнаружением информационных потоков по времени с участием доверенных субъектов возникает необходимость учета данных потоков при анализе защищенности КС. В рамках семейства ДП-моделей возможно введение нового вида ассоциированных сущностей, указывающих на возможность реализации к ним информационных потоков по времени в зависимости от выполняемых субъектом действий. Кроме того, возможно введение новых правил преобразований, а также формулирование и обоснование необходимых и достаточных условий возможности реализации информационных потоков по времени между сущностями КС.

2. Анализ безопасности информационных потоков по памяти в *SELinux*

2.1. Описание средства *SELinux*

В настоящее время распространенным механизмом управления доступом и информационными потоками в ОС семейства *GNU/Linux* является средство *SELinux*. Данное средство представляет собой набор патчей для ядра *Linux* и входит в его стандартную поставку. Стоит отметить, что *SELinux*, а также утилиты его администрирования включены в дистрибутив ОС *Red Hat Enterprise Linux*, сертифицированный ФСТЭК. Средство *SELinux* изначально разрабатывалось Агентством национальной безопасности США, но в конце 2000 года его исходный текст был открыт под лицензией *GPL*, и проект был передан в разработку мировому сообществу.

SELinux позволяет реализовать принудительный контроль доступа в ОС класса *Unix* поверх стандартной дискреционной политики безопасности. С помощью данного средства возможна реализация дискреционного, ролевого, а также мандатного управления доступом. Работа средства *SELinux* основана на сопоставлении каждой программе или процессу, ресурсам (файлу, директории, сокету и т. д.) определенного типа. Тип, сопоставленный процессу, принято называть доменом. Каждый домен представляет собой множество прав доступа, достаточных для нормального функционирования процесса, но не более того. Например, домен может быть ограничен в определенных действиях с заданными файлами. Для того чтобы иметь возможность устанавливать подобные ограничения для конкретных ресурсов, каждый файл помечен определен-

ным контекстом безопасности. Домен не может получить доступ к файлам, имеющим контекст безопасности, отличный от тех, к которым ему непосредственно разрешено получать доступ. При определенных условиях процесс, порождающий новый процесс с помощью запуска исполняемого файла, может покинуть свой домен и перейти в новый. Новый домен может иметь другие привилегии в системе, нежели исходный. Механизм *SELinux*, гарантирующий строгое следование предписанным правилам взаимодействия доменов и типов, получил название *type enforcement*. Также существуют и другие механизмы безопасности, в том числе и ролевое управление доступом (*RBAC*). Определения типов, контекстов безопасности, а также возможных переходов между доменами описываются в политике безопасности на собственном гибком языке. К сожалению, политики зачастую довольно объемны и сложны, что затрудняет их комплексное исследование. В связи с этим возникает задача верификации политик безопасности *SELinux*.

Известно несколько подходов к верификации политик безопасности *SELinux*. В [2] вводится формальная модель описания правил политик безопасности, а также предлагается рекурсивный алгоритм проверки возможности получения субъектом определенного права доступа к объекту по начальному состоянию компьютерной системы. При этом данная модель не учитывает возможность реализации информационных потоков по памяти между сущностями КС и позволяет получить лишь примитивную информацию о возможных правах доступа субъекта. В работе [3] рассматривается программное средство *Apol*, включенное в набор утилит *SETools* для администрирования *SELinux*. Данное средство способно отслеживать информационные потоки по памяти, находить все возможные пути реализации информационного потока между двумя сущностями КС, а также обнаруживать некоторые информационные потоки по времени. Однако для данного средства отсутствует формальная модель, а также нет документации, описывающей алгоритм проверки возможности реализации информационного потока по памяти. Кроме того, *Apol* не учитывает функционально и параметрически ассоциированные с субъектами сущности.

Таким образом, представляется целесообразным разработать средство для анализа политик безопасности *SELinux*, позволяющее анализировать возможность реализации информационного потока по памяти между сущностями КС, которое основано на формальной модели, учитывающей функционально ассоциированные с субъектами сущности. В дальнейшем подобный метод может быть предложен и для анализа возможности реализации информационных потоков по времени.

2.2. Анализ возможности реализации информационного потока по памяти в *SELinux*

В рамках семейства ДП-моделей [1] проводится анализ безопасности компьютерных систем с дискреционным, мандатным и ролевым управлением доступом, формулируются и обосновываются необходимые и достаточные условия получения недоверенным субъектом права доступа владения к доверенному субъекту, а также предлагаются алгоритмы построения замыканий, позволяющих определить истинность предиката *can_share* для всех вершин и прав доступа одновременно. К сожалению, на практике при автоматизированном анализе защищенности реальных компьютерных систем данные алгоритмы малоприменимы из-за их вычислительной сложности. Так, в работе [4] был предложен алгоритм построения замыкания базовой ролевой ДП-модели, а также было показано, что он имеет полиномиальную сложность. Известно, что для построения замыкания с помощью ЭВМ для состояния КС с 60 сущностями, 60 ролями и

30 недоверенными пользователями потребовалось 7 мин. В реальных КС количество сущностей несоизмеримо больше, что затрудняет применение вышеуказанного алгоритма для анализа их защищенности. В связи с этим возникает задача построения алгоритма, пригодного для анализа безопасности компьютерных систем на практике.

Рассмотрим средство управления доступом и информационными потоками в ОС класса *Unix* — *SELinux* и предложим метод проверки возможности реализации информационного потока по памяти между сущностями КС, защищенной с помощью данного средства, пригодный для практического применения.

Как было отмечено выше, для описания политик безопасности в *SELinux* используется собственный язык. Данный язык имеет множество синтаксических конструкций. Ввиду того, что разбор политики безопасности не является основной задачей данной работы, будем рассматривать лишь некоторые конструкции языка, субъективно наиболее важные для проверки возможности реализации информационного потока по памяти между сущностями КС. Такими конструкциями были выбраны определения типов и доменов, а также векторов доступа. Конструкции, отвечающие за определение атрибутов, ограничений, протоколирование, были исключены из рассмотрения. Кроме того, язык политик безопасности *SELinux* позволяет задавать принудительные переходы субъектов КС между доменами. Данные конструкции языка далее также рассматривать не будем. Такие довольно сильные ограничения на исходный текст политики безопасности возможно наложить ввиду того, что существуют работы (например, [2]), в которых предлагаются формальные модели, основанные именно на языке политик безопасности и учитывающие его значимые синтаксические конструкции. При расширении предлагаемого далее метода на весь язык политик можно воспользоваться результатами, изложенными в подобных работах. В связи с вышесказанным дальнейшие рассуждения будем вести в следующих предположениях. Будем рассматривать лишь конструкции языка описания политик безопасности *SELinux*, отвечающие за определение типов, а также векторов доступа и не будем рассматривать информационные потоки по времени между сущностями КС. В связи с тем, что информационные потоки по времени в КС не участвуют в порождении информационных потоков по памяти, а данный раздел посвящен последним, это предположение не ограничивает общности рассуждений.

Рассмотрим подробнее язык описания политик безопасности *SELinux*, а именно те его конструкции, которые были выбраны для анализа возможности реализации информационного потока по памяти.

Определение типа имеет одну из следующих форм (листинг 1):

```
1 type type_id;  
2 type type_id, attribute_id;  
3 type type_id alias alias_id;  
4 type type_id alias alias_id, attribute_id;
```

Листинг 1. Определение типа в *SELinux*

В них

`type` — ключевое слово,

`type_id` — идентификатор типа,

`alias` — ключевое слово,

`alias_id` — необязательный псевдоним (один или несколько) типа `type_id`,

`attribute_id` — один или несколько идентификаторов атрибутов.

Как говорилось выше, конструкции языка, описывающие атрибуты, а также псевдонимы типов в данной работе не рассматриваются.

Определение вектора доступа имеет вид

```
rule_name source_type target_type : class perm_set; (1)
```

В нём

`rule_name` — одно из ключевых слов `allow`, `dontaudit`, `auditallow`, `neverallow`;

`source_type`, `target_type` — один или несколько типов источника/назначения, либо идентификаторов атрибутов;

`class` — один или несколько классов объектов;

`perm_set` — права доступа типа источника к типу назначения.

В соответствии с данным вектором доступа типам из `source_type` будет разрешено обращаться к типам `target_type` как к объектам класса `class` с правами `perm_set`.

Также данная синтаксическая конструкция может включать метки, похожие на регулярные выражения, позволяющие кратко записать, например, все множество прав доступа либо множество без одного элемента.

Далее будем рассматривать лишь те определения векторов доступа, в которых используется ключевое слово `allow`, а права доступа записаны явно, без использования специальных меток.

Для того чтобы учесть функционально ассоциированные сущности при анализе возможности реализации информационного потока по памяти, в язык описания политик безопасности вводится новая синтаксическая конструкция, которая имеет вид

```
fas subj_types : assoc_types; (2)
```

В ней

`subj_types` — один или несколько идентификаторов типов,

`assoc_types` — один или несколько идентификаторов типов.

Данная конструкция указывает на то, что типы, перечисленные в `assoc_types`, являются функционально ассоциированными с типами из `subj_types`.

Сам язык описания политик безопасности *SELinux* не включает в себя подобной синтаксической конструкции. Она служит лишь для анализа возможности реализации информационного потока по памяти и не является правилом, разграничивающим доступ между субъектами и объектами. Определения функционально ассоциированных сущностей с помощью вышеуказанной конструкции могут быть вынесены в отдельный файл таким образом, что оригинальная политика безопасности останется неизменной.

Опишем метод проверки возможности реализации информационного потока по памяти между сущностями КС, защищенной с помощью средства *SELinux*.

Метод состоит в построении по тексту политики безопасности графа информационных потоков по памяти между сущностями КС.

Поскольку сама политика безопасности *SELinux* (в оговоренных выше ограничениях) для разграничения доступа между типами включает лишь правила вида (1), то для анализа возможности реализации информационного потока по памяти будем отталкиваться именно от этих правил. Особенностью данных правил является то, что множество прав доступа не ограничивается традиционными правами на чтение, запись и исполнение. Права доступа основываются на системных вызовах ядра *Linux*, то есть, грубо говоря, определение вектора доступа указывает на то, какие системные вызовы тип источника может применять к типу назначения. Например, возможны следующие определения векторов доступа:

```
allow initrc_t acct_exec_t : file { getattr read execute };
```

```
allow ftpd_t initrc_t : fifo_file { getattr read write append ioctl lock };
```

Более подробную информацию по этому вопросу можно найти в [5].

Идея метода состоит в том, чтобы абстрагироваться от прав доступа субъектов к объектам и построить ориентированный граф, в котором вершины будут сопоставлены сущностям КС, а дуги — информационным потокам по памяти между сущностями, сопоставленными вершинам.

Применительно к политике *SELinux* это означает, что по векторам доступа строится ориентированный граф, в котором вершины будут сопоставлены типам, указанным в векторах доступа, а дуги — возможным информационным потокам по памяти между типами, сопоставленными вершинам. Здесь нужно уточнить, как по политике *SELinux*, а именно по векторам доступа, определить возможные информационные потоки по памяти между типами, указанными в векторах доступа. Поскольку в векторах доступа указывается множество прав доступа между типами, то предлагается определить множество прав доступа, при обладании которыми возможна реализация информационного потока по памяти от домена либо к нему. Для этого предлагается ввести конструкцию вида

```
write_m direction : class perm_set; (3)
```

В ней

`write_m` — ключевое слово;

`direction` — одно из ключевых слов `to`, `from`;

`class` — идентификатор класса;

`perm_set` — множество прав доступа.

Данная конструкция указывает на то, что при обладании доменом любым из прав в `perm_set` возможна реализация информационного потока по памяти между доменом и типом в направлении `direction` (`to` соответствует направлению от домена к типу, `from` — наоборот).

Например, возможны следующие определения информационных потоков по памяти:

```
write_m to : file { write append };
```

```
write_m to : fifo_file { write append };
```

```
write_m from : chr_file { read };
```

Язык описания политик безопасности *SELinux* не включает в себя подобной конструкции, и она вводится лишь для анализа возможности реализации информационного потока по памяти между типами. Данные конструкции могут быть вынесены в отдельный файл так, что оригинальная политика безопасности останется неизменной.

Далее будем использовать термины и обозначения ФАС ДП-модели:

E — множество сущностей;

$S \subset E$ — множество субъектов;

$[s] \subset E$ — множество всех сущностей, функционально ассоциированных с субъектом s (при этом по определению выполняется условие $s \in [s]$, и для каждого субъекта множество сущностей, функционально с ним ассоциированных, не изменяется в процессе функционирования системы);

$R_f = \{write_m\}$ — множество видов информационных потоков, где $write_m$ — информационный поток по памяти на запись в сущность;

$F_a \subset E \times E \times R_f$ — множество информационных потоков между сущностями;

$F \subset E \times E$ — множество информационных потоков по памяти.

Пусть определены множества S , E , F , и пусть для всех $s \in S$ определено множество $[s]$. Определим $G = (E, F)$ — конечный ориентированный граф, в котором элементы множества E являются вершинами, а элементы множества F — дугами.

Метод 1 проверки возможности реализации информационного потока по памяти между сущностями КС.

Пусть определен граф $G = (E, F)$.

1. Для всех $s \in S$, для всех $e \in [s] \setminus \{s\}$ положить $F = F \cup \{(e, s)\}$.
2. Для всех $s \in S$, для всех $f \in [s]$, для всех $e \in E$, если существует путь из e в f , положить $F = F \cup \{(s, e)\}$.
3. Реализация информационного потока по памяти от сущности $e_1 \in E$ к сущности $e_2 \in E, e_2 \neq e_1$, возможна тогда и только тогда, когда в графе G существует путь из e_1 в e_2 (если путь есть — ответ «да», иначе — «нет»).

Покажем теперь, как данный метод может быть применен для анализа возможности реализации информационного потока по памяти между типами, указанными в политике безопасности *SELinux*.

Пусть мы имеем текст политики безопасности *SELinux* и определения функционально ассоциированных сущностей с помощью конструкции (2) и информационных потоков по памяти с помощью конструкции (3).

Задача ставится следующим образом: для данных двух типов, указанных в политике безопасности, определить возможность реализации информационного потока по памяти от первого типа ко второму.

Предложим метод решения данной задачи.

Метод 2 проверки возможности реализации информационного потока по памяти в *SELinux*.

1. Положить $S = E = F = \emptyset$ и $G = (E, F)$.
2. Для каждого определения типа вида листинг 1 (строка 1): если тип является типом субъекта (доменом), то добавить соответствующий ему элемент в множество S , иначе — в множество E . Для каждого определения функционально ассоциированных сущностей вида (2) добавить элементы, соответствующие ассоциированным типам, в множество $[s]$, где s — элемент, соответствующий типу субъекта.
3. Для каждого вектора доступа вида (1): если в E отсутствуют элементы, соответствующие типам, указанным в векторе доступа, то аналогично п. 2 добавить эти типы в множества S или E . Для каждого определения информационного потока вида (3): если вектор доступа содержит хотя бы одно из прав, указанных в (3), то для каждого типа источника и каждого типа назначения в векторе добавить дуги (т.е. элементы в F) по направлению, указанному в определении информационного потока.
4. Для графа $G = (E, F)$ применить метод 1.
5. Реализация информационного потока по памяти от типа — источника к типу — приемнику возможна тогда и только тогда, когда в п. 4 метод 1 выдаёт ответ «да» для пары вершин, соответствующих этим типам.

Пример.

Пусть задана политика безопасности *SELinux*:

```
allow user_t tmp_t : file {read write append};
allow ftpd_t tmp_t : file {write append};
allow ftpd_t ftpd_tmpfs_t:file { create open getattr setattr read write };
allow user_t etc_t : file {getattr};
allow eva_t etc_t : file {write};
```

И пусть заданы определения информационных потоков по памяти и функционально ассоциированных сущностей:

```

write_m to : file {write append};
write_m from : file {read};
fas user_t : {etc_t};

```

Построенный в п. 3 метода 2 граф будет выглядеть, как показано на рис. 1.

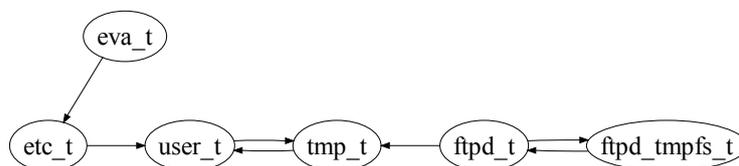


Рис. 1. Граф, полученный в п. 3 метода 2

Если в п. 5 метода 2 рассмотреть не одну пару вершин, а все возможные упорядоченные пары и всякий раз добавлять в граф дугу тогда и только тогда, когда метод 1 выдаёт ответ «да» для данной пары вершин, то получим граф, изображенный на рис. 2.

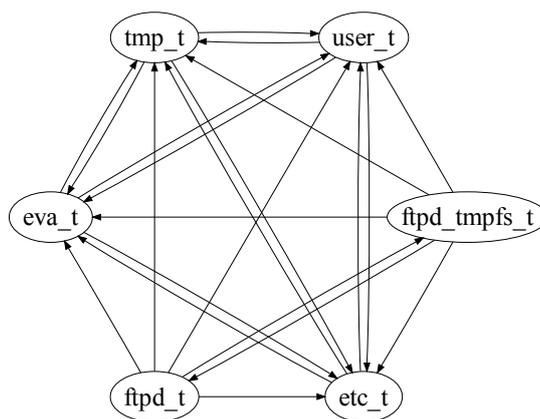


Рис. 2. Граф всех возможных информационных потоков по памяти между типами

2.3. Практическая реализация

Для анализа политик безопасности *SELinux* было разработано программное средство, основанное на методе 2 проверки возможности реализации информационного потока по памяти в *SELinux*. Данное средство позволяет по политике безопасности, а также по определениям информационных потоков по памяти вида (3) и функционально ассоциированных с субъектами сущностей вида (2) ответить на вопрос, возможна ли реализация информационного потока по памяти между типами, указанными в политике безопасности *SELinux*. Входными данными для данного средства являются текст политики безопасности *SELinux*, а также определения собственных конструкций. Кроме того, на вход могут быть поданы имена типов, для которых необходимо проверить возможность реализации информационного потока по памяти. На выходе средство дает информацию о возможности реализации потока либо между двумя указанными типами, либо между всеми типами, указанными в политике безопасности *SELinux*. Также средство способно вывести граф информационных потоков в виде

изображения. Данное средство написано на языке программирования *Python* с использованием библиотек *sepolgen* и *pygraph*. В библиотеке *sepolgen* исправлены некоторые ошибки, а также добавлена новая функциональность.

С помощью данного средства проанализирована часть реальной политики безопасности *SELinux* из стандартного пакета для дистрибутива *Ubuntu 9.04*. Для анализа была выбрана модульная политика для *ftp*-сервиса *ftpd*. По тексту данной политики, а также по определениям информационных потоков по памяти и функционально ассоциированных сущностей построен граф информационных потоков (рис. 3).

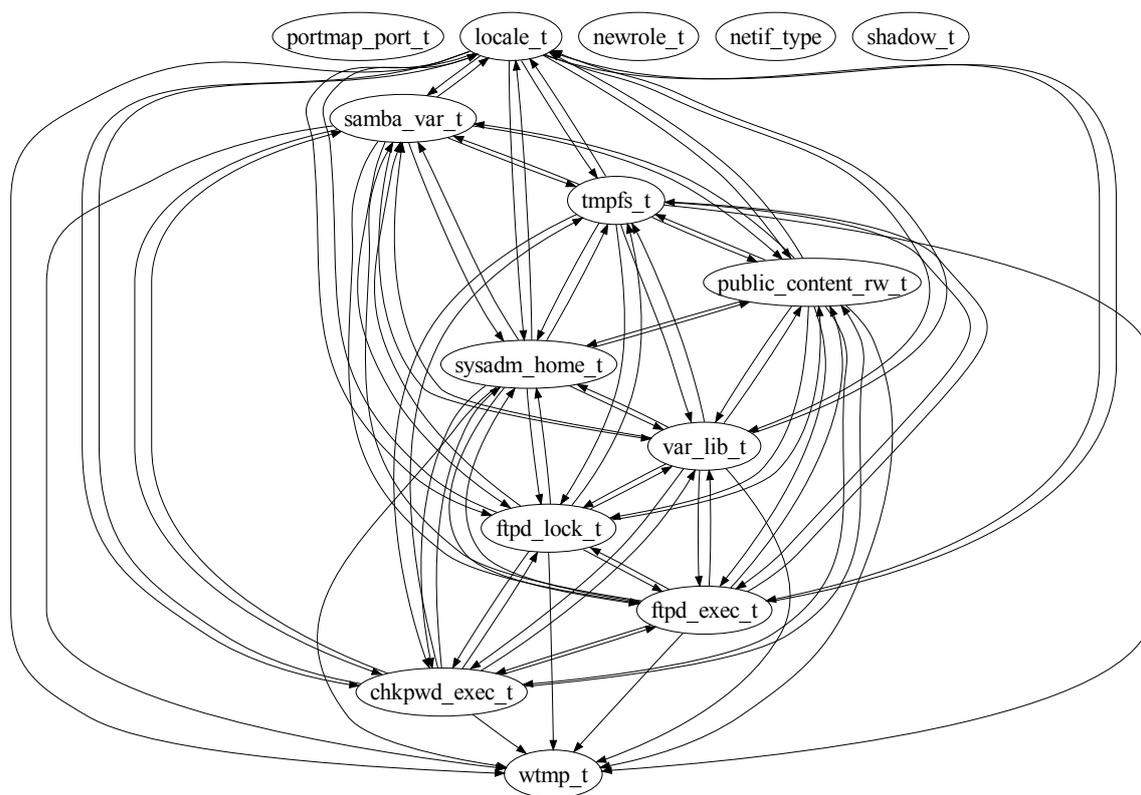


Рис. 3. Часть графа информационных потоков по памяти для политики *ftpd*

Таким образом, для всех типов, указанных в политике безопасности для *ftpd*, удалось определить возможность реализации информационных потоков по памяти между ними.

Заключение

В данной работе рассмотрены вопросы анализа безопасности информационных потоков в ОС семейства *GNU/Linux*. Исследованы информационные потоки по времени в ОС *GNU/Linux*, в том числе и новые информационные потоки по времени с участием доверенных субъектов. Приведены конкретные примеры возможности реализации потоков в ОС *GNU/Linux* с использованием виртуальной файловой системы, сокетов и времени последнего доступа к файлу с подробным описанием и рекомендациями по защите. Рассмотрено распространенное средство реализации политик безопасности в ОС *GNU/Linux* – *SELinux*, предложен метод проверки возможности реализации инфор-

мационного потока по памяти между сущностями КС, а также описано программное средство автоматизированного анализа политики безопасности *SELinux* с примерами его практического применения.

ЛИТЕРАТУРА

1. *Девянин П. Н.* Анализ безопасности управления доступом и информационными потоками в компьютерных системах. М.: Радио и связь, 2006. 176 с.
2. *Zanin G., Mancini L.* Towards a formal model for security policies specification and validation in the selinux system // Proc. of the ninth ACM symposium on Access control models and technologies. NY, USA: ACM, 2004. P. 136–145.
3. <http://selinux-symposium.org/2005/presentations/session5/5-3-macmillan.pdf> — SELinux Symposium. 2005.
4. *Качанов М. А.* Замыкание базовой ролевой ДП-модели // Прикладная дискретная математика. 2009. Приложение № 1. С. 41–44.
5. http://selinuxproject.org/page/Main_Page — SELinux Project Wiki. 2010.

**АУТЕНТИФИКАЦИЯ В МОДЕЛИ ДОВЕРЕННОЙ ПОДСИСТЕМЫ
НА ОСНОВЕ КОММУТАТИВНОГО ШИФРОВАНИЯ¹**

П. А. Паутов

*Томский государственный университет, г. Томск, Россия***E-mail:** __Pavel__@mail.ru

В работе рассматривается подход к организации аутентификации в многоуровневой системе, известный как «модель доверенной подсистемы». Для данного подхода формируются требования безопасности и приводится протокол аутентификации, удовлетворяющий этим требованиям. Описываемый протокол построен с использованием коммутативного алгоритма шифрования. Рассматриваются несколько конкретных коммутативных алгоритмов шифрования, применимых в описываемом протоколе.

Ключевые слова: *многоуровневые системы, аутентификация в многоуровневых системах, коммутативное шифрование.*

Введение

Рассмотрим систему, состоящую из трёх взаимодействующих подсистем: клиент, внешний сервер, внутренний сервер. Клиент взаимодействует только с внешним сервером, внешний сервер взаимодействует как с клиентом, так и с внутренним сервером (внешний сервер является клиентом внутреннего сервера). Внешний сервер взаимодействует с внутренним только для обработки запросов своих клиентов.

В таких многоуровневых системах обычно используется одна из двух следующих моделей организации аутентификации [1]:

- 1) модель делегирования;
- 2) модель доверенной подсистемы.

В модели делегирования внешний сервер взаимодействует с внутренним от имени клиента, т. е. внутренний сервер содержит учётную запись для каждого клиента внешнего сервера. В модели доверенной подсистемы внешний сервер взаимодействует с внутренним от имени фиксированного набора учётных записей, т. е. одна учётная запись внутреннего сервера соответствует нескольким клиентам внешнего сервера. В данной работе рассматривается модель доверенной подсистемы.

1. Постановка задачи

В модели доверенной подсистемы для взаимодействия с внутренним сервером используется учётная запись, соответствующая привилегиям клиента внешнего сервера. Например, на внешнем сервере клиенты делятся на группы по привилегиям: «гости», «операторы», «администраторы». Тогда для взаимодействия с внутренним сервером можно использовать три учётных записи, соответствующих группам клиентов. Если внешний сервер сам выбирает учётную запись для взаимодействия с внутренним сервером, то в случае компрометации первого злоумышленник сможет использовать

¹Работа выполнена в рамках реализации ФЦП «Научные и научно-педагогические кадры инновационной России» на 2009–2013 гг. (гос. контракт № П1010).

учётную запись с максимальными привилегиями. Возникает задача разработки такой схемы аутентификации, при которой внешний сервер смог бы использовать для взаимодействия с внутренним сервером только ту учётную запись, которая соответствует клиенту, и только тогда, когда клиент взаимодействует с внешним сервером. То есть если клиент относится к группе «гости», то внешний сервер может пройти аутентификацию перед внутренним сервером только от имени учётной записи «гость». И внешний сервер не может пройти аутентификацию перед внутренним сервером от имени учётной записи «гость» без помощи клиента, относящегося к группе «гости».

Искомая схема аутентификации должна удовлетворять следующим двум условиям:

- C1. При взаимодействии с клиентом внешний сервер может пройти аутентификацию перед внутренним сервером только от имени учётной записи, соответствующей данному клиенту.
- C2. Внешний сервер не может пройти аутентификацию перед внутренним сервером от имени какой-либо учётной записи без помощи клиента.

В работе автора [2] предложено несколько схем для случая, когда между клиентом и внешним сервером и между внешним сервером и внутренним используется парольная аутентификация. При использовании парольной аутентификации внешнему серверу необходим пароль учётной записи внутреннего сервера для того, чтобы пройти аутентификацию от имени данной учётной записи. То есть если злоумышленник скомпрометирует внешний сервер так, что он получит доступ на чтение к памяти внешнего сервера, то когда клиент инициирует выполнение протокола аутентификации, описанного в [2], злоумышленник получит пароль учётной записи внутреннего сервера, соответствующей привилегиям клиента. После одного сеанса связи клиента со скомпрометированным внешним сервером злоумышленник получает возможность использовать внутренний сервер без помощи клиента.

2. Протокол на основе коммутативного шифрования

Предлагается схема аутентификации, при использовании которой даже после сеанса связи клиента со скомпрометированным внешним сервером злоумышленник не сможет использовать внутренний сервер без помощи клиента.

Для построения предлагаемой схемы необходимы следующие криптографические примитивы:

- 1) E — коммутативный алгоритм шифрования, D — соответствующий алгоритм расшифрования;
- 2) H — хэш-функция.

Каждой учётной записи внутреннего сервера ставится в соответствие некоторый секрет S . Каждому клиенту внешнего сервера ставится в соответствие ключ K алгоритма E . На внешнем сервере для каждого клиента хранится результат шифрования $E_K(S)$, где S соответствует учётной записи внутреннего сервера для данного клиента. Алгоритм аутентификации клиента будет выглядеть следующим образом.

1. Клиент посылает внешнему серверу своё имя.
2. Внешний сервер посылает внутреннему серверу имя учётной записи, соответствующей клиенту.
3. Внутренний сервер генерирует случайный ключ шифрования K_r алгоритма E и передаёт его внешнему серверу.
4. Внешний сервер находит запись $E_K(S)$, соответствующую данному клиенту, вычисляет $E_{K_r}(E_K(S))$ и передаёт полученный результат клиенту.

5. Клиент вычисляет $H(D_K(E_{K_r}(E_K(S)))) \equiv H(E_{K_r}(S))$ и передаёт данное значение внешнему серверу.
6. Внешний сервер передаёт полученное значение внутреннему серверу.
7. Внутренний сервер вычисляет $H(E_{K_r}(S))$ и сравнивает результат со значением, полученным от внешнего сервера. Если значения совпадают, то клиент прошёл аутентификацию перед внешним сервером, а внешний сервер перед внутренним.

Как видно из описания, данный протокол обеспечивает выполнение условий С1, С2. Если внешний сервер попытается использовать запись $E_{K'}(S')$, не соответствующую данному клиенту, то проверка на шаге 7 не выполнится, так как $H(D_K(E_{K_r}(E_{K'}(S')))) \neq H(E_{K_r}(S'))$. Для того чтобы пройти аутентификацию перед внутренним сервером без помощи клиента, внешнему серверу понадобится знание S , но это значение не доступно внешнему серверу в открытом виде.

Хэш-функция используется для того, чтобы предотвратить атаку, в которой внешний сервер генерирует собственное значение K'_r и, получив от клиента $E_{K'_r}(S)$, раскрывает значение S .

Используемый коммутативный алгоритм может быть как симметричным, так и асимметричным. В асимметричном варианте каждому клиенту ставится в соответствие пара ключей: открытый K_e и закрытый K_d . На внешнем сервере для каждого клиента хранятся значение $E_{K_e}(S)$ и открытый ключ клиента K_e . Алгоритм аутентификации аналогичен симметричному варианту (но на шаге 3 внутреннему серверу достаточно сгенерировать только открытый ключ).

3. Операции по управлению пользователями

Применение описанной схемы аутентификации повлияет на операции по управлению пользователями. Рассматриваются следующие операции:

- 1) создание нового пользователя;
- 2) смена ключа пользователем;
- 3) изменение учётной записи внутреннего сервера;
- 4) смена учётной записи внутреннего сервера для данного пользователя системы.

Алгоритмы операций по управлению пользователями строятся по аналогии с [2].

3.1. Операции по управлению пользователями для симметричного E

Создание нового пользователя

1. Администратор генерирует новый ключ K и вычисляет $E_K(S)$ для S , соответствующего новому пользователю.
2. Администратор отправляет K пользователю, а $E_K(S)$ внешнему серверу.

Смена ключа пользователем

1. Пользователь проходит аутентификацию перед внешним сервером.
2. Пользователь запрашивает смену ключа.
3. Внешний сервер генерирует случайный ключ K_r , вычисляет $E_{K_r}(E_K(S))$ и передаёт полученный результат клиенту.
4. Клиент расшифровывает полученное значение на старом ключе и шифрует на новом, отправляет серверу.
5. Сервер расшифровывает полученное от клиента значение с помощью K_r и записывает результат вместо старого значения $E_K(S)$.

Смена учётной записи внутреннего сервера

На внешнем сервере для каждого пользователя хранится запись вида $E_K(S)$. При смене учётной записи внутреннего сервера будет необходимо заменить S на S' . Для этого потребуется зашифровать новое значение S' на ключе пользователя, а так как ключ пользователя известен только пользователю, то провести данную операцию проблематично.

Изменение учётной записи внутреннего сервера для данного пользователя системы

Проведение операции проблематично по тем же причинам, что и проведение операции «Смена учётной записи внутреннего сервера».

3.2. Операции по управлению пользователями для асимметричного E

Создание нового пользователя

1. Пользователь генерирует свои закрытый и открытый ключи и отправляет администратору открытый ключ.
2. Администратор шифрует открытым ключом пользователя соответствующее S и отправляет на внешний сервер.

Смена ключа пользователем

1. Пользователь запрашивает смену ключа.
2. Внешний сервер генерирует случайную пару ключей (K_{er}, K_{dr}) , вычисляет $E_{K_{er}}(E_{K_e}(S))$ и передаёт полученный результат клиенту.
3. Клиент расшифровывает полученное значение на старом ключе и шифрует на новом, отправляет серверу значение и новый открытый ключ.
4. Сервер расшифровывает полученное значение с помощью K_{dr} и записывает результат вместо старого значения $E_{K_e}(S)$.

Смена учётной записи внутреннего сервера

Администратор системы обновляет записи $E_{K_e}(S)$ для каждого пользователя, используя открытые ключи пользователей K_e .

Изменение учётной записи внутреннего сервера для данного пользователя системы

Администратор системы обновляет запись $E_{K_e}(S)$ для данного пользователя, используя соответствующий открытый ключ K_e .

4. Применимые коммутативные алгоритмы шифрования

4.1. Сложение по модулю 2

Сложение по модулю 2 можно рассматривать как симметричный коммутативный алгоритм шифрования. Функции шифрования и расшифрования в данном случае совпадают и имеют вид $E_K(X) = D_K(X) = X \oplus K$, где \oplus — побитовое сложение по модулю 2 (X и K рассматриваются как булевы векторы одинаковой длины n). Злоумышленник, скомпрометировавший внешний сервер, получит доступ к значениям вида $E_K(S)$ для всех клиентов и для всех учётных записей внутреннего сервера. Пусть внешний сервер имеет m клиентов и использует одну учётную запись внутреннего сервера. Тогда для получения доступа к внутреннему серверу злоумышленник должен

будет решить следующую систему уравнений:

$$\begin{cases} K_1 \oplus S = e_1, \\ K_2 \oplus S = e_2, \\ \dots \\ K_m \oplus S = e_m, \end{cases}$$

где K_1, K_2, \dots, K_m (ключи клиентов) и S (секрет внутреннего сервера) являются неизвестными, а e_1, e_2, \dots, e_m — известные значения, хранимые на внешнем сервере. В данной системе S является свободной переменной, и, следовательно, система будет иметь 2^n решений, так как S — булев вектор длины n . Таким образом, злоумышленнику придётся произвести полный перебор всех возможных значений S .

4.2. Возведение в степень по модулю простого числа

В качестве симметричного коммутативного алгоритма шифрования можно выбрать функцию возведения в степень по модулю большого простого числа. Пусть p — большое простое число; p является общеизвестным параметром системы. В качестве ключа выбирается K ($1 \leq K \leq p - 2$), взаимно простое с $p - 1$. Тогда функции шифрования, расшифрования будут выглядеть следующим образом: $E_K(X) = X^K \bmod p$, $D_K(X) = X^{K^{-1}} \bmod p$. Данный алгоритм известен в литературе как алгоритм Полига — Хеллмана [3]. В качестве коммутативного шифра он используется в [4].

4.3. RSA

Как видно из описания предлагаемого протокола, на внешнем и внутреннем серверах необходимо лишь выполнить операцию шифрования S на некотором случайном ключе K_r . Так как операция расшифровки не требуется, то для вычислений на внутреннем и внешнем серверах можно использовать некоторую ключевую хэш-функцию H'_K , коммутативную с алгоритмом E_K (т. е. $H'_{K_1}(E_{K_2}(X)) = E_{K_2}(H'_{K_1}(X))$). С учётом сказанного предлагаемый протокол будет выглядеть следующим образом.

1. Клиент посылает внешнему серверу своё имя.
2. Внешний сервер посылает внутреннему серверу имя учётной записи, соответствующей клиенту.
3. Внутренний сервер генерирует случайный ключ K_r хэш-функции H' и передаёт его внешнему серверу.
4. Внешний сервер находит запись $E_{K_r}(S)$, соответствующую данному клиенту, вычисляет $H'_{K_r}(E_{K_r}(S))$ и передаёт полученный результат клиенту.
5. Клиент вычисляет $H(D_{K_r}(H'_{K_r}(E_{K_r}(S)))) \equiv H(H'_{K_r}(S))$ и передаёт данное значение внешнему серверу.
6. Внешний сервер передаёт полученное значение внутреннему серверу.
7. Внутренний сервер вычисляет $H(H'_{K_r}(S))$ и сравнивает результат со значением, полученным от внешнего сервера. Если значения совпадают, то клиент прошёл аутентификацию перед внешним сервером, а внешний сервер перед внутренним.

В качестве асимметричного варианта алгоритма E можно использовать шифр-систему RSA с модулем n , открытой экспонентой e и закрытой экспонентой d . Каждому клиенту ставится в соответствие пара ключей ($K_e = (e, n)$, $K_d = (d)$). На внешнем сервере для каждого клиента хранится пара $(E_{K_e}(S), K_e) = (S^e \bmod n, (e, n))$ для соответствующего S . Функцию H' можно выбрать как $H'_K(X) = X^K \bmod n$. Тогда предлагаемый протокол будет выглядеть следующим образом.

1. Клиент посылает внешнему серверу своё имя.

2. Внешний сервер посылает внутреннему серверу имя учётной записи и модуль RSA n , соответствующие клиенту.
3. Внутренний сервер генерирует случайное число r и передаёт его внешнему серверу.
4. Внешний сервер находит запись $S^e \bmod n$, соответствующую данному клиенту, вычисляет $S^{er} \bmod n$ и передаёт полученный результат клиенту.
5. Клиент вычисляет $H(S^{er} \bmod n) \equiv H(S^r \bmod n)$ и передаёт данное значение внешнему серверу.
6. Внешний сервер передаёт полученное значение внутреннему серверу.
7. Внутренний сервер вычисляет $H(S^r \bmod n)$ и сравнивает результат со значением, полученным от внешнего сервера. Если значения совпадают, то клиент прошёл аутентификацию перед внешним сервером, а внешний сервер перед внутренним.

Заключение

Рассмотренный в данной работе протокол аутентификации предоставляет более сильные гарантии по сравнению со схемой, описанной в [2]. При использовании данного протокола, даже после сеанса связи клиента со скомпрометированным внешним сервером, злоумышленник не сможет использовать внутренний сервер без помощи клиента. Однако для внедрения данного протокола потребуется добавить его поддержку на все уровни приложения, в то время как описанная в работе [2] схема опирается на широко распространённую парольную аутентификацию. В зависимости от требований к конкретной системе можно использовать тот или иной подход к организации аутентификации.

ЛИТЕРАТУРА

1. Chong F. Trusted Subsystem Design // MSDN. 2006. <http://msdn.microsoft.com/en-us/library/aa905320.aspx>
2. Паутов П. А. Проблема аутентификации в многоуровневых приложениях // Прикладная дискретная математика. 2008. № 2. С. 87–90.
3. Schneier B. Applied Cryptography: Protocols, Algorithms, and Source Code in C. Second Edition. Wiley, 1996. 785 p.
4. Bao F., Deng R. H., Feng P. An Efficient and Practical Scheme for Privacy Protection in the E-Commerce of Digital Goods // LNCS. 2001. V. 2015. P. 162–170.

ПРИКЛАДНАЯ ТЕОРИЯ ГРАФОВ

DOI 10.17223/20710410/9/9

УДК 519.17

СЕМЕЙСТВО ТОЧНЫХ 2-РАСШИРЕНИЙ ТУРНИРОВ

А. А. Долгов

*Саратовский государственный университет им. Н. Г. Чернышевского, г. Саратов***E-mail:** dolgov.a.a@gmail.com

В работе рассматривается семейство турниров, имеющих точное 1- и 2-расширение, но не имеющих точного 3-расширения. Это единственное известное семейство графов с таким свойством и четвертое среди семейств графов, имеющих точное k -расширение при $k > 1$.

Ключевые слова: граф, точное k -расширение, циркулянт.

Введение

Ориентированным графом (орграфом) называется пара $G = (V, \alpha)$, где V — конечное непустое множество, называемое *множеством вершин*, а α — отношение на множестве вершин V , называемое *отношением смежности*.

Граф с симметричным и антирефлексивным отношением смежности называется *неориентированным графом*. Граф с антисимметричным отношением смежности называется *направленным графом*, или *диграфом*. Полный диграф без петель называется *турниром* [1].

Граф H называется *точным (вершинным) k -расширением* графа G , если граф G изоморфен каждому подграфу H , получающемуся путем удаления любых его k вершин и всех связанных с ними дуг (ребер).

Два графа $G_1 = (V_1, \alpha_1)$ и $G_2 = (V_2, \alpha_2)$ называются *изоморфными*, если можно установить взаимно однозначное соответствие $f : V_1 \rightarrow V_2$, сохраняющее отношение смежности: $(u, v) \in \alpha_1 \Leftrightarrow (f(u), f(v)) \in \alpha_2$ для любых $u, v \in V_1$. Изоморфизм графа на самого себя называется *автоморфизмом*. Множество автоморфизмов графа G образует группу, обозначаемую $\text{Aut}(G)$.

Две вершины u и v графа G называются *подобными*, если существует автоморфизм графа G , при котором образом вершины u является вершина v . Граф, все вершины которого подобны, называется *вершинно-симметрическим*.

Циркулянт называется n -вершинный граф G , такой, что, если его вершинам приспаны метки от 0 до $n - 1$, то из вершины i в вершину j проходит дуга тогда и только тогда, когда $(i - j) \bmod n \in S$, где S — некоторое подмножество множества $\mathbb{Z}_n \setminus \{0\}$. Известно, что группа автоморфизмов циркулянта $G = (V, \alpha)$ транзитивна, то есть для всех $v, u \in V$ существует $\varphi \in \text{Aut}(G)$, такой, что $\varphi(v) = u$.

Группа Γ , действующая на множестве X , называется *дважды транзитивной*, если для любых $x_1, x_2, y_1, y_2 \in X$, таких, что $x_1 \neq y_1, x_2 \neq y_2$, в группе Γ найдется такое отображение φ , что $\varphi(x_1) = y_1$ и $\varphi(x_2) = y_2$.

Среди неориентированных графов существует всего два семейства графов, имеющих точное k -расширение при $k > 1$, — это полные и вполне несвязные графы [2]. Сре-

ди ориентированных графов точное k -расширение при $k > 1$ могут иметь только графы, симметризация которых является полным графом [3]. Таким образом, в классе направленных графов точное k -расширение при $k > 1$ могут иметь только турниры. Для турниров на данный момент известно только одно семейство точных k -расширений при $k > 1$ — это турниры с транзитивным отношением смежности (транзитивные турниры) [3]. Кроме того, в работе [4] приводится пара турниров, обладающих интересным свойством. Эти турниры имеют точное 1- и 2-расширение, но не имеют точного k -расширения при $k > 2$. На рис. 1 приведен один из этих турниров и его точные 1- и 2-расширения.

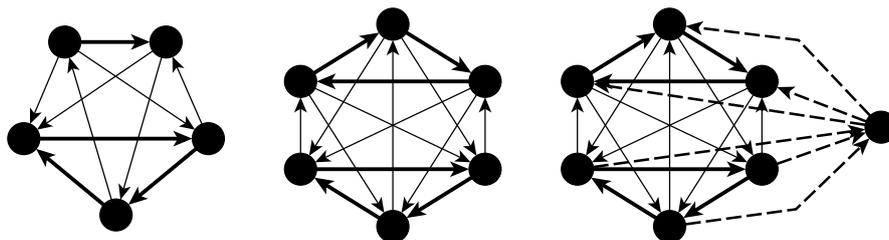


Рис. 1. Турнир и его точные 1- и 2-расширения

Данная работа посвящается описанию семейства точных 2-расширений турниров, к которому принадлежит и упомянутая пара.

Семейство турниров T_n

Рассмотрим p -вершинный граф $G = (V, \alpha)$, где p — простое и $p > 2$. Пусть $V = \{v_0, v_1, \dots, v_{p-1}\}$ — множество вершин G . Из вершины v_i в вершину v_j есть дуга только в том случае, когда $(j - i)$ — квадратичный вычет по модулю p , то есть по теореме Эйлера $(j - i)^{(p-1)/2} \equiv 1 \pmod{p}$. Обозначим полученный граф через T_p .

Рассмотрим общий вид матрицы смежности этого графа:

$$\begin{pmatrix} (1-1)^{(p-1)/2} \pmod{p} & (2-1)^{(p-1)/2} \pmod{p} & \dots & (p-1)^{(p-1)/2} \pmod{p} \\ (1-2)^{(p-1)/2} \pmod{p} & (2-2)^{(p-1)/2} \pmod{p} & \dots & (p-2)^{(p-1)/2} \pmod{p} \\ \dots & \dots & \dots & \dots \\ (1-p)^{(p-1)/2} \pmod{p} & (2-p)^{(p-1)/2} \pmod{p} & \dots & (p-p)^{(p-1)/2} \pmod{p} \end{pmatrix}.$$

Рассмотрим отображение на множестве вершин графа T_p , при котором v_i переходит в $v_{(i+1) \pmod{p}}$. Пусть в исходном графе была дуга из v_i в v_j , значит, $(j-i)^{(p-1)/2} \equiv 1 \pmod{p}$. После отображения получаем $v_j \Rightarrow v_{(j+1) \pmod{p}}$, $v_i \Rightarrow v_{(i+1) \pmod{p}}$. В полученном графе существует дуга из $v_{(i+1) \pmod{p}}$ в $v_{(j+1) \pmod{p}}$, если выполняется соотношение

$$(j+1 - (i+1))^{(p-1)/2} = (j-i)^{(p-1)/2} \equiv 1 \pmod{p}.$$

Значит, отображение $v_i \Rightarrow v_{(i+1) \pmod{p}}$ является автоморфизмом данного графа. Применяя нужное количество раз данный автоморфизм, мы сможем отобразить любую вершину графа в любую другую. Получается, что группа автоморфизмов графа T_p транзитивна. Известно, что граф с транзитивной группой автоморфизмов является точным 1-расширением [1]. Кроме того, то, что циклическая перестановка является автоморфизмом, означает, что T_p является циркулянтном [6].

В работе [5] указано, что при $p = 4n + 3$ граф такого вида является турниром. Рассмотрим пример при $p = 7$. Квадратичными вычетами по модулю 7 являются числа

$1^2 \bmod 7 = 1$, $2^2 \bmod 7 = 4$ и $3^2 \bmod 7 = 2$. Значит, матрица смежности графа T_7 имеет вид

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

Заметим, что граф T_7 изоморфен точному 2-расширению турнира, изображенного на рис 1.

Найдем все автоморфизмы для графа T_p . Для этого воспользуемся алгоритмом, предложенным Морисом в [6] и основывающимся на следующей теореме:

Теорема (Бернсайд, 1901). Пусть Γ — транзитивная группа, действующая на множестве из p элементов, p — простое. Тогда либо Γ — дважды транзитивная группа, либо $\Gamma = \{F_{a,b} : a \in H \subset \mathbb{Z}_p^*, b \in \mathbb{Z}_p\}$, где $F_{a,b}(v_i) = v_{(ai+b) \bmod p}$.

Данная теорема задает вид всех автоморфизмов указанного графа T_p . Алгоритм их получения заключается в нахождении таких $a \in \mathbb{Z}_p^*$, для которых отображение $F_{a,b}(v_i) = v_{(ai+b) \bmod p}$ является автоморфизмом. В результате получим множество H , включающее найденные a . Если множество H совпадает с \mathbb{Z}_p^* , то группа Γ — дважды транзитивная группа.

Рассмотрим отображение множества вершин графа T_p , при котором вершина с номером i переходит в $ia \bmod p$. Пусть в исходном графе есть дуга из v_i в v_j . После отображения получим: $v_i \Rightarrow v_{ia \bmod p}$, $v_j \Rightarrow v_{ja \bmod p}$. В полученном графе существует дуга из v_{ia} в v_{ja} , если выполняется соотношение

$$(ja - ia)^{(p-1)/2} \equiv 1 \pmod{p}.$$

Перепишем полученное соотношение:

$$a^{(p-1)/2}(j - i)^{(p-1)/2} \equiv 1 \pmod{p}.$$

Очевидно, что $(j - i)^{(p-1)/2} \equiv 1 \pmod{p}$, так как в исходном графе есть дуга из v_i в v_j . Следовательно, $a^{(p-1)/2} \equiv 1 \pmod{p}$.

Таким образом, отображение $F_{a,b}(v_i)$ является автоморфизмом T_p только в случае, когда $a \in \mathbb{Z}_p^*$ есть квадратичный вычет. Примечательно, что если a — квадратичный невычет, то отображение $F_{a,b}(v_i)$ является антиавтоморфизмом для данного графа (то есть такой перестановкой вершин, при которой все дуги графа заменяются на обратные).

Так как в \mathbb{Z}_p^* всего $(p - 1)/2$ квадратичных вычетов, то общее число автоморфизмов для T_p получается равным $p(p - 1)/2$. Заметим, что количество автоморфизмов совпадает с количеством всевозможных пар вершин без учета порядка. Если удастся показать, что для любой пары вершин v_i и v_j , $i \neq j$, среди всех автоморфизмов графа T_p можно найти точно один автоморфизм, которому соответствует перестановка вида v_i, v_j, \dots или v_j, v_i, \dots , то получится, что при удалении любой пары вершин мы получим изоморфные графы, а значит, T_p является точным 2-расширением.

То, что у T_p не может быть пары различных автоморфизмов, первые две вершины в которых совпадают, но идут в другом порядке, очевидно, поскольку, поменяв местами две первые вершины у любого турнира, мы изменим направление дуги, которая

связывает эти две вершины, а перестановка остальных вершин на эту дугу никак не влияет.

Остается вопрос, может ли у T_p существовать пара различных автоморфизмов, не меняющих две начальные вершины графа. Мы уже описали вид всех автоморфизмов T_p ; очевидно, интересующий нас вопрос можно сформулировать так: существуют ли $x, y, b \in \mathbb{Z}_p$, $x \neq y$, и $a \in \mathbb{Z}_p^*$, такие, что

$$\begin{aligned}x &\equiv ax + b \pmod{p}; \\y &\equiv ay + b \pmod{p}.\end{aligned}$$

Отсюда получаем $a = 1$ и $b = 0$, то есть таким свойством обладает только тождественный автоморфизм.

Из всего описанного следует, что граф T_p является точным 1-расширением, а если p — простое число вида $4n + 3$, то T_p является точным 1- и 2-расширением для подходящих турниров.

Таким образом, турниры, которые являются точными 2-расширениями, существуют при числе вершин $7, 11, 19, 23, 31, \dots$

ЛИТЕРАТУРА

1. Богомолов А. М., Салий В. Н. Алгебраические основы теории дискретных систем. М.: Наука, 1997.
2. Абросимов М. Б. Минимальные расширения дополнений графов // Теоретические задачи информатики и ее приложений. Саратов: СГУ, 2001. № 4. С. 11–19.
3. Абросимов М. Б. Минимальные расширения транзитивных турниров // Вестник Томского государственного университета. Приложение. 2006. № 17. С. 187–190.
4. Абросимов М. Б., Долгов А. А. Семейства точных расширений турниров // Прикладная дискретная математика. 2008. № 1. С. 101–107.
5. Eplett W. J. R. Self-converse tournaments // Canadian Mathematical Bulletin. 1979. No. 22. P. 23–27.
6. Morris J. Automorphism groups of circulant graphs — a survey // Graph Theory, Trends in Math. 2006. P. 311–325.

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ДИСКРЕТНЫХ АВТОМАТОВ

DOI 10.17223/20710410/9/10

УДК 621.391.1:004.7

РЕШЕНИЕ НЕРАВЕНСТВ НАД АВТОМАТАМИ В ПРОЕКТИРОВАНИИ РЕАКТИВНЫХ СИСТЕМ

А. Н. Чеботарёв

*Институт кибернетики НАН Украины, г. Киев, Украина***E-mail:** ancheb@gmail.com

Рассмотрена задача решения неравенств над автоматами, возникающая при композиционном подходе к проектированию реактивных систем. Задача формулируется и решается на уровне спецификаций автоматов в логическом языке L . Показано, как получить максимальное решение неравенства относительно операции синхронной композиции автоматов.

Ключевые слова: *реактивная система, спецификация в языке L , Σ -автомат, синхронная композиция Σ -автоматов, неравенство над Σ -автоматами, минимальное решение.*

Введение

Проектирование современных систем обработки информации все больше усложняется в связи с усложнением этих систем. Одним из основных способов борьбы со сложностью проектирования является модульный подход, связанный с декомпозицией проектируемой системы на подсистемы, которые взаимодействуют между собой в соответствии с определенными правилами композиции [1–3]. Многие задачи, возникающие при таком подходе к проектированию, могут быть сформулированы следующим образом. Заданы спецификация системы, которая реализуется в виде композиции двух модулей, и спецификация одного из этих модулей. Требуется определить (специфицировать) другой модуль так, чтобы его композиция с заданным модулем удовлетворяла спецификации системы. Для уточнения этой задачи необходимо:

- 1) определить способ представления спецификаций отдельных модулей и всей системы в целом;
- 2) уточнить понятие композиции модулей;
- 3) уточнить отношение, соответствующее понятию «модуль удовлетворяет спецификации».

Указанные уточнения будут сделаны для реактивных систем [4], т. е. систем, постоянно взаимодействующих с окружающей средой. Для спецификации таких систем будет использоваться язык L [5], представляющий собой фрагмент логики предикатов первого порядка с одноместными предикатами, которые интерпретируются на множестве \mathbb{Z} целых чисел. Рассматриваемая задача сводится к решению уравнений (неравенств) над автоматными моделями взаимодействующих модулей, представленных спецификациями в языке L . Поскольку автоматные модели определяют поведение

модуля, работающего потенциально бесконечно, рассматриваются автоматы над бесконечными входными последовательностями, что соответствует понятию циклического автомата.

1. Язык спецификации L

Спецификация в языке L имеет вид формулы $\forall t F(t)$, где $F(t)$ — формула, построенная с помощью логических связок из атомарных формул (атомов) вида $p(t+k)$, где p — одноместный предикатный символ, t — переменная, принимающая значения из множества целых чисел, рассматриваемого как множество моментов дискретного времени, а k — целочисленная константа, называемая *рангом атома*. Разность между максимальным и минимальным значениями рангов атомов, встречающихся в формуле, называется её *глубиной*. В дальнейшем формулы языка L вида $\forall t F(t)$ будем называть L -формулами.

При определении семантики языков спецификации реактивных систем такие языки рассматриваются как формализмы для задания множеств сверхслов (бесконечных слов) в алфавите двоичных векторов, длина которых равна количеству различных предикатных символов, встречающихся в спецификации. Определим необходимые понятия, касающиеся сверхслов.

Пусть Σ — конечный алфавит. Отображения $u : \mathbb{Z} \rightarrow \Sigma$ и $l : \mathbb{N} \rightarrow \Sigma$ называются соответственно двусторонним сверхсловом (обозначается $\dots u(-2)u(-1)u(0)u(1)u(2)\dots$) и сверхсловом (обозначается $l(1)l(2)\dots$) в алфавите Σ . Отрезок $u(\tau)u(\tau+1)\dots u(\tau+k)$ двустороннего сверхслова u обозначается $u(\tau, \tau+k)$. Бесконечный отрезок $u(k+1, \infty)$ будем называть k -суффиксом двустороннего сверхслова u . Множество всех двусторонних сверхслов в алфавите Σ будем обозначать $\Sigma^{\mathbb{Z}}$.

Перейдем теперь к описанию семантики языка L . Каждой L -формуле $F = \forall t F(t)$ ставится в соответствие множество моделей для этой формулы, т. е. множество таких интерпретаций, на которых F истинна. Пусть $\Omega = \{p_1, \dots, p_m\}$ — множество всех предикатных символов, встречающихся в формуле F (сигнатура формулы). Интерпретация формулы F — это упорядоченный набор определенных на \mathbb{Z} одноместных предикатов π_1, \dots, π_m , соответствующих предикатным символам из Ω . Пусть Σ — множество всех двоичных векторов длины m , тогда интерпретацию $I = \langle \pi_1, \dots, \pi_m \rangle$ можно представить в виде двустороннего сверхслова в алфавите Σ , а множество всех моделей для F — в виде множества $M(F)$ двусторонних сверхслов в этом алфавите. В дальнейшем не будем различать интерпретации и соответствующие им двусторонние сверхслова, поэтому будем говорить об истинности формулы F на двустороннем сверхслове $u \in \Sigma^{\mathbb{Z}}$ и значении формулы $F(t)$ в некоторой позиции τ двустороннего сверхслова u , понимая под этим значение формулы $F(\tau)$ в интерпретации u .

При интерпретации формул вида $\forall t F(t)$ на множестве целых чисел для любого $k \in \mathbb{Z}$ справедлива эквивалентность $\forall t F(t) \Leftrightarrow \forall t F(t+k)$, где $F(t+k)$ обозначает формулу, полученную из $F(t)$ путем добавления k к рангам всех её атомов (сдвиг на k). Таким образом, можно ограничиться рассмотрением формул, у которых максимальный ранг атомов равен 0. Будем считать, что L -формула $F = \forall t F(t)$ задает множество 0-суффиксов всех двусторонних сверхслов из $M(F)$. Обозначим это множество сверхслов $W(F)$.

2. Автоматная семантика языка L

Определение 1. Конечный неинициальный X - Y -автомат A — это четверка $\langle X, Y, Q, \chi_A \rangle$, где X, Y, Q — конечные множества соответственно входных символов, выходных символов и состояний; $\chi_A : Q \times X \times Y \rightarrow 2^Q$ — функция переходов автомата.

X - Y -автомат A называется *квазидетерминированным*, если $|\chi_A(q, x, y)| \leq 1$ для любых $q \in Q, x \in X, y \in Y$. Квазидетерминированные X - Y -автоматы удобно рассматривать как детерминированные частичные автоматы без выхода, с входным алфавитом $\Sigma = X \times Y$. Такой автомат $A = \langle \Sigma, Q, \delta_A \rangle$, где $\delta_A : Q \times \Sigma \rightarrow Q$ — частичная функция, будем называть Σ -автоматом.

Определение 2. Σ -автомат $A = \langle \Sigma, Q, \delta_A \rangle$ называется *циклическим*, если для каждого $q \in Q$ существуют такие $\sigma_1, \sigma_2 \in \Sigma$ и $q_1, q_2 \in Q$, что $q_1 = \delta_A(q, \sigma_1)$ и $q = \delta_A(q_2, \sigma_2)$.

В дальнейшем под автоматом будем понимать циклический Σ -автомат. Такой автомат можно однозначно охарактеризовать в терминах допустимых сверхслов.

Определение 3. Сверхслово $l = \sigma_1 \sigma_2 \dots$ в алфавите Σ *допустимо в состоянии* q автомата A , если существует такое сверхслово состояний $q_0 q_1 q_2 \dots$, где $q_0 = q$, что $q_{i+1} = \delta_A(q_i, \sigma_{i+1})$ для любого $i = 0, 1, 2, \dots$. Сверхслово l *допустимо для автомата* A , если оно допустимо хотя бы в одном из его состояний. Множество всех сверхслов, допустимых для автомата A , обозначим $W(A)$.

Два Σ -автомата A_1 и A_2 будем называть *эквивалентными (слабо эквивалентными)*, если $W(A_1) = W(A_2)$.

Предполагается, что символы алфавита Σ представляют собой двоичные векторы длины m , что соответствует кодированию абстрактных символов наборами значений двоичных переменных из $\Omega = \{x_1, \dots, x_m\}$. При использовании языка L для спецификации автоматов предикатные символы соответствуют этим переменным.

Автоматная семантика языка L определяется следующей теоремой.

Теорема 1 [5]. Для всякой непротиворечивой формулы F вида $\forall t F(t)$ существует в общем случае частичный неинициальный циклический автомат A , для которого множество всех допустимых сверхслов совпадает с множеством сверхслов, задаваемых формулой F .

Будем говорить, что автомат A удовлетворяет спецификации F , если $W(A) \subseteq W(F)$. Класс автоматов, специфицируемых в языке L , совпадает с автоматами с конечной памятью [6].

При определении множества сверхслов, задаваемого формулой $\forall t F(t)$, а следовательно, и множества сверхслов, допустимых для специфицируемого ею автомата, удобно использовать понятие пространства состояний для этой формулы [5]. Пусть Ω — сигнатура формулы $F(t)$, а r — её глубина. Обозначим $\Sigma(\Omega)$ множество всех двоичных векторов длины $|\Omega|$, где $|\Omega|$ — мощность множества Ω . Последовательность $s_0 s_1 \dots s_r$ векторов из $\Sigma(\Omega)$ назовем состоянием глубины r , а множество $Q(r, \Omega)$ всех таких последовательностей — пространством состояний глубины r для формулы $F(t)$. На множестве $Q(r, \Omega)$ определим отношение N непосредственного следования так, что за каждым состоянием $q = s_0 s_1 \dots s_r$ непосредственно следуют $2^{|\Omega|}$ состояний вида $s_1 \dots s_r s$, где $s \in \Sigma(\Omega)$. Множество всех состояний, непосредственно следующих за q , будем обозначать $N(q)$. Если компоненты вектора s_i в состоянии $q = s_0 s_1 \dots s_r$ рассматривать как истинностные значения соответствующих атомов ранга $i - r$ при некотором упорядочении множества Ω , то можно говорить о значении формулы $F(t)$ на состоянии q .

Формулу $F(t)$ будем рассматривать как представление множества $Q(F(t))$ состояний из $Q(r, \Omega)$, а именно тех состояний, на которых она истинна. Пусть $G(F)$ — граф ограничения отношения N на множество $Q(F(t))$. Граф $G = \langle V, E \rangle$, где V — множество вершин, а E — множество дуг графа, будем называть циклическим, если для каждой его вершины q существуют такие вершины q_1 и q_2 , что дуги (q_1, q) и (q, q_2) принадлежат E .

Несложно убедиться в справедливости следующей леммы, которую приведем без доказательства.

Лемма 1. Пусть $G^*(F)$ — максимальный циклический подграф графа $G(F)$. Тогда $\forall t F^*(t) \Leftrightarrow \forall t F(t)$, где $F^*(t)$ — формула, задающая множество вершин графа $G^*(F)$.

Если сверхслово l принадлежит $W(F)$, то ему соответствует бесконечный маршрут в графе $G^*(F)$, и наоборот, каждому бесконечному маршруту в $G^*(F)$ соответствует сверхслово, принадлежащее $W(F)$.

3. Синхронная композиция автоматов

Пусть $A_1 = (\Sigma, Q_1, \delta_1)$ и $A_2 = (\Sigma, Q_2, \delta_2)$ — циклические Σ -автоматы с одним и тем же входным алфавитом Σ . *Синхронной композицией* автоматов A_1 и A_2 (обозначается $A_1 \bullet A_2$) назовем максимальный циклический подавтомат автомата $C = \langle \Sigma, Q_C, \delta_C \rangle$, определяемого следующим образом: $Q_C = Q_1 \times Q_2$; значение $\delta_C(\langle q_1, q_2 \rangle, \sigma)$, где $q_1 \in Q_1$, $q_2 \in Q_2$, $\sigma \in \Sigma$, определено тогда и только тогда, когда значения $\delta_1(q_1, \sigma)$ и $\delta_2(q_2, \sigma)$ определены, и равно $\langle \delta_1(q_1, \sigma), \delta_2(q_2, \sigma) \rangle$. Пусть F_A и F_B — спецификации в языке L соответственно автоматов A и B , тогда формула $F_A \& F_B$ специфицирует автомат $A \bullet B$. Заметим, что Σ -автоматы $A = (\Sigma_A, Q_A, \delta_A)$ и $B = (\Sigma_B, Q_B, \delta_B)$, где $\Sigma_A = \Sigma(\Omega_A)$, а $\Sigma_B = \Sigma(\Omega_B)$, можно рассматривать как Σ -автоматы над одним и тем же алфавитом $\Sigma = \Sigma(\Omega_A \cup \Omega_B)$. Это позволяет приведенное выше определение синхронной композиции автоматов естественным образом распространить на Σ -автоматы с различающимися алфавитами.

Вектор $\sigma \in \Sigma(\Omega)$ будем рассматривать как отображение $\sigma : \Sigma \rightarrow \{0, 1\}$. Проекцией $\sigma \in \Sigma(\Omega)$ на $\Omega_1 \subseteq \Omega$ называется ограничение отображения σ на множество Ω_1 .

Определение 4. Ограничением автомата $A = \langle \Sigma(\Omega), Q, \delta \rangle$ на множество переменных (предикатных символов) $\Omega_1 \subseteq \Omega$ будем называть автомат $A_1 = \langle \Sigma_1, Q, \delta_1 \rangle$, где $\Sigma_1 = \Sigma(\Omega_1)$ и $q_1 \in \delta_1(q, \sigma_1)$ тогда и только тогда, когда существует такое $\sigma \in \Sigma(\Omega)$, что σ_1 есть проекция σ на Ω_1 и $\delta(q, \sigma) = q_1$.

Рассмотрим структуру, представленную на рис. 1. Здесь I_1, I_2, U и т. д. — множества двоичных переменных, причем пересечения $I_1 \cap I_2, U \cap O_1, V \cap O_2$ могут быть непустыми.

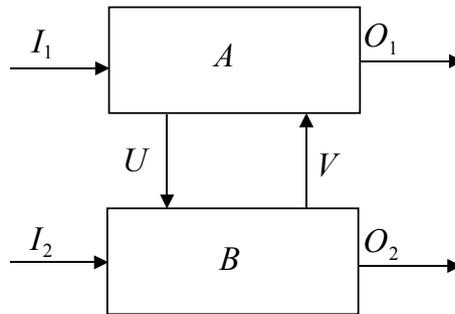


Рис. 1. Синхронная композиция автоматов

Пусть $A = \langle \Sigma(\Omega_A), Q_A, \delta_A \rangle$, где $\Omega_A = I_1 \cup O_1 \cup V \cup U$, а $B = \langle \Sigma(\Omega_B), Q_B, \delta_B \rangle$, где $\Omega_B = I_2 \cup O_2 \cup V \cup U$. Для такой структуры *внешней синхронной композиции* автоматов A и B (обозначается $A \circ B$) будем называть ограничение Σ -автомата $A \bullet B$ на множество переменных $\Omega_C = I_1 \cup I_2 \cup O_1 \cup O_2$.

4. Уравнения над автоматами

На множестве Σ -автоматов с одним и тем же входным алфавитом определим отношение \leq следующим образом: $A \leq B$ тогда и только тогда, когда $W(A) \subseteq W(B)$.

Пусть в приведенной выше структуре заданы Σ -автоматы $A = \langle \Sigma(\Omega_A), Q_A, \delta_A \rangle$, где $\Omega_A = I_1 \cup O_1 \cup V \cup U$, и $C = \langle \Sigma(\Omega_C), Q_C, \delta_C \rangle$, где $\Omega_C = I_1 \cup I_2 \cup O_1 \cup O_2$. Рассмотрим неравенство $A \circ X \leq C$. Здесь X — неизвестное, принимающее значения из множества циклических Σ -автоматов с входным алфавитом $\Sigma = \Sigma(I_2 \cup U \cup O_2 \cup V)$. Таким образом, Σ -автомат B есть решение рассматриваемого неравенства, если $W(A \circ B) \subseteq W(C)$. Задача состоит в нахождении максимального решения неравенства $A \circ X \leq C$ или решения уравнения $A \circ X = C$. Поскольку задача рассматривается на уровне спецификаций автоматов в языке L , решение ищется в классе автоматов с конечной памятью. В этом случае максимальным решением будем называть такое решение B , что не существует никакого другого неэквивалентного ему автомата B_1 , также являющегося решением и такого, что $B \leq B_1$. Чтобы сформулировать эту задачу на уровне спецификаций, определим понятие *минимальной формы спецификации* $F = \forall t F(t)$.

Пусть $F = \forall t F(t)$ — непротиворечивая формула глубины r с сигнатурой $\Omega = \{p_1, \dots, p_q\}$. Формулу $F(t)$ будем рассматривать как пропозициональную формулу с пропозициональными переменными $p_1(t), \dots, p_q(t), p_1(t-1), \dots, p_q(t-1), \dots, p_1(t-r), \dots, p_q(t-r)$.

Пусть $F_1(t)$ и $F_2(t)$ — такие, возможно логически неэквивалентные, формулы, что $\forall t F_1(t)$ и $\forall t F_2(t)$ задают одно и то же множество сверхслов. Тогда формула $\forall t F_1(t) \& F_2(t)$ задает это же множество. Будем рассматривать некоторое каноническое представление формулы $F(t)$, скажем, совершенную дизъюнктивную нормальную форму. Пусть $S^r(F) = \{F_i = \forall t F_i(t) : i = 1, \dots, N\}$ — множество всех спецификаций глубины r , представленных в канонической форме и задающих то же множество сверхслов, что и $F = \forall t F(t)$. Спецификацию $\min^r(F) = F_1 \& \dots \& F_N$ назовем *канонической минимальной формой* порядка r спецификации F . Из определения канонической минимальной формы спецификации следует её единственность.

Пусть $\forall t F_{\min}(t)$ — каноническая минимальная форма спецификации порядка r . Тогда всякую спецификацию $\forall t F(t)$ глубины r , где $F(t) \Leftrightarrow F_{\min}(t)$, назовем *минимальной формой* порядка r этой спецификации. Если $\forall t F_{\min}(t)$ — минимальная форма спецификации $\forall t F(t)$, то $F_{\min}(t) \rightarrow F(t)$. Построение минимальной формы формулы $\forall t F(t)$ состоит в преобразовании формулы $F(t)$, описанном в [7]. Результат такого преобразования будем обозначать $\min^r(F(t))$. Имеется тесная связь между минимальной формой представления формулы $F = \forall t F(t)$ и графом $G(F)$, ассоциируемым с формулой $F(t)$, представленной в пространстве состояний глубины r , а именно $\min^r(F(t))$ задает множество состояний, соответствующее вершинам циклического графа $G^*(F)$.

Теперь определим отношение на спецификациях, соответствующее отношению $A \leq B$ на циклических Σ -автоматах. Пусть $F_A = \forall t F_A(t)$ и $F_B = \forall t F_B(t)$ — формулы, специфицирующие соответственно автоматы A и B . Несложно показать, что $A \leq B$ тогда и только тогда, когда $\min(F_A(t)) \rightarrow F_B(t)$, или, что то же самое, $\min(F_A(t)) \rightarrow \min(F_B(t))$. Если глубина r формулы $F_B(t)$ превышает глубину формулы $F_A(t)$, то для $F_A(t)$ следует рассматривать минимальную форму порядка r .

Пусть $F_A = \forall t F_A(t)$ и $F_C = \forall t F_C(t)$ — спецификации соответственно автоматов $A = \langle \Sigma(\Omega_A), Q_A, \delta_A \rangle$ и $C = \langle \Sigma(\Omega), Q, \delta \rangle$, где $\Omega \subseteq \Omega_A$. Ограничение автомата A на Ω обозначим $[A]_\Omega$. Можно показать, что условию $[A]_\Omega \leq C$ соответствует условие $\min^r(F_A(t)) \rightarrow F_C(t)$, где r — наибольшая из глубин формул $F_A(t)$ и $F_C(t)$.

Теперь рассматриваемую задачу можно переформулировать следующим образом: найти максимальное решение $F_X(t)$ сигнатуры $\Omega_B = I_2 \cup U \cup O_2 \cup V$, удовлетворяющее формуле $\min(F_A(t) \& F_X(t)) \rightarrow F_C(t)$. Здесь, как и раньше, максимальным решением называется такое решение $F_B(t)$, что не существует никакой другой неэквивалентной ему формулы $F(t)$, также являющейся решением, и такой, что $F_B(t) \rightarrow F(t)$. Искомую формулу $F_B(t)$ будем строить следующим образом. Сначала получим максимальное решение $F'_B(t)$ с сигнатурой $\Omega = \Omega_A \cup \Omega_B$, а затем построим максимальную формулу $F_B(t)$ сигнатуры Ω_B , имплицитующую $F'_B(t)$. Решение этой задачи осложняется неоднозначностью представления автомата спецификацией в языке L .

Всякая формула $F(t)$ глубины r задает множество состояний в соответствующем пространстве состояний. *Максимальной формой* формулы $F(t)$ назовем такую формулу, что добавление любого состояния пространства состояний к задаваемому ею множеству состояний приводит к спецификации, не эквивалентной формуле $\forall t F(t)$. Существует одно (с точностью до эквивалентности) минимальное представление спецификации определенной глубины и много различных максимальных представлений. Множество всех максимальных представлений для $F(t)$ обозначим $\text{MAX}(F(t))$.

Теперь покажем, что формула $F'_B(t) = \neg(\min(F_A(t))) \vee \max(\min(F_A(t) \& F_C(t)))$, где $\max(F(t))$ — любое максимальное представление соответствующей формулы, есть решение, т. е. что формула $\min(F_A(t) \& F'_B(t)) \rightarrow F_C(t)$ тождественно истинна. Предварительно приведем некоторые используемые при этом соотношения:

$$\min(\min(F(t))) = \min(F(t)); \quad (1)$$

$$\min(\max(F(t))) = \min(F(t)); \quad (2)$$

$$\min(\min(F_1(t)) \& F_2(t)) = \min(F_1(t) \& F_2(t)); \quad (3)$$

$$\min(F_1(t) \& F_2(t)) \rightarrow (\min(F_1(t)) \& \min(F_2(t))). \quad (4)$$

Несложно показать, что формула $\min(F_A(t) \& F'_B(t)) \rightarrow F_C(t)$ равносильна формуле $\min(F_A(t) \& F'_B(t)) \rightarrow \min(F_A(t) \& F_C(t))$. Поэтому покажем, что приведенное выше значение для $F'_B(t)$ удовлетворяет последней. Согласно (3),

$$\min(F_A(t) \& F'_B(t)) = \min(\min(F_A(t)) \& F'_B(t)).$$

Подставив формулу $F'_B(t)$ в правую часть этой эквивалентности, получим

$$\min(\min(F_A(t)) \& \max(\min(F_A(t) \& F_C(t)))).$$

В силу (4) имеем

$$\begin{aligned} \min(\min(F_A(t)) \& \max(\min(F_A(t) \& F_C(t)))) &\rightarrow \\ &\rightarrow \min(\min(F_A(t)) \& \min(\max(\min(F_A(t) \& F_C(t)))). \end{aligned}$$

В силу (1) и (2) правая часть этой импликации равна $\min(F_A(t)) \& \min(F_A(t) \& F_C(t))$. Очевидно, что $\min(F_A(t)) \& \min(F_A(t) \& F_C(t)) \rightarrow \min(F_A(t) \& F_C(t))$. Таким образом, $\min(F_A(t) \& F'_B(t)) \rightarrow \min(F_A(t) \& F_C(t))$, что и требовалось показать.

Теперь покажем, что для любого решения $F_B(t)$ сигнатуры Ω существует такая максимальная форма формулы $\min(F_A(t)\&F_C(t))$, что

$$F_B(t) \rightarrow (\neg(\min(F_A(t))) \vee \max(\min(F_A(t)\&F_C(t)))) ,$$

т. е. максимальное решение содержится среди $\neg(\min(F_A(t))) \vee \text{MAX}(\min(F_A(t)\&F_C(t)))$. Пусть $F_B(t)$ — произвольное решение. Представим его в виде

$$F_B(t) = \min(F_A(t)\&F_B(t) \vee \neg(\min(F_A(t)))\&F_B(t).$$

Достаточно показать, что $\min(F_A(t)\&F_B(t) \rightarrow \max(\min(F_A(t)\&F_C(t)))$. Будем рассматривать только такие решения $F_B(t)$, для которых

$$\min(F_A(t)\&F_B(t)) = \min(F_A(t)\&F_C(t)),$$

т. е. $\max(\min(F_A(t)\&F_C(t))) = \max(\min(F_A(t)\&F_B(t)))$ для любой максимальной формы. Очевидно, что такие значения для $F_B(t)$ существуют — достаточно в качестве $F_B(t)$ взять формулу $F_C(t)$, рассматриваемую как формулу над $\Omega = \Omega_A \cup \Omega_B$. Всякое такое значение $F_B(t)$ дает максимальное значение для $\min(F_A(t)\&F_B(t))$, удовлетворяющее соответствующей импликации. Неформально это соответствует реализации максимальной части автомата C . Ясно, что всякое максимальное значение $F'_B(t)$ находится среди таких значений.

Следует отметить, что для любой формулы $F(t)$ существует такая максимальная форма формулы $\min(F(t))$, что $F(t) \rightarrow \max(\min(F(t)))$. Таким образом, для $\min(F_A(t)\&F'_B(t))$ существует такая максимальная форма $\max(\min(\min(F_A(t)\&F'_B(t))))$, что $\min(F_A(t)\&F'_B(t) \rightarrow \max(\min(\min(F_A(t)\&F'_B(t)))) = \max(\min(F_A(t)\&F'_B(t)))$, а следовательно, $\min(F_A(t)\&F'_B(t) \rightarrow \max(\min(F_A(t)\&F_C(t)))$. Другими словами, существует такая максимальная форма формулы $\min(F_A(t)\&F_C(t))$, что формула $\neg(\min(F_A(t))) \vee \max(\min(F_A(t)\&F_C(t)))$ будет максимальным решением. Вычисление всех максимальных форм формулы весьма сложно, поэтому будем строить решение, которое не требует вычисления максимальной формы и не сильно отличается от максимального решения. В качестве такого решения можно взять формулу $\neg(\min(F_A(t))) \vee F_C(t)$.

Теперь для получения решения $F_B(t)$ сигнатуры Ω_B необходимо взять \forall -проекцию формулы $F'_B(t)$ на её переменные, определяемые сигнатурой Ω_B .

Пусть $F(x_1, \dots, x_m, y_1, \dots, y_n)$ — пропозициональная формула от переменных $x_1, \dots, x_m, y_1, \dots, y_n$. \forall -проекцией формулы $F(x_1, \dots, x_m, y_1, \dots, y_n)$ на множество переменных $\{x_1, \dots, x_m\}$ называется формула $\forall y_1 \dots \forall y_n F(x_1, \dots, x_m, y_1, \dots, y_n) = F(x_1, \dots, x_m, 0, 0, \dots, 0)\&F(x_1, \dots, x_m, 0, \dots, 0, 1)\&\dots\&F(x_1, \dots, x_m, 1, \dots, 1, 1)$.

Если $\Omega_B = \{p_1, \dots, p_k\}$, то для формулы $F'_B(t)$ глубины r строится её проекция на множество переменных $\{p_1(t), \dots, p_k(t), p_1(t-1), \dots, p_k(t-1), \dots, p_1(t-r), \dots, p_k(t-r)\}$. Для вычисления проекции формулы на подмножество аргументов её удобно представлять в так называемой *нормальной форме* [8]. Нормальная форма имеет вид

$\bigvee_{i=1}^n F_i(t-1)\&f_i(t)$, где $f_i(t)$ — формула, построенная из атомов нулевого ранга, а $F_i(t-1)$ — из атомов, ранг которых не превышает -1 . Кроме того, $F_i(t-1)\&F_j(t-1) \equiv 0$ для всех $i, j = 1, \dots, n, i \neq j$ (условие ортогональности). Нормальная форма вида $\bigvee_{i=1}^n F_i(t-1)\&f_i(t)$ называется *полной*, если $\bigvee_{i=1}^n F_i(t-1) \equiv 1$. Отрицание формулы $F(t)$,

представленной в полной нормальной форме, имеет вид $\bigvee_{i=1}^n F_i(t-1) \& \neg f_i(t)$, что также является нормальной формой. Нормальная форма формулы $\neg(\min(F_A(t))) \vee F_C(t)$ строится из нормальных форм формул $\neg(\min(F_A(t)))$ и $F_C(t)$ при помощи операции дизъюнктивного произведения [8].

Утверждение 1. Чтобы получить ДНФ формулы $\forall y_1 \dots \forall y_n F(x_1, \dots, x_m, y_1, \dots, y_n)$, необходимо в ДНФ формулы $F(x_1, \dots, x_m, y_1, \dots, y_n)$ последовательно осуществить склеивания по переменным y_1, \dots, y_n и после всех возможных склеиваний по y_i удалить все конъюнкции, содержащие y_i или $\neg y_i$.

Основная идея использования нормальной формы для построения \forall -проекции формулы состоит в том, чтобы задачу большой размерности свести к ряду задач существенно меньшей размерности.

Пусть $\{x_1, \dots, x_m, y_1, \dots, y_n\}$ — сигнатура формулы $F(t)$, и необходимо построить проекцию этой формулы (рассматриваемой как пропозициональная формула) на множество переменных, соответствующих символам x_1, \dots, x_m . Обозначим множество этих переменных $\Omega(t)$. Чтобы упростить рассмотрение, будем считать, что $F(t)$ имеет глубину 1. Таким образом, $\Omega(t) = \{x_1(t-1), \dots, x_m(t-1), x_1(t), \dots, x_m(t)\}$. Проекцию на $\Omega(t)$ будем строить в два этапа: сначала построим проекцию на множество $\Omega'(t) = \{x_1(t-1), \dots, x_m(t-1), y_1(t-1), \dots, y_n(t-1), x_1(t), \dots, x_m(t)\}$, а затем — проекцию результата на $\Omega(t)$. В нормальной форме формулы $F(t)$ вида $\bigvee_{i=1}^n F_i(t-1) \& f_i(t)$ только формулы $f_i(t)$ ($i = 1, \dots, n$) зависят от переменных ранга 0. Поскольку $F_i(t-1) \& F_j(t-1) \equiv 0$ для любых $i \neq j$, то для получения проекции на множество переменных $\Omega'(t)$ склеивания по переменным $y_1(t), \dots, y_n(t)$ можно осуществлять независимо в каждой формуле $f_i(t)$ ($i = 1, \dots, n$). Таким образом, построение проекции формулы $F(t)$ на множество переменных $\Omega'(t)$ сводится к построению проекций n существенно более простых формул $f_1(t), \dots, f_n(t)$ на это множество переменных.

После получения проекции $F(t)$ на $\Omega'(t)$ следует построить проекцию этой формулы на множество переменных $\Omega(t)$. Для этого её нужно преобразовать в такую формулу вида $\bigvee_{i=1}^n F_i(t-1) \& f_i(t)$, чтобы для любых $i \neq j$ выполнялось $f_i(t) \& f_j(t) \equiv 0$. Поскольку функции $F_i(t-1)$ ($i = 1, \dots, n$) зависят только от переменных $x_1(t-1), \dots, x_m(t-1), y_1(t-1), \dots, y_n(t-1)$, построение \forall -проекции всей формулы сводится к построению \forall -проекции формул $F_i(t-1)$ в конъюнкциях $F_i(t-1) \& f_i(t)$ на множество $\{x_1(t-1), \dots, x_m(t-1)\}$.

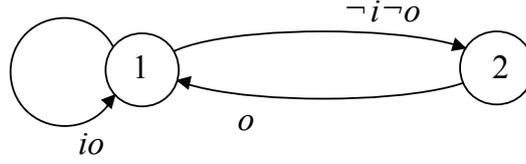
Пример. Пусть в структуре, изображенной на рис. 1, $I_1 = \{i\}$, $O_1 = \{o\}$, $U = \{u\}$, $V = \{v\}$ и $I_2 = O_2 = \emptyset$. Таким образом, $\Omega_C = \{i, o\}$, $\Omega_A = \{i, u, v, o\}$, $\Omega_B = \{u, v\}$.

Для упрощения записи спецификаций автоматов примем следующие соглашения: атом нулевого ранга вида $p(t)$ записывается как p , для атома ранга -1 вида $p(t-1)$ будем использовать запись $[p]$, знаки конъюнкции и квантора всеобщности в формулах опускаются. Обозначение $[p]$ распространим на формулы, построенные из атомов ранга -1 , например, формулу $p_1(t-1) \vee p_2(t-1) \& p_3(t-1)$ будем записывать как $[p_1 \vee p_2 p_3]$. В таких обозначениях спецификации автоматов A и C , представленные в нормальной форме, имеют вид

$$F_A(t) = [o]u(i \vee \neg i \neg o) \vee [\neg v \neg o](\neg v \neg u \vee v u \neg o) \vee [v \neg o] \neg u(v \vee \neg v \neg o),$$

$$F_C(t) = [o](i \vee \neg i \neg o) \vee [\neg o]o.$$

Автомат C , специфицируемый формулой $\forall t F_C(t)$, изображен на рис. 2.

Рис. 2. Автомат C

Спецификацию автомата B сигнатуры $\Omega = \Omega_A \cup \Omega_B$ будем строить в виде $F'_B(t) = \neg(\min(F_A(t))) \vee F_C(t)$.

Минимальная форма формулы $F_A(t)$ имеет вид

$$\min(F_A(t)) = [o(i \vee \neg u)]u(io \vee \neg i \neg o) \vee [\neg v \neg o(\neg i \vee \neg u)](\neg v \neg uo \vee vu \neg o) \vee [vu \neg o] \neg u(vo \vee \neg v \neg o).$$

Для получения полной нормальной формы следует добавить конъюнкцию $[v \neg u \neg o \vee \neg iuo \vee i \neg vu \neg o](0)$. Теперь отрицание формулы $\min(F_A(t))$ получается путём инвертирования всех подформулы вида $f_i(t)$:

$$\neg(\min(F_A(t))) = [o(i \vee \neg u)](\neg u \vee i \neg o \vee \neg io) \vee [\neg v \neg o(\neg i \vee \neg u)](v \neg u \vee uo \vee \neg v \neg o) \vee [vu \neg o](v \neg o \vee \neg vo \vee u) \vee [v \neg u \neg o \vee \neg iuo \vee i \neg vu \neg o](1).$$

Дизъюнктивное произведение формул $\neg(\min(F_A(t)))$ и $F_C(t)$ даёт

$$\begin{aligned} F'_B(t) &= [o(i \vee \neg u)](1) \vee [\neg v \neg o(\neg i \vee \neg u)](\neg u \vee o \vee \neg v) \vee [vu \neg o](u \vee o \vee v) \vee \\ &\quad \vee [v \neg u \neg o \vee \neg iuo \vee i \neg vu \neg o](1) = [o \vee v \neg u \vee i \neg vu](1) \vee \\ &\quad \vee [\neg i \neg v \neg o \vee \neg v \neg u \neg o](\neg u \vee o \vee \neg v) \vee [vu \neg o](u \vee o \vee v). \end{aligned}$$

Построим теперь \forall -проекцию формулы $F'_B(t)$ на $\{[v], [u], v, u\}$.

На первом этапе получим

$$[o \vee v \neg u \vee i \neg vu](1) \vee [\neg i \neg v \neg o \vee \neg v \neg u \neg o](\neg v \vee \neg u) \vee [vu \neg o](v \vee u).$$

Ортогонализация формул $f_i(t)$ даёт $[o \vee \neg v \vee \neg u](\neg v \neg u) \vee [1](\neg vu \vee v \neg u) \vee [o \vee v \vee iu](vu)$.

На втором этапе получим $F_B(t) = [\neg v \vee \neg u](\neg v \neg u) \vee [1](\neg vu \vee v \neg u) \vee [v](vu)$.

Покажем, что полученная формула специфицирует автомат, являющийся решением уравнения $A \circ X = C$. Для этого построим спецификацию композиции автоматов A и B , т. е. $F_A(t) \& F_B(t)$. Перемножив соответствующие формулы, получим

$$[\neg v \neg o](\neg v \neg uo) \vee [v \neg u \neg o](\neg v \neg u \neg o) \vee [o] \neg vu(io \vee \neg i \neg o) \vee [v \neg o](v \neg uo) \vee [vo]vu(io \vee \neg i \neg o). \quad (5)$$

Эта формула специфицирует автомат, изображенный на рис. 3.

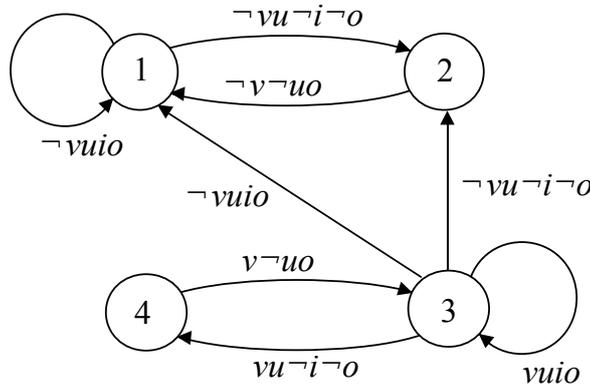


Рис. 3. Автомат, специфицируемый формулой (5)

Нетрудно видеть, что ограничение этого автомата на множество переменных $\{i, o\}$ эквивалентно автомату C .

Заключение

Рассмотрена задача решения неравенств над Σ -автоматами, возникающая при композиционном подходе к проектированию реактивных систем. Такие неравенства характеризуются операцией композиции автоматов и бинарным отношением \leq , определенном на множестве автоматов. Задача формулируется и решается на уровне спецификаций автоматов в логическом языке L , для чего соответствующие понятия определяются для спецификаций. При решении неравенств такого рода обычно интерес представляют наибольшие в смысле указанного отношения решения, однако в нашем случае отношение \leq является частичным порядком, для которого наибольшего решения может не существовать. В связи с этим рассматривается задача отыскания максимального решения. Выбор для спецификации достаточно простого языка позволил свести решение этой задачи к преобразованию пропозициональных формул.

В приложениях, ориентированных на схемную реализацию проектируемой системы, используются другие виды композиции, обеспечивающие невозможность образования порочного цикла при изменении входных и выходных сигналов взаимодействующих модулей. Как показано в [9], многие из таких видов композиции могут быть сведены к рассмотренной в настоящей статье синхронной композиции циклических Σ -автоматов путём простого преобразования одной или обеих спецификаций, участвующих в композиции.

Основная идея предложенного подхода состоит в том, чтобы сначала построить решение для синхронной композиции, т. е. в виде формулы, сигнатура которой состоит из всех предикатных символов, встречающихся в спецификациях, а затем получить решение для внешней композиции, взяв \forall -проекцию полученной формулы на соответствующее множество переменных. Предложен подход к построению такой проекции путём сведения задачи большой размерности к ряду задач существенно меньшей размерности.

Следует заметить, что задача решается для неинициальных спецификаций, хотя на практике, как правило, рассматриваются инициальные системы. Обычно инициализация спецификаций в языке L осуществляется путём задания начального условия в виде формулы $F(t)$ этого же языка, используемой для выделения начального состояния после перехода к процедурному представлению автомата. Нормальная форма представления спецификаций дает возможность сразу учитывать их инициальность, что в ряде случаев сокращает объём вычислений, необходимых для решения задачи.

Автор благодарен Н. В. Евтушенко за полезное обсуждение рассматриваемых в статье вопросов.

ЛИТЕРАТУРА

1. *Yevtushenko N., Villa T., Brayton R., Petrenko A., et. al.* Solution of synchronous language equations for logic synthesis // Вестник Томского госуниверситета. 2002. № 1. С. 132–138.
2. *Buffalov S., El-Fakih K., Yevtushenko N., Bochmann G.* Progressive solutions to a parallel automata equation // LNCS. 2003. V. 2767. P. 367–383.
3. *Yevtushenko N., Zharikova S., Vetrova M.* Multi component digital circuit optimization by solving FSM equations // Euromicro Symposium on Digital System Design, IEEE Computer society, 2003. P. 62–68.

4. Harel D., Pnueli A. On the development of reactive systems // NATO ASI Series. Logic and Models of Concurrent Systems. Berlin: Springer, 1985. F13. P. 477–498.
5. Чеботарев А. Н. Об одном подходе к функциональной спецификации автоматных систем. I // Кибернетика и системный анализ. 1993. № 3. С. 31–42.
6. Брауэр В. Введение в теорию конечных автоматов. М.: Радио и связь, 1987. 392 с.
7. Чеботарев А. Н., Куривчак О. И. Аппроксимация множеств сверхслов формулами языка L // Кибернетика и системный анализ. 2007. № 6. С. 18–26.
8. Капитонова Ю. В., Чеботарев А. Н. Индуктивный синтез автомата по спецификации в логическом языке L // Там же. 2000. № 6. С. 3–13.
9. Чеботарев А. Н. Взаимодействие автоматов // Там же. 1991. № 6. С. 17–29.

ДИСКРЕТНЫЕ МОДЕЛИ РЕАЛЬНЫХ ПРОЦЕССОВ

DOI 10.17223/20710410/9/11

УДК 621.391.1:004.7

КУМУЛЯТИВНЫЙ СИНТЕЗ: КЛЕТОЧНО-АВТОМАТНАЯ МОДЕЛЬ ПРОЦЕССА ОБРАЗОВАНИЯ ПОКРЫТИЯ, НАНОСИМОГО НА МИШЕНЬ С ПОМОЩЬЮ КУМУЛЯТИВНОГО ПОТОКА ЧАСТИЦ¹

О. Л. Бандман*, С. А. Громилов**, С. А. Кинеловский***

Институт вычислительной математики и математической геофизики СО РАН,**Институт неорганической химии им. А. В. Николаева СО РАН,*****Институт гидродинамики им. М. А. Лаврентьева СО РАН,**г. Новосибирск, Россия***E-mail:** bandman@ssd.sscs.ru, grom@niic.ru, skin@hydro.nsc.ru

Предложена дискретная математическая модель стохастического типа, предназначенная для компьютерного моделирования процесса образования покрытия на мишени при обработке её высокоскоростной порошковой струей. Модель относится к классу асинхронных вероятностных клеточных автоматов. Параметры модели определены на основе результатов натуральных экспериментов, а также исходя из известных сведений для аналогичных процессов. Результаты моделирования процесса образования карбида на вольфрамовой мишени при обработке её углеродистой порошковой струей показали, что применение модели может повысить эффективность исследований, которые проводятся в рамках создания методов кумулятивного синтеза новых материалов и структур в Институте гидродинамики СО РАН.

Ключевые слова: компьютерное моделирование, асинхронный вероятностный клеточный автомат, кумулятивная струя, твердое покрытие, карбид вольфрама.

Введение

К настоящему времени существующие методы и подходы к синтезу новых соединений, основанные на ударно-волновом нагружении и взрывном компактировании порошковых смесей, в значительной степени исчерпали свои возможности. Это связано прежде всего с ограничениями по диапазонам давлений, температур и массовых скоростей взаимодействующих частиц. Эти диапазоны удается существенно расширить за счет использования специальных кумулятивных зарядов. При изготовлении облицовки кумулятивного заряда из высокопористого материала, в частности из порошковых смесей, реализуемые при обжати пористых облицовок более высокие, чем при ударно-волновом нагружении, уровни давления и температуры открывают перспективы как для осуществления фазовых переходов в материале, так и для синтеза новых неравновесных структур, отличных от исходного материала [1]. При этом вместо компактной кумулятивной струи образуется разуплотненный поток частиц материала

¹Работа поддержана Программой фундаментальных исследований Президиума РАН №2-6 (2010) и Сибирским отделением РАН, Интеграционный проект 32 (2010).

облицовки, который может быть использован для нанесения покрытия на преграду-подложку (мишень). Кроме того, при взаимодействии кумулятивного потока частиц с поверхностью мишени могут образовываться новые фазы и соединения [1]. Совокупность перечисленных результатов дает основание называть данное технологическое направление кумулятивным синтезом. Из изложенного выше следует, что процесс кумулятивного синтеза имеет две стадии, на которых осуществляется собственно синтез: 1) в высокоскоростном потоке частиц (струе) и 2) на поверхности мишени. Данная работа относится к исследованию процессов, происходящих на второй стадии, а именно при образовании покрытия на мишени, обрабатываемой высокоскоростной порошковой струей.

Теоретическая часть исследований включает в себя анализ физико-химических процессов на базе уравнений состояния вещества, а также кинетики фазовых превращений на основе методов молекулярной динамики [2]. Экспериментальная часть состоит из серии специальных экспериментов по взаимодействию высокоскоростного потока частиц с металлической преградой и последующего проведения рентгенофазового анализа образцов полученных покрытий. Результаты таких экспериментов позволяют оценить зависимости структуры покрытия от различных физических параметров процесса. В проведенных экспериментах было осуществлено взаимодействие потока частиц, полученного из 30-градусной конической облицовки из порошка графита насыпной плотности, с металлической вольфрамовой пластиной.

Поскольку организация и проведение экспериментов требуют больших затрат времени и средств, повышение эффективности исследований путем имитационного компьютерного моделирования чрезвычайно важно. Имитационное моделирование позволяет проследить процесс образования покрытия во времени и в пространстве путем наблюдения его в замедленном режиме на мониторе компьютера. Методы компьютерного моделирования, основанные на дифференциальных уравнениях в частных производных, здесь оказываются неприменимыми из-за дискретного характера изменений состояний (мгновенных химических превращений и фазовых переходов). Методы молекулярной динамики полезны для определения количественных оценок процесса механохимического синтеза прекурсоров [2], но не позволяют наблюдать картины процесса на всем пространстве в целом. Наиболее подходящими являются стохастические методы, основанные на непосредственном отображении в компьютере движений и превращений абстрактных или реальных частиц [3, 4].

Современные компьютеры и суперкомпьютерные системы вполне пригодны для отображения физико-химических взаимодействий на микро- и даже на наноуровнях, при использовании дискретных вероятностных моделей, которые в теории моделирования называются кинетическими асинхронными клеточными автоматами (АКА) [3], а в задачах поверхностной химии — кинетическим методом Монте-Карло [4]. Модель является математическим представлением происходящих в системе событий (адсорбция частицы на поверхность, химические взаимодействия между частицами, фазовые переходы, диффузия по поверхности, проникание внутрь подложки и др.). Каждое событие имитируется изменением состояния покрытия с заданной вероятностью в случайно выбранной точке.

В работе предлагается применить АКА для моделирования процесса образования покрытия, наносимого на металлическую мишень с помощью кумулятивной струи, содержащей порошок из одного или нескольких веществ. В п. 1 дано описание АКА-модели и основанного на ней алгоритма моделирования, а также особенностей программной реализации. В п. 2 приведены результаты проведенного моделирования в усло-

виях, соответствующих указанным выше экспериментам по обработке вольфрамовой мишени высокоскоростной струей из углеродистого порошка. В заключении сделаны выводы о возможностях, целесообразности и перспективах развития методов АКА-моделирования.

1. Описание модели

Поскольку АКА оперирует малыми частицами (размеры — микроны), то, чтобы промоделировать процесс, происходящий на площади в 1 мм^2 , необходим АКА, состоящий из $10^8 - 10^9$ клеток. Даже при очень простых операциях АКА (несколько обращений к генератору псевдослучайных чисел и несколько булевых операций) обработка такого массива данных требует параллельной реализации на суперкомпьютере, состоящем из нескольких десятков или сотен современных процессоров. Это в настоящее время вполне возможно как с точки зрения вычислительной мощности, так и с точки зрения существования эффективного метода распараллеливания АКА [5]. В случае исследования процессов, в которых носителями энергии являются гранулы порошка (далее называемые *частицами*) со средним диаметром $\simeq 10$ мкм, для моделирования процессов на участке мишени размерами 1 см^2 необходим АКА размерами 1000×1000 клеток, что вполне реализуемо на современном персональном компьютере, а для получения картины реаспределения вещества по всей мишени размерами $8 \times 8 \text{ см}^2$ достаточно иметь 4-ядерный компьютер (например, с процессором типа Intel Core i7).

Задача моделирования процесса образования покрытия на мишени при её бомбардировке частицами порошка состоит в следующем. Мишень представляет собой металлическую пластину, на которую направлена кумулятивная струя, состоящая из порошковой смеси одного или нескольких веществ. При подлете к мишени частицы струи сталкиваются с ней, в результате чего происходят следующие события:

- 1) адсорбция вещества на поверхность металла;
- 2) химическая реакция между веществом поверхности и веществом адсорбированной на неё частицы;
- 3) проникание частицы в мишень на некоторую глубину;
- 4) перемещение вещества по поверхности покрытия на место с более низким уровнем энергии (диффузия);
- 5) образование нехимического соединения между веществом поверхности и адсорбированной на него частицы порошка;
- 6) десорбция (сублимация) вещества с поверхности.

Вероятностный клеточный автомат должен имитировать этот процесс таким образом, чтобы в результате компьютерной реализации модели можно было бы получить полную картину процесса, т. е.

- количество вещества каждого типа в покрытии;
- распределение частиц каждого типа по площади мишени и по слоям покрытия.

Для математического описания алгоритма функционирования АКА удобнее всего использовать формализм «Алгоритма параллельных подстановок» [6], который, в отличие от классической теории КА, позволяет явно представлять клеточный асинхронизм (стохастичность изменений как в пространстве моделирования, так и во времени), а также допускает групповые функции изменения состояний, присущие физико-химическим процессам.

Формально АКА определяется тремя понятиями: алфавитом состояний $A = \{a_0, a_1, \dots, a_n\}$, множеством имен клеток $C = \{c\}$ и множеством локальных операторов

$\Theta = \{\theta_1(\mathbf{c}), \dots, \theta_q(\mathbf{c})\}$. Клетка интерпретируется как пара символов (a_k, \mathbf{c}) , где $a_k \in A$ обозначает тип частицы, а $\mathbf{c} = (i, j, k)$ — вектор координат дискретного пространства, в котором $(i, j) \in M \times M$ — координаты клетки в плоскости мишени, $k = 0, \dots, H$ — номер слоя нанесенного покрытия. Номер верхнего (поверхностного) слоя обозначается через h , а $h(i, j)$ — высота нанесенного слоя в точке (i, j) на плоскости мишени. Таким образом, размер массива равен $|C| = H \times M^2$.

Изменение состояний любой клетки зависит от состояний клеток в её локальном окружении, которое задается шаблоном соседства

$$T(\mathbf{c}) = \{\varphi_0(\mathbf{c}), \varphi_1(\mathbf{c}), \dots, \varphi_q(\mathbf{c})\}, \quad \mathbf{c} = (i, j, k), \quad i, j = 0, \dots, M, \quad k = 0, \dots, h(i, j). \quad (1)$$

При этом принимается $\varphi_0(\mathbf{c}) = \mathbf{c}$. В кинетических АКА функции $\varphi_l(\mathbf{c})$ имеют следующий вид:

$$\varphi_l(\mathbf{c}) = \mathbf{c} + (\alpha, \beta, \gamma), \quad l = 0 \dots, q,$$

где $\alpha, \beta, \gamma \in \{-1, 0, 1\}$.

Множество локальных операторов $\Theta = \{\theta_1(\mathbf{c}), \dots, \theta_g(\mathbf{c}), \dots, \theta_n(\mathbf{c})\}$ соответствует множеству событий в моделируемом процессе. Локальный оператор $\theta_g(\mathbf{c}) \in \Theta$ меняет состояния в некоторых клетках из своего соседства $\{(v_{g_1}, \varphi_{g_1}), \dots, (v_{g_r}, \varphi_{g_r}(\mathbf{c}))\}$, $\varphi_{g_l} \in T(\mathbf{c})$, $g_l \in \{0, \dots, r\}$, $r \leq q$, на новые состояния v'_{g_l} , равные значениям функций перехода от состояний всех клеток соседства $T(\mathbf{c})$, т. е.

$$\theta_g(\mathbf{c}) : \{(v_{g_1}, \varphi_{g_1}(\mathbf{c})), \dots, (v_{g_r}, \varphi_{g_r}(\mathbf{c}))\} \xrightarrow{p_g} \{(v'_{g_1}, \varphi_{g_1}(\mathbf{c})), \dots, (v'_{g_r}, \varphi_{g_r}(\mathbf{c}))\}, \quad (2)$$

где p_g — вероятность выполнения $\theta_g(\mathbf{c})$, $v'_{g_l} = f_{g_l}(v_0, v_1, \dots, v_q)$. Так, например, оператор адсорбции меняет состояние поверхностной клетки $\mathbf{c} = (i, j, h(i, j))$ с $v = \emptyset$ на $v' = \mathbf{C}$. Оператор реакции меняет состояние клетки на результат реакции, если в двух смежных клетках оказываются реагенты, при этом одна из клеток опустошается. Оператор диффузии перемещает частицу в ближайшую клетку, если это перемещение уменьшает энергию системы. Выполнение каждого оператора ограничено условиями, которые вычисляются на основе следующих физических соображений и допущений.

1) Все гранулы порошка имеют одинаковый диаметр D_p . В каждой клетке АКА может находиться не более одной гранулы. Физический размер клетки в пространственной решетке равен $D_p \times D_p$. В клеточном автомате все клетки имеют размеры $1 \times 1 \times 1$ при плотном прилегании клеток друг к другу.

2) В струе, падающей на мишень, функции распределения скорости $u(i, j, t)$ и плотности $d(i, j, t)$ частиц по пространству и времени выражаются либо в виде таблиц, полученных путем компьютерного моделирования струи, либо в виде аппроксимирующих их функций, которые могут быть представлены следующим образом:

$$z(r, t) = z_{\max} t^a \exp(-(br^2 + ct)), \quad (3)$$

где $z(r, t) = u(r, t)$ или $z(r, t) = d(r, t)$; $r = \sqrt{(M-i)^2 + (M-j)^2}$ — расстояние от центра мишени; a, b, c — константы, различные для $u(r, t)$ и $d(r, t)$.

3) Кинетическая энергия удара частицы о мишень равна

$$E_k(i, j, t) = \frac{mu^2(i, j, t)}{2}. \quad (4)$$

Эта энергия расходуется на химическую реакцию между веществом упавшей частицы и соприкасающейся с ней на мишени, на деформацию поверхности при проникании

упавшей частицы в глубь мишени, на выделение тепла и, возможно, на образование некоторого нехимического соединения.

4) Вероятность адсорбции частицы на поверхность равна плотности частиц в струе в момент её касания с мишенью

$$P_{\text{ads}}(i, j, t) = d(i, j, t). \quad (5)$$

5) Химические реакции происходят между частицами, если реактанты находятся в смежных клетках и выполнены условия

$$E_k(i, j, t) > E_{\text{reac}}(i, j, t), \quad T_{\text{reac-low}} < T(i, j, t) < T_{\text{reac-high}}, \quad (6)$$

где E_{reac} — энергия активации реакции; $T_{\text{reac-low}}$, $T_{\text{reac-high}}$ — границы температурного диапазона, при котором реакция возможна.

6) Перемещение частицы из одной клетки в другую (диффузия) происходит, если после этого перемещения энергия её связей с соседними частицами увеличивается. Поскольку энергию связей между частицами подсчитать практически невозможно, предлагается применить упрощенный вариант диффузии, в котором выполняется только «сглаживание» получаемой поверхности. Иными словами, перемещение частицы происходит с более высокого уровня относительно поверхности покрытия на более низкий, что качественно соответствует принципу уменьшения свободной энергии системы. При этом допустимая разница в уровнях соседних клеток должна быть выбрана исходя из свойств получаемого покрытия. Энергия при диффузии не расходуется.

7) Поскольку химические реакции эндотермические, то можно считать, что когда условия (6) для химической реакции не выполняются или вблизи падающей частицы нет соответствующего реактанта, тогда часть кинетической энергии расходуется на деформацию и нагрев материала:

$$E_k(i, j, t) - E_{\text{reac}}(i, j, t) = E_{\text{def}}(i, j, t) + E_{\text{heat}}(i, j, t). \quad (7)$$

Если не учитывать отвода тепла в течение процесса, то можно считать, что нагревается один слой мишени (толщиной D_p) на участке непосредственно под упавшей на него частицей. При таком допущении повышение температуры можно подсчитать исходя из теплоёмкости материала мишени и массы нагреваемого материала.

8) Энергия деформации поверхности расходуется на проникание в глубь мишени и на пластическую деформацию вблизи падения частицы, т. е.

$$E_{\text{def}}(i, j, t) = E_{\text{surf}} \cdot S_{\text{br}}(i, j, t) + E_{\text{duc}} \cdot V_{\text{duc}}(i, j, t), \quad (8)$$

где E_{surf} — поверхностная энергия изменяющего свою форму материала; S_{br} — изменение площади поверхности; E_{duc} — энергия пластической деформации; V_{duc} — объём пластически деформируемого материала. Величины энергий E_{surf} и E_{duc} характеризуют хрупкость и пластичность материала.

2. Реализация программы и результаты моделирования

Модель испытывалась на примере вольфрамовой мишени, обрабатываемой углесодержащей кумулятивной струей. Исходные данные для моделирования соответствуют экспериментам, которые были проведены в Институте неорганической химии и Институте гидродинамики СО РАН: размер мишени 80×80 мм²; максимальные скорость и плотность порошка, падающего на мишень, составляют $u_{\text{max}} = 2000$ мс⁻¹, $d_{\text{max}} = 0,4$;

средний размер гранул $d = 10$ мкм; струя содержит $\simeq 2,8 \cdot 10^9$ гранул порошка углерода с массой гранулы $m = 1,8 \cdot 10^{-9}$ г.

Программная реализация отлаживалась на фрагменте размерами 6×6 мм². При линейном размере модельной клетки $l = 10$ мкм размер фрагмента составляет 600×600 клеток в плоскости мишени при максимальной высоте наносимого покрытия $H(i, j) = 30$. Такие размеры АКА позволяют провести экспериментальную отладку программы на персональном компьютере типа Pentium IV при выводе на экран монитора моделируемой поверхности на каждой итерации. Более того, если предположить, что выбранный фрагмент находится в центре мишени, то значения скорости и плотности в струе можно считать постоянными и равными своим максимальным значениям. Именно эти значения можно сравнивать с данными дифрактограмм результатов натуральных испытаний. При этом предположении и с учетом известных экспериментальных данных определены следующие параметры модели.

1) Кинетическая энергия, приносимая падающей частицей: $E_k = 3,6 \cdot 10^{-6}$ Дж.

2) Длительность существования струи $t = 25$ мкс при расстоянии от облицовки до мишени $L = 50$ мм.

3) Вероятность адсорбции $P_{\text{ads}} = \rho_{\text{max}} = 0,4$ равна плотности струи при соприкосновении с мишенью.

4) Вероятности реакций могут быть рассчитаны исходя из соотношений значений энергий активации (энтальпий) реакций карбидов $E_a(\text{WC})$ и $E_a(\text{W}_2\text{C})$ при заданных условиях (температуре, давлении, количестве избыточного углерода). Однако поскольку, кроме предположительной температуры в центральной части струи $T_{\text{max}} = 1600\text{--}2000$ °С, других данных для локальных условий моделируемого процесса нет, то приходится ориентироваться на литературные данные [7, 8] и данные натуральных испытаний авторов, из которых можно заключить, что при заданной температуре и избытке углерода соотношение между количествами получаемых карбидов и остающимся вольфрамом равно $Q(\text{W})/Q(\text{WC})/Q(\text{W}_2\text{C}) = 2/3/4$, что позволяет принять следующие величины вероятностей образования карбидов: $P(\text{W}_2\text{C}) = 0,44$, $P(\text{WC}) = 0,33$.

5) Диффузия выполняется с вероятностью $P_{\text{diff}} = 1$ в тех клетках массива, где разница высоты покрытия в соседних клетках $\Delta h \geq 2$. Акт диффузии состоит в том, что частица с более высокого уровня перемещается на более низкий, причем если существует несколько возможных вариантов перемещений, то один из них выбирается равновероятно.

6) Из экспериментов известно, что проникание углерода в глубь вольфрамовой мишени происходит не более чем на несколько десятков микрометров. Таким образом, энергия деформации E_{def} рассчитывается по (8) при условии проникания поверхностной клетки на два слоя вниз и вытеснения аморфного вольфрама на поверхность. При $E_{\text{surf}}(\text{W}) = 1,7$ Дж·м⁻², $E_{\text{duc}}(\text{WC}) = 7$ МПа·Н/м^{3/2} и $S_{\text{br}} = 0,9 \cdot 10^{-9}$ м², $V_{\text{duc}} = 10^{-12}$ м³, рассчитанных по данным из [8] и $T = 1600$ °С, энергия составит $E_{\text{def}} = 2,2 \cdot 10^{-9}$ Дж. Остальная энергия расходуется на повышение температуры в месте соударения. Зная теплоемкость вольфрама c_{W} и принимая нагреваемую массу мишени m_{W} равной массе вещества в одной клетке, получаем повышение температуры $\Delta T = E_{\text{term}}/(c_{\text{W}} \cdot m_{\text{W}})$. Эти расчетные величины могут быть приняты в качестве ориентировочных, так как они зависят от свойств и получаемых карбидов, и мишени, которые значительно различаются от образца к образцу.

Функционирование АКА происходит в соответствии со следующим стохастическим итерационным алгоритмом.

В начальном состоянии ($t = 0$) все клетки массива $\Omega(0)$ пусты:

$$\forall(i, j, k) \in \mathbf{C} \quad (v(i, j, 0) = \emptyset, h(i, j) = 0).$$

Каждая t -я итерация состоит из трех частей:

1. Для всех $(i, j) \in \mathbf{C}$, выбираемых в случайном порядке, выполняется следующее:
 - с вероятностью $P_{\text{ads}} = \rho_{\text{max}} = 0,4$ применяется оператор адсорбции;
 - если условия (6) для реакций выполнены, то в соответствии с вероятностями $P(W2C)$ и $P(WC)$ применяется одна из химических реакций: $W+C \rightarrow WC$ или $W+W+C \rightarrow W2C$;
 - если условия для химических реакций не выполнены, то применяются операторы деформации и нагрева.
2. Для всех $(i, j) \in \mathbf{C}$, выбираемых в случайном порядке, выполняется операция диффузии.
3. Через несколько итераций покрытие оказывается сформированным, так как поверхность оказывается покрытой слоем углерода и химических реакций не происходит. Процесс заканчивается полностью, когда иссякает поток порошка.

Программная реализация предусматривает получение следующей информации о процессе.

- 1) Вывод в файл зависимости процентных соотношений количеств W , WC и $W2C$ в поверхностном слое и во всем покрытии от времени (на каждой итерации).
- 2) Вывод на экран монитора на каждой итерации состояния поверхностного слоя и разреза мишени, причем каждое состояние клетки (вещество) представлено своим цветом.

На рис. 1 приведены зависимости количеств полученных веществ в поверхностном слое от времени моделирования. Время выражено в количестве итераций, каждая итерация соответствует реальному промежутку времени $\Delta t = 10^{-8}$ с.

На рис. 2 показано распределение веществ по уровням (слоям) покрытия, отсчитываемым от $h(i, j) = 0$, причем номер слоя указывает на расстояние от нулевого уровня, т. е. уровня поверхности вольфрама до начала процесса. Из рис. 2 видно, что наибольший процент карбидов находится в слоях ниже нулевого, что объясняется тем, что получающийся в результате реакции карбид занимает место участвовавшего в реакции вольфрама.

Моделирование процесса на мишени размерами 80×80 мм² проводилось на компьютере с 4-ядерным процессором типа Intel® Core™ i7. Клеточное пространство размером 8000×8000 было разделено на четыре домена размерами 4000×4000 каждый. Распараллеливание асинхронного алгоритма клеточно-автоматной эволюции проводилось с применением блочно-синхронного преобразования АКА [7]. При этом учитывалось распределение скорости и плотности частиц в струе по пространству, которые рассчитывались в соответствии с формулой (3). Константы в (3) $a = 1,86$, $c = 0,64$, $b_u = 30$ и $b_d = 12$ были вычислены путем подстановки в (3) следующих граничных значений: $u(r = 0, t = 2) = u_{\text{max}}$, $d(r = 0, t = 2) = d_{\text{max}}$, $u(r = 0, t = 20) = 0$, $d(r = 0, t = 20) = 0$, $u(r = M/2) = d(r = M/2) = 0$ для всех $t = 0, \dots, 20$.

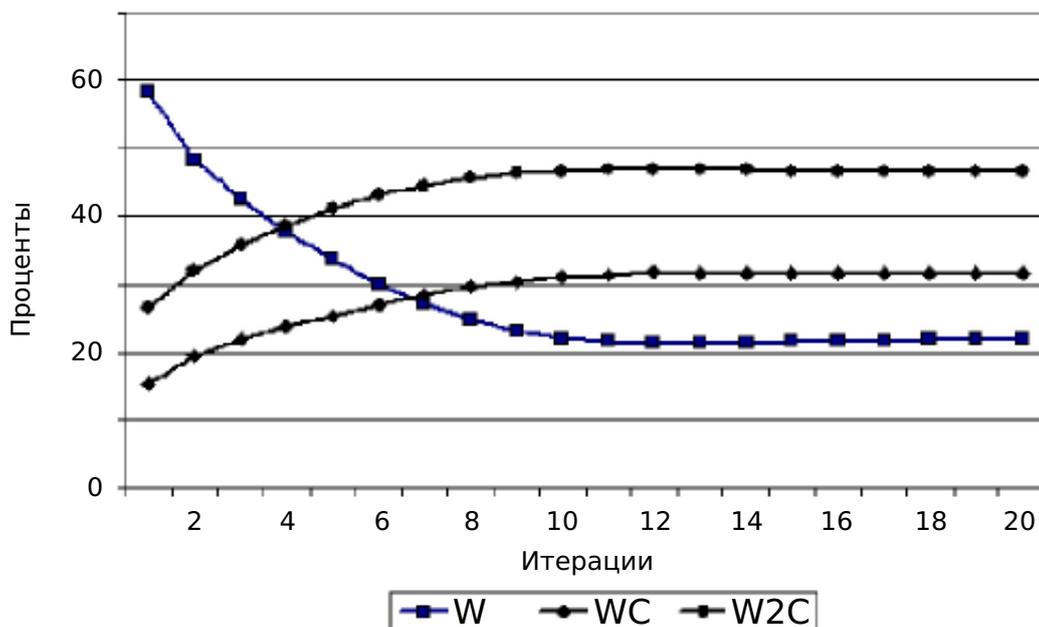


Рис. 1. Зависимость количества веществ в поверхностном слое от времени

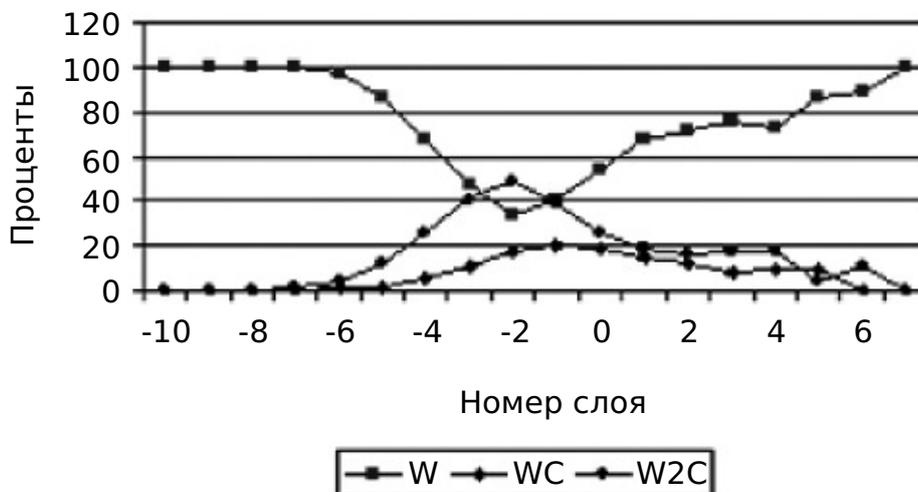


Рис. 2. Распределение веществ в покрытии по слоям

При моделировании применялся тот же алгоритм, что и для центрального фрагмента. Но для каждой точки с координатами (i, j) вычислялись значения $u(i, j)$ и $d(i, j)$ и, в зависимости от них, вероятности

$$P_{\text{ads}}(i, j) = d(i, j), \quad P_{\text{WC}}(i, j) = 0,4 \exp(1,4 - v^2(i, j)), \quad P_{\text{W2C}} = 0,6 \exp(-2,4 - v^2(i, j)),$$

где $v(i, j) = u(i, j) \cdot 10^{-3}$. Полученные в результате моделирования распределения скорости и плотности струи вдоль её оси показаны на рис. 3.

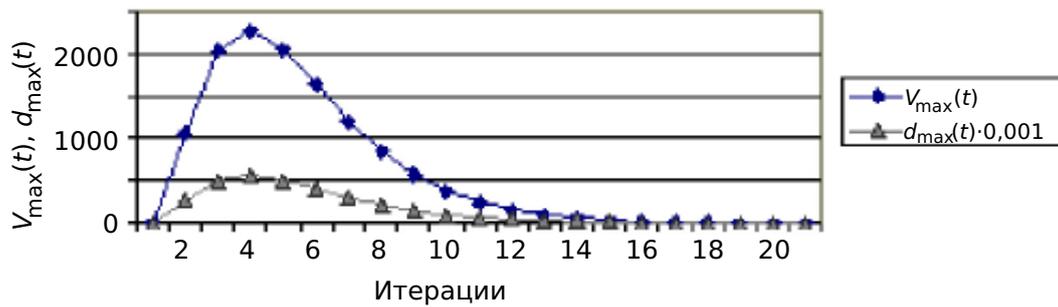


Рис. 3. Зависимость скорости и плотности в центре струи от времени ее касания с мишенью

На рис. 4 показаны полученные в результате моделирования соотношения плотности веществ в зависимости от расстояния от центра мишени. Время вычисления на компьютере Intel® Core™ i7 составляет 0,5 ч, что убеждает в возможности проводить в приемлемое время моделирование процессов с порошками размером 1 мкм и менее.

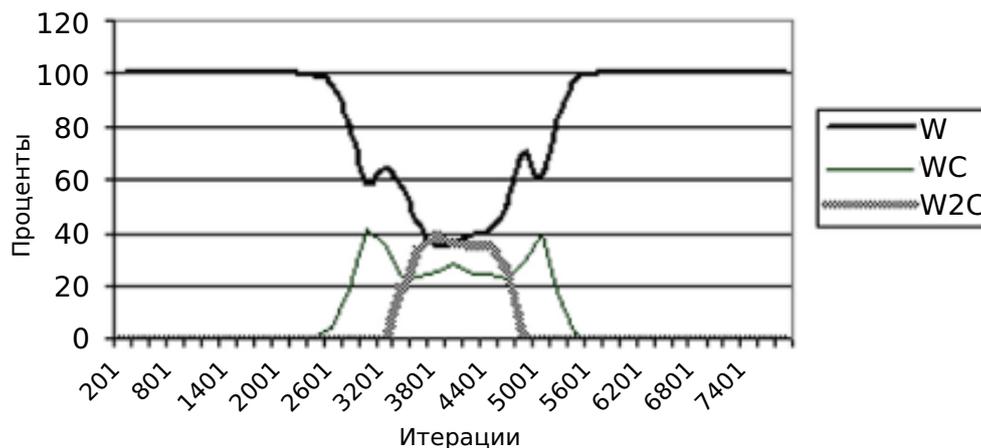


Рис. 4. Распределение веществ по пространству мишени после окончания процесса

Заключение

Представлена дискретная математическая модель стохастического типа, относящаяся к классу асинхронных клеточных автоматов, предназначенная для моделирования процесса образования покрытия на мишени при обработке её высокоскоростной порошковой струей. Основные параметры модели, а именно вероятности адсорбции и химических реакций, скорость, плотность и температура порошковой струи приняты в модели исходя из натуральных экспериментов и полученных на их основе дифрактограмм. Однако часть необходимых данных (поверхностная энергия, энергия пластической деформации), а также допущений (упрощенная диффузия, функция распределения скорости и плотности порошка в струе) были выбраны исходя из известных сведений для аналогичных процессов [7, 8]. На самом деле, для каждого набора порошков и материала мишени они должны быть уточнены путем проведения серии натуральных и соответствующих вычислительных экспериментов. Программные реализации,

разработанные для малого фрагмента мишени и для мишени в целом, показали, что она может стать полезным инструментом в исследованиях процессов кумулятивного синтеза.

ЛИТЕРАТУРА

1. Громилов С. А., Кинеловский С. А., Попов Ю. Н., Тришин Ю. А. О возможности физико-химических превращений веществ при кумулятивном нанесении покрытий // Физика горения и взрыва. 1997. Т. 33. №6. С. 127–130.
2. Псахье С. Г., Коростелев С. Ю., Смолин А. Ю. и др. Метод подвижных клеточных автоматов как инструмент физической мезомеханики материалов // Физическая мезомеханика. 1998. Т. 1. №1. С. 1–15.
3. Бандман О. Л. Клеточно-автоматные модели пространственной динамики // Системная информатика. Новосибирск: СО РАН, 2006. Вып. 10. С. 59–113.
4. Jansen A. P. J. An Introduction to Monte Carlo Simulation Of Surface Reaction // arXiv: cond-mat/0303028v1 [cond-mat.stat-mech]
5. Bandman O. Parallel Simulation of Asynchronous Cellular Automata Evolution // LNCS. 2006. V. 4173. P. 41–48.
6. Achasova S., Bandman O., Markova V., Piskunov S. Parallel Substitution Algorithm: Theory and Application. Singapore: World Scientific, 1994. 180 p.
7. Itaka I., Aoki Y. Quantitative separation of WC from W₂C and tungsten, and the conditions of formation of two carbides // Bulletin of Chemical Society. 1982. No. 4. P. 108–114.
8. http://www.wolfram_at/wDeutsch/produkte/carbid/WC_eigenschaft. Common Properties — Tungsten Carbide.

СВЕДЕНИЯ ОБ АВТОРАХ

АГИБАЛОВ Геннадий Петрович — профессор, доктор технических наук, заведующий кафедрой защиты информации и криптографии Томского государственного университета, г. Томск. E-mail: agibalov@isc.tsu.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.sccc.ru, bandman@academ.org

ГРОМИЛОВ Сергей Александрович — доктор физико-математических наук, заведующий лабораторией Института неорганической химии им. А. В. Николаева СО РАН, г. Новосибирск. E-mail: grom@niic.nsc.ru

ДОЛГОВ Александр Алексеевич — аспирант Саратовского государственного университета им. Н. Г. Чернышевского, г. Саратов. E-mail: dolgov.a.a@gmail.com

ЕГОРОВ Владимир Николаевич — старший научный сотрудник, кандидат физико-математических наук, старший научный сотрудник Института проблем информационной безопасности, Московский государственный университет им. М. В. Ломоносова, г. Москва. E-mail: egorov49@inbox.ru

КАЧАНОВ Марк Александрович — студент кафедры защиты информации и криптографии Томского государственного университета, г. Томск. E-mail: m.a.kachanov@gmail.com

КИНЕЛОВСКИЙ Сергей Анатольевич — профессор, доктор физико-математических наук, ведущий научный сотрудник Института гидродинамики им. М. А. Лаврентьева СО РАН, г. Новосибирск. E-mail: skin@hydro.nsc.ru

ЛОГАЧЕВ Олег Алексеевич — кандидат физико-математических наук, заведующий отделом Института проблем информационной безопасности, Московский государственный университет им. М. В. Ломоносова, г. Москва. E-mail: logol@iisi.msu.ru

ПАНКРАТОВА Ирина Анатольевна — кандидат физико-математических наук, доцент кафедры защиты информации и криптографии Томского государственного университета, г. Томск. E-mail: pank@isc.tsu.ru

ПАРВАТОВ Николай Георгиевич — кандидат физико-математических наук, доцент кафедры защиты информации и криптографии Томского государственного университета, г. Томск. E-mail: parvatov@mail.tsu.ru

ПАУТОВ Павел Александрович — аспирант кафедры защиты информации и криптографии Томского государственного университета, г. Томск. E-mail: ___Pavel___@mail.ru

СМЫШЛЯЕВ Станислав Витальевич — студент Московского государственного университета им. М. В. Ломоносова, г. Москва. E-mail: smyshsv@gmail.com

ТРЕНЬКАЕВ Вадим Николаевич — доцент, кандидат технических наук, доцент кафедры защиты информации и криптографии Томского государственного университета, г. Томск. E-mail: tvnik@sibmail.com

ЧЕБОТАРЕВ Анатолий Николаевич — доктор технических наук, ведущий научный сотрудник Института кибернетики НАН Украины, г. Киев.
E-mail: ancheb@gmail.com

АННОТАЦИИ СТАТЕЙ НА АНГЛИЙСКОМ ЯЗЫКЕ

Egorov V. N. **ON AUTOMORPHISM GROUPS OF MATRICES.** In this paper we consider the groups of the left (right) automorphisms of matrices and their automorphism groups. Without loss of generality one can take square matrices over the ring of integers. For such a matrix, we suggest the notion of a quasiautomorphism and the correspondent notion of its quasiautomorphism group. The description of doubly transitive groups of the left (right) automorphisms is given with the help of the block designs. The knowledge of the structure of the balanced block designs is used for the calculation of the left (right) automorphisms and the quasiautomorphism groups of circulants. The problem that is under consideration is closely connected with the description of the graph automorphisms, the graph isomorphism problem, and also with the group classification of Boolean functions.
Keywords: *(quasi)automorphism groups of matrices, circulants, block designs.*

Logachev O. A. **ON VALUES OF AFFINITY LEVEL FOR ALMOST ALL BOOLEAN FUNCTIONS.** In the current paper, we consider asymptotic form of values of one parameter of Boolean functions, namely affinity level (generalized affinity level). We prove that asymptotically (with $n \rightarrow \infty$) values of affinity level (generalized affinity level) for almost all Boolean functions are all in the segment $[n - \log_2 n, n - \log_2 n + 1]$.
Keywords: *affinity level, generalized affinity level, Boolean equations, cryptography.*

Parvatov N. G. **POINT FUNCTIONS ON SEMILATTICES.** Fundamental functions classes on semilattices are studied.
Keywords: *semilattice, quasimonotonic function, monotonic function, point function, strongly point function.*

Smyshlyaev S. V. **CONSTRUCTION OF PERFECTLY BALANCED FUNCTIONS WITHOUT BARRIERS.** From the results of the previous papers dedicated to the set of perfectly balanced Boolean functions, one can conclude that the subset of Boolean functions without barriers is of prior interest in this set. Such a subset was considered previously, and the nonemptiness of it was proven, but no nontrivial estimations of the cardinality of this subset were found. In the current paper, some methods for constructing perfectly balanced Boolean functions without barriers are considered. They are based on the composition of Boolean functions of a special form and on certain important properties of such composition.
Keywords: *perfectly balanced functions, barriers of Boolean functions, filtering generator, cryptography.*

Agibalov G. P., Pankratova I. A. **STATISTICAL APPROXIMATION THEORY FOR DISCRETE FUNCTIONS WITH APPLICATION IN CRYPTANALYSIS OF ITERATIVE BLOCK CIPHERS.** A statistical approximation of a discrete function is defined as a Boolean equation being satisfied with a probability and accompanied by a Boolean function being statistically independent on a subset of variables. Properties of this notion are studied. A constructive test for the statistical independence is formulated. Methods for designing linear statistical approximations for functions used in iterative block symmetric ciphers are considered. Cryptanalysis algorithms based on solving systems of

statistical approximations being linear or nonlinear ones are proposed for symmetric ciphers. The algorithms are based on the maximum likelihood method. Definitions, methods and algorithms are demonstrated by examples taken from DES. Particularly, it is shown that one of the cryptanalysis algorithms proposed in the paper allows to find 34 key bits for full 16-round DES being based on two known nonlinear approximate equations providing 26 key bits only by Matsui's algorithm.

Keywords: *iterative block ciphers, statistical approximations, linear cryptanalysis, nonlinear cryptanalysis, DES.*

Trenkaev V. N. **ZAKREVSKIJ'S CIPHER BASED ON RECONFIGURABLE FSM.** The paper presents Zakrevskij's cipher realization based on reconfigurable finite state machine (FSM). The reconfigurable FSM generates a ciphering automaton according to a key. The ciphersystem can resist the brute-force attack and has key length which is acceptable in practice. The ciphersystem is shown can not resist the chosen-plaintext attack when a cryptanalyst knows the initial state of the ciphering automaton and has many copies of the cipher machine.

Keywords: *Zakrevskij's cipher, automata ciphersystem, invertible finite automata, automata with bijective output function, reconfigurable finite state machine, multiple unconditional experiments with automata.*

Kachanov M. A. **SECURITY ANALYSIS OF INFORMATION FLOWS IN GNU/LINUX OPERATING SYSTEMS.** The paper addresses to information flow analysis in *GNU/Linux* operating systems. Information flows by time with the participation of legal subjects are described, and some examples are given. A method for checking the possibility of information flows by memory between entities in a computer system hardened by SELinux is suggested.

Keywords: *information flows, Linux, security.*

Pautov P. A. **AUTHENTICATION IN TRUSTED SUBSYSTEM MODEL USING COMMUTATIVE ENCRYPTION.** The paper considers the peculiarities of authentication in multi-tier environment and corresponding security problems. The authentication protocol for multi-tier system based on commutative encryption is provided. Also, some specific commutative encryption algorithms are considered.

Keywords: *multi-tier systems, authentication in multi-tier systems, commutative encryption.*

Dolgov A. A. **A FAMILY OF EXACT 2-EXTENSIONS OF TOURNAMENTS.** The family of tournaments which have exact 1- and 2-extensions but haven't exact 3-extension is introduced. It is the only known family of graphs with such a property, and it is the fourth family of graphs which have exact k -extension for $k > 1$.

Keywords: *graph, exact k -extension, circulant.*

Chebotarev A. N. **SOLVING INEQUALITIES OVER FINITE STATE MACHINES IN THE REACTIVE SYSTEMS DESIGN.** The problem of solving inequalities over finite state machines (FSMs) is considered. This problem arises in compositional approach to the design of reactive systems. The problem is formulated and solved at the level of FSMs specifications in the logical language L . We show how to compute the maximal solution to the inequality with respect to the operation of synchronous composition of FSMs.

Keywords: *reactive system, language L specification, Σ -automaton, synchronous composition of Σ -automata, inequality over Σ -automata, maximal solution.*

Bandman O. L., Gromilov S. A., Kinelovsky S. A. **CUMULATIVE SYNTHESIS: A CELLULAR-AUTOMATA MODEL OF TARGET COATING FORMATION BY MEANS OF CUMULATIVE FLOW OF PARTICLES.** A discrete mathematical model of stochastic type is proposed. The model is intended for computer simulating of coating formation on metallic target under the influence of a high speed powder jet. The model belongs to the class of asynchronous probabilistic cellular automata. The model parameters are determined both from the actual test and from the known facts of similar processes. Program implementation and computer simulation of carbide formation on tungsten target being processed by carbon powder jet showed that the model is a useful instrument in the investigations aimed at creation methods for cumulative synthesis of new materials and structures.

Keywords: *computer simulation, asynchronous probabilistic cellular automaton, cumulative jet, hard coating, tungsten carbide.*

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 надежности вычислительных и управляющих систем*
- *Прикладная теория кодирования*
- *Прикладная теория автоматов*
- *Прикладная теория графов*
- *Логическое проектирование дискретных автоматов*
- *Математические основы информатики и программирования*
- *Вычислительные методы в дискретной математике*
- *Дискретные модели реальных процессов*
- *Математические основы интеллектуальных систем*
- *Исторические очерки по дискретной математике и ее приложениям*