ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ им. М. В. КЕЛДЫША
РОССИЙСКОЙ АКАДЕМИИ НАУК

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. М. В. ЛОМОНОСОВА
МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

МАТЕРИАЛЫ VI МОЛОДЕЖНОЙ НАУЧНОЙ ШКОЛЫ ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ И ЕЕ ПРИЛОЖЕНИЯМ

(Москва, 16–21 апреля 2007 г.)

ЧАСТЬ III

Москва 2007

МАТЕРИАЛЫ
VI МОЛОДЕЖНОЙ НАУЧНОЙ ШКОЛЫ
ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ И ЕЕ ПРИЛОЖЕНИЯМ

(Москва, 16–21 апреля 2007 г.)

ЧАСТЬ III

Москва 2007

М34
УДК 519.7

Издание осуществлено при поддержке Российского фонда фундаментальных исследований по проекту 07–01–06018

М34 Материалы VI молодежной научной школы по дискретной математике и ее приложениям (Москва, 16–21 апреля 2007 г.). Часть III. Под редакцией А. В. Чашкина. 2007. — 56 с.

Сборник содержит материалы VI молодежной научной школы по дискретной математике и ее приложениям, проходившей в Москве с 16 по 21 апреля 2007 г. при поддержке Российского фонда фундаментальных исследований (проект 07–01–06018). Для студентов, аспирантов и научных работников в области дискретной математики и математической кибернетики.

Научное издание
МАТЕРИАЛЫ
VI МОЛОДЕЖНОЙ НАУЧНОЙ ШКОЛЫ
ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ И ЕЕ ПРИЛОЖЕНИЯМ
(Москва, 16–21 апреля 2007 г.)

Под общей редакцией А. В. ЧАШКИНА

Ответственный за выпуск *Ф. М. Ковалев*

СОДЕРЖАНИЕ

ИЕРАРХИЯ КЛАССОВ БЕНТ-ФУНКЦИЙ КРАТНОЙ НЕЛИНЕЙНОСТИ

Н. Н. Токарева (Новосибирск)

В работе предлагается иерархия мер нелинейности булевых функций от четного числа m переменных. В ее основе лежит понятие *максимально k-нелинейной* (k-*бент*) функции — функции максимально нелинейной в k различных смыслах одновременно. Обычные бент-функции представляют класс 1-бент-функций. Для $k > j \geq 1$ класс k-бент-функций является собственным подклассом класса j-бент-функций. Для каждого допустимого k приводятся способы построения k-бент-функций и рассматриваются некоторые их свойства; при этом $k = 1, \ldots, m/2$.

1. Введение. Одной из важных характеристик булевой функции в криптографии является мера ее нелинейности. Линейность и близкие к ней свойства булевой функции, как правило, представляют собой богатый источник информации о многих других ее свойствах, что в криптографии, безусловно, является нежелательным. С целью максимизировать меру нелинейности булевой функции в криптографии выделяют класс *максимально нелинейных* (или *бент-*) функций — функций, определяемых как функции, удаленные от множества всех аффинных функций на наибольшее возможное расстояние (см. обзоры методов построения таких функций в [1] и [2]). В геометрической интерпретации векторы значений всех аффинных булевых функций $\langle \mathbf{u}, \mathbf{v} \rangle \oplus a$ от m переменных образуют двоичный линейный код Адамара длины 2^m, а векторы значений бент-функций удалены от этого кода на максимально возможное расстояние $2^{m-1} - 2^{(m/2)-1}$. Говоря неформально, каждая функция f из класса бент-функций "плохо" аппроксимируется аффинными функциями. Именно это свойство булевых функций, использующихся в блочных шифрах, способствует предельному повышению стойкости этих шифров к методам линейного и дифференциального криптоанализа, см. например [3].

Однако, принадлежность функции f классу бент-функций не исключает того, что f может оказаться "хорошо" аппроксимируемой функциями некоторого другого класса, являющимися нелинейными, но обладающими свойством "скрытой линейности— линейности в некотором другом смысле. Тогда использование таких бент-функций, например, в блочном шифре может обнаружить его слабость к соответствующим модификациям вышеупомянутых методов криптоанализа. С целью избежать подобные ситуации мы рассмотрим бент-функции с более сильными свойствами нелинейности, а именно бент-функции от m переменных максимально нелинейные при k различных смыслах линейности одновременно, где k меняется от 1 до $m/2$.

Поясним, что мы подразумеваем под "скрытой линейностью". С 90-х годов в теории кодирования активно стали исследоваться нелинейные коды, образы которых под действием подходящих (как правило, взаимно-однозначных и изометричных) отображений в другие метрические пространства линейны (см. [4], [5], [6]).Такие "скрыто линейные" коды среди всех кодов с некоторыми фиксированными параметрами, зачастую, немногочисленны и по своим свойствам близки к линейным.

Рассмотрим $\mathbb{Z}_2$- и $\mathbb{Z}_4$-линейные коды Адамара. Известно, что $\mathbb{Z}_2$-линейный (т. е. просто линейный) двоичный код Адамара длины 2^m единствен с точностью до эквивалентности. Д. С. Кротовым [7] было показано, что существуют в точности $\lfloor m/2 \rfloor$ попарно неэквивалентных $\mathbb{Z}_4$-линейных кодов Адамара длины 2^{m+1} при $m \geq 3$. Опираясь на данную Д. С. Кротовым [7] классификацию всех таких кодов, рассмотрим серию некоторых "скрыто линейных" двоичных кодов Адамара A_m^k, $1 \leq k \leq \lfloor m/2 \rfloor$ длины 2^m. Множество булевых функций, векторами значений которых являются кодовые векторы кода A_m^k, представляют собой аналог множества аффинных функций — это функции вида $\langle \mathbf{u}, \mathbf{v} \rangle_k \oplus a$, где операция $\langle , \rangle_k$ играет роль скалярного произведения. В рассматриваемой серии кодов каждый код A_m^k получается из линейного четверичного кода $\mathcal{A}_m^k$ заменой элементов $0, 1$ на 0 и элементов $2, 3$ на 1 в каждой координате, где $\mathcal{A}_m^k$ — подкод соответствующего линейного четверичного кода Адамара типа $4^k 2^{m-2k}$, состоящий из всех кодовых векторов, имеющих в первой координате только 0 или 2. При этом код A_m^1 линеен, и все коды $A_m^1, \ldots, A_m^{\lfloor m/2 \rfloor}$ попарно неэквивалентны. Такие "скрыто линейные" коды Адамара выбраны для того, чтобы возникающие новые скалярные произведения $\langle , \rangle_k$ обладали многими свойствами обычного скалярного произведения (см. утверждение 1) и на их основе оказались возможными конструктивные построения. Булеву функцию f от четного числа переменных m назовем *максимально k-нелинейной* (k-*бент*) функцией, $1 \leq k \leq m/2$, если вектор значений функции f удален на наибольшее возможное расстояние $2^{m-1} - 2^{(m/2)-1}$ от каждого кода Адамара A_m^i, $i = 1, \ldots, k$. Обычные бент-функции представляют собой класс 1-бент-функций $\mathfrak{B}_m^1$. Для $k > j \geq 1$ класс k-бент-функций $\mathfrak{B}_m^k$ является собственным подклассом класса j-бент-функций $\mathfrak{B}_m^j$. Для каждого k, $k = 1, \ldots, m/2$, в работе приводятся способы построения k-бент-функций и рассматриваются некоторые их свойства.

2. Необходимые определения. Пусть $\langle \mathbf{u}, \mathbf{v} \rangle$ — обычное скалярное произведение двоичных векторов $\mathbf{u}$ и $\mathbf{v}$. Множество всех булевых функций от m переменных обозначим через $\mathfrak{F}_m$. Через $\mathfrak{A}_m$ обозначим класс всех аффинных булевых функций от m переменных. Каждой булевой функции $f \in \mathfrak{F}_m$ соответствует двоичный вектор $\mathbf{f}$ ее значений длины 2^m. Всюду далее векторы, в отличие от функций, будем выделять полужирным шрифтом. Вес Хэмминга и расстояние Хэмминга обозначим через $wt_H(\cdot)$ и $d_H(\cdot, \cdot)$ соот-

ветственно. Под расстоянием $dist(\cdot,\cdot)$ между булевыми функциями понимается расстояние Хэмминга между соответствующими векторами значений. Напомним, что для функции $f \in \mathfrak{F}_m$ целочисленная функция W_f, заданная на множестве $\mathbb{Z}_2^m$ равенством $W_f(\mathbf{v}) = \sum_{\mathbf{u}\in\mathbb{Z}_2^m}(-1)^{\langle \mathbf{u},\mathbf{v}\rangle \oplus f(\mathbf{u})}$, называется *преобразованием Уолша—Адамара* (или *дискретным преобразованием Фурье*) функции f. Имеет место равенство Парсеваля: $\sum_{\mathbf{v}\in\mathbb{Z}_2^m}\left(W_f(\mathbf{v})\right)^2 = 2^{2m}$, из которого следует, что $\max\limits_{\mathbf{v}\in\mathbb{Z}_2^m}|W_f(\mathbf{v})| \geq 2^{m/2}$. Под *нелинейностью* N_f булевой функции f понимается расстояние от данной функции до множества $\mathfrak{A}_m$, т. е. $N_f = dist(f,\mathfrak{A}_m) = 2^{m-1} - \frac{1}{2}\max\limits_{\mathbf{v}\in\mathbb{Z}_2^m}|W_f(\mathbf{v})|$. Функция $f \in \mathfrak{F}_m$ называется *максимально нелинейной* (m любое), если параметр N_f принимает наибольшее возможное значение, и *бент-функцией* (m четное), если для любого $\mathbf{v} \in \mathbb{Z}_2^m$ справедливо $W_f(\mathbf{v}) = \pm 2^{m/2}$. При четном m эти определения совпадают.

Пусть $\langle \mathbb{Z}_2^n, d_H\rangle$ — метрическое пространство на множестве двоичных векторов длины n с метрикой Хэмминга. Непустое множество $C \subseteq \mathbb{Z}_2^n$ мощности M с минимальным расстоянием d между его различными элементами называется *двоичным* $(n, M, d)_2$*-кодом*, а его элементы — *кодовыми словами*. Параметры n и d называются соответственно *длиной* и *кодовым расстоянием* кода. Код называется *линейным*, если он образует линейное подпространство в $\mathbb{Z}_2^n$. Пусть $\beta, \gamma : \mathbb{Z}_4 \to \mathbb{Z}_2$ — отображения такие, что: $\beta(0) = \beta(1) = 0$, $\beta(2) = \beta(3) = 1$ и $\gamma(0) = \gamma(3) = 0$, $\gamma(1) = \gamma(2) = 1$. Пусть $\phi : \mathbb{Z}_4 \to \mathbb{Z}_2^2$ — отображение Грея: $\phi(c) = (\beta(c), \gamma(c))$ для $c \in \mathbb{Z}_4$. Отображения β, γ и ϕ покоординатно продолжаются до отображений $\beta, \gamma : \mathbb{Z}_4^i \to \mathbb{Z}_2^i$ и $\phi : \mathbb{Z}_4^i \to \mathbb{Z}_2^{2i}$ для любого целого i. Напомним, что ϕ согласно [5] является изометрией, т. е. для любых $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_4^i$ выполняется $d_L(\mathbf{x}, \mathbf{y}) = d_H(\phi(\mathbf{x}), \phi(\mathbf{y}))$. Четверичный код длины n *линеен*, если он является подгруппой группы $\mathbb{Z}_4^n$. Двоичный код C называется $\mathbb{Z}_4$*-линейным*, если код $\phi^{-1}(C)$ линеен. Пусть m, k положительные целые числа, причем $0 \leq k \leq m/2$. *Ядром* двоичного кода C, содержащего нулевой вектор, называется максимальный линейный подкод $Ker(C)$ кода C такой, что выполняется $\mathbf{x} \oplus C = C$ для любого вектора $\mathbf{x} \in Ker(C)$.

3. Коды Адамара A_m^k. Всюду далее пусть $n = 2^m$. Пусть $\mathbf{G}_m^k$ — четверичная $(m-k) \times n$ – матрица, состоящая из лексикографически упорядоченных столбцов $\mathbf{z}^T$, где $\mathbf{z} \in \{0,1,2,3\}^k \times \{0,2\}^{m-2k}$. Например,

$$\mathbf{G}_1^0 = \begin{pmatrix}02\end{pmatrix}, \mathbf{G}_2^1 = \begin{pmatrix}0123\end{pmatrix}, \mathbf{G}_2^0 = \begin{pmatrix}0022\\0202\end{pmatrix}, \mathbf{G}_3^1 = \begin{pmatrix}00112233\\02020202\end{pmatrix}.$$

Матрицы такого вида впервые рассматривались Д. С. Кротовым в работах [8] и [7] для построения $\mathbb{Z}_4$-линейных кодов Адамара длины $2n$ и получения их полной классификации. Определим взаимно-однозначное отображение

$\phi_k : \mathbb{Z}_4^k \times \mathbb{Z}_2^{m-2k} \to \mathbb{Z}_2^m$ по правилу:

$$\phi_k : (\mathbf{u}', \mathbf{u}'') \to (\phi(\mathbf{u}'), \mathbf{u}'') \text{ для любых векторов } \mathbf{u}' \in \mathbb{Z}_4^k, \mathbf{u}'' \in \mathbb{Z}_2^{m-2k}.$$

Аналогично тому как это было сделано в [6] определим бинарную операцию $\star : \mathbb{Z}_2^m \times \mathbb{Z}_2^m \to \mathbb{Z}_2^m$ следующим образом:

$$\mathbf{u} \star \mathbf{v} = \phi_k(\phi_k^{-1}(\mathbf{u}) + \phi_k^{-1}(\mathbf{v})) \text{ для любых векторов } \mathbf{u} \in \mathbb{Z}_2^m, \mathbf{v} \in \mathbb{Z}_2^m.$$

Пусть $\mathbf{C}_m^k = (c_{\mathbf{u},\mathbf{v}}^k)$, $\mathbf{u}, \mathbf{v} \in \mathbb{Z}_2^m$, — четверичная $n \times n$-матрица, строками которой являются всевозможные векторы $\mathbf{h}^{\mathbf{u}} = \phi_k^{-1}(\mathbf{u}) \cdot \mathbf{G}_m^k$, расположенные в порядке лексикографического возрастания векторов $\phi_k^{-1}(\mathbf{u})$. Считаем, что нумерация столбцов матрицы $\mathbf{C}_m^k$ также производится в порядке лексикографического возрастания векторов $\phi_k^{-1}(\mathbf{v})$. Например,

$$\mathbf{C}_1^0 = \begin{pmatrix} 00 \\ 02 \end{pmatrix}, \mathbf{C}_2^0 = \begin{pmatrix} 0000 \\ 0202 \\ 0022 \\ 0220 \end{pmatrix}, \mathbf{C}_2^1 = \begin{pmatrix} 0000 \\ 0123 \\ 0202 \\ 0321 \end{pmatrix}, \mathbf{C}_3^0 = \begin{pmatrix} 00000000 \\ 02020202 \\ 00220022 \\ 02200220 \\ 00002222 \\ 02022020 \\ 00222200 \\ 02202002 \end{pmatrix}, \mathbf{C}_3^1 = \begin{pmatrix} 00000000 \\ 02020202 \\ 00112233 \\ 02132031 \\ 00220022 \\ 02200220 \\ 00332211 \\ 02312013 \end{pmatrix}.$$

Пусть четверичный линейный код $\mathcal{A}_m^k$ состоит из векторов $\mathbf{h}^{\mathbf{u}}$ и $\mathbf{h}^{\mathbf{u}} + \mathbf{2}$, где $\mathbf{h}^{\mathbf{u}}$ — строка матрицы $\mathbf{C}_m^k$. Определим двоичный код $A_m^k = \beta(\mathcal{A}_m^k)$. Отметим, что на множестве A_m^k отображение β обратимо, что, вообще говоря, неверно на всём множестве $\mathbb{Z}_2^n$. Определим бинарную операцию $\bullet : A_m^k \times A_m^k \to A_m^k$, согласованную с операцией $+$ на $\mathcal{A}_m^k$:

$$\mathbf{x} \bullet \mathbf{y} = \beta(\beta^{-1}(\mathbf{x}) + \beta^{-1}(\mathbf{y})) \text{ для любых векторов } \mathbf{x}, \mathbf{y} \in A_m^k.$$

Нетрудно видеть, что $(A_m^k, \bullet)$ является абелевой группой.

Теорема 1. *При любом k, $0 \le k \le m/2$, двоичный код A_m^k с заданной на нем групповой операцией $\bullet$ является $(n, 2n, n/2)_2$-кодом Адамара. Коды A_m^0, A_m^1 линейны, при $k \ge 2$ код A_m^k нелинеен, причем размерность ядра кода A_m^k равна $m - k + 1$.*

4. Аналог скалярного произведения $\langle \cdot, \cdot \rangle_k$. Для любого k, $0 \le k \le m/2$, определим бинарную операцию $\langle , \rangle_k : \mathbb{Z}_2^m \times \mathbb{Z}_2^m \to \mathbb{Z}_2$ (аналог скалярного произведения) следующим образом: $\langle \mathbf{u}, \mathbf{v} \rangle_k = \beta(c_{\mathbf{u},\mathbf{v}}^k)$ для любых $\mathbf{u}, \mathbf{v} \in \mathbb{Z}_2^m$. Операция $\langle , \rangle_0$ совпадает с обычным скалярным произведением, т. е. $\langle \mathbf{u}, \mathbf{v} \rangle_0 = \langle \mathbf{u}, \mathbf{v} \rangle$. Пусть π_k обозначает подстановку $(1,2)(3,4)\ldots(2k-1,2k)$ на m элементах, представленную в виде произведения транспозиций.

Утверждение 1. *Пусть $m \ge 0, k$ — целые, $0 \le k \le m/2$. Для любых векторов $\mathbf{u}, \mathbf{v} \in \mathbb{Z}_2^m$ выполняется:*

(i) $\langle \mathbf{u}, \mathbf{v} \rangle_k = \langle \mathbf{v}, \mathbf{u} \rangle_k$;

(*ii*) $\langle a\mathbf{u}, \mathbf{v}\rangle_k = a\langle \mathbf{u}, \mathbf{v}\rangle_k$ *для любого* $a \in \mathbb{Z}_2$;

(*iii*) $\sum_{\mathbf{w}\in\mathbb{Z}_2^m}(-1)^{\langle \mathbf{u},\mathbf{w}\rangle_k \oplus \langle \mathbf{v},\mathbf{w}\rangle_k} = \begin{cases} 2^m, & \text{если } \mathbf{u} = \mathbf{v}, \\ 0, & \text{иначе.} \end{cases}$

(*iv*) $\langle (\mathbf{u}, a), (\mathbf{v}, b)\rangle_k = \langle \mathbf{u}, \mathbf{v}\rangle_k \oplus ab$ *для любых* $a, b \in \mathbb{Z}_2$;

(*v*) $\langle (a, a'), (b, b')\rangle_1 = \langle (a', a), (b, b')\rangle_0$ *для любых* $a, a', b, b' \in \mathbb{Z}_2$;

(*vi*) $\langle (a, a', \mathbf{u}), (b, b', \mathbf{v})\rangle_{k+1} = \langle (a, a'), (b, b')\rangle_\varepsilon \oplus \langle \mathbf{u}, \mathbf{v}\rangle_k$, *для любых* $a, a', b, b' \in \mathbb{Z}_2$, *где параметр* $\varepsilon \in \mathbb{Z}_2$ *определяется равенством* $\varepsilon = \langle \mathbf{u}, \mathbf{v}\rangle_k \oplus \langle \pi_k(\mathbf{u}), \mathbf{v}\rangle_k \oplus 1$;

(*vii*) $\langle \mathbf{u}, \mathbf{w}\rangle_k \oplus \langle \mathbf{v}, \mathbf{w}\rangle_k = \langle \mathbf{u} \star \mathbf{v}, \mathbf{w}\rangle_k \oplus \Big(\langle \mathbf{u}, \mathbf{w}\rangle_k \oplus \langle \pi_k(\mathbf{u}), \mathbf{w}\rangle_k\Big)\Big(\langle \mathbf{v}, \mathbf{w}\rangle_k \oplus \langle \pi_k(\mathbf{v}), \mathbf{w}\rangle_k\Big)$.

Для каждого k, $0 \le k \le m/2$, целочисленную функцию W_f^k, заданную на множестве $\mathbb{Z}_2^m$ равенством $W_f^k(\mathbf{v}) = \sum_{\mathbf{u}\in\mathbb{Z}_2^m}(-1)^{\langle \mathbf{u},\mathbf{v}\rangle_k \oplus f(\mathbf{u})}$ для любого $\mathbf{v} \in \mathbb{Z}_2^m$, назовем *k-преобразованием Уолша — Адамара* булевой функции $f \in \mathfrak{F}_m$. Заметим, что W_f^0 совпадает с W_f. Нетрудно видеть, что для W_f^k имеет место аналог равенства Парсеваля, из которого следует неравенство $\max\limits_{\mathbf{v}\in\mathbb{Z}_2^m} |W_f^k(\mathbf{v})| \ge 2^{m/2}$. Пусть каждому вектору $\mathbf{g}$ кода A_m^k отвечает функция $g \in \mathfrak{F}_m$, для которой вектор $\mathbf{g}$ является вектором значений, причем $g(\mathbf{v}) = \langle \mathbf{u}, \mathbf{v}\rangle_k \oplus a$ для некоторых $\mathbf{u} \in \mathbb{Z}_2^m$, $a \in \mathbb{Z}_2$ и произвольного $\mathbf{v} \in \mathbb{Z}_2^m$. Множество всех таких функций от m переменных назовем множеством *k-аффинных функций* и обозначим через $\mathfrak{A}_m^k$. Расстояние между булевой функцией $f \in \mathfrak{F}_m$ и множеством функций $\mathfrak{A}_m^k$ назовем *k-нелинейностью* функции f и обозначим через N_f^k. Можно показать, что имеет место равенство $N_f^k = 2^{m-1} - \frac{1}{2}\max\limits_{\mathbf{v}\in\mathbb{Z}_2^m} |W_f^k(\mathbf{v})|$. Для произвольного k, $1 \le k \le m/2$, булеву функцию $f \in \mathfrak{F}_m$ назовем *максимально k-нелинейной*, если каждый параметр N_f^j, $j = 1, \ldots, k$, принимает наибольшее возможное значение; и *k-бент-функцией*, если все коэффициенты $W_f^j(\mathbf{v})$, $j = 1, \ldots, k$, равны $\pm 2^{m/2}$ (m четно). В случае четного m эти определения эквивалентны. Наибольшее число k, для которого бент-функция является k-бент-функцией, назовем *кратностью нелинейности* этой функции. Класс всех k-бент-функций от m переменных обозначим через $\mathfrak{B}_m^k$. Из утверждения 1 несложно следует, что класс $\mathfrak{B}_m^1$ является классом обычных бент-функций $\mathfrak{B}_m$.

5. Построение k-бент-функций и их свойства. С помощью компьютера нами было проверено, что $|\mathfrak{B}_4^1| = 448$, $|\mathfrak{B}_4^2| = 192$. Функция $\xi(u_1, u_2, u_3, u_4) = u_1u_2 \oplus u_2u_3 \oplus u_3u_4$ представляет пример функции из $\mathfrak{B}_4^1 \setminus \mathfrak{B}_4^2$.

Пусть $\mathfrak{F}_2^1$ — множество всех симметрических функций от двух переменных. Приведем индуктивный способ построения k-бент-функций.

Теорема 2. *Пусть $m, r \in \mathbb{N}$ четны, $k, j \in \mathbb{N}$ — любые, причем $1 \le k \le m/2$. Пусть функция $f \in \mathfrak{F}_{2j+m+r}$ представима в виде*

$$f(a_1, a'_1, \ldots, a_j, a'_j, \mathbf{u}', \mathbf{u}'') = \left(\bigoplus_{i=1}^{j} s_i(a_i, a'_i) \right) \oplus p(\mathbf{u}') \oplus q(\mathbf{u}''),$$

где $s_1, \ldots, s_j \in \mathfrak{F}_2^1, p \in \mathfrak{F}_m$ и $q \in \mathfrak{F}_r$ — функции с непересекающимися множествами переменных. Тогда f принадлежит классу $\mathfrak{B}_{2j+m+r}^{j+k}$ тогда и только тогда, когда $s_1, \ldots, s_j \in \mathfrak{B}_2^1$, $p \in \mathfrak{B}_m^k$ и $q \in \mathfrak{B}_r^1$.

Следствие 1. *Справедливо $\mathfrak{B}_m^k \neq \emptyset$ для любого четного $m \ge 2$ и любого $k \le m/2$.*

Следствие 2. *Для четного $m \ge 2$ справедливо $\mathfrak{B}_m^1 \supset \mathfrak{B}_m^2 \supset \cdots \supset \mathfrak{B}_m^{m/2}$.*

Рассмотрим следующую взаимосвязь k-бент-функций с обычными бент-функциями. Обозначим через S_m^k подгруппу группы S_m подстановок на m координатах, порожденную k транспозициями: $(1,2), (3,4), \ldots, (2k-1, 2k)$. Пусть $\mathfrak{F}_m^k$ обозначает множество всех функций $f \in \mathfrak{F}_m$, постоянных на каждой орбите множества $\mathbb{Z}_2^m$ под действием группы S_m^k. Несложно проверить, что $|\mathfrak{F}_m^k| = 2^{2^{m-k\log\frac{4}{3}}}$, где log обозначает $\log_2$.

Теорема 3. *Справедливо равенство $\mathfrak{F}_m^k \cap \mathfrak{B}_m^k = \mathfrak{F}_m^k \cap \mathfrak{B}_m^1$ для четного $m \ge 2$ и любого k, $1 \le k \le m/2$.*

Однако, функциями из $\mathfrak{F}_m^k \cap \mathfrak{B}_m^1$ весь класс $\mathfrak{B}_m^k$ не исчерпывается. Интересным для дальнейшего исследования представляется вопрос: какие значения принимает кратность нелинейности для функций из известных классов бент-функций?

Работа выполнена при финансовой поддержке интеграционного проекта СО РАН № 35 «Древовидный каталог математических Интернет-ресурсов» и Российского фонда фундаментальных исследований (проект 07-01-00248).

Список литературы

1. Логачев О. А., Сальников А. А., Ященко В. В. Булевы функции в теории кодирования и криптологии // Москва, 2004.

2. Dobbertin H. and Leander G. A Survey of Some Recent Results on Bent Functions // Proc. Third Int. Conf. "Sequences and Their Applications" SETA, 2004. LNCS 3486. P. 1–29.

3. Молдовян А. А., Молдовян Н. А., Гуц Н. Д., Изотов Б. В. Криптография: скоростные шифры // СПб.: БХВ-Петербург, 2002. 496 с.

4. Нечаев А. А. Код Кердока в циклической форме // Дискретная математика. Т. 1. Вып. 4. 1989. С. 123–139.

5. Hammons A. R., Kumar P. V., Calderbank A. R., Sloane N. J. A., Solé P. The $\mathbb{Z}_4$-Linearity of Kerdock, Preparata, Goethals, and Related Codes // IEEE Trans. Inform. Theory. 1994. V. 40. № 2. P. 301–319.

6. Borges J., Phelps K. T., Rifa J., Zinoviev V. A. On $\mathbb{Z}_4$-Linear Preparata-Like and Kerdock-Like Codes // IEEE Trans. Inform. Theory. 2003. V. 49. № 11. P. 2834–2843.

7. Krotov D. S. $\mathbb{Z}_4$-linear Hadamard and extended perfect codes // Proc. of the Int. Workshop on Coding and Cryptography WCC 2001, Jan. 8–12, 2001. Paris, France. P. 329–334.

8. Кротов Д. С. $\mathbb{Z}_4$-линейные совершенные коды // Дискретный анализ и исследование операций. Сер. 1. Новосибирск: Ин-т математики СО РАН. 2000. Т. 7. № 4. С. 78–90.

О ВЕРХНЕЙ ОЦЕНКЕ ЧИСЛА РАВНОМЕРНО УПАКОВАННЫХ ДВОИЧНЫХ КОДОВ

Н. Н. Токарева (Новосибирск)

В работе рассматриваются равномерно упакованные (в широком смысле) двоичные коды длины n с кодовым расстоянием d и радиусом покрытия ρ. Показано, что любой такой код однозначно определяется множеством своих кодовых слов весов $\lceil \frac{n}{2} \rceil - \rho, \ldots, \lfloor \frac{n}{2} \rfloor + \rho$, и в случае нечётного d число различных таких кодов не превышает числа $2^{2^{n - \frac{d}{2} \log_2 n + o(\log_2 n)}}$.

1. Введение. Пусть E^n — метрическое пространство на множестве двоичных векторов длины n с метрикой Хемминга $d(\cdot, \cdot)$ (расстояние между двумя векторами равно числу координат, в которых векторы различаются). *Вес Хемминга* $wt(\cdot)$ вектора из E^n определяется как число его ненулевых координат (т. е. как расстояние до нулевого вектора $\mathbf{0}$). Непустое подмножество C в пространстве E^n с минимальным расстоянием d между его различными элементами называется *двоичным* (n, d)*-кодом*, где n — *длина*, а d — *кодовое расстояние* кода. *Радиусом покрытия* ρ двоичного кода C длины n называется максимальное расстояние, на которое может быть удалён от кода C двоичный вектор длины n, т. е. $\rho = \max_{x \in E^n} d(x, C)$. Согласно работе Л. А. Бассалыго, Г. В. Зайцева и В. А. Зиновьева [2] двоичный (n, d)-код C с радиусом покрытия ρ называется *равномерно упакованным в широком смысле*, если существуют действительные числа $\alpha_0, \alpha_1, \ldots, \alpha_\rho$ такие, что для любого двоичного вектора x длины n выполняется равенство $\sum_{i=0}^{\rho} \alpha_i f_i(x) = 1$, где $f_i(x)$ — число кодовых слов кода C, находящихся на расстоянии i от вектора x, $i = 0, 1, \ldots, \rho$. Пусть $d = 2t + 1$. Далее под термином "равномерно упакованный" будем понимать "равномерно упакованный

в широком смысле". С. В. Августиновичем [1] было показано, что каждый двоичный совершенный код длины n с кодовым расстоянием 3 однозначно определяется множеством своих кодовых слов веса $(n-1)/2$. Используя это свойство, в [1] было показано, что число различных совершенных двоичных кодов не превосходит $2^{2^{n-\frac{3}{2}\log n+o(\log n)}}$ (здесь и далее log обозначает логарифм по основанию 2).

Рассмотрим произвольный класс $\mathbb{L}(n,d,\rho;\alpha_0,\ldots,\alpha_\rho)$ двоичных равномерно упакованных (в широком смысле) (n,d)-кодов с радиусом покрытия ρ и параметрами равномерной упаковки $\alpha_0,\ldots,\alpha_\rho$. Считаем, что d и ρ — константы. Число различных кодов в классе $\mathbb{L}(n,d,\rho;\alpha_0,\ldots,\alpha_\rho)$ обозначим через $L_{n,d}$. Используя границу сферической упаковки для мощности (n,d)-кода, несложно получить следующую тривиальную оценку: $L_{n,d}\le 2^{2^{n-\frac{d-1}{2}\log n+o(\log n)}}$. Обобщая метод работы [1], покажем, что любой код из класса $\mathbb{L}(n,d,\rho;\alpha_0,\ldots,\alpha_\rho)$ однозначно определяется множеством своих кодовых слов весов $\lceil n/2\rceil-\rho,\ldots,\lfloor n/2\rfloor+\rho$, и в случае нечётного d имеет место оценка:

$$L_{n,d}\le 2^{2^{n-\frac{d}{2}\log n+o(\log n)}}.$$

Заметим, что параметры ρ и $\alpha_0,\ldots,\alpha_\rho$ в полученную оценку не входят.

2. Необходимые утверждения. Пусть x,y — любые двоичные векторы длины n, и пусть $d(x,y)=k$. Известно (см., например, [4, гл. 21]), что число векторов $z\in E^n$ таких, что $d(x,z)=i$ и $d(y,z)=j$, не зависит от выбора векторов x и y, а зависит лишь от чисел i,j,k,n. Обозначим это число через p_{ijk} (подразумевая также зависимость этого параметра от n). Ясно, что $p_{ijk}=\binom{k}{(i-j+k)/2}\binom{n-k}{(i+j-k)/2}$, если число $i+j-k$ — чётное. В случае нечётного $i+j-k$ имеем $p_{ijk}=0$. Будем считать, что параметр p_{ijk} определён для любых значений i,j и k, $0\le i,j,k\le n$, и равен нулю, если соответствующее множество векторов z пусто.

Пусть C — произвольный код из класса равномерно упакованных кодов $\mathbb{L}(n,d,\rho;\alpha_0,\ldots,\alpha_\rho)$. Обозначим через C_i и E_i множества векторов веса i кода C и пространства E^n соответственно, где $i=0,1,\ldots,n$. Пусть μ_C^i — мощность множества C_i. Набор $\mu(C)=\{\mu_C^0,\mu_C^1,\ldots,\mu_C^n\}$ называется *весовым спектром* кода C, а числа μ_C^i, $i=0,1,\ldots,n$, — *спектральными значениями* кода. В работе [2] приведена формула для вычисления весового спектра (более точно: весовой функции) произвольного равномерно упакованного кода, содержащая ρ неизвестных констант. Для определения этих констант требуется знать любые ρ спектральных значений кода, при которых возможно решение соответствующей системы линейных уравнений (см. подробнее [2]).

Лемма 1. *Весовой спектр произвольного кода C из класса равномерно упакованных кодов $\mathbb{L}(n, d, \rho; \alpha_0, \ldots, \alpha_\rho)$ однозначно определяется значениями $\mu_C^0, \ldots \mu_C^{\rho-1}$.*

Доказательство. Покажем как с помощью известных значений $\mu_C^0, \ldots,$ $\mu_C^{j+\rho-1}$ при любом $j = 0, 1, \ldots, n-\rho$ восстановить значение $\mu_C^{j+\rho}$. При любом $i = 0, 1, \ldots, \rho$ имеет место следующее равенство

$$\sum_{x \in E_j} f_i(x) = \sum_{k=\max\{0, j-i\}}^{j+i} p_{ijk}\ \mu_C^k. \tag{1}$$

Действительно, каждый кодовый вектор веса k находится на расстоянии i в точности от p_{ijk} двоичных векторов веса j. Заметим, что в каждом соотношении (1) при $i = 0, 1, \ldots, \rho-1$ участвуют лишь известные спектральные значения $\mu_C^{\max\{0, j-\rho+1\}}, \ldots, \mu_C^{j+\rho-1}$, а при $i = \rho$ единственным неизвестным спектральным значением является $\mu_C^{j+\rho}$, причём оно входит в это равенство с ненулевым коэффициентом. В силу равномерной упакованности кода C справедливо равенство $\sum_{x \in E_j} \sum_{i=0}^{\rho} \alpha_i f_i(x) = \binom{n}{j}$. Меняя местами знаки суммирования в этом равенстве и пользуясь (1), получаем

$$\sum_{i=0}^{\rho} \alpha_i \sum_{k=\max\{0, j-i\}}^{j+i} p_{ijk}\ \mu_C^k = \binom{n}{j}.$$

Отсюда однозначно определяется значение $\mu_C^{j+\rho}$. Таким образом последовательно восстанавливаются значения $\mu_C^\rho, \ldots, \mu_C^n$.

Следующая лемма является обобщением одного свойства совершенных двоичных кодов, приведённого в работе [1].

Лемма 2. *Множество $X = C_{\lceil n/2 \rceil - \rho} \cup \cdots \cup C_{\lfloor n/2 \rfloor + \rho}$ однозначно определяет код C из класса равномерно упакованных кодов $\mathbb{L}(n, d, \rho; \alpha_0, \ldots, \alpha_\rho)$.*

Доказательство. Для кода C обозначим через A и B следующие множества: $A = C_0 \cup \cdots \cup C_{\lceil n/2 \rceil - \rho - 1}$ и $B = C_{\lfloor n/2 \rfloor + \rho + 1} \cup \cdots \cup C_n$. Тогда $C = A \cup X \cup B$. Несложно заметить, что расстояние между множествами A и B не меньше $2\rho + 1$ и, следовательно, не меньше d. Предположим, что существует другой код $C' = A' \cup X \cup B'$ из класса $\mathbb{L}(n, d, \rho; \alpha_0, \ldots, \alpha_\rho)$, и пусть $B \neq B'$. Тогда код C'', полученный из C заменой множества B на B', также имеет кодовое расстояние d. Покажем, что C'' принадлежит классу $\mathbb{L}(n, d, \rho; \alpha_0, \ldots, \alpha_\rho)$, т. е. является равномерно упакованным кодом с параметрами $\alpha_0, \ldots, \alpha_\rho$. Для произвольного вектора $x \in E^n$ рассмотрим сумму

$$\sum_{i=0}^{\rho} \alpha_i f_i(x), \tag{2}$$

где $f_i(x)$ — число кодовых слов кода C'', находящихся на расстоянии i от вектора x. Обозначим через $T_\rho^D(x)$ множество всех кодовых слов произвольного кода D длины n, содержащихся в шаре радиуса ρ с центром в вершине x, т. е. $T_\rho^D(x) = \{\, y \in D \mid d(x,y) \le \rho \,\}$. По построению кода C'' имеем

$$T_\rho^{C''}(x) = \begin{cases} T_\rho^C(x), & \text{если } wt(x) \le \lfloor n/2 \rfloor, \\ \\ T_\rho^{C'}(x), & \text{если } wt(x) \ge \lceil n/2 \rceil. \end{cases}$$

Так как коды C и C' являются равномерно упакованными с параметрами $\alpha_0, \ldots, \alpha_\rho$, то для любого вектора $x \in E^n$ сумма (2) равна 1. Таким образом, код C'' принадлежит классу равномерно упакованных кодов $\mathbb{L}(n, d, \rho; \alpha_0, \ldots, \alpha_\rho)$.

Поскольку $B \ne B'$, без ограничения общности можно считать, что найдётся вектор $y \in E^n$ такой, что $y \in B$ и $y \notin B'$. Пусть $z = y \oplus \mathbf{1}$, где $\mathbf{1}$ — вектор со всеми координатами, равными 1, и $\oplus$ обозначает покоординатное сложение векторов по модулю 2. Тогда, как нетрудно заметить, выполняется неравенство $wt(z) \le \lceil n/2 \rceil - \rho - 1$, поэтому

$$T_\rho^C(z) = T_\rho^{C''}(z).$$

Отсюда следует, что для равномерно упакованных кодов $z \oplus C$ и $z \oplus C''$ (сдвигов кодов C и C'' соответственно на вектор z) первые $\rho + 1$ спектральных значений одинаковы, т. е.

$$\mu^0_{z\oplus C} = \mu^0_{z\oplus C''}, \ldots, \mu^\rho_{z\oplus C} = \mu^\rho_{z\oplus C''}.$$

Тогда согласно лемме 1 коды $z \oplus C$ и $z \oplus C''$ имеют одинаковые весовые спектры. Но поскольку $\mathbf{1} \in z \oplus C$ и $\mathbf{1} \notin z \oplus C''$, имеем $\mu^n_{z\oplus C} \ne \mu^n_{z\oplus C''}$. Полученное противоречие доказывает лемму.

Лемма 3. *Для любого двоичного кода C длины n с кодовым расстоянием $d = 2t + 1$ при любом $i = t, \ldots, n - t$ справедливо неравенство $|C_i| \le \frac{2^t t!}{n^t} \binom{n}{i}$.*

3. Верхняя оценка. Основным результатом работы является

Теорема 1. *Для числа $L_{n,d}$ различных кодов из класса равномерно упакованных кодов $\mathbb{L}(n, d, \rho; \alpha_0, \ldots, \alpha_\rho)$ при нечётном d справедлива оценка $L_{n,d} < 2^{2^{n - \frac{d}{2}\log n + o(\log n)}}$.*

Доказательство. Из леммы 2 следует, что

$$L_{n,d} \le \begin{pmatrix} |E_{\lceil n/2 \rceil - \rho}| + \cdots + |E_{\lfloor n/2 \rfloor + \rho}| \\ |C_{\lceil n/2 \rceil - \rho}| + \cdots + |C_{\lfloor n/2 \rfloor + \rho}| \end{pmatrix}. \tag{3}$$

Имеем $|E_{\lceil n/2\rceil-\rho}| + \cdots + |E_{\lfloor n/2\rfloor+\rho}| \le (2\rho+1)\binom{n}{\lfloor n/2\rfloor}$. По лемме 3 для произвольного двоичного кода C длины n с кодовым расстоянием d выполняется неравенство $|C_{\lceil n/2\rceil-\rho}| + \cdots + |C_{\lfloor n/2\rfloor+\rho}| \le \frac{\lambda}{n^t}\binom{n}{\lfloor n/2\rfloor}$, где $\lambda = (2\rho+1)\cdot 2^t \cdot t!$ и $t = (d-1)/2$. Применяя формулу Стирлинга получаем $\binom{n}{\lfloor n/2\rfloor} \le 2^{n-\frac{1}{2}\log n+2}$. Тогда в силу (3) имеем

$$L_{n,d} < \begin{pmatrix} 2^{n-\frac{1}{2}\log n+(2+\log(2\rho+1))} \\ 2^{n-\frac{d}{2}\log n+(2+\log\lambda)} \end{pmatrix}.$$

Поскольку d и ρ — константы, отсюда и из неравенств $\binom{a}{b} < \left(\frac{3a}{b}\right)^b$ и $c! \le \left(\frac{c+1}{2}\right)^c$ для любых $a > b > 1$, $c \ge 1$, вытекает требуемое неравенство.

Приведём примеры классов двоичных кодов, к которым применима теорема 1.

1) Двоичные *совершенные коды* длины $n = 2^m - 1$ ($m \ge 2$), мощности $2^{n-\log(n+1)}$ с кодовым расстоянием $d = 3$ и параметрами равномерной упаковки $\alpha_0 = \alpha_1 = 1$ (см. [5]). Этот частный случай теоремы 1 был доказан в [1]. Другие примеры равномерно упакованных кодов с $d = 3$ можно найти в [7] (см. также [2] и [8]).

2) Двоичные *коды Препараты* длины $n = 2^m - 1$ ($m \ge 4$ чётно), мощности $2^{n-2\log(n+1)+1}$ с кодовым расстоянием $d = 5$ и параметрами упаковки

$$\alpha_0 = \alpha_1 = 1, \alpha_2 = \alpha_3 = 3/n$$

(см. [5] и [2]).

Следствие 1. *Число различных двоичных кодов Препараты длины n с кодовым расстоянием 5 не превосходит величины $2^{2^{n-\frac{5}{2}\log n+o(\log n)}}$.*

Отметим, что для числа кодов одного специального подкласса кодов Препараты имеет место более точная оценка. А именно, согласно [6, следствие 2], число неэквивалентных четверичных линейных кодов Препараты длины n с кодовым расстоянием 6 не превосходит величины $2^{n\log n}$.

3) Двоичные примитивные *коды типа БЧХ* длины $n = 2^m - 1$ ($m \ge 5$ нечётно), мощности $2^{n-2\log(n+1)}$ с кодовым расстоянием $d = 5$, радиусом покрытия $\rho = 3$ и параметрами

$$\alpha_0 = \alpha_1 = 1, \alpha_2 = \alpha_3 = \frac{6}{n-1}$$

(см. [2]).

4) Двоичные *коды Гетала* (или *коды типа Гетала*) длины $n = 2^m - 1$ ($m \ge 4$ чётно), мощности $2^{n-3\log(n+1)+2}$ с кодовым расстоянием $d = 7$,

радиусом покрытия $\rho = 5$ и параметрами упаковки

$$\alpha_0 = \alpha_1 = 1, \alpha_2 = \alpha_3 = \frac{15}{2n}, \alpha_4 = \alpha_5 = \frac{30}{n(n-3)}$$

(см. [7] и [3]).

Следствие 2. *Число различных двоичных кодов Гетапса длины n с кодовым расстоянием 7 не превосходит величины* $2^{2^{n-\frac{7}{2}\log n + o(\log n)}}$.

5) Двоичные примитивные *коды типа БЧХ* длины $n = 2^m - 1$ ($m \geq 5$ нечётно) мощности $2^{n-3\log(n+1)}$ с кодовым расстоянием $d = 7$, радиусом покрытия $\rho = 5$ и параметрами упаковки

$$\alpha_0 = \alpha_1 = 1, -\alpha_2 = -\alpha_3 = \alpha_4 = \alpha_5 = \frac{120}{(n-1)(n-7)}$$

(см. [7]).

Автор благодарен Д. С. Кротову за ценные замечания, позволившие существенно расширить множество кодов, для которых справедлива теорема. Работа выполнена при финансовой поддержке интеграционного проекта СО РАН № 35 «Древовидный каталог математических Интернет-ресурсов» и Российского фонда фундаментальных исследований (проект 07-01-00248).

Список литературы

1. Августинович С. В. Об одном свойстве совершенных двоичных кодов // Дискрет. анализ и исслед. операций. 1995. Т. 2, № 1. С. 4–6.

2. Бассалыго Л. А., Зиновьев В. А., Зайцев Г. В. О равномерно упакованных кодах // Проблемы передачи информации. 1974. Т. 10, вып. 1. С. 9–14.

3. Зиновьев В. А., Хеллесет Т. О весовых спектрах сдвигов кодов типа Гетапса // Проблемы передачи информации. 2004. Т. 40, вып. 2. С. 19–36.

4. Мак-Вильямс Ф. Дж., Слоэн Н. Дж. А. Теория кодов, исправляющих ошибки, М: Связь, 1979.

5. Семаков Н. В., Зиновьев В. А., Зайцев Г. В. Равномерно упакованные коды // Проблемы передачи информации. 1971. Т. 7, вып. 1. С. 38–50.

6. Токарева Н. Н. Представление $\mathbb{Z}_4$-линейных кодов Препараты с помощью векторных полей // Проблемы передачи информации. 2005. Т. 41, вып. 2. С. 50–62.

7. Goethals J. M., Van Tilborg H. C. A. Uniformly packed codes // Philips Res. Repts. 1975. V. 30. P. 9–36.

8. Rifa J., Zinoviev V. A. On completely regular codes from perfect codes // Proc. Tenth Int. Workshop "Algebraic and Combinatorial Coding Theory", Zvenigorod, Russia, P. 225–229, September, 3–9, 2006.

СЛОЖНОСТЬ ПРОБЛЕМЫ ВЫПОЛНИМОСТИ ДЛЯ ОДНОГО ЯЗЫКА С ФУНКЦИОНАЛЬНЫМИ БУЛЕВЫМИ ПЕРЕМЕННЫМИ

В. С. Федорова (Москва)

Пусть $X = \{x_1, x_2, \ldots\}$ — множество индивидных переменных, $F = \{f_1^{(n_1)}, f_2^{(n_2)}, \ldots\}$, где n_i, $i = 1, 2, \ldots$, — натуральные числа, есть множество функциональных переменных, $C = \{\&, \vee, \neg\}$ — множество функциональных констант. Будем рассматривать язык L, содержащий индивидные переменные X, функциональные переменные F, функциональные константы C, скобки и запятую.

Определим синтаксис языка L. Назовем термом всякое слово в языке L, удовлетворяющее следующим условиям:

1. Если x принадлежит множеству индивидных переменных X, то x является термом.

2. Если $f_i^{(n)}$ принадлежит множеству функциональных переменных F, а $t_1, t_2, \ldots, t_n$ — термы, то $f_i^{(n)}(t_1, t_2, \ldots, t_n)$ является термом.

3. Если t_1, t_2 — термы, то $t_1 \& t_2$, $t_1 \vee t_2$, $\overline{t_1}$ — также термы.

Если t_1, t_2 — термы, то $t_1 = t_2$ есть равенство. Пусть T — конечная система равенств. Будем говорить, что данные значения всех функциональных переменных, входящих в систему T, выполняют эту систему, если все равенства системы верны при этих значениях функциональных переменных и всех значениях индивидных переменных, входящих в систему T. Конечная система равенств T выполнима, если на множестве всех булевых функций P_2 существуют значения всех функциональных переменных, которые выполняют систему T.

Для языка L можно сформулировать следующую проблему: по произвольной конечной системе равенств T выяснить, является ли T выполнимой. В данной работе получены верхняя и нижняя оценки временной сложности решения этой проблемы.

Получим верхнюю оценку.

Условимся, что все индивидные переменные, участвующие в системе равенств T, занумерованы по возрастанию без пропусков, то есть если в системе T используется индивидная переменная x_i, то в T найдется и переменная x_j, где $1 \leqslant j < i$.

Пусть система T состоит из t равенств, в которых участвуют m различных функциональных переменных

$$f_1^{(n_1)},\ f_2^{(n_2)}, \ldots, f_m^{(n_m)}.$$

Тогда справедлива следующая

Теорема 1. *Существует алгоритм, проверяющий выполнимость системы T за время, не превосходящее по порядку*

$$t \cdot l^3 \cdot 2^l \cdot \left(2^{2^l}\right)^m,$$

где l — длина входа данного алгоритма.

Доказательство. Построим такой алгоритм в классе одноленточных машин Тьюринга с алфавитом $\Psi = \{0, 1, *, \#, e, f, \Lambda\}$. Представим систему равенств T как слово длины l в алфавите Ψ. Для этого закодируем индивидные переменные x_i, $i = 1, 2, \ldots$, их номерами i в двоичной системе счисления; функциональные переменные — равномерным двоичным кодом длины $[\log_2 m] + 1$ с добавлением в начало каждого кода символа $f \in \Psi$; функциональные константы $\&, \vee, \neg$ — соответственно наборами символов $\#, \#\#, \#\#\#$; все запятые и закрывающиеся скобки — символом $*$; символ $=$ заменим e, а между различными равенствами системы T поставим ee. Легко заметить, что общее число индивидных переменных строго меньше l.

Искомая машина Тьюринга перебирает все возможные сочетания (возможно, с повторениями) из m булевых функций, зависящих соответственно от $n_1, n_2, \ldots, n_m$ переменных, и для каждого такого набора — все возможные 2^l значений индивидных переменных (в худшем случае). Теперь для того, чтобы проверить каждое из t равенств системы T, машина Тьюринга копирует код системы T правее на ленту (это займет порядка l^2 тактов) и начинает вычисление, заменяя коды индивидных переменных их запомненными значениями и сдвигая после каждой такой замены правый остаток кода влево за линейное по l число тактов, чтобы избежать разрывов кода. Всего замен может быть не больше l. При этом, если все аргументы функции являются константами, то ее код заменяется на ее значение. Для вычисления всех правых и левых частей равенств системы T потребуется не больше l проходов по скопированному коду. Таким образом, время работы машины Тьюринга не превосходит по порядку

$$\left(2^{2^l}\right)^m \cdot 2^l \cdot (l^2 + t \cdot l^3),$$

что и требовалось доказать.

Для получения нижней оценки временной сложности решения поставленной проблемы будут использоваться конечные однородные структуры. По произвольной конечной однородной структуре (далее — ОС) будет построена система равенств, принадлежащая описанному выше классу. Тогда временная сложность проверки этой системы равенств с использованием ОС и будет являться искомой нижней оценкой.

Введем необходимые понятия. Пусть $A = (Q, q)$ — конечный автомат с множеством состояний $Q = \{q_0, q_1, \ldots, q_{k-1}\}$ и функцией переходов $g : Q^3 \to Q$ (автомат A имеет два входа и два выхода). Для любого натурального числа m через M_m обозначим линейно упорядоченную последовательность из m копий $A_1, A_2, \ldots, A_m$ автомата A, в которой каждый автомат A_i, $1 < i < m$, связан с автоматами A_{i-1} и A_{i+1}. Автоматы A_1 и A_m связаны соответственно только с автоматами A_2 и A_{m-1}. ОС M_m работает в дискретном времени $t = 1, 2, \ldots$ В каждый момент времени $t + 1$ состояние автомата A_i, $1 < i < m$, определяется с помощью функции g состояниями автоматов A_{i-1}, A_i, A_{i+1} в момент времени t. Будем считать, что при вычислении состояний автоматов A_1 и A_m вместо соответственно первого и третьего аргументов в функцию g всегда подставляются значения q_1 и q_2 из Q соответственно.

Согласно приведенным определениям функционирование ОС M_m происходит следующим образом. В начальный момент времени автоматы $A_1, A_2, \ldots, A_m$ устанавливаются в некоторые состояния $q_{i_1}, q_{i_2}, \ldots, q_{i_m}$. Назовем этот набор состояний *инициальным*. В следующий момент времени вектор-состоянием (или конфигурацией) ОС M_m будет набор

$$(g(q_1, q_{i_1}, q_{i_2}), g(q_{i_1}, q_{i_2}, q_{i_3}), \ldots,$$
$$g(q_{i_{m-2}}, q_{i_{m-1}}, q_{i_m}), g(q_{i_{m-1}}, q_{i_m}, q_2)).$$

Затем к полученным состояниям вновь применяется функция g и так далее.

Выделим состояние $q_0 \in Q$ и назовем его заключительным. Также наложим ограничения на функцию переходов g: $g(q_0, q_i, q_j) = q_0$, $g(q_i, q_0, q_j) = q_0$, $g(q_i, q_j, q_0) = q_0$, где $q_i, q_j \in Q$. Тогда если все автоматы ОС M_m придут в заключительное состояние q_0, то в дальнейшем с течением времени конфигурация ОС M_m не поменяется. В этом случае будем считать, что ОС M_m закончила работу, а такую конфигурацию назовем *заключительной*.

Назовем функционирование ОС M_m при заданном инициальном наборе состояний $q_{i_1}, q_{i_2}, \ldots, q_{i_m}$ *правильным*, если ОС преобразовывает этот набор состояний в заключительную конфигурацию. Очевидно, что для того, чтобы выяснить, является ли функционирование данной ОС при заданном инициальном наборе состояний правильным, достаточно проверить лишь первые k^m тактов.

Пусть $Q_I = (q_{i_1}, q_{i_2}, \ldots, q_{i_m})$, $q_{i_j} \in Q$, $j = 1, 2, \ldots, m$, — произвольное инициальное вектор-состояние ОС M_m. Обозначим через $\Pi(A, m, Q_I)$ следующую проблему: функционирует ли ОС, составленная из m копий автомата A, правильно при инициальной конфигурации Q_I.

Теорема 2. *Существует алгоритм, сводящий проблему* $\Pi(A, m, Q_I)$ *к проблеме выполнимости некоторой системы равенств* T *за время порядка* m^2 *так, что система равенств* T *выполнима тогда и только тогда, когда*

ОС, составленная из m копий автомата A, функционирует правильно при инициальном вектор-состоянии Q_I.

Доказательство. Закодируем состояния $q_0, q_1, \ldots, q_{k-1}$ автомата A двоичными наборами длины $l = [\log_2 k]+1$ так, чтобы код заключительного состояния q_0 являлся единичным булевым вектором, и построим по функции переходов g соответствующие булевы $(2l, l)-$ и $(3l, l)-$операторы G_1, G_2, G_3 следующим образом:

1. Если наборы $(a_1, a_2, \ldots, a_l)$, $(b_1, b_2, \ldots, b_l)$, $(c_1, c_2, \ldots, c_l)$ суть коды состояний q_a, q_b, q_c автомата A, $g(q_a, q_b, q_c) = q_d$ и набор $(d_1, d_2, \ldots, d_l)$ является кодом состояния q_d, то положим

$$G_2(a_1, a_2, \ldots, a_l, b_1, b_2, \ldots, b_l, c_1, c_2, \ldots, c_l) = \\ = (d_1, d_2, \ldots, d_l).$$

На остальных двоичных наборах длины $3l$ (если они есть) оператор G_2 определяется произвольным образом.

2. Если набор $(e_1, e_2, \ldots, e_l)$ кодирует состояние q_1, то в соответствии с соглашением о функционировании автомата A_1 в ОС M_m оператор G_1 задается следующим образом:

$$G_1(x_1, x_2, \ldots, x_{2l}) = G_2(e_1, e_2, \ldots, e_l, x_1, x_2, \ldots, x_{2l}).$$

3. Если набор $(e'_1, e'_2, \ldots, e'_l)$ кодирует состояние q_2, то аналогично получаем

$$G_3(x_1, x_2, \ldots, x_{2l}) = G_2(x_1, x_2, \ldots, x_{2l}, e'_1, e'_2, \ldots, e'_l).$$

Назовем объединение кодов m состояний автоматов $A_1, A_2, \ldots, A_m$ кодом соответствующей конфигурации ОС M_m. Пусть B_1 — булев вектор длины lm, кодирующий некоторый инициальный набор состояний. Тогда при функционировании ОС M_m под действием операторов G_1, G_2, G_3 вектор B_1 будет преобразовываться с течением времени в вектора $B_2, B_3, \ldots, B_{2^{lm}}$. Поскольку различных булевых векторов длины lm ровно 2^{lm}, дальнейшие преобразования будут повторением приведенных выше векторов или их части. Объединим вектора $B_1, B_2, \ldots, B_{2^{lm}}$ в один вектор B длины $lm \cdot 2^{lm}$.

Для упрощения изложения пусть числа l и m являются степенями двойки: $l = 2^{l_1}$, $m = 2^{m_1}$, где l_1, m_1 — целые. В этом случае длина вектора B есть $lm \cdot 2^{lm} = 2^{lm+l_1+m_1}$, и его можно рассматривать как вектор-столбец значений некоторой булевой функции f, зависящей от $n = lm + l_1 + m_1$ переменных.

Построим систему равенств, описывающую вычисление всех значений функции f:

1. Вектор, состоящий из первых lm значений функции f, расположенных в лексикографическом порядке, есть в точности вектор B_1.

2. Поделим вектор значений функции f на блоки длины lm. Тогда любые два соседних блока суть коды двух последовательных конфигураций ОС M_m.

3. Последний блок вектора значений функции f есть в точности код заключительной конфигурации ОС M_m.

Выпишем соответствующие равенства. Здесь $I_i^l(x_1, x_2, \ldots, x_l) = x_i$ — селекторная функция, $1 \leqslant i \leqslant l$, код заключительной конфигурации Q_0 есть единичный булев вектор. Также во всех равенствах необходимо заменить константы 0 и 1 соответственно на термы $x_1 \& \overline{x_1}$ и $x_1 \vee \overline{x_1}$, а функцию эквивалентности $x \sim y$ и импликацию $x \to y$ разложить по системе функциональных констант $\{\&, \vee, \neg\}$ следующим образом: $x \sim y = x\&y \vee \overline{x}\&\overline{y}$, $x \to y = x\&y \vee \overline{x}$.

1. $f(0,0,\ldots,0,0) = I_1^{lm}(B_1)$
   $f(0,0,\ldots,0,1) = I_2^{lm}(B_1)$
   $\ldots$
   $f(0,\ldots,0,\underbrace{1,\ldots,1}_{l_1+m_1}) = I_{lm}^{lm}(B_1)$

3. $f(1,\ldots,1,\underbrace{0,\ldots,0,0}_{l_1+m_1}) = 1$
   $f(1,\ldots,1,\underbrace{0,\ldots,0,1}_{l_1+m_1}) = 1$
   $\ldots$
   $f(1,1,\ldots,1) = 1$

2. Пусть наборы индивидных переменных $(x_1, x_2, \ldots, x_{lm})$ и $(y_1, y_2, \ldots, y_{lm})$ задают номера конфигураций ОС M_m, причем $(x_1, x_2, \ldots, x_{lm})_2$ есть номер не заключительной конфигурации. Это утверждение представляется термом

$$T_1 = (x_1 \& x_2 \& \cdots \& x_{lm}) \sim 0.$$

Также пусть $(y_1, y_2, \ldots, y_{lm})_2$ — номер конфигурации, непосредственно следующий за номером конфигурации $(x_1, x_2, \ldots, x_{lm})_2$, т. е. $(y_1, y_2, \ldots, y_{lm})_2 = (x_1, x_2, \ldots, x_{lm})_2 \oplus 1$ (сложение по модулю 2). Это описывает терм

$$T_2 = \Big(y_{lm} \sim \overline{x_{lm}}\Big)\& \cdots \&\Big(y_i \sim (\overline{x_i}\& x_{i+1}\& \cdots \& x_{lm} \vee x_i \& \overline{x_{i+1}\& \cdots \& x_{lm}})\Big)\&$$

$$\& \cdots \&\Big(y_1 \sim (\overline{x_1}\& x_2 \& \cdots \& x_{lm} \vee x_1 \& \overline{x_2 \& \cdots \& x_{lm}})\Big).$$

Тогда код состояния крайнего левого автомата A_1 ОС M_m есть значение булева $(2l, l)$-оператора G_1, взятое от кодов состояний автоматов A_1 и A_2 в предыдущий момент времени.

$$T_3 = \Big(f(y_1, \ldots, y_{lm}, \underbrace{0, \ldots, 0}_{l_1+m_1}) \sim$$

$$\sim I_1^l(G_1(f(x_1,\ldots,x_{lm},\underbrace{0,\ldots,0}_{l_1+m_1}),\ldots,f(x_1,\ldots,x_{lm},\underbrace{0,\ldots,0,1}_{m_1},\underbrace{1,\ldots,1}_{l_1})))\Big)\&$$

$$\&\cdots\&\Big(f(y_1,\ldots,y_{lm},\underbrace{0,\ldots,0}_{m_1},\underbrace{1,\ldots,1}_{l_1})\sim$$

$$\sim I_l^l(G_1(f(x_1,\ldots,x_{lm},\underbrace{0,\ldots,0}_{l_1+m_1}),\ldots,f(x_1,\ldots,x_{lm},\underbrace{0,\ldots,0,1}_{m_1},\underbrace{1,\ldots,1}_{l_1})))\Big).$$

Аналогично для крайнего правого автомата A_m ОС M_m:

$$T_4=\Big(f(y_1,\ldots,y_{lm},\underbrace{1,\ldots,1}_{m_1},\underbrace{0,\ldots,0}_{l_1})\sim$$

$$\sim I_1^l(G_3(f(x_1,\ldots,x_{lm},\underbrace{1,\ldots,1,0}_{m_1},\underbrace{0,\ldots,0}_{l_1}),\ldots,f(x_1,\ldots,x_{lm},\underbrace{1,\ldots,1}_{l_1+m_1})))\Big)\&$$

$$\&\cdots\&\Big(f(y_1,\ldots,y_{lm},\underbrace{1,\ldots,1}_{l_1+m_1})\sim$$

$$\sim I_l^l(G_3(f(x_1,\ldots,x_{lm},\underbrace{1,\ldots,1,0}_{m_1},\underbrace{0,\ldots,0}_{l_1}),\ldots,f(x_1,\ldots,x_{lm},\underbrace{1,\ldots,1}_{l_1+m_1})))\Big).$$

Вышесказанное суммирует равенство

$$(T_1\&T_2)\to(T_3\&T_4)=1.$$

Для наборов переменных $(z_1^1,\ldots,z_{m_1}^1)$, $(z_1^2,\ldots,z_{m_1}^2)$ и $(z_1^3,\ldots,z_{m_1}^3)$ аналогичным образом выписываются термы T_5 и T_6, показывающие, что

$$(z_1^2,\ldots,z_{m_1}^2)_2=(z_1^1,\ldots,z_{m_1}^1)_2\oplus 1,\quad (z_1^3,\ldots,z_{m_1}^3)_2=(z_1^2,\ldots,z_{m_1}^2)_2\oplus 1.$$

Тогда терм T_7, утверждающий, что в конфигурации с номером $(y_1,...,y_{lm})$ код любого не крайнего автомата получается из кодов соответствующих трех автоматов в предыдущий момент времени применением $(3l,l)$-оператора G_2, выглядит следующим образом:

$$T_7=\Big(f(y_1,\ldots,y_{lm},z_1^2,\ldots,z_{m_1}^2,\underbrace{0,\ldots,0}_{l_1})\sim$$

$$\sim I_1^l(G_2(f(x_1,\ldots,x_{lm},z_1^1,\ldots,z_{m_1}^1,\underbrace{0,\ldots,0}_{l_1}),\ldots,$$

$$\ldots,f(x_1,\ldots,x_{lm},z_1^3,\ldots,z_{m_1}^3,\underbrace{1,\ldots,1}_{l_1})))\Big)\&$$

$$\& \cdots \& \Big(f(y_1, \ldots, y_{lm}, z_1^2, \ldots, z_{m_1}^2, \underbrace{1, \ldots, 1}_{l_1}) \sim$$

$$\sim I_l^l(G_2(f(x_1, \ldots, x_{lm}, z_1^1, \ldots, z_{m_1}^1, \underbrace{0, \ldots, 0}_{l_1}), \ldots,$$

$$\ldots, f(x_1, \ldots, x_{lm}, z_1^3, \ldots, z_{m_1}^3, \underbrace{1, \ldots, 1}_{l_1}))) \Big).$$

Объединяя, получаем равенство

$$(T_1 \& T_2 \& T_5 \& T_6) \to T_7 = 1.$$

Назовем системой равенств T объединение $2lm + 2$ равенств из пунктов 1, 2, 3.

Длина полученной системы равенств T (число всех символов) есть m^2 по порядку. Таким образом, алгоритм сводит проблему $\Pi(A, m, Q_I)$ к проблеме выполнимости системы T за время порядка m^2, поскольку равномерное кодирование состояний автомата A может быть осуществлено эффективно с линейной сложностью.

Следствие 1. *Нижняя оценка временной сложности решения проблемы выполнимости системы равенств длины l по порядку не меньше $d^{\sqrt{l}}$, $d > 1$.*

Доказательство. Сложность решения проблемы $\Pi(A, m, Q_I)$ по порядку логарифма совпадает со сложностью вычисления функций на одноленточных машинах Тьюринга, работающих с линейной зоной [3]. Таким образом, проблему $\Pi(A, m, Q_I)$ нельзя решить за время, по порядку меньшее, чем d^m, $d > 1$, то есть существенно проще непосредственного перебора. Отсюда с учетом теоремы 2 получаем требуемое утверждение.

Список литературы

1. Марченков С. С. Итерация булевых $(n, n)-$операторов. Вестник Московского Университета. Серия 15. Вычислительная математика и кибернетика. 2006. № 4, стр. 36–41.

2. Кудрявцев В. Б., Подколзин А. С., Болотов А. А. Основы теории однородных структур. Москва. Наука. 1990.

3. Катериночкина Н. Н. Об эквивалентности некоторых вычислительных устройств. Кибернетика. 1970. №5, стр. 27–31.

ПРЕДСТАВИМОСТЬ ЯЗЫКОВ ДВУХСТОРОННИМИ АВТОМАТАМИ

К. Р. Хадиев (Казань)

1. Описание техники

Опишем технику нахождения количества классов эквивалентности для вероятностных автоматов, описанную в статье Anne Condon, Bounded Error Probabilistic Finite State Automata. Далее распространим ее на другие виды автоматов, рассматривая их как частные случаи или расширения вероятностных. **Граф, связанный с автоматом.** Рассмотрим двусторонний вероятностный автомат A, который работает на слове xy и имеет c состояний. Сейчас мы произвольно разбиваем слово на две части, однако в последствии это разбиение будет сознательным. Опишем поведение автомата на границе между словами x и y. Для этого сконструируем граф специального вида.

Рассмотрим граф, обозначим его $M[x]$, — двудольный, в первую долю поместим вершины типа $(q, 1)$, которые будут соответствовать конфигурациям автомата A, когда он находится в состоянии q и читающая головка считывает последний символ слова x. Во вторую долю поместим вершины типа $(q', 2)$, которые будут соответствовать конфигурациям автомата A, когда он находится в состоянии q' и читающая головка считывает первый символ слова y. Добавим дуги из вершин первой доли в вершины второй доли. На дуге, ведущей из вершины $(q, 1)$ в вершину $(q', 2)$ напишем вес p, если p — это вероятность того, что детерминировано начиная работу из состояния q и обозревая последний символ слова x, автомат будет как-то двигаться лишь по слову x и при первом пересечении границы он окажется именно в состоянии q', а значит в конфигурации, соответствующей вершине $(q', 2)$. Добавим в первую долю Вершину $Init$, соответствующую начальной конфигурации автомата A. Из нее будут исходить дуги во вторую долю по тому же принципу: на дуге в вершину $(q', 2)$ напишем вес p, если p — это вероятность того, что детерминировано начиная работу из начальной конфигурации, автомат будет как-то двигаться лишь по слову x и при первом пересечении границы он окажется именно в состоянии q'. Заметим, что все дуги этого графа зависят лишь от слова x, т. к. описывают поведения автомата именно на этой части входного слова.

Рассмотрим еще один граф построенный аналогичным образом, обозначим его через $M[y]$. В первую долю поместим вершины типа $(q, 1)$, которые будут соответствовать конфигурациям автомата A, когда он находится в состоянии q и читающая головка считывает последний символ слова x. Во

вторую долю поместим вершины типа $(q', 2)$, которые будут соответствовать конфигурациям автомата A, когда он находится в состоянии q' и читающая головка считывает первый символ слова y. Добавим дуги из вершин второй доли в вершины вида $(*, 1)$ первой доли. На дуге, ведущей из вершины $(q', 2)$ в вершину $(q, 1)$ напишем вес p, если p — это вероятность того, что детерминировано начиная работу из состояния q' и обозревая первый символ слова y, автомат будет как-то двигаться лишь по слову y и при первом пересечении границы он окажется именно в состоянии q, а значит в конфигурации, соответствующей вершине $(q, 1)$. Добавим в первую долю вершины *Accept* и *Reject*, соответствующие конфигурациям принятия и отвержения автомата A, соответственно. В них будут входить дуги из второй доли по тому же принципу: на дуге из вершину $(q', 2)$, например, в вершину *Accept* напишем вес p, если p — это вероятность того, что детерминировано начиная работу, обозревая первый символ y и находясь в состоянии q', автомат достигнет принимающей конфигурации, двигаясь лишь по слову y. Для *Reject* аналогично. Также для вершин *Accept* и *Reject* добавим петли с весом 1. Заметим, что все дуги этого графа зависят лишь от слова y, т. к. описывают поведения автомата именно на этой части входного слова.

Объединим эти два графа в граф $M[x, y]$. Полученный граф будет иметь $2 * c + 3$ вершин: в первой доле c вершин, соответствующих каждому из состояний, а также вершины $Init$, *Accept* и *Reject*, во второй доле c, для каждого из состояний. Множество дуг графа $M[x, y]$ — это объединение дуг графов $M[x]$ и $M[y]$. Такой граф полностью описывает поведение автомата на слове xy. Пронумеруем вершины от 1 до $2c + 3$, начиная с первой доли и продолжая второй и составим матрицу весов M для графа $M[x, y]$, она будет иметь следующий вид:

$$\left(\begin{array}{c|c} 0 & x \\ \hline y & 0 \end{array}\right).$$

Часть матрицы, помеченная буквой x зависит лишь от слова x, часть y — от y. Каждая из строк является вектором распределения вероятностей перехода из данной конфигурации в во все остальные (представленные в графе).

2. Цепь Маркова

Если мы запишем в ряд те вершины, которые соответствуют конфигурациям, посещенным автоматом при чтении слова xy, то это будет след вычислений $K_0K_1K_2...K_M$, где K_0=$Init$, а K_M — это терминальная конфигурация, а именно *Accept* или *Reject*. Причем Каждая K_i — это случайная

величина, зависящая от K_{i-1}. Этот процесс называется цепью Маркова. Поведение цепи Маркова задает матрица смежности графа $M[x, y]$. Обозначим за $a(xy)$ вероятность того, что цепь Маркова окончится принимающим состоянием, т. е. K_M=*Accept*.

О мере близости. Обычная мера близости не подойдет для двусторонних автоматов, здесь введем новую меру.

Определение. *Назовем два числа p и p' β-близкими, для $\beta \geq 1$, если для них выполнится одно из следующих соотношений:$p = p' = 0$ или $p \neq 0$, $p' \neq 0$ и $\beta^{-1} \leq \frac{p}{p'} \leq \beta$ или, что то же самое $|\log(p) - \log(p')| \leq \beta$*

Определим аналогичную меру близости для матриц.

Определение. *Назовем две матрицы $[p_{ij}]$ и $[p'_{ij}]$ β-близкими, при $\beta \geq 1$, для каждого i и j p_{ij} и p'_{ij} β-близки.*

Из теории цепей Маркова известно, что если матрицы весов для графов $M[x, y]$ и $M[x', y]$ β-близки и имеют по m вершин, то величины $a(x'y)$ и $a(xy)$ β^{2m}-близки.

3. Двусторонние конечные детерминированные автоматы могут распознать лишь регулярные языки

Рассмотрим слово xy ему соответствует цепь Маркова M с матрицей вероятностей переходов $[p_{ij}]$. Если эта матрица строится для 2-КДА, то элементами матрицы могут быть лишь 0 или 1. В каждой строке сумма элементов должна быть равна 1, значит в каждой строке находиться одна и только одна 1.

Лемма 1. *Если два слова x и x' — ε-близки тогда и только тогда, когда матрицы весов соответствующих им графов $M[x]$ и $M[x']$, построенных по 2-КДА, совпадают.*

Определение. *Класс языков 2DFA это все языки, для которых существует 2-х сторонний КДА (2-КДА) их распознающий.*

Определение. *Пусть есть P — 2-КДА, которому соответствуют некоторый язык $L(P)$. Два слова x и x', будем называть эквивалентными относительно языка $L(P)$, если для любого слова y P принимает xy, тогда и только тогда, когда принимает слово $x'y$.*

Теорема 1. *2DFA = Reg.*

Доказательство. Рассмотрим два слова x и x', и соответствующие им $M[x]$ и $M[x']$. Если мы возьмем любое слово y, то поведение автомата на словах xy и $x'y$ полностью определяется матрицами смежности $M[x, y]$ и

$M[x', y]$. Если они совпадают, то x и x' эквивалентны. значит количество классов эквивалентности может быть не больше количества различных матриц смежности $M[x]$(ведь именно этот граф определяет элементы, графа $M[x, y]$, зависящие от x). Таких матриц конечное число, ведь в каждой ячейке могут стоять лишь 0 или 1, и размерность матрицы конечна.

Раз количество классов эквивалентности конечно, то язык, определяемый каждым из 2-КДА регулярен.

Свойство. *Если c — количество состояний 2-КДА, то c^{c+1} — количество состояний 1-КДА.*

Доказательство. В каждой строке матрицы весов графа $M[x]$ стоит только одна 1. Ее возможных положений всего c, т. к. столько вершин во второй доли, а словом x определяются дуги, которые идут именно в эту долю. Дуги из $c+1$ вершин определяются словом x($Init$ и c — конфигурации на конце слова x), значит различных матриц, может быть всего c^{c+1}.

4. Двусторонние конечные недетерминированные автоматы могут распознать лишь регулярные языки

Определение. *Класс языков 2NFA — это все языки, для которых существует 2-х сторонний КНА (2-КНА) их распознающий.*

Определение. *Класс языков $2PFA_0$ — это все языки, для которых существует 2-х сторонний КВА с точкой сечения 0 (2-КВА$_0$) их распознающий.*

Лемма 2. *$2NFA = 2PFA_0$.*

Доказательство. Возьмем произвольное слово x и 2-КДА A. По A построим 2-КВА$_0$ A', просто на каждом шаге выбирая путь с помощью датчика случайных символов. Если слово x принимается автоматом A, значит есть путь в состояние принятия, следовательно вероятность принятия слова x автоматом A' $P_{Accept}(x) > 0$ и оно примется автоматом A'.

Если слово x не принимается автоматом A, значит нет пути в состояние принятия, следовательно вероятность принятия слова x автоматом A' $P_{Accept}(x) = 0$ и оно не примется автоматом A'. И наоборот.

Теорема 2. *$2NFA = Reg$.*

Доказательство. Докажем, что $2PFA_0 = \mathrm{Reg}$, тогда из леммы будет следовать утверждение теоремы. То, что $Reg \in 2PFA_0$ очевидно, т.к.2-КДА — частный случай 2-КВА$_0$. Докажем, что $2PFA_0 \in Reg$.

Все вероятности переходов лежат в интервале $[2^{-cn}, 1]$, а значит логарифмы от вероятностей в интервале $[-cn, 0]$. Разобьем отрезок $[-cn, 0]$ на

подинтервалы длинны ε. Два слова x и x' ε-близки, если 2^{ε}-близки соответствующие $M[x]$ и $M[x']$, а значит логарифмы их вероятностей $\log(p)$ и $\log(p')$ попадают в один подинтервал. Тогда 2^{ε}-близки $M[x,y]$ и $M[x',y]$, а вероятности достижения вершины принятия $a(xy)$ и $a(x'y)$ $2^{2(2+3)\varepsilon}$-близки. Это означает что,

$$\frac{a(xy)}{a(x'y)} \geq 2^{-2(2+3)\varepsilon}.$$

Если $x'y$ принимается, то $a(x'y) > 0$, значит

$$a(xy) \geq 2^{-2(2c+3)\varepsilon} a(x'y) > 0,$$

причем это равенство выполняется для любого ε. Отсюда следует, что любые два слова x и x' такие, что для соответствующих им $M[x]$ и $M[x']$ логарифмы их вероятностей $\log(p)$ и $\log(p')$ из $[-cn, 0]$, то эти слова эквивалентны. Значит не эквивалентными могут быть только ε не сравнимые слова, у которых матрицы весов графов $M[x]$ и $M[x']$ не совпадают по позициям нулевых компонент, а таких различных матриц конечное количество, ввиду конечности размеров матрицы весов.

Свойство. *Если c — количество состояний 2-КНА, то $2^{c(c+1)}$ — количество состояний 1-КДА, а значит различных классов эквивалентности не может превышать это число.*

Доказательство. В каждой строке матрицы весов графа $M[x]$ могут стоять 1 или 0. их можно расположить 2^c способами, т.к. столько вершин во второй доли, а словом x определяются дуги, которые идут именно в эту долю. Дуги из $+1$ вершин определяются словом x($Init$ и c — конфигурации на конце слова x), значит различных матриц, может быть всего $(2^{c+1})^c = 2^{c(c+1)}$.

5. Двусторонние конечные вероятностные автоматы могут распознать нерегулярные языки

Более подробно можно посмотреть информацию в статье Anne Condon, Bounded Error Probabilistic Finite State Automata. Приведем лишь результаты.

Количество классов эквивалентности растет как полином n^k, где n — длина слова x, а k — константа, зависящая лишь от c — количества состояний автомата и δ — на сколько изолирована точка сечения.

Языки, классы эквивалентности которых растут быстрее, не распознаются 2-КВА. Например, не распознаются языки с суперполиномиальным $n^{\log n}$ ростом числа классов.

6. Двусторонние конечные квантовые автоматы

Рассмотрим модель 2-qfa(2-ККА), с изолированной точкой сечения. Особенность этой модели в том, что количество различных состояний зависит от n — длинны слова. Пусть $c = dn$, где $d = const$. В вероятностном случае i-я строка матрицы, задающей поведение автомата, была распределением вероятностей попадания в соответствующую конфигурацию из i-й. Здесь вероятность попадания в соответствующую конфигурацию — это вектор из квадратов амплитуд, которые будут для каждого состояния при пересечении границы. А значит, говоря о близости слов, надо говорить о близости матриц, составленных из квадратов амплитуд.

Определим в каких приделах лежат элементы полученной матрицы. Любой двухсторонний вероятностный автомат с рациональными вероятностями, можно представить в виде автомата, все вероятности в котором из множества $\{0, 0.5, 1\}$. Это достигается представлением вероятности в двоичном виде и используя это организуются вероятности переходов. Здесь можно применить тот же способ. Каждый элемент матрицы переходов в квантовом автомате представим в двоичном виде. А так как $2^{-1} = (\sqrt{2})^{-2}$, $2^{-2} = (\sqrt{2})^{-4}$, $2^{-3} = (\sqrt{2})^{-6}$ и т. д., то каждое рациональное число можно разложить по степеням $\sqrt{2}$, т. е. можно рассматривать автомат, элементы матрицы переходов которого из $\{-1, -\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}, 1\}$. Тогда минимальное значение амплитуды по абсолютному значению будет, $2^{-nc/2}$, а квадрата амплитуды 2^{-nc}. Значит логарифмы всех не нулевых элементов матриц квадратов снизу ограничено величиной $-nc$, т. е. $\log p \in [-nc, 0]$, тогда длинна такого отрезка dn^2. Матрицы, которые соответствуют измерениям, не могут увеличить этот отрезок, т. к. после их применения, одни из амплитуд увеличиваются, а другие становятся равными 0.

Теорема 3. *Количество классов эквивалентности в языках, распознаваемых 2-ККА, растет как* $(\frac{d^2n^3}{\gamma} + 1)^{dn(dn+1)}$, *где* γ — *константа.*

Доказательство. Разобьем отрезок $[-cn, 0]$ на подинтервалы длинны ε. Два слова x и x' ε-близки, если 2^{ε}-близки соответствующие $M[x]$ и $M[x']$, а значит логарифмы их вероятностей $\log(p)$ и $\log(p')$ попадают в один подинтервал. Тогда 2^{ε}-близки $M[x, y]$ и $M[x', y]$, а вероятности достижения вершины принятия $a(xy)$ и $a(x'y)$ $2^{2(2+3)\varepsilon}$-близки. Это означает что,

$$\frac{a(xy)}{a(x'y)} \geq 2^{-2(2+3)\varepsilon}.$$

Если $x'y$ принимается, то $a(x'y) > \frac{1}{2} + \delta$, значит и $a(x'y) > \frac{1}{2} + \delta + \delta_1$ для некоторого не нулевого δ_1, тогда справедливо следующее

$$a(xy) \geq 2^{-2(2c+3)\varepsilon} a(x'y) > 2^{-2(2c+3)\varepsilon} (\frac{1}{2} + \delta + \delta_1).$$

Подберем ε так, чтобы слово xy принималось, т. е. $a(xy) > \frac{1}{2} + \delta$. Тогда

$$2^{-2(2c+3)\varepsilon}(\frac{1}{2} + \delta + \delta_1) > \frac{1}{2} + \delta.$$

Откуда

$$\varepsilon < -\log_2 \frac{\frac{1}{2} + \delta}{\frac{1}{2} + \delta + \delta_1} \Big/ 2(2c+3).$$

Можно сказать, что асимптотически $\varepsilon < \frac{\gamma}{c}$.

Заменим в матрицах графов $M[x]$ и $M[x']$ не нулевые вероятности на номера ε-интервалов, в которые они попадают, тогда ε-близость слов эквивалентно равенству таких матриц. В каждой ячейки могут быть числа от 0 до cn/ε, таких позиций $c(c+1)$, а значит различных матриц $(\frac{d^2n^3}{\gamma} + 1)^{dn(dn+1)} = 2^{dn(dn+1)(3\log n + \gamma_1)}$.

Рассмотрим случай, когда время работы автомата ограничивается полиномом $t(n)$. В этом случае все элементы матрицы весов графа M[x], меньшие $t(n)^{-2}$ можно заменить на 0. В худшем случае через границу автомат пройдет $t(n)$ раз, а значит в вероятность принятия такие элементы внесут вклад не больше, чем $t(n)^{-1}$, на это число можно уменьшить ошибку и тогда они не будут ни как влиять на принимаемость слова.

Таким образом все элементы p матрицы весов графа $M[x]$ находятся в интервале $[t(n)^{-2}, 1]$, а их логарифмы в интервале $[-2\log(t(n)), 0]$. Тогда количество ε-интервалов равно $\frac{2\log(t(n))}{\varepsilon} = \frac{2\log(t(n))c}{\gamma} = \frac{2\log(t(n))dn}{\gamma}$, а значит количество различных классов эквивалентности

$$(\frac{2\log(t(n))dn}{\gamma} + 1)^{dn(dn+1)} = 2^{dn(dn+1)(\log(n) + \log(\log(t(n))) + \gamma_1)}.$$

Этого достаточно, чтобы распознавать такие языки как эквивалентность или палиндром.

О РАСШИРЕНИИ ТИПОВ ИГРОВОГО ВЗАИМОДЕЙСТВИЯ В ЯЗЫКЕ ИГРОВЫХ ПРОГРАММ

Р. В. Хелемендик (Москва)

Игровая программа (ИП) представляет собой специальный граф, который описывает выигрышную стратегию при взаимодействии двух сторон. Рассматривается задача синтеза ИП для заданных условий: начальных значений переменных, набора функций (“ходов”), типа взаимодействия и цели,

записываемой формулой логики ветвящегося времени. При этом стратегия, описываемая ИП (в случае её существования), считается выигрышной, если для этой ИП выполнены все компоненты зафиксированного условия.

При исследовании вопроса о выразительных возможностях языков ИП (см. [1]) и логики ветвящегося времени (см. [2]) выяснилось, что первый язык слабее второго. А именно: не для всякой выполнимой формулы, являющейся целью в условии взаимодействия, можно подобрать начальные значения переменных, игровое взаимодействие и ходы сторон, чтобы существовала ИП, удовлетворяющая этому условию. В связи с этим в настоящей работе для языка ИП введено расширение типов игрового взаимодействия (введены тип "максимального выбора" и "вершина-выбиратель") и установлено, что при таком расширении выразительные возможности языков ИП и логики ветвящегося времени эквивалентны.

Игровые правила

Обозначим через $Y = \{y_1, \ldots, y_n\}$, $n \geq 4$, конечное множество переменных, $\overline{y} =< y_1, \ldots, y_n >$ — набор переменных $y_1, \ldots, y_n$; $\mathrm{A} = \{0, \ldots, \mathrm{k}-1\} \cup \{2, 3\}$, $k \geq 2$, — конечную область значений переменных из множества Y, $\overline{\alpha}^{\delta} =< \alpha_1^{\delta}, \ldots, \alpha_n^{\delta} >$ — набор значений этих переменных, $0 \leq \alpha_1^{\delta} \leq 2$, $0 \leq \alpha_j^{\delta} \leq 3$, $2 \leq j \leq 3$, $0 \leq \alpha_j^{\delta} \leq k-1$, $4 \leq j \leq n$; W, B — конечные множества частичных n-мерных функций называемых, соответственно, *множествами ходов белых и чёрных*. Через F_{δ}^{w} (F_{δ}^{b}) обозначим множество функций из W (B), определённых на наборе $\overline{\alpha}^{\delta}$. Выделенный набор $\overline{\alpha}^{\delta_0}$ назовем *начальным*. Пятёрку $R = \langle Y, \mathrm{A}, \overline{\alpha}^{\delta_0}, \mathrm{W}, \mathrm{B} \rangle$ будем называть *игровыми правилами* или *R-правилами*.

Игровое взаимодействие

Игровое взаимодействие характеризует класс ИП согласно их структуре. Переменная y_1 управляет очередностью ходов белых и чёрных. Если $y_1 = 0$ ($y_1 = 1$), то ход белых (чёрных), а если $y_1 = 2$, то может быть как ход белых, так и ход чёрных. Переменная y_2 (y_3) определяет наши взаимотношения с выбором ходов белых (чёрных). Если $y_2 = 0$ ($y_3 = 0$), то это отношение "доверия", когда мы можем выбрать наиболее удобный ход белых (чёрных) с точки зрения достижения цели (см. ниже). Если $y_2 = 1$ ($y_3 = 1$), то это отношение "просчитывания", когда для достижения нашей цели придётся предусмотреть каждый из возможных ходов белых (чёрных). Если $y_2 = 2$ ($y_3 = 2$), то это отношение "максимального выбора", когда рассматривается любое подмножество возможных ходов белых (чёрных), причём каждый ход из этого множества может быть продублирован конечное число раз. Если $y_2 = 3$ ($y_3 = 3$), то возможен любой из этих вариантов. С использованием переменных y_1, y_2, y_3 теперь можно задавать игровыми правилами произвольный тип взаимодействия, в том числе и такие типы, которые управляются ходами сторон.

Игровая программа

Игровая программа (ИП), удовлетворяющая R-правилам, есть связный конечный ориентированный граф $\mathcal{P}$ с вершинами следующих видов: преобразователями, ветвителями, выбирателями и финальными вершинами. Каждая не финальная вершина является преобразователем (ветвителем, выбирателем) белых, либо чёрных.

Функционирование ИП определяется по индукции. Для каждой вершины $v_r^{\overline{\alpha}^\delta, x_r}$ этого графа нижний индекс уникален, $\overline{\alpha}^\delta$ есть набор значений переменных в этой вершине, а x_r является одним из следующих выражений, определяющих вид данной вершины.

• x_r есть $\otimes$. Тогда вершина $v_r^{\overline{\alpha}^\delta, \otimes}$ *финальная*. Она не имеет выходящих дуг.

• x_r есть f_t^{w} (f_t^{b}), где $f_t^{\mathrm{w}} \in F_\delta^{\mathrm{w}} \subseteq \mathrm{W}$ $(f_t^{\mathrm{b}} \in F_\delta^{\mathrm{b}} \subseteq \mathrm{B})$. Тогда вершина $v_r^{\overline{\alpha}^\delta, f_t^{\mathrm{w}}}$ $(v_r^{\overline{\alpha}^\delta, f_t^{\mathrm{b}}})$ называется белым (чёрным) *преобразователем*. Из этого преобразователя выходит единственная дуга, помеченная символом f_t^{w} (f_t^{b}), и эта дуга входит в некоторую вершину $v_{r'_t}^{\overline{\alpha}^{\delta'_t}, x_{r'_t}}$ со значением $\overline{\alpha}^{\delta'_t} = f_t^{\mathrm{w}}(\overline{\alpha}^\delta)$ $(\overline{\alpha}^{\delta'_t} = f_t^{\mathrm{b}}(\overline{\alpha}^\delta))$. Отметим, что в случае $|F_\delta^{\mathrm{w}}| > 1$ $(|F_\delta^{\mathrm{b}}| > 1)$ в качестве функции f_t^{w} (f_t^{b}) допускается любой элемент F_δ^{w} (F_δ^{b}).

• x_r есть F_δ^{w} (F_δ^{b}), и $|F_\delta^{\mathrm{w}}| \geq 1$ $(|F_\delta^{\mathrm{b}}| \geq 1)$. Тогда вершина $v_r^{\overline{\alpha}^\delta, F_\delta^{\mathrm{w}}}$ $(v_r^{\overline{\alpha}^\delta, F_\delta^{\mathrm{b}}})$ называется белым (чёрным) *ветвителем*. Тогда из этой вершины выходит $h = |F_\delta^{\mathrm{w}}|$ $(h = |F_\delta^{\mathrm{b}}|)$ дуг, помеченных соответственно символами f_t^{w} (f_t^{b}), где $f_t^{\mathrm{w}} \in F_\delta^{\mathrm{w}} \subseteq \mathrm{W}$ $(f_t^{\mathrm{b}} \in F_\delta^{\mathrm{b}} \subseteq \mathrm{B})$, и эти дуги входят соответственно в вершины $v_{r'_t}^{\overline{\alpha}^{\delta'_t}, x_{r'_t}}$ со значениями $\overline{\alpha}^{\delta'_t} = f_t^{\mathrm{w}}(\overline{\alpha}^\delta)$ $(\overline{\alpha}^{\delta'_t} = f_t^{\mathrm{b}}(\overline{\alpha}^\delta))$.

• x_r есть F_τ^{w} (F_τ^{b}), и $F_\tau^{\mathrm{w}} \subseteq F_\delta^{\mathrm{w}}$, $|F_\tau^{\mathrm{w}}| \geq 1$ $(F_\tau^{\mathrm{b}} \subseteq F_\delta^{\mathrm{b}}, |F_\tau^{\mathrm{b}}| \geq 1)$. Тогда вершина $v_r^{\overline{\alpha}^\delta, F_\tau^{\mathrm{w}}}$ $(v_r^{\overline{\alpha}^\delta, F_\tau^{\mathrm{b}}})$ называется белым (чёрным) *выбирателем*. Тогда из этой вершины выходит $h = |F_\tau^{\mathrm{w}}|$ $(h = |F_\tau^{\mathrm{b}}|)$ видов дуг, помеченных соответственно символами f_t^{w} (f_t^{b}), где $f_t^{\mathrm{w}} \in F_\delta^{\mathrm{w}} \subseteq \mathrm{W}$ $(f_t^{\mathrm{b}} \in F_\delta^{\mathrm{b}} \subseteq \mathrm{B})$, и эти дуги входят соответственно в вершины $v_{r'_t}^{\overline{\alpha}^{\delta'_t}, x_{r'_t}}$ со значениями $\overline{\alpha}^{\delta'_t} = f_t^{\mathrm{w}}(\overline{\alpha}^\delta)$ $(\overline{\alpha}^{\delta'_t} = f_t^{\mathrm{b}}(\overline{\alpha}^\delta))$. При этом каждая из h видов дуг f_t^{w} (f_t^{b}) может быть продублирована конечное число раз и вести как в уже имеющуюся, так и в новую вершину, значением которых является $\overline{\alpha}^{\delta'_t}$.

Если $\alpha_1^\delta = 0$ и $F_\delta^{\mathrm{w}} = \emptyset$, или $\alpha_1^\delta = 1$ и $F_\delta^{\mathrm{b}} = \emptyset$, или $\alpha_1^\delta = 2$ и $F_\delta^{\mathrm{w}} \cup F_\delta^{\mathrm{w}} = \emptyset$, то вершина с набором $\overline{\alpha}^\delta$ является финальной: $v_r^{\overline{\alpha}^\delta, \otimes}$.

В противном случае вершина $v_{x_r}^{\overline{\alpha}^\delta, \otimes}$ финальной не является, и возможный ее вид (индекс x_r) находится по таблице 1, охватывающей все возможные комбинации значений α_1^δ, α_2^δ, α_3^δ.

Цель в ИП

Цель в ИП есть формула Θ^* логики ветвящегося времени (см. [2]), в которой пропозициональными переменными являются утверждения вида $(y_j = l)$, где $y_j \in Y$, $j \leq 4$, $l \leq (k-1)$.

Постановка задачи синтеза ИП

Пусть $\mathcal{U} = \langle R, \Theta^* \rangle$. Будем говорить, что ИП $\mathcal{P}_\mathcal{U}$ удовлетворяет условию $\mathcal{U}$, если она удовлетворяет R-правилам и в ней истинна формула Θ^*. Существует ли ИП $\mathcal{P}_\mathcal{U}$, удовлетворяющая условию $\mathcal{U}$? Если существует, то необходимо построить хотя бы одну ИП.

Таблица 1

α_1^δ	α_2^δ	α_3^δ	x_r
0	0	0,1,2,3	f_t^{w}
1	0,1,2,3	0	f_t^{b}
0	1	0,1,2,3	F_δ^{w}
1	0,1,2,3	1	F_δ^{b}
0	2	0,1,2,3	F_τ^{w}
1	0,1,2,3	2	F_τ^{b}
0	3	0,1,2,3	f_t^{w} или F_δ^{w} или F_τ^{w}
1	0,1,2,3	3	f_t^{b} или F_δ^{b} или F_τ^{b}
2	0	0	f_t^{w} или f_t^{b}
2	0	1	f_t^{w} или F_δ^{b}
2	0	2	f_t^{w} или F_τ^{b}
α_1^δ	α_2^δ	α_3^δ	x_r
2	1	0	F_δ^{w} или f_t^{b}
2	1	1	F_δ^{w} или F_δ^{b}
2	1	2	F_δ^{w} или F_τ^{b}
2	2	0	F_τ^{w} или f_t^{b}
2	2	1	F_τ^{w} или F_δ^{b}
2	2	2	F_τ^{w} или F_τ^{b}
2	0	3	f_t^{w} или f_t^{b} или F_δ^{b} или F_τ^{b}
2	1	3	F_δ^{w} или f_t^{b} или F_δ^{b} или F_τ^{b}
2	2	3	F_τ^{w} или f_t^{b} или F_δ^{b} или F_τ^{b}
2	3	0	f_t^{w} или F_δ^{w} или F_τ^{w} или f_t^{b}
2	3	1	f_t^{w} или F_δ^{w} или F_τ^{w} или F_δ^{b}
2	3	2	f_t^{w} или F_δ^{w} или F_τ^{w} или F_τ^{b}
2	3	3	f_t^{w} или F_δ^{w} или F_τ^{w} или f_t^{b} или F_δ^{b} или F_τ^{b}

Теоремы о соответствии

Пусть задано условие $\mathcal{U} = \langle R, \Theta^* \rangle$, $m = \lceil \log_2 k \rceil$. Введём $m(n-3)$ пропозициональных переменных $p_1, \ldots, p_{m(n-3)}$. Заменим в формуле Θ^* каждое утверждение $(y_j = l)$ на конъюнкцию m множителей $p^{\sigma_{j,l,i}}_{m(j-4)+i}$, $1 \le i \le m$, где $p^{\sigma_{j,l,i}}_{m(j-4)+i}$ есть $p_{m(j-4)+i}$, если в двоичной записи m разрядами числа l на i-м месте стоит 1 (в этом случае переменную $p_{m(j-4)+i}$ будем называть положительно входящей для утверждения $y_j = l$), и $p^{\sigma_{j,l,i}}_{m(j-4)+i}$ есть $\neg p_{m(j-4)+i}$ в противном случае. Полученную формулу Θ логики ветвящегося времени назовём *формулой, соответствующей* Θ^*.

Теорема 1. *Если существует ИП* $\mathcal{P}_{\mathcal{U}}$*, удовлетворяющая условию* $\mathcal{U} = \langle R, \Theta^* \rangle$*, то формула* Θ*, соответствующая* Θ^**, выполнима.*

Пусть дана формула логики ветвящегося времени Θ, и $p_1, \ldots, p_m$ - все встречающиеся в ней различные пропозициональные переменные. Положим $Y = \langle y_1, \ldots, y_n \rangle$, $n = m + 3$, $A = \{0, 1, 2, 3\}$, $k = 2$. Заменим каждую подформулу формулы Θ с чётным (в частности, нулевым) числом отрицаний, непосредственно стоящих перед переменной p_j, на подформулу $(y_{j+3} = 1)$, а каждую подформулу с нечётным числом отрицаний, непосредственно стоящих перед переменной p_j – на подформулу $(y_{j+3} = 0)$. Полученную формулу обозначим через Θ^*. В условии $\mathcal{U} = \langle R, \Theta^* \rangle$ тройку $K = \langle Y, A, \Theta^* \rangle$ с зафиксированными и определёнными выше компонентами назовём *каркасом* (условия $\mathcal{U}$), соответствующим формуле Θ.

Теорема 2. *Если выполнимой формуле* Θ *соответствует каркас* $K = \langle Y, A, \Theta^* \rangle$ *условия* $\mathcal{U}$*, то возможно такое доопределение условия* $\mathcal{U}$*, что существует ИП* $\mathcal{P}_{\mathcal{U}}$*, удовлетворяющая этому условию.*

Из теорем 1 и 2 следует, что теперь в языках ИП и логики ветвящегося времени могут быть решены одни и те же задачи. В то же время язык ИП предоставляет более широкие средства для структуризации записи задачи в рамках следующего подхода: указание начальной ситуации; определение правил её изменения и типа взаимодействия; постановка цели.

Работа выполнена при финансовой поддержке программы фундаментальных исследований Отделения математических наук РАН “Алгебраические и комбинаторные методы математической кибернетики” (проект “Оптимальный синтез управляющих систем”).

Список литературы

1. Хелемендик Р. В. О расширении логического языка игровых программ и решении задачи синтеза. // Синтаксис и семантика логических систем: Материалы российской школы-семинара. — Иркутск, Издательство ГОУ

ВПО “Иркутский государственный педагогический университет”, 2006. С. 108–112.

2. Хелемендик Р. В. Алгоритм распознавания формул логики ветвящегося времени и эффективный алгоритм построения выводов общезначимых формул из аксиом. // Математические вопросы кибернетики. Вып. 15: Сборник статей / Под ред. О.Б.Лупанова. — М.: Физматлит, 2006. С. 217–266.

О МНОГОЯРУСНЫХ ФОРМУЛАХ

Д. Ю. Черухин (Москва)

Рассматриваются булевы схемы из функциональных элементов (СФЭ, [1]). Ветвлением вершины называется число рёбер, исходящих из этой вершины; в частности, в ветвлении учитывается число выходов, которым соответствует эта вершина (на которые "подаётся сигнал"из этой вершины). Например, если из вершины выходит 2 ребра, ведущих в другие вершины схемы и, кроме того, вершина соответствует трём выходам схемы, то её ветвление равно 5.

Вершина называется узловой, если либо она является входом схемы, либо её ветвление больше единицы. СФЭ S называется *k-ярусной* формулой, если в любом ориентированном пути в S содержится не более k узловых вершин. Другими словами, если узловые вершины можно разбить на k ярусов так, что внутри каждого яруса вершины попарно несравнимы в смысле естественного порядка в ориентированном графе.

Одноярусные формулы соответствуют обычным формулам, т. е. схемам без ветвления (могут ветвиться только входы). Сложностью k-ярусной формулы является сумма ветвлений узловых вершин. Такая мера сложности естественна для формул; в случае $k = 1$ она совпадает с числом вхождений переменных в формулу. Пусть L_B^k — мера сложности функций (операторов) в классе k-ярусных формул в конечном базисе B, $\mathcal{L}_B^k$ — соответствующая функция Шеннона [1].

Теорема 1. *В любом базисе B при $k \geqslant 2$*

$$\mathcal{L}_B^k(n) \sim \frac{2^n}{n}.$$

Теорема 1 показывает, что начиная с 2-х ярусов функция Шеннона асимптотически совпадает с аналогичной функцией для СФЭ (СФЭ можно

рассматривать как формулу с неограниченным числом ярусов). Заметим, что [1]

$$\mathcal{L}_B^1(n) \sim \frac{2^n}{\log_2 n}.$$

Пусть $F = (f_1, \ldots, f_m)$ — оператор, зависящий от двух наборов переменных: $X = (x_1, \ldots, x_k)$ и $Y = (y_1, \ldots, y_l)$. Разложим f_j в полином Жегалкина [8] по переменным X:

$$f_j = f_j^0 \oplus f_j^1 x_1 \oplus \ldots \oplus f_j^k x_k \oplus f_j',$$

где каждая из функций f_j^i зависит только от Y, f_j' — нелинейная по X часть.

Пусть $M^1, \ldots, M^k$, $M_1, \ldots, M_m$ — множества функций, зависящих от переменных Y и обладающих свойством: каждая функция f_j^i вычислима через функции из множества $M^i \cup M_j$, т. е. представима в виде $h(g_1, \ldots, g_t)$, где $\{g_1, \ldots, g_t\} \subseteq M^i \cup M_j$, $h \in P_2$. Тогда набор $(M^1, \ldots, M^k;\ M_1, \ldots, M_m)$ назовём *F-таблицей*. Через $L'(F)$ обозначим минимум по всем F-таблицам суммы $\sum\limits_i |M^i| + \sum\limits_j |M_j|$.

Теорема 2. *В любом базисе B*

$$L_B^2(F) \geqslant L'(F).$$

Следствие 1. *Пусть F_n — любой из операторов: умножение матриц, умножение многочленов, циклическая свёртка; n — число его входов. Тогда в любом базисе B*

$$L_B^2(F_n) = \Omega(n^{3/2}).$$

Теорема 2 может быть обобщена [5, 6] на класс СФЭ глубины 2 в базисе из всех булевых функций. Известные для этого класса нижние оценки вида $\Omega(n\frac{\ln^2 n}{\ln\ln n})$ следуют из результатов теории графов [10].

Пусть $F = (F_1, \ldots, F_k)$ — булев оператор, разбитый на k операторов и зависящий от набора переменных $X = (X_1, \ldots, X_k)$, разбитого на k наборов. Обозначим через $\mathcal{D}(F_i, X_j)$ множество подоператоров, полученных из F_i при всевозможных подстановках констант вместо всех переменных, не входящих в набор X_j.

Теорема 3. *Для любого базиса B существует константа c'_B такая, что для любого оператора F, существенно завясящего от всех переменных из X*

$$L_B^2(F) \geqslant c'_B \sum_{i=1}^{k} \log_2 |\mathcal{D}(F_i, X_i)|.$$

Теорема 3 является обобщением метода Нечипорука для формул [2]. Она позволяет получать нижние оценки сложности вида $\Omega(\frac{n^2}{\log n})$, например, для оператора, состоящего из одинаковых функций Нечипорука (число функций равно числу переменных).

Обозначим $\Lambda_n = x_1 \oplus \ldots \oplus x_n$. Для любого $k \geqslant 1$ введём функцию

$$\varphi_k(x) = \frac{1}{1-(1-1/x)^k}.$$

Теорема 4. [7] *Пусть B — базис, $\gamma > 1$. Тогда:*

а) если

$$L_B^1(\Lambda_n) = \mathcal{O}(n^\gamma),$$

то для любого k

$$L_B^k(\Lambda_n) = \mathcal{O}(n^{\varphi_k(\gamma)}).$$

б) если базис B обладает экспонентой сжатия (shrinkage exponent) [7, 9] γ, то для любого k

$$L_B^k(\Lambda_n) = \Omega(n^{\varphi_k(\gamma)}).$$

Теорема 4 устанавливает связь между нижними и верхними оценками сложности функции Λ_n в классе формул и соответствующими оценками в классе многоярусных формул. Заметим, что нижние оценки сложности функции Λ_n в классе формул традиционно получаются с помощью метода Субботовской [3,4] и из подобного доказательства обычно извлекается информация о нижней оценке экспоненты сжатия.

Пусть M^* — множество булевых функций, возрастающих или убывающих по каждой переменной. Для мер сложности μ_1, μ_2 обозначим через $\mu_1 \leqslant \mu_2$ отношение частичного порядка $\mu_1 = \mathcal{O}(\mu_2)$ (т. е. $\forall f \in P_2\ \mu_1(f) = \mathcal{O}(\mu_2(f))$).

Теорема 5. [7] *а) Для любого базиса $B \subseteq M^*$ существует последовательность $k_1, k_2, \ldots$ такая, что*

$$L_B^{k_1} > L_B^{k_2} > \ldots$$

б) Для базиса $B_0 = \{\&, \vee, \neg\}$

$$L_{B_0}^1 > L_{B_0}^2 > \ldots$$

в) Для каждого k существует последовательность базисов $B_1, B_2 \ldots$ такая, что

$$L_{B_1}^k > L_{B_2}^k > \ldots$$

Теорема 5 следует из теоремы 4 и известных результатов об экспоненте сжатия для различных базисов [3,9].

Теорема 6. *Для любого базиса B и любых k, l*

$$L_B^k = (L_B^l)^{\mathcal{O}(1)}.$$

Обозначим $\ln^{(s)}(x) = \ln\ln\ldots\ln(x)$, где число логарифмов равно s, $s \geqslant 0$. Скажем, что меры сложности μ_1 и μ_2 s-эквивалентны, если

$$\exists C\ \forall f \in P_2\ |\ln^{(s)}(\mu_1(f)) - \ln^{(s)}(\mu_2(f))| \leqslant C.$$

Скажем, что меры *слабо* эквивалентны, если для некоторого s они s-эквивалентны.

Заметим, что 1-эквивалентность есть совпадение по порядку (это отношение исследуется в теореме 5), 2-эквивалентность есть полиномиальная эквивалентность (как в теореме 6). Можно показать, что меры сложности многоярусных формул и контактно-вентильных схем (а также их ограничений — контактных схем, ветвящихся программ и т. д.) 3-эквивалентны. В то же время, вопрос о слабой эквивалентности этих мер с мерой сложности СФЭ является открытым.

Работа выполнена при финансовой поддержке РФФИ (проект 05–01–00994), программы "Университеты России"(проект УР.04.02.528) и программы поддержки ведущих научных школ РФ (проект НШ–1807.2003.1).

Список литературы

1. Лупанов О. Б. Асимптотические оценки сложности управляющих систем. М.: Изд-во МГУ, 1984.

2. Нечипорук Э. И. Об одной булевской функции // ДАН СССР. 1966. Т. 169, N 4. с. 765–766.

3. Перязев Н. А. Сложность представлений булевых функций формулами в немонолинейных базисах. Дискретная математика и информатика. Вып. 2. — Иркутск: Изд-во Иркут. ун-та, 1995.

4. Субботовская Б. А. О реализации линейных функций формулами в базисе $\vee, \&, -$. ДАН СССР. 1961. Т. 136, N. 3. с. 784–787.

5. Черухин Д. Ю. О схемах из функциональных элементов с ограниченной глубиной ветвления // Докл. РАН. Т. 405, N. 4. 2005. с. 467–470.

6. Черухин Д. Ю. Нижняя оценка сложности в классе схем глубины 2 без ограничений на базис // Вестн. МГУ. Сер. 1. N 4. 2005. с. 54–56.

7. Черухин Д. Ю. О схемах из функциональных элементов конечной глубины ветвления // Дискретная математика. Т. 18, вып. 4. 2006. с. 73–83.

8. Яблонский С. В. Введение в дискретную математику. М.: Наука, 1986.

9. Hastad J. The shrinkage exponent of de Morgan formulas is 2. SIAM J. Comput. 1998. V. 27. p. 48–64.

10. Radhakrishnan J. Ta-Shma A. Bounds for dispersers, extractors, and depth-two superconcentrators // SIAM J. of Discrete Mathematics. 2000. V. 13, No 1. p. 2–24.

О СЛОЖНОСТИ ФУНКЦИЙ С "МАЛЫМ ЧИСЛОМ ЕДИНИЦ" В КЛАССЕ КНФ

С. Е. Черухина (Москва)

В работе [1] для функций $f(x_1, \ldots, x_n)$ из семейства $R_{n,k}$ (обращающихся в единицу на k наборах) было доказано, что в классе формул в базисе $\{\&, \vee, \neg\}$ их сложность не превосходит $2n + k2^{k-1}$. Используя конструкцию автора работы [1], нетрудно показать, что в классе КНФ сложность будет ограничена величиной $2n + 2^{k-2}2^{2^{k-1}}$.

Применяя метод, рассмотренный в [2], верхняя оценка существенно понижается до $2n + ck^2 2^k + k2^{k-1}$. Нижняя оценка для функционала $L(n,k) = \max L(f)$ (по всем функциям из $R_{n,k}$) следует из доказываемой далее теоремы.

Тогда для небольших значений k (точнее, для $k \leqslant \log n + \frac{1}{2}\log\log n$) верно

$$2n + c_1 \frac{2^k}{\sqrt{k}} \leqslant L(n,k) \leqslant 2n + c_2 k^2 2^k.$$

В частности, для $k \leqslant \log n - 2\log\log n - \psi(n), \psi(n) \to \infty, n \to \infty$, верно

$$L(n,k) \sim 2n.$$

Мы будем рассматривать некоторую булеву функцию $f(x_1, \ldots, x_n)$, принимающую значение 1 на $k+2$ наборах. Эти наборы составляют *матрицу* M функции f. Оценим снизу сложность такой функции формулами вида КНФ.

Для удобства доказательства будем рассматривать не саму функцию f, а ее отрицание — $\bar{f}$ и реализацию $\bar{f}$ формулами ДНФ. При этом матрица M содержит наборы, на которых $\bar{f}$ равна нулю.

Определение. *Два столбца матрицы M находятся в общем положении, если в строках подматрицы, ими составляемой, встречаются все четыре возможные комбинации из* 0 *и* 1.

Лемма 1. *Пусть* x_i *— переменная функции* $\bar{f}(x_1, \ldots, x_n), n \geqslant 2$, *а также*:

а) расстояние между любыми двумя строками матрицы M *не меньше* 2;

б) столбец, соответствующий переменной x_i, *находится в общем положении с любым другим столбцом из* M.

Тогда в любой ДНФ для $\bar{f}$ *как переменная* x_i, *так и ее отрицание встречаются не менее двух раз.*

Доказательство. Заметим, что по крайней мере по одному разу x_i и $\bar{x}_i$ встречаются в любой ДНФ, реализующей $\bar{f}$. Действительно, в силу п. б), функция $\bar{f}$ равна 0 на некотором наборе $\tilde{\alpha}$ с $\alpha_i = 1$. На наборе $\tilde{\beta}$, отличающемся от набора $\tilde{\alpha}$ только i-й координатой ($\beta_i = 0$), в силу п. а), $\bar{f}(\tilde{\beta}) = 1$. Следовательно, функция $\bar{f}$ не возрастает по переменной x_i, а значит, $\bar{x}_i$ содержится в любой ДНФ для $\bar{f}$. Аналогично существуют наборы $\tilde{\alpha}$ с координатой $\alpha_i = 0$ и $\tilde{\beta}$ с координатой $\beta_i = 1$, $\rho(\tilde{\alpha}, \tilde{\beta}) = 1$, на которых $\bar{f}(\tilde{\alpha}) = 0$ и $\bar{f}(\tilde{\beta}) = 1$, из чего следует, что x_i также должно присутствовать в любой ДНФ для $\bar{f}$.

Далее, предположим, что

$$x_i \text{ входит в ДНФ для } \bar{f} \text{ ровно 1 раз.} \tag{1}$$

Пусть K — конъюнкция, содержащая x_i. Рассмотрим два случая:

1) $K = x_i$. Тогда при $x_i = 1$ функция $\bar{f}$ обращается в 1, следовательно, матрица M должна содержать нулевой столбец, соответствующий переменной x_i, что противоречит п. б) условия леммы.

2) $K = x_i x_j^{\sigma} \cdots$ (без ограничения общности, можем считать, что $\sigma = 1$).

Пусть $\tilde{\alpha}$ — любой набор такой, что $\alpha_i = \alpha_j = 0$. Покажем, что $\bar{f}(\tilde{\alpha}) = 1$. Предположим, что $\bar{f}(\tilde{\alpha}) = 0$. Рассмотрим набор $\tilde{\beta}$ такой, что $\rho(\tilde{\alpha}, \tilde{\beta}) = 1$, $\beta_i = 1, \beta_j = 0$. Тогда $\bar{f}(\tilde{\beta}) = 1$, т. к. из п. а) следует, что соседних нулей у $\bar{f}$ нет. Набор $\tilde{\beta}$ должен быть накрыт некоторой конъюнкцией K', причем $K' \neq K$, поскольку $K(\tilde{\beta}) = 0$. Конъюнкция K' должна содержать x_i, т. к. K' не убывает по x_i ($K'(\tilde{\alpha}) = 0, K'(\tilde{\beta}) = 1$). Но по предположению (1) K — единственная конъюнкция, содержащая x_i. Противоречие. Следовательно, $\bar{f}(\tilde{\alpha}) = 1$. Это верно для любого набора $\tilde{\alpha}$, у которого $\alpha_i = \alpha_j = 0$, поэтому в матрице M отсутствуют строки, в i-м и j-м столбцах которых стоят одновременно 0, а это противоречит тому, что i-ый и j-ый столбцы находятся в общем положении (п. б)). Итак, x_i входит в ДНФ, как минимум, два раза. Случай $\bar{x}_i$ аналогичен.

Утверждение 1. *Если в матрице* M *функции* f *в столбце, соответствующем переменной* x_i, *есть* 0 *и* 1 *и расстояние между любыми двумя строками не меньше* 2, *то переменная* x_i *и ее отрицание входят в любую КНФ функции* f *по крайней мере по одному разу.*

Доказательство следует из первой части доказательства леммы.

Теорема 1. *Для любых $n, k, k \geqslant 4$, $n \geqslant C_k^{\lfloor k/2 \rfloor}$, найдется функция от n переменных, принимающая значение 1 на $k+2$ наборах, такая, что*

$$L(f) \geqslant 2n + 2C_k^{\lfloor k/2 \rfloor} - 2.$$

Доказательство. Матрица M функции $\bar{f}$ будет устроена следующим образом. Рассмотрим $C_k^{\lfloor k/2 \rfloor}$ столбцов, представляющие из себя всевозможные наборы длины k, содержащие ровно $\lfloor k/2 \rfloor$ единиц; из них составим матрицу A. Последний (например) столбец матрицы A продублируем $n - C_k^{\lfloor k/2 \rfloor}$ раз. Эти одинаковые столбцы образуют матрицу B. Также добавим к матрицам A и B по две строки: "нулевую"и "единичную", вместе с которыми образуются новые матрицы A' и B'. Наконец, $M = (A'|B')$. Тогда любые два столбца из A' находятся в общем положении (т. к. соответствующие векторы из A несравнимы). И любой столбец из A', кроме последнего, находится в общем положении с любым другим столбцом из M. Расстояние между любыми двумя строками из M не меньше двух; это следует из того, что в матрице M, кроме двух последних строк, в качестве столбцов имеются все варианты наборов длины k, содержащие $\lfloor k/2 \rfloor$ единиц. Поэтому для всех столбцов из A', кроме последнего, применима лемма, а для остальных — утверждение. Следовательно,

$$L(f) \geqslant 4(C_k^{\lfloor k/2 \rfloor} - 1) + 2(n - C_k^{\lfloor k/2 \rfloor} + 1) = 2n + 2C_k^{\lfloor k/2 \rfloor} - 2.$$

Замечание. Для больших значений k, а именно, для k таких, что $n \leqslant C_k^{\lfloor k/2 \rfloor}$ и $k \leqslant 2^{n-1}$ можно доказать, что $L(n, k) \geqslant 4n$.

Список литературы

1. Фиников Б. И. Об одном семействе классов функций алгебры логики и их реализации в классе П-схем. ДАН СССР 115, 2, 1957, с. 247–248.

2. Черухина С. Е. О сложности функций с малым числом единиц. Труды 6 Международной конференции "Дискретные модели в теории управляющих систем" (Москва, 7-11.12.2004). Изд. отд. ф-та ВМиК МГУ, 2004, с. 95.

ОБ ЭФФЕКТИВНОЙ РЕАЛИЗАЦИИ ФУНКЦИЙ, ПОСТРОЕННЫХ ПО РЕКУРСИВНОЙ КОНСТРУКЦИИ СПЕЦИАЛЬНОГО ВИДА

С. Г. Шипунов (Москва)

Введение

Ю.В. Таранниковым была разработана рекурсивная конструкция, позволяющая строить m-устойчивые функции с нелинейностью близкой к максимально возможной для данного класса функций. Однако, основной проблемой ограничивающей возможность применения данной конструкции при создании потоковых шифраторов является нетривиальная в общем случае реализация функции, порожденных этой конструкцией. В данной работе будет представлен алгоритм, позволяющий эффективно реализовывать эти функции.

Данная статья состоит из двух частей. В первой части приводятся базовые определения и алгоритм построения рассматриваемой конструкции. Во второй части приводится разработанный нами алгоритм эффективной реализации в общем случае.

1. Базовые определения

Рассмотрим векторное пространство F_n^2 наборов длины n с компонентами из F_2 — конечного поля из двух элементов 0 и 1, операции сложения и умножения в котором вводятся как обычные операции сложения и умножения по модулю 2.

Будем говорить, что булева функция

$$f = f(x_1, x_2, ..., x_n)$$

зависит от пары переменных (x_i, x_j) квазилинейно, если $f(X') \neq f(X'')$ для любых двух наборов X' и X'', различающихся только в i-й и j-й компонентах. Пара (x_i, x_j) в этом случае называется парой квазилинейных переменных. В [1] показывается, что пара (x_i, x_j) является парой квазилинейных переменных в булевой функции

$$f = f(x_1, x_2, ..., x_n)$$

в том и только в том случае, когда f может быть представлена в виде

$$f = g(x_1, ..., x_{j-1}, x_{j+1}, ..., x_{i-1}, x_{i+1}, ..., x_n, x_i \oplus x_j) \oplus x_j \ .$$

Конструкция 1. *Пусть* $f_0, ..., f_{2^k-1}$ *булевы функции на* F_n^2. *Пусть* $\sigma_1, ..., \sigma_k$ — *двоичное представление* r.*Пусть* $C = (c_1, ..., c_k)$ *произвольный двоичный набор. Обозначим* $s = \sum_{i=1}^{i=k} c_i$. *Введем множества*

$$X = \{x_i | i = 1, ..., n\}, \quad Y = \{y_i | i = 1, ..., k\}, \quad Z = \{z_i | i = 1, ..., k\}.$$

Определим

$$F(X, Y, Z) = \bigoplus_{j=1}^{j=2^k-1} f_{\sigma_1,...,\sigma_k}(X)(y_1 \oplus c_1 z_1 \oplus \sigma_1) \ldots$$

$$\ldots (y_k \oplus c_k z_k) \oplus c_1 z_1 \oplus \ldots \oplus c_k z_k.$$

Определение. *Матрица* $B_{k_0,k,p,t} = b_{ij}$ *размера* $2^k \times p$, *состоящая из* $(*, 1, 2)$, *называется подходящей, если*:

1) *Для любых двух строк* i_1, i_2 *существует такой столбец* j, *что* $b_{i_1,j} = 1$, $b_{i_2,j} = 2$ *или* $b_{i_1,j} = 2$, $b_{i_2,j} = 1$;

2) *сумма всех элементов в любой строке не превосходит* t, $*$ *считается как* 0;

3) *число единиц в любой строке не превосходит* k_0.

Обозначим $S_{n,m,k}$ множество булевых функций, такое что для каждого $s, 0 \leq s \leq k$, множество $S_{n,m,k}$ содержит $(m + s)$ -устойчивую функцию на F_2^{n+s} , которая имеет s непересекающихся пар квазилинейных переменных.

Были введены все необходимые определения, теперь приведем теорему Ю. В. Таранникова, позволяющую строить m-устойчивые функции, для сколько угодно больших m.

Теорема 1. *При* $2p - t \leq n$, *по системе функций* S_{n,m,k_0} *и подходящей матрице* $B_{k_0,k,p,t}$ *можно построить систему функций* $S_{n+k+t,m+t,k_0}$.

Приведем только конструктивную часть доказательства. Ее можно разбить на три части.

1) Рассмотрим каждую строку матрицы $B_{k_0,k,p,t}$. Пусть рассматриваем i-ю строку и пусть в ней s единиц. Из системы функций S_{n,m,k_0} возьмем $(m + s)$-устойчивую функцию с s парами неперсекающихся квазилинейных переменных. Добавим к ней $t - s$ линейных переменных. И обозначим результат f_i'.

2) Переименуем переменные в f_i' таким образом, чтобы стоящей в матрице $B_{k_0,k,p,t}$ в i-й строке на j месте единице соответствовала пара квазиленейных переменных (x_{2j-1}, x_{2j}), а двойке соответствовала пара линейных переменных (x_{2j-1}, x_{2j}). Получим функцию f_i''.

3) К полученным функциям применим конструкцию 1 для всех $C, 0 \leq wt(C) \leq k$, и получим искомую систему функций $S_{n+k+t,m+t,k_0}$.

Важно учитывать, что приведенная конструкция рекурсивная, то есть по системе S_{n,m,k_0} после неоднократного последовательного применения теоремы с различными подходящими матрицами $B_{k_0^i,k^i,p^i,t^i}$, можно получить функции со сколько угодно высокой устойчивостью, а при правильном подборе параметров k_0, k, p, t, и со сколько угодно близкой к максимально возможной нелинейностью. Но перестановка на втором шаге чрезвычайно усложняет реализацию функций из системы, полученных с помощью последовательного применения данной теоремы для произвольной фиксированной последовательности подходящих матриц $B_{k_0^i,k^i,p^i,t^i}, 0 \le i \le n$. Далее будет предложен метод, позволяющий обойти эту проблему.

2. Об эффективной реализации конструкции

Пусть есть произвольная последовательность

$$B_{k_0^i,k^i,p^i,t^i}, 0 \le i \le N,$$

с условием $k_0^n \le k^{n-1}$, и произвольная система S_{n,m,k_0^0}. Построим с помощью последовательного применения теоремы 1 систему

$$S_{n+\sum_{i=1}^{i=N} k^i+\sum_{i=1}^{i=N} t^i,m+\sum_{i=1}^{i=N} t^i,k^N}.$$

Зафиксируем произвольным образом $s^N, 0 \le s^N \le k^N$. Рассмотрим применение третьего шага из теоремы 1 для последней матрицы $B_{k_0^N,k^N,p^N,t^N}$ и системы

$$S_{n+\sum_{i=1}^{i=N-1} k^i+\sum_{i=1}^{i=N-1} t^i,m+\sum_{i=1}^{i=N-1} t^i,k^n}.$$

В нем

$$f_s^N(X,Y,Z) = \bigoplus_{\sigma_1,\dots,\sigma_{k^N}} f''_{\sigma_1,\dots,\sigma_{k^N}}(X)(y_1^N \oplus z_1^N \oplus \sigma_1)\dots$$

$$\dots(y_s^N \oplus z_s^N \oplus \sigma_s)(y_{s+1}^N \oplus \sigma_{s+1})\dots(y_{k^N}^N \oplus \sigma_{k^N}^N) \oplus \bigoplus_{i=1}^{i=s} z_i^N.$$

Заметим, что на произвольно заданном наборе переменных Y, Z только один из множителей

$$(y_1^N \oplus z_1^N \oplus \sigma_1)\dots(y_s^N \oplus z_s^N \oplus \sigma_s)(y_{s+1}^N \oplus \sigma_{s+1})\dots(y_{k^N}^N \oplus \sigma_{k^N}^N)$$

не равен нулю, а значит можно определить индекс $\sigma_1, \dots, \sigma_{k^N}$

функции $f''_{\sigma_1,\ldots,\sigma_{k_N}}$, а именно

$$\begin{aligned}
\sigma_1 &= y_1^N \oplus z_1^N \oplus 1\\
&\ldots\ldots\ldots\ldots\\
\sigma_s &= y_s^N \oplus z_s^N \oplus 1,\\
\sigma_{s+1} &= y_{s+1}^N \oplus 1,\\
&\ldots\ldots\ldots\ldots\\
\sigma_{k^N} &= y_{k^N}^N \oplus 1.
\end{aligned}$$

А значит и номер строки матрицы $B_{k_0^N,k^N,p^N,t^N}$, согласно которой происходила перестановка во втором шаге. Так же можно вычислить линейную прибавку

$$L^N = \bigoplus_{i=1}^{i=s} z_i^N.$$

Перейдем теперь ко второму шагу. Пусть в данной строке матрицы $B_{k_0^N,k^N,p^N,t^N}$ s единиц, тогда в ней $\frac{t^N-s}{2}$ двоек и $p-\frac{t^N+s}{2}$ символов *. Зафиксируем переименование переменных следующим образом:
1) Пусть на j месте в интересующей нас i строке стоит 2 и до нее в этой строке было r_2 двоек. В первом шаге теоремы 1 было добавлено t^N-s линейных переменных, а именно $(u_1^N,\ldots,u_{t^N-s}^N)$. Переименуем

$$(u_{2r_2-1}^N, u_{2r_2}^N) \text{ в } (x_{2j-1}^N, x_{2j}^N).$$

2)Пусть на j месте в интересующей нас i строке стоит 1 и до нее в этой строке было r_1 единиц. В третьем шаге теоремы 1 для матрицы

$$B_{k_0^{N-1},k^{N-1},p^{N-1},t^{N-1}}$$

и системы

$$S_{n+\sum_{i=1}^{i=N-2} k^i+\sum_{i=1}^{i=N-2} t^i, m+\sum_{i=1}^{i=N-2} t^i, k^{N-2}}$$

было добавлено s переменных z и k^{N-1} переменных y, а именно

$$(z_1^{N-1},\ldots,z_s^{N-1}) \text{ и } (y_1^{N-1},\ldots,u_{y^{N-1}}^{N-1})$$

Из них $(z_i^{N-1}, y_i^{N-1}), 0 \le i \le s$, образуют пары квазилинейных переменных. Переименуем

$$(y_{r_1}^{N-1}, z_{r_1}^{N-1}) \text{ в } (x_{2j-1}^N, x_{2j}^N).$$

3)Переименование оставшихся переменных должно быть сделано произвольным фиксированным образом. В частности переменные $(y_{s+1}^{N-1}, ..., y_{k^{N-1}}^{N-1})$, не входящие в упомянутые ранее пары квазилинейных переменных переименовываются в какие-то $(X_{i_1}^N, ..., x_{s_{k^{N-1}-s}}^N)$.

Переименование переменных есть всего лишь подстановка вместо

$$U = (u_1^N, \ldots, u_{t^N-s}^N),$$
$$Z = (z_1^{N-1}, \ldots, z_s^{N-1}),$$
$$Y = (y_1^{N-1}, \ldots, y_{k^{N-1}}^{N-1})$$

определенных переменных из X. Вариант такой подстановки был описан ранее, поэтому по номеру строки можно определить параметр s(число пар квазилинейных переменных) и подстановку переменных вместо наборов U, Y, Z. А зная это можно перейти к третьему шагу теоремы 1 для системы

$$S_{n+\sum_{i=1}^{i=N-2} k^i + \sum_{i=1}^{i=N-2} t^i, m+\sum_{i=1}^{i=N-2} t^i, k^N}$$

и матрицы

$$B_{k_0^{N-1}, k^{N-1}, p^{N-1}, t^{N-1}}.$$

Рассмотрим первый шаг теоремы. На этом шаге нам известны значения переменных

$$U = (u_1^N, \ldots, u_{t^N-s}^N)$$

из второго шага. Остается лишь прибавить их к L^N.

Таким образом, мы перешли от вычисления функции из

$$S_{n+\sum_{i=1}^{i=N} k^i + \sum_{i=1}^{i=N} t^i, m+\sum_{i=1}^{i=N} t^i, k^N}$$

к вычислению функции из

$$S_{n+\sum_{i=1}^{i=N-1} k^i + \sum_{i=1}^{i=N-1} t^i, m+\sum_{i=1}^{i=N-1} t^i, k^{N-1}}.$$

Данный процесс можно продолжить вплоть до S_{n,m,k_0^0}. В итоге мы получили алгоритм, позволяющий эффективно вычислять функции, построенные по произвольно заданной последовательности подходящих матриц $B_{k_0^i, k^i, p^i, t^i}, 0 \leq i \leq N$, при этом на каждом шаге алгоритма вычисляется всего одна функция из 2^{k^i} возможных. И даже у этой одной функции вычисляется лишь линейная часть, а вместо нелинейной части считается номер предыдущей функции. В совокупности это обеспечивает линейную программную скорость.

Список литературы

1. Тарасников Ю. В. О корреляционно-имунных и устойчивых булевых функциях// Математические вопросы кибернетики. Вып 11.— М.: Физматлит, 2002. — С. 91–148.

ОБЩИЙ ПОДХОД К ПРОБЛЕМЕ ЭКВИВАЛЕНТНОСТИ ПРОГРАММ НА ШКАЛАХ, СВЯЗАННЫХ С ОБРАБОТКОЙ ПРЕРЫВАНИЙ

В. Л. Щербина (Москва)

Эквивалентность программ на шкалах — понятие существенно более сильное, чем функциональная эквивалентность (по сути, это эквивалентность на целом классе семантик). С одной стороны, это делает возможным строить эффективные алгоритмы, с другой — выражать характерные особенности модели вычислений. Метод критериальных систем, позволяющий устанавливать эквивалентность для некоторых шкал, требует явного построения полугруппы -критериальной системы. Универсальный способ построения критериальной системы для составной шкалы, включающей действия основной программы и действия по обработке прерываний, не известен. В данной работе предлагается подход, базирующийся на методе критериальных систем, позволяющий решать проблему эквивалентности для широкого класса шкал за полиномиальное время. Идея его состоит в том, чтобы рассматривать серии объединённых определённым образом операторов как единичные команды.

Рассмотрим конечное множество $\tilde{A}$, которое назовем алфавитом *действий. Операторной цепочкой* называется слово в алфавите $\tilde{A}$. Для пустой операторной цепочки будет использоваться обозначение λ.

Рассмотрим также непустое конечное множество $\mathcal{C}$, элементами которого обозначаются всевозможные комбинации значений логических условий, различаемых в описываемой модели программ.

Пропозициональная операторная программа сигнатуры $(\tilde{A}, \mathcal{C})$ задается системой переходов $\langle V, \textbf{вход}, \textbf{выход}, \textbf{тупик}, B, T\rangle$, где V — конечное множество вершин-преобразователей, **вход** — *начальная вершина*, **выход** — *заключительная вершина*, **тупик** — *вершина пустого цикла*, $B\colon V \to \tilde{A}$ — *функция привязки*, сопоставляющая каждому преобразователю программы некоторый оператор, $T\colon (V \cup \{\textbf{вход}\}) \times \mathcal{C} \to (V \cup \{\textbf{выход}, \textbf{тупик}\})$. *Размером* $|\pi|$ программы π назовем число вершин-преобразователей в ней.

Детерминированной динамической шкалой (или просто *шкалой*) сигнатуры $\tilde{A}$ назовем тройку $\mathcal{F} = \langle S, s_0, R\rangle$, состоящую из непустого множества *состояний* S, *начального состояния* $s_0, s_0 \in S$, и *функции преобразования* $R\colon S \times \tilde{A} \to S$.

Назовем шкалу *упорядоченной*, если отношение достижимости состояний является частичным порядком на S.

Для каждой операторной цепочки обозначим через $[h]$ состояние $s = R^*(s_0, h)$, где R^* — естественное расширение функции R на множество конечных последовательностей операторов. Мы ограничимся рассмотрением

только таких шкал, в которых каждое состояние достижимо из начального. Если для любой четверки операторных цепочек h_1, h_2, h_3, h_4 равенства $[h_1] = [h_3]$, $[h_2] = [h_4]$ влекут $[h_1 h_2] = [h_3 h_4]$, то шкала называется *полугрупповой*. Можно рассматривать ее как полугруппу $(S, *)$, в которой операция определяется соотношением $[h_1] * [h_2] = [h_1 h_2]$.

Детерминированной динамической моделью (или просто *моделью*) M сигнатуры $(\tilde{\mathcal{A}}, \mathcal{C})$ назовем пару $\langle \mathcal{F}, \xi \rangle$, в которой $\mathcal{F} = \langle S, s_0, R \rangle$ — шкала сигнатуры $\tilde{\mathcal{A}}$, а $\xi \colon S \to \mathcal{C}$ — *означивание* логических условий.

Пусть $\pi = \langle V, \textbf{вход}, \textbf{выход}, \textbf{тупик}, B, T \rangle$ — некотороя пропозициональная операторная программа, и $M = \langle \mathcal{F}, \xi \rangle$ — модель, базирующаяся на шкале $\mathcal{F} = \langle S, s_0, R \rangle$. Тогда последовательность четверок (конечная или бесконечная)

$$r = (v_0, a_0, s_0, \delta_0), (v_1, a_1, s_1, \delta_1), \ldots, (v_m, a_m, s_m, \delta_m), \ldots$$

называется *вычислением* программы π на модели M, если она удовлетворяет следующим требованиям:

1) $v_0 = \textbf{вход}, a_0 = \lambda, s_0$ — начальное состояние, $\delta_0 = \xi(s_0)$;

2) каждая четверка $(v_i, a_i, s_i, \delta_i), i \geqslant 1$, состоит извершины преобразователя v_i(состояния программы), оператора a_i, состояния данных s_i и логического условия δ_i;

3) для каждого i, $i \geqslant 1$, выполняются соотношения $v_i = T(v_{i-1}, \delta_{i-1})$, $a_i = B(v_i)$, $s_i = R(s_{i-1}, a_i)$, $\delta_i = \xi(s_i)$;

4) эта последовательность оканчивается четверкой $(v_m, a_m, s_m, \delta_m)$ тогда и только тогда, когда $v_m \in \{\textbf{выход}, \textbf{тупик}\}$.

Таким образом определенное вычисление будем обозначать $r(\pi, M)$. Если $r(\pi, M)$ оканчивается четверкой $(\textbf{выход}, a_m, s_m, \delta_m)$, то будем называть это вычисление *терминальным*, а состояние s_m — его *результатом*, который будем обозначать $[r(\pi, M)]$. В остальных случаях вычисление считается безрезультатным, и значение $[r(\pi, M)]$ полагается неопределенным.

Программы π' и π'' назовем *эквивалентными на шкале* $\mathcal{F}$ (и обозначим этот факт $\pi'' \sim_{\mathcal{F}} \pi''$), если для всякой модели $M = \langle \mathcal{F}, \xi \rangle$ выполняется $[r(\pi', M)] = [r(\pi'', M)]$.

Пусть фиксированы 2 шкалы: $\mathcal{F}_{\mathcal{A}} = \langle S_{\mathcal{A}}, s_{\mathcal{A}0}, R_{\mathcal{A}} \rangle$ сигнатуры $\mathcal{A}$ и $\mathcal{F}_{\mathcal{B}} = \langle S_{\mathcal{B}}, s_{\mathcal{B}0}, R_{\mathcal{B}} \rangle$ сигнатуры $\mathcal{B}$, $\mathcal{A} \cap \mathcal{B} = \emptyset$. Шкалу $intr(\mathcal{F}_{\mathcal{A}}, \mathcal{F}_{\mathcal{B}}) = \langle S, s_0, R \rangle$ сигнатуры $\tilde{\mathcal{A}} = \mathcal{A} \cup \mathcal{B}$, где $S = S_{\mathcal{A}} \times S_{\mathcal{B}}$, $s_0 = (s_{\mathcal{A}0}, s_{\mathcal{B}0})$,

$$R((s_{\mathcal{A}}, s_{\mathcal{B}}), a) = \begin{cases} (s_{\mathcal{A}}, R_{\mathcal{B}}(s_{\mathcal{B}}, a)), & a \in \mathcal{B}, \\ (R(s_{\mathcal{A}}, a), s_{\mathcal{B}0}), & a \in \mathcal{A}, \end{cases}$$

назовём *шкалой* основных действий $\mathcal{F}_{\mathcal{A}}$ *с прерываниями* $\mathcal{F}_{\mathcal{B}}$.

Выбор такого определения обусловлен изучаемым свойством обработчиков прерываний — незаметностью их действий для прикладной программы. Рассмотрим выполнение программы, работающей в вычислительной среде, в которой возможно возникновение прерываний (как засчет генерации их самой программой, так и вследствие внешних причин). На время обработки прерывания выполнение основной программы приостанавливается. Как правило, в процессе этой обработки выполняются служебные действия (перераспределение памяти, управление устройствами), не вляющие существенно на основное вычисление. Это случай успешной обработки прерывания. Но возможна ситуация, когда после завершения обработки прерывания управление не возвращается в основную программу (например, при возникновении аварийной ситуации, последствия которой невозможно исправить, нехватке ресурсов, и т.д.). Тогда результат вычисления определяется действиями, выполненными основной программой и действиями последнего цикла обработки прерываний.

В ряде случаев полиномиальный алгоритм для проблемы эквивалентности на упорядоченных шкалах можно получить методом критериальных систем с теми или иными модификациями. Основная идея состоит в том, чтобы отслеживать всевозможные совместные (на одной модели) вычисления пары программ, представляя различие в их состояниях данных в процессе выполнения как элемент особой структуры. Для учёта повторяющихся фрагментов вычисления в процессе работы алгоритма строится граф, каждая вершина которого вершины которого помечена вершинами-преобразователями программ и элементом критериальной системы. Пути в таком графе отвечают совместным вычислениям. О неэквивалентности программ свидетельствует достижимость из начальной вершины вершины, в которой обе программы завершили вычисление с различными состояниями данных. При выполнении определённых требований можно избежать рассмотрения бесконечного множества состояний во всевозможных моделях.

Пусть $\mathcal{F} = \langle S, s_0, R\rangle$ — упорядоченная полугрупповая шкала, и $\prec$ — некоторое отношение строгого частичного порядка на S, отношение достижимости. Рассмотрим некоторую полугруппу W с бинарной операцией $\circ$ и единицей e, в которой выделена подполугруппа U и пара элементов w^+, w^*. Пятерку $K = \langle W, U, w^+, w^*, \varphi\rangle$, где φ — гомоморфизм полугруппы $\mathcal{F} \times \mathcal{F}$ в полугруппу U назовем k_0 *-критериальной системой* для шкалы $\mathcal{F}$, если K и $\mathcal{F}$ удовлетворяют следующим требованиям:

1) для всякой пары состояний s', s'' из S выполняется

$$s' = s'' \Leftrightarrow w^+ \circ \varphi(\langle s', s''\rangle) \circ w^* = e;$$

2) для любых состояний $s_1, s_2, s_3, s_4 \in S$ таких, что $w^+ \circ \varphi(\langle s_1, s_2\rangle) = w^+ \circ \varphi(\langle s_3, s_4\rangle)$, выполняется

$$s_1 \prec s_2 \Leftrightarrow s_3 \prec s_4, \qquad s_1 \succ s_2 \Leftrightarrow s_3 \succ s_4,$$

3) для любого элемента $u'' \in U \circ w^*$ существует не более k_0 различных элементов $u' \in w^+ \circ U$ таких, что $u' \circ u'' = e$

Обоснование и примеры применения метода критериальных систем такого типа можно найти в [1]. Модификации метода описаны в [2, 3].

Идея общего алгоритма проверки эквивалентности на шкалах с прерываниями базируется на том, что в большинстве случаев действия по обработке прерываний не играют никакой роли, т.к. поглощаются основными действиями. Это позволяет разбить задачу на две слабо связанные между собой части. Проблема эквивалентности решается для программ, получающихся из исходных выкидыванием действий по обработке прерываний (с учётом возможности наличия серии непоглощённых операторов обработки прерываний в конце). Для корректного построения переходов в таких программах необходимо решать проблему достижимости или совместной достижимости для фрагментов исходных программ, содержащих только действия по обработке прерываний.

Будем говорить, что у программы $\pi = \langle V, \textbf{вход}, \textbf{выход}, \textbf{тупик}, B, T\rangle$ сигнатуры $(\tilde{\mathcal{A}}, \mathcal{C})$ *заданы финалы* функцией f, если каждому ребру отвечающей этой программе системы переходов, ведущему в вершину **выход**, поставлен в соответствие элемент некоторого множества (*множества финалов*) — т.е. функция f определена на множестве $\{(v, \delta) \mid v \in \{\textbf{вход}\} \cup V, \delta \in \mathcal{C}, T(v, \delta) = \textbf{выход}\}$.

Финалом результативного *вычисления* программы π на модели M назовём элемент множества финалов, отвечающий последнему шагу данного вычисления (обозначим $fin(\pi, v, M)$).

Задача поиска финалов формулируется следующим образом. Пусть дана программа $\pi = \langle V, \textbf{вход}, \textbf{выход}, \textbf{тупик}, B, T\rangle$ сигнатуры $(\tilde{\mathcal{A}}, \mathcal{C})$, у которой заданы финалы и выбрана некоторая вершина $v_0 \in \{\textbf{вход}\} \cup V$. Требуется построить множество $finals(\pi, v_0) = \bigcup_\xi \{fin(\pi, v_0, \langle \mathcal{F}, \xi\rangle)\}$, где ξ пробегает по всем функциям означивания. По определению положим $finals(\pi, \textbf{тупик}) = \{\textbf{тупик}\}$.

Задача поиска совместных финалов формулируется так. Пусть даны 2 программы $\pi_i = \langle V_i, \textbf{вход}, \textbf{выход}, \textbf{тупик}, B_i, T_i\rangle, i = 1, 2$ сигнатуры $(\tilde{\mathcal{A}}, \mathcal{C})$, у которых заданы финалы (причем множества финалов могут содержать элемент **выход** — ему придается особый статус) и выбраны некоторые вершины $v_{i0} \in \{\textbf{вход}\} \cup V_i$. Требуется построить множество $\bigcup_\xi \{t(\xi)\}$, где ξ

пробегает по всем функциям означивания, а

$$t(\xi) = \begin{cases} (\textbf{выход}, \textbf{выход}, 1), & fin(\pi_1, v_{10}, \langle \mathcal{F}, \xi \rangle) = fin(\pi_2, v_{20}, \langle \mathcal{F}, \xi \rangle) = \textbf{выход}, \\ & [r(\pi_1, v_{10}, \langle \mathcal{F}, \xi \rangle)] = [r(\pi_2, v_{20}, \langle \mathcal{F}, \xi \rangle)], \\ (\textbf{выход}, \textbf{выход}, 0), & fin(\pi_1, v_{10}, \langle \mathcal{F}, \xi \rangle) = fin(\pi_2, v_{20}, \langle \mathcal{F}, \xi \rangle) = \textbf{выход}, \\ & [r(\pi_1, v_{10}, \langle \mathcal{F}, \xi \rangle)] \neq [r(\pi_2, v_{20}, \langle \mathcal{F}, \xi \rangle)], \\ (fin(\pi_1, v_{10}, \langle \mathcal{F}, \xi \rangle), & \\ \ fin(\pi_2, v_{20}, \langle \mathcal{F}, \xi \rangle)), & \text{иначе.} \end{cases}$$

Теорема 1. *Задачи поиска финалов и поиска совместных финалов разрешимы на упорядоченной полугрупповой шкале, для которой имеется k_0 — критериальная система.*

Теорема 2. *Пусть фиксированы 2 шкалы: $\mathcal{F}_\mathcal{B}$ сигнатуры $\mathcal{B}$ и полугрупповая упорядоченная шкала $\mathcal{F}_\mathcal{A}$ сигнатуры $\mathcal{A}$. Пусть существует алгоритмы, решающие задачи поиска финалов и поиска совместных финалов для шкалы $\mathcal{F}_\mathcal{B}$ с временной сложностью $O(f(n))$, где n — размер входных программ. Тогда проблема эквивалентности на шкале $intr(\mathcal{F}_\mathcal{A}, \mathcal{F}_\mathcal{B})$ разрешима за время $O(n^2 f(n))$.*

Это позволяет эффективно проверять эквивалентность программ на составных шкалах, для которых не построено критериальных систем, если критериальные системы имеются для компонент таких шкал. Например, для шкалы на базе полугруппы, порожденной тождествами поглощения $[ba] = [a]$, $a \in \mathcal{A}$, $b \in \mathcal{B}$ и тождествами перестановочности $[a_1 a_2] = [a_2 a_1], (a_1, a_2) \in Comm$, где $Comm \subseteq (\mathcal{A} \times \mathcal{A}) \cup (\mathcal{B} \times \mathcal{B})$, предложенный алгоритм имеет сложность $O(n^4)$, где n — размер входных программ.

Список литературы

1. Захаров В.А. Быстрые алгоритмы разрешения эквивалентности пропозициональных операторных программ на упорядоченных полугрупповых шкалах. // Вестник Московского университета, сер. 15, Вычислительная математика и кибернетика, N 3, 1999, часть 1, с. 29–35.

2. Захаров В.А. Быстрые алгоритмы разрешения эквивалентности операторных программ на уравновешенных шкалах. // Математические вопросы кибернетики, Физматлит, 1998, вып. 7, с. 303–324.

3. Zakharov V., Zakharyaschev I. On the equivalence-checking problem for a model of programs related with multi-tape automata. // Lecture Notes in Computer Science, v. 3317, 2005, p.293–305.

НЕСУЩЕСТВОВАНИЕ ДВОИЧНЫХ КОДОВ, РАВНОМЕРНО РАСПРЕДЕЛЕННЫХ ПО ШАРАМ ПОЧТИ ВСЕХ МОЩНОСТЕЙ

М. С. Ярыкина (Москва)

Двоичные коды, равномерно распределенные по подкубам, изучались в нескольких областях математики и в приложениях, например, двоичные коды с наибольшим дуальным расстоянием, корреляционно-иммунные и устойчивые функции, ортогональные массивы. Такие коды используются для генерации псевдослучайных последовательностей, в криптографии и т. д. Равномерное распределение двоичных наборов по шарам не изучалось ранее работы [1], хотя представляется, что коды, двоичные наборы которых равномерно распределены по сферам, могут иметь некоторые полезные приложения. Например, такие коды можно использовать в качестве хеширующей функции, а также когда мы хотим, чтобы все слова на выходе связи имели примерно одинаковую вероятность декодирования.

Кодом C (множеством двоичных наборов) назовем произвольное подмножество булевого куба размерности n. *Мощностью* $|C|$ называется число двоичных наборов в нем. *Расстоянием* $d(x, y)$ между двумя наборами x и y называется число компонент, в которых эти наборы различаются. *Шаром* $S_r(x)$ с центром $x \in V^n$ радиуса r называется множество $S_r(x) = \{y \in V^n \,|\, d(x, y) \leqslant r\}$. *Весом* $wt(S_r(x), C)$ шара $S_r(x)$ для кода C называется мощность множества $S_r(x) \cap C$.

Определение. *Пусть l — натуральное. Код $C \subseteq V^n$ называется* равномерно распределенным *по шарам со степенью l (или l-РРШ кодом), если для любых $x, y \in V^n, 0 \leqslant r \leqslant n$, выполняется*

$$|wt((S_r(x), C) - wt((S_r(y), C)| \leqslant l.$$

Полное описание 1-РРШ кодов было дано в [1], в т. ч. было получено количество 1-РРШ кодов для любого n.

Утверждение 1. [1] *Пусть $C \subseteq V^n$, $|C| \leqslant 2^{n-1}$. Если C — 1-РРШ код, то выполняется один из случаев:*

$$1)\ |C| \leqslant 2; \qquad 2)\ n \leqslant 4; \qquad 3)\ n = 6, \quad |C| = 4.$$

Теорема 1. (*Коды малой мощности*) *Пусть $l \in \mathbb{N}$ и $m = m(n) \geqslant 2l + 1$. Тогда, для достаточно больших n не существует l-*РРШ *кодов мощности m при* $\frac{m}{\sqrt{n}e^{\frac{n}{4^{l+1}}}} \underset{n\to\infty}{\longrightarrow} 0$.

Доказательство полностью приведено в [2]. Идея доказательства состоит в том, что в условиях теоремы доказывается существование шаров радиуса R веса 0 и веса не менее $l+1$, где $R = \lambda(n)n$ и $\lambda(n) \to 1/2-0$.

Теорема 2. (*Коды средней мощности*) *Пусть l — фиксированное натуральное число. Тогда существует некоторое $k_0(l)$ такое, что при достаточно больших n не существует кодов, равномерно распределенных по шарам со степенью l, мощности m, удовлетворяющего неравенствам:*

$$8nl < m < \frac{2^n}{\sum\limits_{i=0}^{k_0(l)} \binom{n}{i}}.$$

Замечание. В случае $l = 2$ имеем $k_0 = 52$.

Для доказательства теоремы нам понадобятся следующие леммы.

Лемма 1. *Пусть все шары радиуса k имеют вес не более, чем l. Тогда существует шар радиуса $\left\lfloor \frac{2^l}{2^l-1} k \right\rfloor$ имеющий вес не более, чем l.*

Лемма 2. *Пусть l — фиксированное натуральное число. Тогда найдется такое k_0, что при $k > k_0$, $n \geqslant 2\frac{2^l}{2^l-1}k + 2^l + 2$ выполнено неравенство*

$$\sum_{i=0}^{\lfloor \frac{2^l}{2^l-1} k \rfloor} \binom{n}{i} > 2l \sum_{i=0}^{k+1} \binom{n}{i}.$$

Замечание. В случае $l = 2$ имеем $k_0 = 52$.

Лемма 3. *Булев куб размерности n можно покрыть (не более, чем) $4n$ шарами радиуса $R = \frac{n}{2} - 2^l - 1$ для достаточно больших n.*

Доказательство теоремы 2. Предположим, что l-РРШ код (в булевом кубе размерности n) веса m существует. Из условия следует, что существует шар радиуса $k = k_0$ веса 0. Докажем по индукции, что существует шар радиуса $k+1$ веса 0. База индукции: $k = k_0$.

Шаг индукции. Если существует шар радиуса k веса 0, то по лемме 1, существует шар радиуса $R \geqslant \left\lfloor \frac{2^l}{2^l-1} k \right\rfloor$ веса не более, чем l, поэтому любой шар радиуса R имеет вес не больше, чем $2l$.

Если $n < 2 \cdot \frac{2^l}{2^l-1} k + 2^l + 2$, то по лемме 3 весь булев куб покрываем $4n$ шарами радиуса R. Значит, вес кода не превосходит $2l + l2(4n-1)$, что противоречит условию теоремы. Если $n \geqslant 2 \cdot \frac{2^l}{2^l-1} k + 2^l + 2$, то по лемме 2

средний вес шара радиуса $k+1$ меньше, чем $\frac{1}{2l}$ среднего веса $R_C^{\frac{2^l}{2^l-1}k}$ шара радиуса не менее $\frac{2^l}{2^l-1}k$. Значит, существует шар радиуса $k+1$ веса 0.

Поскольку k растет, а n фиксированно, когда-нибудь выполнится условие $n < 2 \cdot \frac{2^l}{2^l-1}k + 2^l + 2$, то есть на каком-то шаге индукции мы придем к противоречию. Теорема доказана.

Коды большой мощности. В случае кодов большой мощности мы выделим два семейства мощностей. Для каждого семейства мощностей у нас будет свой способ доказательства.

Теорема 3. (*Первое семейство*) *Пусть* $l \in \mathbb{N}$, $s \in \mathbb{N}$, $u > 1$, c_s — *некоторая константа* (*своя для каждого* s) *и* m *такое что*

$$\frac{ul^2}{4}\frac{2^n}{\sum\limits_{i=0}^{s}\binom{n}{i}} \leqslant m \leqslant \left(\frac{n}{s^2}+c_s\right)\frac{2^n}{\sum\limits_{i=0}^{s}\binom{n}{i}}, \qquad \text{для } s \geqslant 2,$$

$$\frac{ul^2}{4}\cdot\frac{2^n}{n+1} \leqslant m \leqslant 2^{n-1}, \qquad \text{для } s = 1.$$

Тогда для достаточно больших n *не существует* l-РРШ *кодов мощности* m. *Кроме того, в случае* $s = 1$ *не существует* l-РРШ *кодов для* n, *удовлетворяющих следующим условиям:*

$$n > \frac{u}{u-1}\left(3l+1+\frac{ul^2}{4}\right), \quad n \geqslant 6l+3+\frac{ul^2}{2}.$$

Доказательство теоремы приведено в [2]. Заметим, что коды мощности m, удовлеторяющие случаю $s = 1$ — это почти все двоичные коды размерност n. Уравновешенные коды также относятся к этому случаю.

Лемма 4. *Пусть* $x_1, \ldots, x_k$ — *произвольные натуральные числа, такие, что* $x_1 + \cdots + x_k = S > 0$ *и* $\max\{x_i\} - \min\{x_i\} \leqslant l$. *Тогда*

$$\frac{S^2}{k} \leqslant x_1^2 + \cdots + x_k^2 \leqslant \frac{S^2}{k} + \frac{kl^2}{4}.$$

Теорема 4. (*Второе семейство*) *Пусть* $l \in \mathbb{N}$, $s \in \mathbb{N}$, λ_1, λ_2 — *некоторые положительные числа и* m *такое что*

$$\lambda_1\frac{2^n}{\sum\limits_{i=0}^{s}\binom{n}{i}} \leqslant m \leqslant \lambda_2\frac{2^n}{\sum\limits_{i=0}^{s}\binom{n}{i}}$$

Тогда для достаточно больших n *не существует* l-РРШ *кодов мощности* m.

Замечание. Числа λ_1 и λ_2 выбирает так, чтобы первое и второе семейства мощностей пересекались.

Доказательство. Пусть C — l-РРШ код размерности n, мощность которого равна m. Оценим число пар кодовых слов во всех шарах радиуса s двумя способами. Обозначим через V_k объем шара радиуса k.

Первый способ. Обозначим x_1, x_2, …, x_{2^n} — веса шаров радиуса s. Так как каждое кодовое слово содержится ровно в $V_s = \sum\limits_{i=0}^{s} \binom{n}{i}$ шарах радиуса s, то $S := \sum\limits_{i=0}^{2^n} x_i = mV_s$. Число пар кодовых слов в шаре радиуса s веса x_i равно $\frac{x_i(x_i-1)}{2}$, соответственно, число пар кодовых слов во всех шарах радиуса s равно $N = \sum\limits_{i=0}^{2^n} \frac{x_i(x_i-1)}{2} = \frac{1}{2}\sum\limits_{i=0}^{2^n} x_i^2 - \frac{1}{2}\sum\limits_{i=0}^{2^n} x_i$.

Поскольку C является l-РРШ кодом, то $\max\{x_i\} - \min\{x_i\} \leqslant l$. Значит, по лемме 1 и выражая S^2 через m, n и s получаем:

$$N \leqslant \frac{m^2}{2\cdot 2^n}\cdot\frac{1}{(s!)^2}\left(n^{2s} + (3s-s^2)n^{2s-1} + O(n^{2s-2})\right) - \\ - \frac{m}{2}\cdot\frac{1}{s!}\left(n^s + \frac{3s-s^2}{2}n^{s-1} + O(n^{s-2})\right) + \frac{2^n l^2}{8}.$$

Второй способ. В шарах радиуса s содержатся пары кодовых слов на расстоянии от 1 до $2s$. Мы рассмотрим шары радиуса $2s$ с центром в кодовых словах. Разобьем эти шары на сферы радиусом от 1 до $2s$, оценим вес каждой сферы, и соответственно, оценим число пар кодовых слов на расстоянии 1, 2 и так далее до $2s$ отдельно.

В дальнейшем, для более компактной записи выкладок, мы будем писать $t = x \pm 2l$ вместо двойного неравенства $x - 2l \leqslant t \leqslant x + 2l$.

Средний вес шара радиуса k равен $P_k = \frac{mV_k}{2^n}$, вес произвольного шара $S_k(x)$ равен $wt(S_k(x)) = \frac{mV_k}{2^n} \pm l$, а вес произвольной сферы ρ_k равен $wt(\rho_k) = \frac{mV_k}{2^n} - \frac{mV_{k-1}}{2^n} \pm 2l$.

Обозначим через N_k число пар двоичных наборов кода C на расстоянии k, а p_k — число шаров радиуса s, в которых одновременно содержится пара двоичных наборов на расстоянии k. Искомое число N пар кодовых слов в сумме во всех шарах радиуса s равно

$$N = \sum_{k=1}^{2s} N_k p_k = \frac{m^2}{2\cdot 2^n}\cdot\frac{1}{(s!)^2}\left(n^{2s} + (3s-s^2)\,n^{2s-1} + O(n^{2s-2})\right) \pm \frac{m^2}{2\cdot 2^n}\cdot 2l\cdot O(n^{s-1}).$$

Рассмотрим случай $m = \lambda\cdot\frac{2^n}{n^{s-1}}\cdot(s-1)!$. Тогда мы получаем, что оценки $N = N'$, полученная первым способом и $N = N''$, полученная вторым способом, имеют вид

$$N \leqslant N' = \frac{\lambda^2 \cdot 2^n}{2s^2}\left(n^2 + (3s - s^2)n - \frac{s}{\lambda}n + O(1)\right),$$

$$N = N'' = \frac{\lambda^2 \cdot 2^n}{2s^2}\left(n^2 + \left(3s - s^2\right)n + O(1)\right).$$

Поскольку одна оценка отличается от другой наличием слагаемого $-\frac{s}{\lambda}n$, начиная с некоторого n мы получим, что оценка N' меньше оценки N''. Противоречие. Теорема доказана.

Список литературы

1. Таранников Ю. В. О классе булевых функций, равномерно распределенных по шарам со степенью 1. Вестник Московского Университета. Серия 1. Математика. Механика. 1997, вып. 52, №5, стр. 18–22.

2. Ярыкина М. С. Применение оценок для сумм биномиальных коэффициентов при решении некоторых задач теории кодирования и криптографии. Математические вопросы кибернетики. Вып. 12. — М.: Физматлит, 2003. — С. 87–108.