Московский государственный университет им. М. В. Ломоносова
Казанский (Приволжский) федеральный университет
Институт прикладной математики им. М. В. Келдыша РАН
Вычислительный центр им. А. А. Дородницына РАН

Проблемы теоретической кибернетики

Материалы XVII международной конференции
(Казань, 16 – 20 июня 2014 г.)

Казань, 2014

ББК 22.18
П 78
УДК 519.7

Издание осуществлено при финансовой поддержке Российского фонда фундаментальных исследований по проекту 14-01-06036-г

П78 Проблемы теоретической кибернетики. Материалы XVII международной конференции (Казань, 16 – 20 июня 2014 г.). Под редакцией Ю.И. Журавлева. — Казань: Отечество, 2014. — 307 с.

ISBN 978-5-9222-0861-1

Сборник содержит доклады XVII международной конференции “Проблемы теоретической кибернетики” (Казань, 16 – 20 июня 2014 г.), организованной при поддержке Российского фонда фундаментальных исследований (проект 14-01-06036-г).

Для научных работников и специалистов в области математической кибернетики, дискретной математики, информатики и их приложений.

Научное издание

ПРОБЛЕМЫ ТЕОРЕТИЧЕСКОЙ КИБЕРНЕТИКИ
МАТЕРИАЛЫ XVII МЕЖДУНАРОДНОЙ КОНФЕРЕНЦИИ
(Казань, 16 – 20 июня 2014 г.)

Под общей редакцией академика РАН Ю. И. Журавлева

Редакционная группа
Ф. М. Аблаев, В. Б. Алексеев, О. М. Касим-Заде

Ответственный за выпуск: Ф. М. Аблаев

ISBN 978-5-9222-0861-1

Оглавление

Квантовое хеширование

Ф. М. Аблаев, А. В. Васильев

alexander.ksu@gmail.com

Казанский федеральный университет, Казань

Квантовое хеширование

Нами предлагается метод криптографического квантового хеширования, позволяющий представлять классическую информацию в виде квантовой суперпозиции специального вида, т.е. являющийся *классически-квантовой функцией.*

Классически-квантовой функцией будем называть функцию вида

$$\psi : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes s}, \tag{1}$$

где $(\mathcal{H}^2)^{\otimes s} = \mathcal{H}^2 \otimes \cdots \otimes \mathcal{H}^2 = \mathcal{H}^{2^s}$ обозначает 2^s-мерное гильбертово пространство, описывающие состояния s кубит. Наряду с обозначением выше удобно применять часто используемое в работах по квантовой информатике обозначение

$$\psi : w \mapsto |\psi(w)\rangle \tag{2}$$

для ψ.

Опишем некоторые свойства, которыми должна обладать функция ψ, чтобы считаться квантовой криптографической хеш-функцией.

В первую очередь, квантовая хеш-функция ψ должна быть квантовой односторонней функцией, т.е.

— ψ эффективно вычислима. Термин "эффективно вычислима" означает, что существует полиномиальный (от длины входного набора) по времени алгоритм, который на входном наборе w по фиксированному начальному состоянию $|\psi(e)\rangle$ квантовой системы формирует (строит) состояние $|\psi(w)\rangle$.

— функцию ψ тяжело (вычислительно сложно) обратить, т.е., имея лишь $|\psi(w)\rangle$, невозможно эффективно достоверно получить w.

Кроме того, необходимо наличие дополнительного свойства, обеспечивающего устойчивость к квантовым коллизиям. Необходимость введения отдельного понятия *квантовая коллизия* заключается в том, что при квантовом хешировании коллизии в классическом понимании могут отсутствовать, т.к. порождаемые квантовой хеш-функцией состояния различны для различных исходных сообщений. Однако сравнение квантовых хеш-кодов подразумевает выполнение вероятностной процедуры измерения квантовых состояний, что может приводить к ошибкам, связанным с коллизиями.

Квантовой коллизией будем называть ситуацию, когда процедура, проверяющая равенство квантовых хеш-кодов, ошибочно выдает совпадение исходных сообщений, в то время как они были различны. Такой процедурой может быть хорошо известный SWAP-тест [1] или специфический для квантовой хеш-функции алгоритм. В любом случае такая проверка связана с понятием различимости квантовых состояний. И поскольку неортогональные

состояния не могут быть достоверно различены, потребуем, чтобы они были "почти ортогональны". Для формализации данного понятия вводится следующее определение.

Определение 1. Состояния $|\psi_1\rangle$ и $|\psi_2\rangle$ называются *δ-ортогональными*, если

$$|\langle\psi_1|\psi_2\rangle| < \delta. \tag{3}$$

Таким образом, для квантовой хеш-функции важна δ-ортогональность квантовых хеш-кодов различных слов, т.е. они должны успешно проходить тесты на неравенство. Рассмотрим подробнее возможные процедуры проверки равенства.

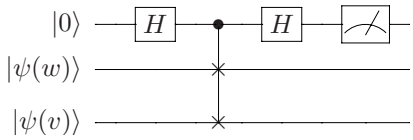
REVERSE-тест. В тех случаях, когда необходимо проверить, является ли квантовое состояние $|\psi(w)\rangle$ хеш-кодом некоторого классического слова v, можно применить процедуру, которую мы называем *REVERSE-тест*. Суть теста заключается в применении инвертированной процедуры создания квантового хеша, т.е. его "раскручивание до начального" состояния.

Формально, пусть процедура создания хеш-кода слова w состоит в применении унитарного преобразования $U(w)$, к начальному состоянию $|0\rangle$, т.е. $|\psi(w)\rangle = U(w)|0\rangle$. Тогда Reverse-тест заключается в применении $U^{-1}(v)$ к к квантовому хешу $|\psi(w)\rangle$ и проверке полученного состояния. Если $v = w$, то результатом преобразования $U^{-1}(v)|\psi(w)\rangle$ всегда будет $|0\rangle$, и REVERSE-тест выдаст равенство, в противном случае результирующее состояние будет δ-ортогонально к $|0\rangle$, поскольку унитарные преобразования сохраняют скалярное произведение.

Предлагаемый вариант квантового хеширования позволяет применять REVERSE-тест со сколь угодно малой вероятностью ошибки.

SWAP-тест.

Для сравнения двух квантовых квантовых состояний (в частности, хеш-кодов) часто используется SWAP-тест [1], задаваемый следующей схемой:



Данный тест выдает результат $|\psi(w)\rangle = |\psi(v)\rangle$, если при измерении первый кубит оказывается в состоянии $|0\rangle$.

Свойство 1. *Вероятность получения состояния $|0\rangle$ в результате SWAP-теста равна*

$$\frac{1}{2}\left(1 + |\langle\psi(w)|\psi(v)\rangle|^2\right). \tag{4}$$

Таким образом, при $|\psi(w)\rangle = |\psi(v)\rangle$ данный тест не ошибается, а в случае $|\psi(w)\rangle \neq |\psi(v)\rangle$ вероятность ошибки зависит от скалярного произведения $|\psi(w)\rangle$ и $|\psi(v)\rangle$ – она минимальна (близка к $1/2$), если эти состояния будут ортогональны или "почти ортогональны" [2].

Таким образом, свойство δ-ортогональности квантовых состояний является важным качеством для обеспечения устойчивости к квантовым коллизиям, поэтому вводится понятие δ-устойчивости.

Определение 2 (δ-устойчивость). Назовем хеш-функцию $\psi : w \mapsto |\psi(w)\rangle$ δ-устойчивой, если для любой пары сообщений w, w', $w \neq w'$ выполняется:

$$|\langle\psi(w)\,|\,\psi(w')\rangle| < \delta, \tag{5}$$

т.е. их образы δ-ортогональны.

На основе понятий односторонности и δ-устойчивости введем следующее определение классически-квантовой хеш-функции.

Определение 3 ((n, s, δ)-квантовая хеш-функция). Назовем функцию $\psi : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes s}$ (n, s, δ)-квантовой хеш-функцией (кратко (n, s, δ)-КХФ), если она является квантовой односторонней и δ-устойчивой функцией.

На основе предложенного нами ранее квантового метода отпечатков нами построен пример функции, удовлетворяющей Определению 3.

Пусть $q = 2^n$, $B = \{b_i : b_i \in \{0, \ldots, q-1\}\}$. Классически-квантовая функция $\psi_{q,B} : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes(\log|B|+1)}$ определяется следующим образом. Для сообщения $w \in \{0,1\}^n$ полагают

$$|\psi_{q,B}(w)\rangle = \frac{1}{\sqrt{|B|}} \sum_{i=1}^{|B|} |i\rangle \left(\cos\frac{2\pi b_i w}{q}|0\rangle + \sin\frac{2\pi b_i w}{q}|1\rangle\right). \tag{6}$$

Теорема 1. *Для произвольного $\delta > 0$ существует такое множество $B \subset \mathbb{Z}_q$ такое, что $|B| = \lceil(2/\delta^2)\ln(2q)\rceil$ и функция $\psi_{q,B}$ является $(n, O(\log n + \log 1/\delta), \delta)$-квантовой хеш-функцией.*

Предложенная выше квантовая хеш-функция совместно с представлением булевых функций в виде характеристических полиномов [3] позволяет строить эффективные алгоритмы в модели квантовых один раз читающих ветвящихся программ (квантовых OBDD) [4]. А именно, справедлива следующая теорема.

Теорема 2. *Пусть для булевой функции f существует линейный характеристический полином над кольцом $\mathbb{Z}_q$. Тогда для произвольного $\delta \in (0,1)$ функция f может быть вычислена квантовой OBDD с использованием $O(\log\log q + \log 1/\delta)$ кубит.*

Работа выполнена при поддержке РФФИ, проект № 14-07-00878-а.

Литература

[1] Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Phys. Rev. Lett.*, 87(16):167902, Sep 2001.

[2] Daniel Gottesman and Isaac Chuang. Quantum digital signatures. Technical Report arXiv:quant-ph/0105032, Cornell University Library, Nov 2001.

[3] Farid Ablayev and Alexander Vasiliev. Algorithms for quantum branching programs based on fingerprinting. *Electronic Proceedings in Theoretical Computer Science*, 9:1–11, 2009.

[4] Farid Ablayev, Aida Gainutdinova, and Marek Karpinski. On computational power of quantum branching programs. In *FCT*, pages 59–70, 2001.

Квантовое хеширование на основе классических ε-универсальных хеш-семейств

Ф. М. Аблаев, М. Ф. Аблаев

fablayev@gmail.com, mablayev@gmail.com

Казанский Федеральный Университет, Казань

Квантовые модели алгоритмов обратили на себя особое внимание после разработки П. Шором быстрого (полиномиального по времени) квантового алгоритма факторизации, представляющего угрозу (в случае создании полномасштабного квантового компьютера) современным системам передачи данных с открытым ключом типа RSA. Ответом криптографического сообщества является разработка направления пост-квантовая криптография (Post-quantum cryptography), которая, в частности, ведет разработки систем цифровой подписи на основе хеш-функций.

В данной работе мы предлагаем систему построения квантовых хеш-функций, которые могут быть использованы для построения квантовых систем цифровой подписи. Мы используем следующую систему понятий и обозначений из работы [3].

— Функцию $f : \mathbb{X} \to \mathbb{Y}$, с $|\mathbb{X}| = K$ и $|\mathbb{Y}| = M$ называют хеш-функцией, если $K > M$.

— Семейство $F = \{f_1, \ldots, f_N\}$ хеш-функций $f_i : \mathbb{X} \to \mathbb{Y}$ называют ε-универсальным, если для любой пары w, w' различных элементов из $\mathbb{X}$ выполняется $|f \in F : f(w) = f(w')| \leqslant \varepsilon N$.
Применяют обозначение ε-U $(N; K, M)$ для такого ε-универсального семейства F.

— Через $(\mathcal{H}^2)^{\otimes s}$ обозначают 2^s-мерное комплексное пространство состояний системы, образованной из s кубитов.

— Функция $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ (другое обозначение $\psi : w \mapsto |\psi(w)\rangle$) называется $(K; s)$ классически-квантовой односторонней функцией, если ψ эффективно вычисляется и является необратимой.

Лемма 1. *Если* $\log K > s$, *то* $(K; s)$ *классически-квантовая функция* ψ *является односторонней.*

— Функцию $\psi : w \mapsto |\psi(w)\rangle$ будем называть δ-устойчивой (resistant), если для каждой пары w, w' различных элементов из $\mathbb{X}$ выполняется

$$|\langle \psi(w) \mid \psi(w') \rangle| \leqslant \delta.$$

Теорема 1. *Если функция* $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ δ-*устойчива. то*

$$s \geqslant \log\log|\mathbb{X}| - \log\log\left(1 + \sqrt{2/(1-\delta)}\right) - 1.$$

Определение 1 (Квантовая хеш-функция). Пусть $K = |\mathbb{X}|$, $s \geqslant 1$. Одностороннюю δ-устойчивую функцию $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ будем называть δ-устойчивой $(K; s)$-квантовой хеш-функцией (δ-R $(K; s)$-квантовой хеш-функцией).

Квантовый хеш генератор Рассмотрим функцию $g : \mathbb{X} \to \mathbb{F}_q$, пусть $\ell \geqslant 1$. Пусть ψ_g классически-квантовая функция $\psi_g : \mathbb{X} \to (\mathcal{H}^2)^{\otimes \ell}$ определяется условием

$$\psi_g : w \mapsto |\psi_g(w)\rangle = \sum_{i=1}^{2^\ell} \alpha_i(g(w))|i\rangle, \tag{1}$$

при этом амплитуды $\alpha_i(g(w))$, $i \in \{1, \ldots, 2^\ell\}$, состояния $|\psi_g(w)\rangle$ определяются функцией $g(w)$ и (по крайней мере) должны удовлетворять общему условию

$$\sum_{i=1}^{2^l} |\alpha_i(g(w))|^2 = 1.$$

Определение 2 (Квантовый хеш генератор). Пусть $K = |\mathbb{X}|$ и $G = \{g_1, \ldots, g_D\}$ – это семейство функций вида $g_j : \mathbb{X} \to \mathbb{F}_q$. Пусть $\ell \geqslant 1$. Для $g \in G$ пусть ψ_g классически-квантовая функция $\psi_g : \mathbb{X} \to (\mathcal{H}^2)^{\otimes \ell}$. Пусть $d = \log D$.

Определим классически-квантовую функцию $\psi_G : \mathbb{X} \to (\mathcal{H}^2)^{\otimes (d+\ell)}$ следующим образом

$$\psi_G : w \mapsto |\psi_G(w)\rangle = \frac{1}{\sqrt{D}} \sum_{j=1}^{D} |j\rangle \big|\psi_{g_j}(w)\big\rangle. \tag{2}$$

Будем говорить, что семейство G является δ-R $(K; d + \ell)$ квантовым хеш генератором, если классически-квантовая функция ψ_G является δ-R $(K; d + \ell)$ квантовой хеш функцией.

В работе [1] предложена конструкция, которая для произвольного $\delta \in (0, 1)$, по числу q, которое является степенью простого числа задает семейство функций $H_{\delta,q} = \{h_1, \ldots, h_T\}$, $h_i : \mathbb{F}_q \to \mathbb{F}_q$, являющихся в соответствии с приведенным выше определением δ-R $(q; \log T + 1)$ квантовым хеш генератором.

Квантовые хеш генераторы на основе конструкций ε-универсальных хеш семейств и приложения Пусть $K = |\mathbb{X}|$, $M = |\mathbb{Y}|$. Пусть $F = \{f_1, \ldots, f_N\}$ семейство функций вида $f_i : \mathbb{X} \to \mathbb{Y}$.

Пусть $\mathbb{F}_q$ – конечное поле (q – степень простого числа). Пусть $H = \{h_1, \ldots, h_T\}$ – семейство функций вида $h_j : \mathbb{Y} \to \mathbb{F}_q$. Для функций $f \in F$ $h \in H_B$ рассмотрим их композицию $g = f \circ h$, $g : \mathbb{X} \to \mathbb{F}_q$, вида $g(w) = (f \circ h)(w) = h(f(w))$. Определим семейство G как композицию двух семейств F and H условием $G = F \circ H = \{g = f \circ h : f \in F, h \in H\}$.

Доказана следующая теорема.

Теорема 2. *Пусть* $F = \{f_1, \ldots, f_N\}$ *– это* ε*-универсальное хеш семейство* (ε-$U\ (N; K, M)$). *Пусть* $\ell \geqslant 1$. *Пусть семейство* $H = \{h_1, \ldots h_T\}$ *является* δ-R $(M; \log T + \ell)$ *квантовым хеш генератором. Пусть* $\log K > \log N + \log T + \ell$.

Тогда семейство $G = F \circ H$ *является* Δ-R $(K; s)$ *квантовым хеш генератором с параметрами:*

$$s = \log N + \log T + \ell \quad \text{и} \quad \Delta \leqslant \varepsilon + \delta.$$

Пусть $\mathbb{F}_q$ – конечное поле. Пусть $C : (\mathbb{F}_q)^k \to (\mathbb{F}_q)^n$ – линейный $[n, k, d,]_q$ код. Следствием теоремы 2 является следующее утверждение.

Теорема 3. *Пусть* C *– линейный* $[n, k, d]_q$ *код. Тогда для произвольного* $\delta \in (0, 1)$ *функция* $G = F_C \circ H_{\delta,q}$ *является* Δ-R $(q^k; s)$ *квантовым хеш генератором, где* F_C *универсальное хеш семейство, порождаемое кодом* C, *а* $\Delta = (1 - d/n) + \delta$ *and* $s \leqslant \log n + \log\log q + 2\log 1/\delta + 4$.

Например, выберем $[n, k, n-(k-1)]_q$ код Рида-Соломона $C_{RS} : (\mathbb{F}_q)^k \to (\mathbb{F}_q)^n$ с $n \in [ck, c'k]$. Тогда для $\delta \in (0, 1)$ имеем $\Delta \leqslant 1/c + \delta$ и далее в силу Теоремы 1 имеем

$$\log(q\log q) - \log\log\left(1 + \sqrt{2/(1-\Delta)}\right) - \log c'/2 \leqslant s \leqslant \log(q\log q) + 2\log 1/\Delta + 4.$$

Последнее соотношение показывает, что коды Рида-Соломона обеспечивают близкие к оптимальным значения Δ надежности и число s используемых кубит (см. Теорему 1) для конструкции квантовой хеш функции ψ_{RS}.

Литература

[1] F. Ablayev, A. Vasiliev : Cryptographic quantum hashing, Laser Physics Letters Volume 11 Number 2, 2014

[2] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf : Quantum fingerprinting. Phys. Rev. Lett. 87, 167902 (2001)

[3] D.R. Stinson. Universal hash families and the leftover hash lemma, and applications to cryptography and computing. Journal of Combinatorial Mathematics and Combinatorial Computing, Vol. 42 (2002), pp. 3-31

К вопросу о числе дуг в минимальном вершинном 1-расширении турнира

М. Б. Абросимов, О. В. Моденова

mic@rambler.ru, oginiel@rambler.ru

Саратовский государственный университет имени Н.Г. Чернышевского, Саратов

Граф $G^* = (V^*, \alpha^*)$ называется *минимальным вершинным* k*-расширением* (k — натуральное) n-вершинного графа $G = (V, \alpha)$, если выполняются следующие условия:

1. G^* является вершинным k-расширением G, то есть граф G вкладывается в каждый подграф графа G^*, получающийся удалением любых его k вершин;

2. G^* содержит $n+k$ вершин, то есть $|V^*| = |V|$;
3. α^* имеет минимальную мощность при выполнении условий 1) и 2).

Понятие минимального вершинного k-расширения введено на основе понятия оптимальной k-отказоустойчивой реализации, которое было предложено Хейзом в работе [1] при построении модели отказоустойчивости, основанной на графах. Основные определения используются согласно [2].

В работе мы будем рассматривать случай $k = 1$. Через $ec(G)$ обозначим количество дополнительных рёбер (дуг) в минимальном вершинном 1-расширении графа G по сравнению с самим графом G.

Легко убедиться, что единственным минимальным вершинным 1-расширением графа K_n является граф K_{n+1}, причём

$$ec(K_n) = n.$$

Граф $G^* = (V^*, \alpha^*)$ называется *точным вершинным k-расширением* n-вершинного графа $G = (V, \alpha)$, если граф G изоморфен каждому подграфу графа G^*, получающемуся из графа G^* путем удаления любых его k вершин и всех связанных с ними дуг (ребер).

Можно заметить, что минимальное вершинное 1-расширением графа K_n является и его точным вершинным 1-расширением.

В работе [3] доказываются некоторые результаты о связи между минимальными вершинным 1-расширениями неориентированных и ориентированных графов.

Лемма. *Пусть орграф G^* есть минимальное вершинное k-расширение орграфа G. Тогда симметризация орграфа G^* является вершинным k-расширением симметризации орграфа G.*

Следствие. *Число дополнительных дуг минимального вершинного k-расширения орграфа G не менее числа дополнительных ребер минимального вершинного k-расширения симметризации орграфа G.*

Если ориентировать каждое ребро полного графа K_n, то мы получим некоторый турнир $\overrightarrow{T}_n$. Согласно следствию можно сделать вывод:

$$ec(\overrightarrow{T}_n) \geqslant n,$$

то есть число дополнительных дуг минимального вершинного 1-расширения произвольного турнира $\overrightarrow{T}_n$ не может быть меньше n, причем в этом случае минимальное вершинное 1-расширение будет и точным вершинным 1-расширением. Такие турниры существуют, например, транзитивные и некоторые другие турниры (см. [4]). Однако удалось доказать, что не существует турниров с числом дополнительных дуг в минимальном вершинном 1-расширении $n+1$:

Теорема 1. *Если минимальное вершинное 1-расширение турнира $\overrightarrow{T}_n$ не является его точным вершинным 1-расширением, то справедливо неравенство:*

$$ec(\overrightarrow{T}_n) > n+1.$$

Литература

[1] *Hayes J. P.* A graph model for fault-tolerant computing system // IEEE Trans. Comput. — 1976. — Vol.C.25. № 9. — P. 875–884.

[2] *Абросимов М. Б.* Графовые модели отказоустойчивости. — Саратов: Изд-во Сарат. ун-та, 2012. — 192 с.

[3] *Абросимов М. Б.* Минимальные вершинные расширения направленных звезд // Дискрет. матем. — 2011. — Т. 23, № 2. — С. 93–102.

[4] *Абросимов М. Б., Долгов А. А.* Семейства точных расширений турниров // Прикладная дискретная математика. — 2008. — № 1. — С. 101–107.

О билинейной сложности умножения матриц размеров 5×2 и 2×2

В. Б. Алексеев

vbalekseev@rambler.ru

Московский государственный университет им. М.В. Ломоносова

Еще в 1969 году Ф. Штрассен [1] построил первый алгоритм умножения матриц, асимптотически более быстрый, чем стандартный алгоритм «строка на столбец». В последующие 20 лет верхняя оценка сложности умножения двух матриц порядка n была понижена до $O\left(n^{2.38}\right)$ [2], однако дальше (уже 25 лет) существенных продвижений в этой задаче нет. Чтобы лучше понять ситуацию в этой задаче, исследуются различные близкие проблемы, в частности: чему равно минимальное число умножений элементов для перемножения двух матриц малого размера. Это связано, в частности, с тем, что результат Штрассена был основан на найденном им способе перемножения двух матриц размера 2×2 с использованием только 7 умножений вместо 8 в обычном алгоритме.

Более точно, речь идет о билинейных алгоритмах, в которых можно умножать не только элементы исходных матриц, но и линейные комбинации элементов первой матрицы на линейные комбинации элементов второй матрицы. При этом минимально возможное число таких умножений называют билинейной сложностью задачи.

Оказалось,что даже в задачах перемножения двух матриц достаточно малого размера не удается установить точное значение билинейной сложности. Например, для задачи перемножения двух матриц размера 3×3 к настоящему моменту известно только, что билинейная сложность заключена между 19 и 23 [3, 4]. Для задачи перемножения двух матриц размера 4×4 верхняя оценка 49 на число умножений (вместо обычных 64) получается двукратным использованием алгоритма Штрассена, и эта оценка пока не понижена. Для задачи перемножения двух матриц размера 5×5 наилучшим остается алгоритм из [5] с числом умножений 100 вместо обычных 125. Из недавних результатов интересен результат А.В. Смирнова [6], который построил алгоритм для умножения матрицы размера 3×3 на матрицу размера 3×6 с 40 умножениями (вместо обычных 54).

Еще тяжелее обстоит дело с нижними оценками. Для задачи перемножения двух матриц размера 2×2 достаточно быстро было доказано, что оценка 7 на число умножений неулучшаема над произвольным полем [7]. Однако к данному моменту для билинейной сложности умножения матрицы размера $m \times n$ на матрицу размера $n \times p$ нижние оценки над произвольным полем, совпадающие с верхней оценкой, установлены только для нескольких значений параметров m, n, p.

Обозначим через $< m, n, p >_F$ задачу умножения матрицы размера $m \times n$ на матрицу размера $n \times p$ над некоторым полем F. А через $rk_F < m, n, p >$ обозначим билинейную сложность этой задачи. Теорема о двойственности [8] утверждает, что $rk_F < m, n, p >$ не изменяется при любой перестановке чисел m, n, p.

Нетрудно показать, что $rk_F < m, 1, p >= mp$. Из результата Штрассена легко получается, что $rk_F < m, 2, 2 >\leqslant \lceil \frac{7m}{2} \rceil$ для произвольного поля F. В работе [9], была получена такая же нижняя оценка, но только для поля из 2 элементов. Автором в работе [10] был рассмотрен случай $m = 3$ и доказано, что $rk_F < 3, 2, 2 >= 11$ для произвольного поля F. В статье [11] доказано, что $rk_F < 4, 2, 2 >= 14$ для произвольного поля F. Пока только для этих параметров и двойственных к ним установлено точное значение для $rk_F < m, n, p >$ над произвольным полем F. В данной работе рассмотривается величина $rk_F < 5, 2, 2 >$, для которой доказано следующее утверждение.

Теорема 1. *Любой билинейный алгоритм для умножения матрицы размера* 5×2 *на матрицу размера* 2×2 *над произвольным полем имеет билинейную сложность не менее* 17. *Тот же результат справедлив для умножения матрицы размера* 2×5 *на матрицу размера* 5×2 *и для умножения матрицы размера* 2×2 *на матрицу размера* 2×5.

Эта теорема доказывается с помощью методов, являющихся развитием методов автора из [10]. Отметим, что известная верхняя оценка для этих задач равна 18.

Кроме билинейных алгоритмов рассматривают также более общий класс так называемых приближенных билинейных алгоритмов и соответствующую приближенную билинейную сложность (определения см., например, в [11]). Для задачи умножения матрицы размера 5×2 на матрицу размера 2×2 приближенная билинейная сложность не превосходит 16 [11, 6]. Поэтому из теоремы 1 вытекает, что для задачи умножения матрицы размера 5×2 на матрицу размера 2×2 билинейная сложность и приближенная билинейная сложность различаются. Ранее различие билинейной сложности и приближенной билинейной сложности для задачи умножения матриц было установлено только для случаев $< 3, 2, 2 >$ [12, 10] и $< 4, 2, 2 >$ [11] (и двойственных к ним). Совпадение билинейной сложности и приближенной билинейной сложности доказано для задачи $< 2, 2, 2 >$ [13].

Работа выполнена при финансовой поддержке РФФИ, проект № 12-01-91331-ННИО-а.

Литература

[1] *Strassen V.* Gaussian elimination is not optimal // Numer. Math. — 1969. — V. 13. — P. 354–356.

[2] *Coppersmith D., Winograd S.* Matrix Multiplication via Arithmetic Progressions // J. Symbolic Computation. — 1990. — V. 9, № 3. — P. 251–280.

[3] *Laderman J. D.* A noncommutative algorithm for multiplying 3×3 matrices using 23 multiplications // Bull. Amer. Math. Soc. — 1976. — V. 82, № 1. — P. 126–128.

[4] *Bläser M.* On the complexity of the multiplication of matrices of small formats // J. Complexity. — 2003. — V. 19. — P. 43–60.

[5] *Макаров О. М.* Некоммутативный алгоритм умножения квадратных матриц пятого порядка, использующий сто умножений // Журн. выч. матем. и матем. физики. — 1987. — Т. 27, № 2. — С. 311–315.

[6] *Смирнов А. В.* О билинейной сложности и практических алгоритмах умножения матриц // Журн. выч. матем. и матем. физики. — 2013. — Т. 53, № 12. — С. 1970–1984.

[7] *Winograd S.* On multiplication of 2×2 matrices // Linear Algebra and Appl. — 1971. — V. 4. — P. 381–388.

[8] *Hopcroft J. E., Musinski J.* Duality applied to the complexity of matrix multiplication and other bilinear forms // SIAM J. Comput. — 1973. — V. 2, № 3. — P. 159–173.

[9] *Hopcroft J. E., Kerr L. R.* On minimizing the number of multiplications necessary for matrix multiplication // SIAM J. Appl. Math. — 1971. — V. 20, № 1. — P. 127–148.

[10] *Alekseyev V. B.* On the complexity of some algorithms of matrix multiplication // Journal of Algorithms. — 1985. — V. 6, № 1. — P. 71–85.

[11] *Алексеев В. Б., Смирнов А. В.* О точной и приближенной билинейных сложностях умножения матриц размеров 4×2 и 2×2 // Современные проблемы математики. — 2013. — Вып. 17. — С. 135–152.

[12] *Bini D., Capovani M., Lotti G., Romani F.* $O\left(n^{2.7799}\right)$ complexity for approximate matrix multiplication // Inform. Process. Lett. — 1979. — V. 8, № 5. — P. 234–235.

[13] *Landsberg J. M.* The border rank of the multiplication of 2×2 matrices is seven // J. Amer. Math. Soc. — 2006. — V. 19, № 2. — P. 447–459 (electronic).

Синтез надежных схем в P_3

М. А. Алехина, О. Ю. Барсукова

ama@sura.ru, kuzya_7@mail.ru

Пензенский государственный университет, Пенза

Пусть $n \in \mathbf{N}$, а P_3 – множество всех функций трехзначной логики, т. е. функций $f(x_1, ..., x_n) : \{0, 1, 2\}^n \to \{0, 1, 2\}$. Рассмотрим реализацию функций трехзначной логики схемами из ненадежных функциональных элементов в произвольном полном конечном базисе B. Предполагается, что элементы схемы переходят в неисправные состояния независимо друг от друга, а сами неисправности могут быть произвольными (например, инверсными или константными).

Будем считать, что схема из ненадежных элементов реализует функцию $f(\tilde{x}^n)(\tilde{x}^n = (x_1, ..., x_n))$, если при поступлении на входы схемы набора $\tilde{a}^n$ при отсутствии неисправностей в схеме на ее выходе появляется значение $f(\tilde{a}^n)$.

Пусть схема S реализует функцию $f(\tilde{x}^n)$, $\tilde{a}^n$ – произвольный входной набор схемы S, $f(\tilde{a}^n) = \tau$. Обозначим через $P_i(S, \tilde{a}^n)$ вероятность появления значения i $(i \in \{0, 1, 2\})$ на выходе схемы S при входном наборе $\tilde{a}^n$, а через $P_{f(\tilde{a}^n)\neq\tau}(S, \tilde{a}^n)$ – вероятность появления ошибки на выходе схемы S при входном наборе $\tilde{a}^n$. Ясно, что $P_{f(\tilde{a}^n)\neq\tau}(S, \tilde{a}^n) = P_{\tau+1}(S, \tilde{a}^n) + P_{\tau+2}(S, \tilde{a}^n)$ (в выражениях $\tau+1$ и $\tau+2$ сложение осуществляется по $\bmod 3$). Например, если входной набор $\tilde{a}^n$ схемы S такой, что $f(\tilde{a}^n) = 0$, то вероятность появления ошибки на этом наборе равна $P_{f(\tilde{a}^n)\neq 0}(S, \tilde{a}^n) = P_1(S, \tilde{a}^n) + P_2(S, \tilde{a}^n)$.

Ненадежностью схемы S, реализующей функцию $f(\tilde{x}^n)$, будем называть число $P(S)$, равное наибольшей из вероятностей появления ошибки на выходе схемы S. *Надежность* схемы S равна $1 - P(S)$.

Пусть $\tilde{\alpha}^m, \tilde{\beta}^m$ – некоторые троичные наборы длины m $(m \geqslant 3)$. Обозначим через $\rho(\tilde{\alpha}^m, \tilde{\beta}^m)$ число координат, в которых наборы $\tilde{\alpha}^m$ и $\tilde{\beta}^m$ различаются.

Пусть функция $g(\tilde{x}^m)$ обладает следующими свойствами одновременно: существуют такие троичные наборы $\tilde{\alpha}^m$, $\tilde{\beta}^m$, $\tilde{\gamma}^m$, что

1) значения $g(\tilde{\alpha}^m)$, $g(\tilde{\beta}^m)$, $g(\tilde{\gamma}^m)$ попарно различны;

2) для любого набора $\tilde{\alpha}_1^m$ $(\rho(\tilde{\alpha}^m, \tilde{\alpha}_1^m) \leqslant 1)$ верно $g(\tilde{\alpha}_1^m) = g(\tilde{\alpha}^m)$; для любого набора $\tilde{\beta}_1^m$ $(\rho(\tilde{\beta}^m, \tilde{\beta}_1^m) \leqslant 1)$ верно $g(\tilde{\beta}_1^m) = g(\tilde{\beta}^m)$; для любого набора $\tilde{\gamma}_1^m$ $(\rho(\tilde{\gamma}^m, \tilde{\gamma}_1^m) \leqslant 1)$ верно $g(\tilde{\gamma}_1^m) = g(\tilde{\gamma}^m)$.

Наборы $\tilde{\alpha}^m, \tilde{\beta}^m, \tilde{\gamma}^m$ будем называть *характеристическими* наборами функции $g(\tilde{x}^m)$.

Замечание 1. Если $\tilde{\alpha}^m, \tilde{\beta}^m, \tilde{\gamma}^m$ – характеристические наборы функции $g(\tilde{x}^m)$, то $\rho(\tilde{\alpha}^m, \tilde{\beta}^m) \geqslant 3$, $\rho(\tilde{\alpha}^m, \tilde{\gamma}^m) \geqslant 3$, $\rho(\tilde{\beta}^m, \tilde{\gamma}^m) \geqslant 3$.

Обозначим через G_m множество функций $g(\tilde{x}^m)$ с перечисленными свойствами. Нетрудно проверить, что для $|G_m|$ справедливы неравенства.

Теорема 1. $3^{3^m-3m-3} - 3^{3^m-4m-2}(2m^2+1) \leqslant |G_m| \leqslant 3^{3^m-3m-3}$.

Пусть $G = \bigcup_{m=3}^{\infty} G_m$. Пусть функция $g(\tilde{x}^m) \in G$, $\tilde{\alpha}^m, \tilde{\beta}^m, \tilde{\gamma}^m$ – ее характеристические наборы, а S_g – любая схема, реализующая функцию $g(\tilde{x}^m)$. Пусть $f(\tilde{x}^n)$ – произвольная функция. Рассмотрим функции $\varphi_i(y)$ $(i \in \{1, 2, ..., m\})$, которые определяются формулами:

$$\varphi_i(y) = \begin{cases} \alpha_i, & y = 0; \\ \beta_i, & y = 1; \\ \gamma_i, & y = 2. \end{cases}$$

Пусть Φ_i – схема, реализующая функцию $\varphi_i(f(\tilde{x}^n))$ $(i \in \{1, 2, ..., m\})$. Возьмем схемы $\Phi_1, \Phi_2, ..., \Phi_m$ и соединим их выходы со входами схемы S_g. Построенную схему обозначим через A (см. рис. 1).

Теорема 2. *Предположим, что любую функцию $f \in P_3$ можно реализовать схемой с ненадежностью не больше p. Пусть схема S_g реализует функцию $g(\tilde{x}^m) \in G$ с ненадежностью $P(S_g)$, причем v^0, v^1, v^2 – вероятности ошибок схемы S_g на характеристических наборах $\tilde{\alpha}^m, \tilde{\beta}^m, \tilde{\gamma}^m$ соответственно и*

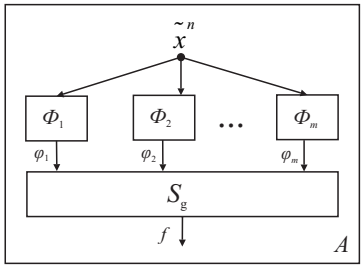


Рис. 1. Схема A

$g(\tilde{\alpha}^m) = 0, g(\tilde{\beta}^m) = 1, g(\tilde{\gamma}^m) = 2$. *Тогда функцию* f *можно реализовать схемой* A *(см. рис. 1), ненадежность которой*

$$P(A) \leqslant \max\{v^0, v^1, v^2\} + mpP(S_g) + (2^m - m - 1)p^2.$$

Доказательство проводится непосредственной проверкой с использованием формулы полной вероятности.

Покажем, как воспользоваться теоремой 2 в случае инверсных неисправностей на выходах базисных элементов. Эти неисправности характеризуются тем, что базисный элемент с функцией $\varphi(\tilde{x}^m)$ на любом входном наборе $\tilde{a}^m$ таком, что $\varphi(\tilde{a}^m) = \tau$, с вероятностью $1 - 2\varepsilon$ ($\varepsilon \in (0, 1/4)$) выдает значение τ, с вероятностью ε выдает значение $\tau + 1 \pmod 3$ и с вероятностью ε выдает значение $\tau + 2 \pmod 3$.

Нетрудно видеть, что ненадежность любого базисного элемента равна 2ε, а надежность – $1 - 2\varepsilon$.

Пусть $P_\varepsilon(f) = \inf P(S)$, где инфимум берется по всем схемам S из ненадежных элементов, реализующим функцию f в базисе B. Схема A из ненадежных элементов, реализующая функцию f, называется *асимптотически оптимальной по надежности*, если $P(A) \sim P_\varepsilon(f)$ при $\varepsilon \to 0$.

Очевидно, что любую из функций $x_1, ..., x_n$ можно реализовать абсолютно надежно, не используя функциональных элементов. Если же схема содержит хотя бы один функциональный элемент, то справедлива теорема 3.

Теорема 3. *Пусть* B *– произвольный конечный полный базис в* P_3, $f(\tilde{x}^n)$ *– любая функция из* P_3, *отличная от функций* $x_1, ..., x_n$, *а* S *– любая схема, реализующая функцию* f. *Тогда при всех* $\varepsilon \in (0, 1/4)$

$$P(S) \geqslant 2\varepsilon.$$

Верхняя оценка ненадежности схем получена в теореме 4.

Теорема 4. *Пусть* B *– конечный полный базис в* P_3. *Тогда любую функцию* $f \in P_3$ *можно реализовать такой схемой* A, *что при всех* $\varepsilon \in (0, 1/(\lambda_1 10^4)]$ *верно неравенство*

$$P(A) \leqslant 2\lambda_2\varepsilon + k_1\varepsilon^2,$$

где λ_1 *– число элементов в схеме, реализующей функцию Вебба,* λ_2 *– число элементов в схеме, реализующей* $g(\tilde{x}^m) \in G$, $k_1 = 17m\lambda_1^2\lambda_2 + 65(2^m - m - 1)\lambda_1^4$.

Доказательство проводится с помощью теоремы 2 и результатов статьи [1].

Из теорем 3 и 4 следует, что в произвольном полном конечном базисе любую функцию трехзначной логики, не равную переменной, можно реализовать схемой, ненадежность которой по порядку равна ε.

В теории надежности управляющих систем *надежной* называют схему, ненадежность которой по порядку равна ненадежности одного элемента. Следовательно, в произвольном полном конечном базисе любую функцию, не равную переменной, можно реализовать надежной схемой.

Схемы, удовлетворяющие условиям теоремы 4, в некоторых базисах и для некоторых функций могут быть не просто надежными, а асимптотически оптимальными по надежности.

Следствие 1. *Пусть B – конечный полный базис в P_3, и пусть $B \cap G_m \neq \varnothing$ при некотором $m \geqslant 3$. Тогда любую функцию f можно реализовать такой схемой A, что при всех $\varepsilon \in (0, 1/(\lambda_1 10^4)]$ верно неравенство*

$$P(A) \leqslant 2\varepsilon + k_2\varepsilon^2,$$

где λ_1 – число элементов в схеме, реализующей функцию Вебба, $k_2 = 17m\lambda_1^2 + 65(2^m - m - 1)\lambda_1^4$.

Таким образом, из теоремы 3 и следствия 1 получаем следующий результат: если конечный полный базис B таков, что $B \cap G \neq \varnothing$, то 1) любую функцию из P_3 можно реализовать схемой, ненадежность которой асимптотически (при $\varepsilon \to 0$) не больше 2ε; 2) для любой функции, отличной от переменной, такая схема является асимптотически оптимальной по надежности и функционирует с ненадежностью, асимптотически равной 2ε при $\varepsilon \to 0$.

Работа выполнена при финансовой поддержке РФФИ (номера проектов 14-01-00273, 14-01-31360).

Литература

[1] *Алехина М. А., Барсукова О. Ю.* Верхняя оценка ненадежности схем в базисе, состоящем из функции Вебба // Материалы девятой международной молодежной школы по дискретной математике и ее приложениям. — М.: Изд-во Института прикладной математики им. М.В. Келдыша РАН, 2013. — С. 9–12.

Метод ветвей и границ для задачи вогнутого программирования

А. А. Андрианова, И. В. Коннов

aandr78@mail.ru, konn-igor@yandex.ru

Казанский (Приволжский) федеральный университет, Казань

Постановка задачи

Рассматривается задача вогнутого программирования в пространстве R_n:

$$\min\{f(x), x \in D\}, \tag{1}$$

где $D = \{x : x \in R_n, f_i(x) \leqslant 0, i = 1 \dots m\}$. Здесь $f(\cdot)$, $f_i(\cdot)$ $i = 1 \dots m$ – вогнутые непрерывно дифференцируемые функции в R_n.

Задача (1), как правило, имеет множество локальных минимумов, отличных от глобального, что серьезно затрудняет решение данной задачи. В данном сообщении предлагается один способ получения глобального решения, основанный на эквивалентности необходимых условий оптимальности задачи (1) и некоторой задачи выпуклого программирования, для определения которой нужно выбрать ее множество допустимых решений. Предлагаемый способ имеет те же основания, что и процедура, примененная в [1] для задачи вогнутого программирования с линейными ограничениями.

Необходимые условия оптимальности задачи вогнутого программирования

Сформулируем для задачи (1) известные (см., например, [2]) необходимые условия точки локального минимума (стационарной точки), которые принято называть принципом Лагранжа:

Теорема 1. *Если точка $x^* \in D$ является точкой локального минимума задачи (1), тогда существуют числа $y_i^* \leqslant 0$ $i = 0 \dots m$ такие, что имеют место условия:*

$$y_0^* f'(x^*) + \sum_{i=1}^{*} y_i^* f_i'(x^*) = 0 \tag{2}$$

$$y_i^* f_i(x^*) = 0 \quad i = 1 \dots m. \tag{3}$$

Очевидно, что условия (2) и (3) эквивалентны одному условию:

$$y_0^* f'(x^*) + \sum_{i \in I(x^*)} y_i^* f_i'(x^*) = 0, \tag{4}$$

где $I(x^*) = \{i : i = 1 \dots m, f_i(x^*) = 0\}$.

Поставим вспомогательную задачу:

$$\max\{f(x), x \in D_1\}, \tag{5}$$

где $D_1 = \{x : x \in R_n, f_i(x) \geqslant 0, i = 1 \dots I\}$, I — некоторое множество индексов.

Задача (5) является задачей выпуклого программирования. Поэтому каждое ее локальное решение является точкой глобального максимума функции $f(x)$. Необходимые и достаточные условия оптимальности точки $x^* \in D_1$ для задачи (5) имеют вид:

$$y_0^* f'(x^*) + \sum_{i \in I} y_i^* f_i'(x^*) = 0 \tag{6}$$

$$y_i^* f_i(x^*) = 0 \quad i \in I. \tag{7}$$

Видно, что при выполнении $I = I(x^*)$ необходимое условие стационарной точки (4) для задачи (1) эквивалентно необходимым и достаточным условиям оптимальности (6), (7) для вспомогательной задачи (5).

Аналогично [1], было доказано, что любая точка локального минимума задачи (1) может быть получена как решение задачи вида (5) при некотором множестве $I \subset \{1, \dots m\}$. Лучшая точка локального минимума (с минимальным значением целевой функции $f(\cdot)$) будет являться глобальным решением задачи (1).

Вспомогательную задачу (5) легче решить с помощью ее двойственной задачи:

$$\min\{\varphi(y), y \in Y\}, \tag{8}$$

где $\varphi(y) = \sup\limits_{x \in R_n} L(x, y)$, $L(x, y) = y_0 f(x) + \sum\limits_{i \in I} y_i f_i(x)$, $Y = \{y \in R_n : y_i \geqslant 0, i \in I, y_i = 0, i \notin I\}$. С помощью двойственной задачи (8) можно сделать вывод как о совместности допустимого множества задачи (5), так и об оптимальном решении этой задачи. Простой вид множества Y позволяет использовать простые процедуры для решения задачи (8)(например, некоторые из них можно найти в [1]).

Метод ветвей и границ для задачи вогнутого программирования

Согласно эквивалентности условий оптимальности для задач (1) и (5), задачу (1) можно решить посредством перебора всех возможных подмножеств I и определения тех из них, которые определяют стационарные точки задачи (1). Такой перебор предлагается осуществить посредством метода ветвей и границ, который определяется следующими положениями.

Каждый узел дерева решений характеризуется некоторым подмножеством индексов $I \subset \{1 \dots m\}$, которое определяет возможное множество активных индексов задачи (1). Ветвление от каждого узла характеризуется добавлением нового индекса к этому множеству. В каждом узле дерева по зафиксированному множеству индексов I формулируется и решается задача (5) (или двойственная ей задача (8)). В зависимости от результатов решения этой вспомогательной задачи происходит отклонение данного узла и исходящего из него поддерева (например, множество допустимых решений задачи (5) несовместно или оптимальное значение задачи (5) хуже уже полученного рекордного значения) или последующее ветвление.

Описание численного эксперимента

Численный эксперимент проводился на наборе сгенерированных задач частного вида при вогнутой квадратичной целевой функции и линейных функциях-ограничениях. Принципы генерации задач описаны в [1]. Поскольку задача (5) в этом случае является задачей выпуклого квадратичного программирования, она решается с помощью конечных алгоритмов.

Эксперимент показал принципиальную пригодность предложенного метода ветвей и границ для получения решения задачи (1). Преимуществом данного метода является гарантированное получение глобального решения задачи. Однако для более сложных видов целевой функции и функций-ограничений

способы решения задачи (5) могут быть заметно более вычислительно трудоемкими, что потребует, например, разработки приближенных методов решения или применения параллельных алгоритмов.

Литература

[1] *Konnov I. V.* Sign reversion approach to concave minimization problems // Optim Lett. — 2010. — No. 4. — P. 491—500.

[2] *Сухарев А. Г., Тимохов А. В., Федоров В. В.,* Курс методов оптимизации. — М.:Наука, 1986. — 328 с.

Модель задачи негильотинного размещения набора прямоугольников на полуполосе

А. А. Андрианова, Т. М. Мухтарова, В. Р. Фазылов

aandr78@mail.ru, tmm116@yandex.ru, vfazylov@gmail.com

Казанский (Приволжский) федеральный университет, Казань

Постановка задачи

Пусть даны набор n прямоугольников (деталей) с размерами $a_i \times b_i$ ($a_i \geqslant b_i$), $i = 1, \ldots, n$, и полуполоса с шириной B, $B \geqslant \max\limits_{1 \leqslant i \leqslant n} b_i$. Требуется определить длину куска полуполосы, необходимую и достаточную для размещения данного набора деталей, и соответствующий план размещения, т. е. координаты размещения на куске полуполосы каждого из прямоугольников и его ориентация (вдоль или поперек полуполосы, ориентация прямогольника "наискосок" не допускается).

Модель негильотинного размещения набора прямоугольников на полуполосе

Определим переменные модели. Пусть A — искомая длина полуполосы. Началом координат будем считать левый нижний угол полуполосы. Ось X направлена вдоль полуполосы. Обозначим через (x_i, y_i) координаты левого нижнего угла i-й детали, а через z_i — ориентацию i-й детали на листе:

$$z_i = \begin{cases} 0, & \text{если деталь ориентирована вдоль полуполосы,} \\ 1, & \text{если деталь ориентирована поперек полуполосы.} \end{cases}$$

Для формулировки условий непересечения деталей требуется введение новых переменных, задающих расположение двух деталей друг относительно друга. Условия непересечения двух деталей задается через систему из четырех альтернативных ограничений (i-я деталь лежит левее j-ой или i-я деталь лежит правее j-ой, i-я деталь лежит выше j-ой или i-я деталь лежит ниже j-ой). Для сведения системы альтернативных неравенств к совместной системе линейных неравенств использованы два набора булевых переменных $s_{ij}, t_{ij}, i = 1 \ldots n-1, j = i+1 \ldots n$.

Модель компактного негильотинного размещения набора прямоугольников на полуполосе имеет следующий вид:

$$\min A \tag{1}$$

$$x_i \geqslant 0, \quad x_i + (b_i - a_i)z_i - A \leqslant -a_i, \quad i = 1, \ldots, n, \tag{2}$$

$$y_i \geqslant 0, \quad y_i + (a_i - b_i)z_i \leqslant B - b_i, \quad i = 1, \ldots, n, \tag{3}$$

$$-x_i + x_j - (b_i - a_i)z_i + \bar{A}t_{ij} + \bar{A}s_{ij} \geqslant a_i, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n, \tag{4}$$

$$x_i - x_j - (b_j - a_j)z_j + \bar{A}t_{ij} - \bar{A}s_{ij} \geqslant a_j - \bar{A}, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n, \tag{5}$$

$$-y_i + y_j - (a_i - b_i)z_i - \bar{B}t_{ij} + \bar{B}s_{ij} \geqslant b_i - \bar{B}, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n, \tag{6}$$

$$y_i - y_j - (a_j - b_j)z_j - \bar{B}t_{ij} - \bar{B}s_{ij} \geqslant b_j - 2\bar{B}. \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n, \tag{7}$$

$$z_i \in \{0, 1\}, \quad i = 1, \ldots, n, \tag{8}$$

$$s_{ij} \in \{0, 1\}, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n, \tag{9}$$

$$t_{ij} \in \{0, 1\}, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n. \tag{10}$$

где $\bar{A}$ — верхняя оценка оптимального значения A (например, $\bar{A} = \sum_{i=1}^{n} a_i$), $\bar{B} \geqslant B$.

В качестве целевой функции выбрана неизвестная длина полуполосы. Ограничения (2), (3) задают условия размещения отдельных деталей на листе размером $A \times B$. Ограничения (4)–(7) являются условиями попарного непересечения деталей. Подробное обоснование модели можно найти в [1].

Таким образом, задача компактного негильотинного размещения набора прямоугольников на полуполосе сформулирована в виде задачи линейного частично булевого программирования (1)–(10).

Метод решения задачи компактного негильотинного размещения набора прямоугольников на полуполосе

Для решения задачи (1)–(10) был применен метод Лэнд и Дойг ([2]). Данный метод относится к классу методов ветвей и границ и является универсальным методом решения задач частично целочисленного линейного программирования с двусторонними ограничениями на целочисленные переменные. Идея метода заключается в решении набора задач линейного программирования, в которых ограничения (8)–(10) заменены на двусторонние неравенства вида:

$$0 \leqslant z_i \leqslant 1, \quad i = 1, \ldots, n,$$

$$0 \leqslant s_{ij} \leqslant 1, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n,$$

$$0 \leqslant t_{ij} \leqslant 1, \quad i = 1, \ldots, n-1, \quad j = i+1, \ldots, n.$$

и вводятся по ходу метода дополнительные ограничения на целочисленные переменные, которые в нашем случае, фактически означают фиксацию значений булевых переменных.

Задача компактного негильотинного размещения набора прямоугольников на полуполосе имеет очевидную нижнюю границу целевой функции, равную $\left\lceil \tilde{B}^{-1} \sum\limits_{i=1}^{n} \frac{a_i b_i}{D} \right[\cdot D$, где D — наибольший общий делитель всех размеров деталей a_i, b_i, $i = 1 \ldots n$, $\tilde{B} = [B/D] \cdot D$ ($[\cdot]$, $]\cdot[$ — операции округления до ближайшего меньшего и большего целого соответственно). Поэтому в случае ее совпадения с текущим рекордным значением целевой функции можно прекратить вычисления, так как текущий рекорд будет оптимальным решением задачи.

Для выбора переменной ветвления мы использовали правило первой нецелой переменной, причем в ходе численных экспериментов было установлено, что для уменьшения трудоемкости вычислений целесообразно сделать следующее:

1) предварительно отсортировать детали по невозрастанию площадей, а детали с равными площадями — по невозрастанию длин деталей;

2) по отсортированному списку деталей сформировать следующий список булевых переменных:

$$z_1, z_2, s_{12}, t_{12}, z_3, s_{13}, t_{13}, s_{23}, t_{23}, z_4, \ldots, s_{n-1,n}, t_{n-1,n},$$

и в качестве переменной ветвления выбирать первую по этому списку нецелую переменную. Такой порядок переменных и правило выбора первой нецелой переменной отражает стремление искать оптимальное размещение путем добавления очередной детали к уже полученному частичному размещению.

Результаты численного эксперимента

Численный эксперимент проводился на наборе, состоящем из 300 сгенерированных задач. В наборы входило от 10 до 15 прямоугольников размеров из интервала $[1, 20]$. В качестве показателя трудоемкости решения задачи использовалось количество рассмотренных задач линейного программирования. Подробно данный эксперимент описан в [1].

Было замечено, что трудоемкость вычислений сильно возрастает для задач, в которых нижняя граница недостижима. Часто при решении этих задач оптимальное размещение бывает получено довольно рано. Однако на дальнейший последовательный просмотр и отсечение оставшихся ветвей дерева решений приходится большая часть трудоемкости.

По результатам эксперимента было замечено, что для получения оптимального решения в среднем достаточно было бы просмотреть около 9% дерева. В связи с этим был применен следующий эвристический критерий остановки: последний нефиктивный рекорд считается эвристическим решением задачи, если текущая трудоемкость превысила 100 000 узлов и выполняется одно из условий: 1) оценка просмотренной доли дерева превышает 10 %; 2) трудоемкость процесса вычислений с момента получения последнего рекорда превысила 50 % от текущей трудоемкости.

На основе эксперимента получены следующие результаты. Досрочная остановка вычислений по эвристическому критерию произошла в 63% случаев.

Среди них в 67% случаев наблюдалось уменьшение общей трудоемкости более 50%, а в 30% случаев — более 90%. В основном (92% от общего количества задач) было получено оптимальное размещение деталей на полуполосе. Средняя погрешность полученных размещений в оставшихся 8% задач составила менее 4%.

Литература

[1] *Андрианова А. А., Мухтарова Т. М., Фазылов В. Р.* Модели задачи негильотинного размещения набора прямоугольников на листе и полуполосе // Уч. зап. Казан. ун-та. Сер. физ.-матем.науки. — 2013. — Т. 155, кн. 3. — С. 5—18.

[2] *Land A. H., Doig A. G.* An automatic method of solving discrete programming problems // Econometrica. — 1960. — V. 28, No. 3. — P. 497—520.

Метод функций Ляпунова и проблема оптимизации процесса управления в системах с переменной структурой

О. Г. Антоновская, В. И. Горюнов

olga.antonovsckaja@yandex.ru, pmk@unn.ac.ru

НИИ прикладной математики и кибернетики Нижегородского государственного университета им. Н.И. Лобачевского, Нижний Новгород

Известно, что проблема выбора оптимальных значений параметров в таких нелинейных динамических системах как синтезаторы частоты [1] при работе в заданном диапазоне частот, является центральной [2] и, в том числе, при синтезе комбинированного управления [3].

Настоящий доклад связан с практически важной прикладной задачей реализации надежной радиосвязи, в основе которой лежит использование управляемых синтезаторов частот (СЧ) [1], построенных на базе импульсных систем фазовой синхронизации. В таких системах используется широтно-импульсная модуляция управляющего сигнала, и поэтому их математические модели (ММ) являются частным случаем систем с переменной структурой. Порядок смены дифференциальных уравнений в системах определяется динамическими свойствами фазовых траекторий и реализуется в моменты переключения управляющих импульсов. Поэтому изучение динамики ММ таких СЧ осуществляется на основе применения метода точечных отображений.

В докладе [4] отмечались основные моменты решения проблемы синтеза оптимального управления в системах управления указанного класса с помощью корневого критерия. В отличие от обычного подхода к использованию корневого критерия в случае работы синтезатора в заданном (широком)диапазоне параметры характеристического уравнения, определяющие не только устойчивость, но и скорость переходных процессов, являются интервальными. А поэтому для исследования устойчивости можно было воспользоваться рекомендациями и приемами исследования робастных систем [5].

Анализ точечных отображений различных сечений подпространств, соответствующих постоянству величины управляющего сигнала в СЧ показал [6],

что в зависимости от параметров системы основному рабочему режиму соответствует неподвижная точка одного из двух точечных отображений, для которых можно получить аналитические выражения функций последования. Но тогда левая часть исходного характеристического уравнения принимает вид полинома, в котором коэффициенты, вследствие зависимости от интервальных параметров, также интервальны [4].

В связи с тем, что увеличение диапазона интервальности приводит к расширению области фазового пространства, в которой реализуются переходные движения [4], принципиально существенной становится необходимость учета нелинейности исследуемой динамической системы. В настоящем докладе предлагается использование для этой цели концепции применения функций Ляпунова к интервально-неопределенным системам [7]. Учет интервальности параметров с оценкой сверху размеров соответствующей области фазового пространства позволяет трансформировать постановку вопроса в задачу построения функции Ляпунова, удовлетворяющей заданному ограничению [7].

Известно, что положительно определенная квадратичная форма будет функцией Ляпунова дискретной динамической системы в том и только том случае, когда

$$\max_{i=1,\dots,n}\{|z_i|^2-1\}\leqslant\max_{V=V_0}\frac{\Delta V}{V}<0,$$

где z_i – корни характеристического уравнения, определяющего устойчивость системы, $|z_i|<1, i=1,2,\dots,n$. При этом всегда существует квадратичная функция Ляпунова, для которой обеспечивается заданный запас знакоотрицательности первой разности в силу линеаризованных уравнений системы [8], отвечающий значениям

$$\max_{V=V_0}\frac{\Delta V}{V}=\delta,\ \max_{i=1,\dots,n}\{|z_i|^2-1\}\leqslant\delta<0,$$

в том числе и максимальный ее запас [9]. Удобнее всего строить такую функцию Ляпунова путем перехода к каноническим координатам [10], поскольку в этом случае можно предложить аналитические соотношения для определения параметров квадратичной формы в случае произвольного n, подобные приведенным в [8] для случая $n=2$. Например, для точечного отображения

$$\bar{x}_i=z_ix_i\,(i=1,2,\dots,n),$$

где $|z_n|<|z_{n-1}|<\dots<|z_2|<|z_1|<1$, причем все z_i действительны, квадратичную функцию Ляпунова можно искать в виде

$$V(x_1,\dots,x_{n-2},x_{n-1},x_n)=K_{11}x_1^2+\dots+K_{n-2,n-2}x_{n-2}^2+$$

$$+K_{n-1,n-1}x_{n-1}^2+2K_{n-1,n}x_{n-1}x_n+K_{n,n}x_n^2,$$

где величины $K_{11}>0,\dots,K_{n-2,n-2}>0$, а $K_{n-1,n-1}>0, K_{n,n}>0, K_{n-1,n}$ связаны соотношением

$$K_{n-1,n}^2=(1-R(\delta))K_{n-1,n-1}K_{n,n},$$

в котором

$$R(\delta) = (1+\delta)(z_n - z_{n-1})^2(z_{n-1}z_n - 1 + \delta)^{-2}$$
$$(z_n^2 - 1 \leqslant \delta < 0).$$

Использование квадратичной функции Ляпунова подобного вида позволило получить такие интервальные оценки параметров системы, при которых построенная функция является функцией Ляпунова системы с интервально изменяющимися параметрами [4]. Оценки содержат абсолютные величины корней характеристического уравнения при одном конкретном значении параметров системы из области устойчивости, величину δ, и зависят от элементов матрицы перехода к каноническим координатам.

Литература

[1] *Левин В. А., Малиновскай В. И., Романов С. К.* Синтезаторы частот с импульсно-фазовой автоподстройкой. — М.: Радио и связь, 1989. — 314 с.

[2] *Goryunov V. I.* On the feedback parameter optimization for the robust stable frequency synthesizer // Progress of nonlinear science: Proceedings of International Conference, dedicated to the 100-th Annivesary of A.A. Andronov. — Nizhny Novgorod, 2002. — V. 3. — P. 159–163.

[3] *Antonovskaya O. G., Goryunov V. I., Palochkin Yu. P.* On the synthesis of combined control by nonlinear phenomena analysis for frequency synthesizer working in wide band // 2-nd International Conference on Circutes and Systems of Communications. Proceedings. — Moscow, 2004. — P. 54–62.

[4] *Антоновская О. Г., Горюнов В. И.* К проблеме оптимизации процесса управления в системвх с переменной структурой // Проблемы теоретической кибернетики. Материалы XVI Междунвродной конференции. — Нижний Новгород: Издательство Нижегородского госуниверситета, 2011. — С. 38–41.

[5] *Джури Э. И.* Робастность дискретных систем // Автоматика и телемеханика. — 1990. — № 5. — С. 3–28.

[6] *Антоновская О. Г., Горюнов В. И.* О влиянии диссипации энергии на динамику астатической системы с широтно-импульсной модуляцией управляющего сигнала // Вестник ННГУ — Н. Новгород: Изд-во ННГУ, 2009. — С. 141–145.

[7] *Антоновская О. Г., Горюнов В. И.* К оптимизации алгоритма использования прямого метола Ляпунова при исследовании процессов в интервальнонеопределенных системах // VII Международнвй семинар "Устойчивость и колебания нелинейных систем управления". Тезисы докладов — М.: ИПУ РАН, 2004. — С. 12–13.

[8] *Антоновская О. Г.* О построении квадратичной функции Ляпунова с заданными свойствами // Дифференциальные уравнения. — 2013. — Т. 49, № 9. — С. 1220–1224.

[9] *Антоновская О. Г.* О максимальном ограничении первой производной (первой разности) квадратичной функции Ляпунова // Дифференциальные уравнения. — 2003. — Т. 39,№ 11. — С. 1562–1563.

[10] *Неймарк Ю. И.* Метод точечных преобразований в теории нелинейных колебаний // Изв. вузов: Радиофизике. — 1958. — Т. 1, № 1. — С. 41–66.

Автоматная сложность булевых функций

Д. Н. Бабин, М. А. Кибкало

d.n.babin@mail.ru, mkibkalo@gmail.com

Кафедра МаТИС, мех-мат МГУ, Москва

Рассматривается сложность представления булевых функций конечными автоматами [1] и устанавливаются асимптотические оценки функции Шеннона для всех замкнутых классов булевых функций. Булевой функции $f \in P_2^n$ сопоставим конечный язык $L(f)$ по правилу: слово $\tilde{\alpha} = \alpha_1 \dots \alpha_n \in L(f) \Leftrightarrow f(\tilde{\alpha}) = f(\alpha_1, \dots, \alpha_n) = 1$. Пусть $V_q = (E, Q, E, \varphi, \psi, q)$ инициальный конечный автомат (ИКА) [2] представляющий язык $L(f)$. Сложностью автомата назовем число его состояний. Автоматной сложностью $S(f, n)$ булевой функции $f \in P_2^n$ назовем наименьшую сложность ИКА, представляющего язык $L(f)$. Пусть $\mathcal{K} \subseteq P_2$ - класс булевых функций, $\mathcal{K}(n) = \mathcal{K} \cap P_2^n$. Функцией Шеннона класса $\mathcal{K}$ назовем $S(\mathcal{K}, n) = max_{f \in \mathcal{K}(n)} S(f, n)$. Будем пользоваться нотацией классов Поста, введенной в [3].

Теорема 1. *Имеют место следующие оценки:*

1. Пусть $\mathcal{K}$ - один из классов $C_i, i = 1, 2, 3, 4$, D_1, D_3, $F_i^\infty(n)$, $F_i^\mu(n), i = 1, 4, 5, 8, \mu > 1, \mu \in \mathbb{N}$. Тогда:

$$S(\mathcal{K}, n) \asymp \frac{2^n}{n}$$

2. Пусть $\mathcal{K}$ - один из классов $A_i, i = 1, 2, 3, 4$, D_2, $F_i^\infty(n)$, $F_i^\mu(n), i = 2, 3, 6, 7, \mu > 1, \mu \in \mathbb{N}$. Тогда:

$$S(\mathcal{K}, n) \asymp \frac{2^n}{n \cdot \sqrt{\log n}}$$

3. Пусть $\mathcal{K}$ - один из классов $L_i, i = 1, 2, 3, 4, 5$, $S_i, i = 1, 3, 5, 6$, $P_i, i = 1, 3, 5, 6$, $O_i, i = 1, 2, 4, 5, 6, 7, 8, 9$. Тогда:

$$S(\mathcal{K}, n) \asymp n$$

4. $S(O_3, n) = 1$

Литература

[1] *Кузьмин В. А.* Реализация функций алгебры логики автоматами, нормальными алгорифмами и машинами Тьюринга. // Проблемы кибернетики. — 1955. — Т. 13.— С. 75–96.

[2] *Кудрявцев В. Б., Алешин С. В., Подколзин А. С.* // Введение в теорию автоматов. — М.: Наука, 1985.

[3] *Яблонский С. В., Гаврилов Г. П., Кудрявцев В. Б.* Функции алгебры логики и классы Поста. — М.: Наука, 1966.

[4] *Кибкало М. А.* О сложности представления коллекции языков в конечных автоматах // Интеллектуальные системы. — 2010. — Т. 13.— С. 347–360.

О некоторых минимальных ультраклонах

С. А. Бадмаев, И. К. Шаранжаев

badmaevsa@mail.ru, goran5@mail.ru

Бурятский государственный университет, Улан-Удэ

Пусть $E_k = \{0, \ldots, k-1\}$, $F_k = 2^{E_k} \setminus \{\varnothing\}$. Отображение $f : E_k^n \to F_k$ называется n-местной ультрафункцией на E_k.

Отображение $e_n^i : (x_1, \ldots, x_n) \to \{x_i\}$ называется n-местной проекцией.

Константной ультрафункцией будем называть ультрафункцию, принимающую на всех наборах одинаковое значение.

Так как множество ультрафункций не является замкнутым относительно обычной суперпозиции, в [1] вводится «специальная» суперпозиция, относительно которой замкнутость выполняется.

Для того, чтобы суперпозиция $f(f_1(x_1, \ldots, x_m), \ldots, f_n(x_1, \ldots, x_m))$ определяла некоторую ультрафункцию $g(x_1, \ldots, x_m)$, зададим значения ультрафункции на наборах из подмножеств множества E_k.

Если $(\alpha_1, \ldots, \alpha_m) \in E_k^m$, то по определению

$$g(\alpha_1, \ldots, \alpha_m) = \begin{cases} \bigcap\limits_{\beta_i \in f_i(\alpha_1, \ldots, \alpha_m)} f(\beta_1, \ldots, \beta_n), & \text{если пересечение не пусто;} \\ \bigcup\limits_{\beta_i \in f_i(\alpha_1, \ldots, \alpha_m)} f(\beta_1, \ldots, \beta_n), & \text{в противном случае.} \end{cases}$$

Это определение позволяет вычислить значение ультрафункции $f(x_1, \ldots, x_n)$ на любом наборе $(\sigma_1, \ldots, \sigma_n) \in F_k^n$.

Клоном на E_k называется замкнутое множество функций k-значной логики, содержащее все проекции.

Ультраклоном на E_k называется множество ультрафункций на E_k, замкнутое относительно введенной суперпозиции и содержащее все проекции.

Авторами доказано следующее

Утверждение. 1) Любой минимальный клон на E_k является минимальным ультраклоном на E_k;

2) Любая константная ультрафункция на E_k порождает минимальный ультраклон на E_k.

3) Других минимальных ультраклонов на E_2 нет.

Следствие. Существует ровно 8 минимальных ультраклонов на E_2 и не менее 88 минимальных ультраклонов на E_3.

Данное следствие получается с учетом описания всех минимальных клонов на E_2 и E_3 [2, 3].

Литература

[1] *Пантелеев В. И.* О некоторых максимальных частичных гипер- и ультраклонах // Материалы XVIII Международной школы-семинара «Синтез и сложность управляющих систем», – М.: Изд-во мех.-мат. МГУ, 2009. – Стр. 73-75.

[2] *Post E. L.* Two-valued iterative systems of mathematical logic // Annals of Math. Studies. — Princeton: Univer. Press, 1941. — Vol. 5. — 122 p.

[3] *Csákány B.* All minimal clones on the three-element set // Acta cybernetica. — 1983. — №6. — P. 227–238.

Статистики спусков и средних на множествах слов

Л. Н. Бондаренко, М. Л. Шарапова

leobond5@mail.ru, msharapova@list.ru

Пензенский государственный университет, Московский государственный университет им. М. В. Ломоносова, механико–математический факультет

Статистикой на конечном множестве Ω называется неотрицательная целочисленная функция stat, определенная для любого слова $\omega{\in}\Omega$. Статистика индуцирует производящий многочлен

$$P(t) = \sum_{\omega\in\Omega} t^{\mathrm{stat}(\omega)} = \sum_k P_k\, t^k$$

с коэффициентами $P_k{=}\#\{\omega{\in}\Omega : \mathrm{stat}(\omega){=}k\}$, а при равномерном распределении слов на Ω и распределение коэффициентов нормированного многочлена $\overline{P}(t){=}P(t)/P(1)$. Этому распределению отвечает случайная величина X, имеющая математическое ожидание $E(X){=}(\overline{P}(t))'$ и дисперсию $D(X){=}(\overline{P}(t))''{+}(\overline{P}(t))'(1{-}(\overline{P}(t))')$.

Так, статистика $\mathrm{des}(\sigma){=}\#\{i{\in}N_n : \sigma_i{>}\sigma_{i+1}, \sigma_{n+1}{=}0\}$ на множестве всех перестановок S_n над алфавитом $N_n{=}\{1,\ldots,n\}, \#S_n{=}n!$ описывает число спусков перестановки $\sigma{\in}S_n$ и индуцирует производящий многочлен Эйлера $A_n(t){=}\sum_{k=1}^n A_{n,k}\, t^k$ с коэффициентами $A_{n,k}{=}\#\{\sigma{\in}S_n : \mathrm{des}(\sigma){=}k\}$ [1], задаваемый рекуррентным соотношением

$$A_0(t) = 1,\ A_n(t) = ntA_{n-1}(t) + t(1-t)A'_{n-1}(t)\,. \tag{1}$$

Статистика des естественно продолжается с S_n на множество W_n всех слов $\omega{=}\omega_1\ldots\omega_n$ над алфавитом N_n, т. е. $\mathrm{des}(\omega){=}\#\{i{\in}N_n : \omega_i{>}\omega_{i+1}, \omega_{n+1}{=}0\}$, причем $\#W_n{=}n^n$, и ее можно подсчитывать как своеобразное среднее

$$\mathrm{des}(\omega) = \frac{1}{n+1}\sum_{i=0}^{n}(\omega_{i+1}-\omega_i),\ \ \omega_0 = \omega_{n+1} = 0\,,$$

в котором разности под знаком суммы вычисляются по $\mathrm{mod}\,(n{+}1)$ на числовом множестве $\{0,1,\ldots,n\}$.

Вводя биекции $\mathbf{c} : W_n \to W_n$ и $\mathbf{r} : W_n \to W_n$ выражениями $\mathbf{c}\,\omega_i{=}n{+}1-\omega_i$ и $\mathbf{r}\,\omega_i{=}\omega_{n+1-i}$, $i{\in}N_n$, $\omega{\in}W_n$ получаем для дополнительного $\mathbf{c}\,\omega$ и обратного $\mathbf{r}\,\omega$ слов равенства $\mathrm{des}(\mathbf{c}\,\omega){=}\mathrm{des}(\mathbf{r}\,\omega)$ и $\mathrm{des}(\omega){+}\mathrm{des}(\mathbf{c}\,\omega){=}n{+}1$.

Определим среднее значение символа $\mathrm{mes}(\omega)$ слова $\omega{\in}W_n$ формулой

$$\mathrm{mes}(\omega) = \left[\frac{1}{n}\sum_{i=1}^{n}\omega_i\right],$$

где $[\,\cdot\,]$ — целая часть числа.

Тогда несложно получить $\mathrm{mes}(\omega)=\mathrm{mes}(\mathbf{r}\,\omega)$ и $\mathrm{mes}(\sigma)=[(n+1)/2]$ при $\sigma\in S_n$. Поэтому соответствующую статистику на S_n определим с помощью биекции $\mathbf{v}:W_n\to W_n$, задаваемой соотношениями $\mathbf{v}\,\omega_i=\omega_i+n+1-i \pmod n$, $i\in N_n$, где $\mathbf{v}\,\omega_i$ — наименьший положительный вычет. Распределения статистик $\mathrm{mes}(\mathbf{v}\,\omega)$ и $\mathrm{mes}(\omega)$ на W_n совпадают, а $\mathrm{mes}(\mathbf{v}\,\sigma)=\mathrm{ivp}(\sigma)$ при $\sigma\in S_n$, причем для статистики ivp на S_n [2] имеем $\mathrm{ivp}(\mathbf{c}\,\sigma)=\mathrm{exc}(\sigma)$, $\mathrm{ivp}(\sigma)+\mathrm{exc}(\sigma)=n+1$, а $\mathrm{exc}(\sigma)=\#\{i\in N_n:\mathbf{v}\,\sigma_i>n\}=\#\{i\in N_n:\sigma_i\geqslant i\}$.

Естественно возникает задача нахождения производящих многочленов для статистик ivp и exc на S_n, а также des и mes на W_n. Для des и exc на S_n коэффициенты производящих многочленов вычисляются по одному и тому же рекуррентному соотношению

$$A_{0,k}=\delta_{0k},\ A_{n,k}=kA_{n-1,k}+(n-k+1)A_{n-1,k-1}\,,$$

в котором δ_{ij} — символ Кронекера, и легко проверяемому по определению статистик методом математической индукции.

Так как $A_{n,k}=A_{n,n-k+1}$, то производящие многочлены для статистик exc и ivp на S_n также совпадают с многочленами Эйлера (1), причем определение этих статистик позволяет представить многочлены $A_n(t)$ двумя разными способами через перманенты функциональных матриц [2].

Свойства коэффициентов $A_{n,k}$ описывают следующие взаимно обратные соотношения эйлеровского типа, отсутствующие в [3].

Теорема 1. *При $n\geqslant 1$ и $j,m=1,\ldots,n$ справедливы соотношения*

$$F_{n,m}=\sum_{k=1}^{m}\binom{m+n-k}{m}G_{n,k},\quad G_{n,j}=\sum_{i=1}^{j}(-1)^{j-i}\binom{n+1}{j-i}F_{n,i}\,. \tag{2}$$

Доказательство. Подстановка первого выражения (2) во второе приводит к соотношению ортогональности $\sum_{k=0}^{r}(-1)^k\binom{n+1}{k}\binom{r+n-k}{n}=\delta_{r0}$, проверяемому с помощью разностного оператора Δ, так как при $r=1,\ldots,n$ левая часть этого соотношения равна $\Delta^{n+1}\binom{r-1}{n}=0$. ∎

В частности, если в (2) положить $G_{n,k}=A_{n,k}$, то $F_{n,m}=m^n$ [4], причем можно записать $A_{n,n-k+1}=\Delta^{n+1}k^n_+$, где u^n_+ — усеченная степенная функция с периодом $p=n+1$, т. е. $(u+p)^n_+=u^n$ при $u\geqslant 0$ и $u^n_+=0$ при $u<0$.

Теорема 2. *Для статистик* des *и* mes *на W_n производящие многочлены $\widetilde{A}_n(t)=\sum_{k=1}^{n}\widetilde{A}_{n,k}\,t^k$ одинаковы, а коэффициенты $\widetilde{A}_{n,k}$ находятся по формуле*

$$\widetilde{A}_{n,k}=\sum_{i=1}^{k}(-1)^{k-i}\binom{n+1}{k-i}\binom{(i+1)n-1}{n}, \tag{3}$$

т. е. в выражении (2) *числа $G_{n,k}=\widetilde{A}_{n,k}$, если $F_{n,m}=\binom{(m+1)n-1}{n}$, причем можно записать $\widetilde{A}_{n,n-k+1}=\Delta^{n+1}\binom{(k+1)n-1}{n}_+$, где усеченный биномиальный коэффициент $\binom{(u+p+1)n-1}{n}_+=\binom{(u+1)n-1}{n}$ при $u\geqslant n^{-1}$ и $\binom{(u+1)n-1}{n}_+=0$ при $u<n^{-1}$, а соответствующий период $p=n+1$.*

Доказательство. Результаты для статистики mes получаются с помощью производящей функции $(t+\ldots+t^n)^n$, описывающей число слов $\omega\in W_n$ с одинаковой суммой $\sum_{i=1}^n \omega_i$, и теоремы 1. Для статистики des также используется теорема 1, а производящий многочлен находится с помощью цепочки рекуррентных соотношений, позволяющих получить требуемый результат. ∎

Многочлены $A_n(t)$ и $\widetilde{A}_n(t)$ являются f–эйлеровыми многочленами, так как справедливо равенство [1]

$$\sum_{k=1}^{\infty} F_{n,k}\, z^k = \frac{G_n(z)}{(1-z)^{n+1}}\,,$$

в котором $G_n(z)=A_n(z)$, если $F_{n,k}=k^n$, и $G_n(z)=\widetilde{A}_n(z)$, если $F_{n,k}=\binom{(k+1)n-1}{n}$. Другим методом описания рассматриваемых производящих многочленов может служить z–преобразование [4].

Теорема 3. *Пусть случайные величины X_n и Y_n соответствуют распределениям коэффициентов нормированных производящих многочленов для статистик* des *(*ivp*) на S_n и* des *(*mes*) на W_n. Тогда эти распределения асимптотически нормальны с параметрами*

$$E(X_n)=\frac{n+1}{2}\sim\frac{n}{2};\;\; D(X_n)=\frac{n+1}{12}\sim\frac{n}{12},\; n\geqslant 2\,;$$

$$E(Y_n)=\frac{n^2+1}{2\,n}\sim\frac{n}{2};\;\; D(Y_n)=\frac{(n+1)(n^2-1)}{12\,n^2}\sim\frac{n}{12}\,.$$

Доказательство. Математические ожидания и дисперсии случайных величин X_n и Y_n находятся с помощью формул (1) и (3). Напомним, что нормированная случайная величина $\overline{X}_n$ называется асимптотически нормальной, если выполняется равенство $\lim\limits_{n\to\infty}\Pr(\overline{X}_n\leqslant x)=(2\pi)^{-1/2}\int_{-\infty}^{x} e^{-t^2/2}dt$. Можно показать, что все нули рассматриваемых нормированных производящих многочленов неположительны и различны, что и влечет асимптотическую нормальность соответствующих распределений. ∎

Отметим, что статистика $\mathrm{exc}(\omega)=\#\{i\in N_n : \omega_i\geqslant i\}$ на W_n индуцирует производящий многочлен

$$\widehat{A}_n(t)=n!\prod_{k=0}^{n-1}\left(t+\frac{k}{n-k}\right)\,,$$

а распределение коэффициентов нормированного многочлена асимптотически нормально с математическим ожиданием $(n+1)/2$ и дисперсией $(n^2-1)/(6n)$.

Литература

[1] *Стенли Р.* Перечислительная комбинаторика.— Т. 1.— М.: Мир, 1990. — 440 с.

[2] *Бондаренко Л. Н., Шарапова М. Л.* Статистики на группе перестановок и перманенты // Материалы XI Международного семинара «Дискретная математика и ее приложения», посвященного 80–летию со дня рождения академика О. Б. Лупанова (Москва, МГУ, 18-23 июня 2012 г.) — М: Изд–во механико–математического факультета МГУ, 2012. — С. 234–237.

[3] *Риордан Дж.* Комбинаторные тождества. — М.: Наука, 1982. — 256 с.

[4] *Бондаренко Л. Н., Шарапова М. Л.* Применение обобщенной формулы Родрига в комбинаторном анализе // Известия высших учебных заведений. Поволжский регион. Физико–математические науки. — 2011. — № 4 (20). — С. 44–58.

Об эквивалентности отображений многообразий

О. П. Бондарь

marina_semenyuta@mail.ru

Кировоградская летная академия НАУ, г. Кировоград

Систему, параметры которой описываются дискретными, непрерывными или дифференцируемыми функциями, можно представить, соответственно, дискретным (не связным), топологическим или гладким многообразием, которое называется многообразием состояний системы. Сингулярности функций на многообразии можно интерпретировать, как подмножество параметров, при которых состояние системы резко меняется. Количество и вид сингулярностей определяют структуру системы, позволяя прогнозировать ее поведение при соответствующем изменении ее параметров.

Если функции на многообразиях рассматривать, как частный случай отображений многообразий (см., напр., [1], [2]), то задача описания возможных состояний системы сводится к задаче нахождения отображений, эквивалентных заданным, т.е. прогнозирующим поведение системы. Здесь указан один из способов нахождения эквивалентных непрерывных (дифференцируемых) отображений многообразий.

Теорема 1. *Два непрерывных (дифференцируемых) отображения с изолированными сингулярными точками топологически (дифференцируемо) эквивалентны, если прообразы их обобщенных графов Кронрода-Риба, допускающих склеивающую перестройку, изоморфны.*

Доказательство. Под графом Кронрода-Риба гладкой функции с изолированными критическими точками на замкнутой поверхности понимают ([1]) граф, образованный стягиванием каждой компоненты связности линий уровня функции в точку, с заданным порядком на вершинах, отвечающим порядку возрастания значений функции в соответствующих точках. Графы Кронрода-Риба называют изоморфными, если существует изоморфизм, сохраняющий порядок на вершинах.

Под обобщенным графом Кронрода-Риба непрерывного (дифференцируемого) отображения многообразий с изолированными сингулярными точками на замкнутом многообразии будем понимать граф, образованный стягиванием каждой компоненты связности линий уровня отображения в точку, с заданным порядком на вершинах, отвечающем порядку значений точек образа отображения.

Обобщенный граф Кронрода-Риба непрерывного (дифференцируемого) отображения многообразий допускает склеивающую перестройку, если существует непрерывное (дифференцируемое) отображение многообразий, хотя бы

на одной из сингулярных линий уровня которого расположено не менее двух сингулярных точек.

Прообразом обобщенного графа Кронрода-Риба, допускающего склеивающую перестройку, будем полагать обобщенный граф Кронрода-Риба, на каждом из сингулярных уровней которого расположена либо одна сингулярная точка, либо множество сингулярных точек, полученное при стягивании в точку связной компоненты образа отображения.

Существование прообразов классического графа Кронрода-Риба гладкой функции с невырожденными критическими точками (функции Морса) известно ([1], [2]).

Доказательство состоит в нахождении прообразов обобщенных графов Кронрода-Риба и определении их изоморфности.

Литература

[1] *Шарко В. В.* Гладкая и топологическая эквивалентность функций на поверхностях // Украинский матем. Журнал. — 2003.— Т. 55, № 5.— С. 687–700.

[2] *Арнольд В. И., Варченко А. Н., Гусейн-заде С. М.* Особенности дифференцируемых отображений. — М.: Наука, 1982. — 303 с.

Нижняя оценка длины полного проверяющего теста в базисе $\{x|y\}$

Ю. В. Бородина

jborodina@inbox.ru

ИПМ им. М. В.Келдыша РАН, Москва

Будем рассматривать схемы из функциональных элементов [1, 2] в некотором конечном базисе B. В качестве неисправностей предполагаем константные неисправности типа "1" на выходах элементов (при переходе в неисправное состояние элемент выдает значение 1 независимо от входных данных).

Пусть S — некоторая схема из функциональных элементов, реализующая булеву функцию $f(\tilde{x})$, $\tilde{x} = (x_1, x_2, \ldots, x_n)$ в базисе B.

Функция, реализуемая на выходе схемы при наличии в схеме неисправного элемента, называется *функцией неисправности*. Всякое множество T входных наборов схемы S называется *полным проверяющим тестом* для этой схемы, если для любой функции неисправности $g(\tilde{x})$, не равной тождественно $f(\tilde{x})$, в T найдется хотя бы один такой набор $\tilde{\sigma}$, что $f(\tilde{\sigma}) \neq g(\tilde{\sigma})$ [3, 4]. Число наборов, составляющих этот тест, называется *длиной* теста.

Введем обозначения [3, 4]: $D(T)$ — длина теста T; $D(S) = \min D(T)$, где минимум берется по всем полным проверяющим тестам T для схемы S; $D(f, B) = \min D(S)$, где минимум берется по всем схемам S в данном базисе B, реализующим функцию f; $D(n, B) = \max D(f, B)$; где максимум берется по всем булевым функциям f от n переменных. Функция $D(n, B)$ называется *функцией Шеннона* длины полного проверяющего теста для базиса B.

В работе [5] было показано, что $D(n, \{\&, \vee, {}^-\}) = 2$ для всех $n \geqslant 2$. Возникает естественный вопрос: верно ли, что для всякого конечного базиса B $D(n, B) \leqslant C(B)$, где $C(B)$— константа, не зависящая от n? Оказывается, это не так.

Теорема 1. *Для всякого $n \geqslant 2$ имеет место равенство*

$$D(x_1 \vee x_2 \vee \cdots \vee x_n, \{|\}) = n + 1$$

(здесь | обозначает штрих Шеффера).

Следствие. $D(n, \{|\}) \geqslant n + 1$, *при* $n \geqslant 2$.

Литература

[1] *Лупанов О. Б.* Асимптотические оценки сложности управляющих систем. — М.: МГУ, 1984.

[2] *Яблонский С. В.* Введение в дискретную математику. — М.: Высшая школа, 2002.

[3] *Яблонский С. В.* Некоторые вопросы надежности и контроля управляющих систем // Математические вопросы кибернетики. — 1988. — № 1. — С. 5–25.

[4] *Редькин Н. П.* Надежность и диагностика схем. — М.: МГУ, 1992.

[5] *Бородина Ю. В.* О синтезе легкотестируемых схем в случае однотипных константных неисправностей на выходах элементов // Вестник Московского университета. —Серия 15. Вычислительная математика и кибернетика. — 2008. — № 1. — С. 40–44.

Обзор современной теории нормализации в реляционных базах данных

Д. Б. Буй, А. В. Пузикова

buy@unicyb.kiev.ua, anna_inf@mail.ru

Киевский национальный университет имени Тараса Шевченко, Киев

Рассматривается эволюция классических нормальных форм, исторические попытки их улучшения и различные варианты неклассических нормальных форм в реляционных базах данных (БД).

Впервые термин «нормализация» был применен в 1970 г. Э. Коддом для названия процедуры избавления от непростых доменов [1]. В последующие годы среди профессионалов развязалась полемика относительно понятия «непростого домена», что влияло на интерпретацию первой нормальной формы (1НФ) [2, 3, 4]. Возможность использовать в качестве элементов домена группы (groups) и отношения отстаивали авторы работ [5, 6].

В 1971 г. Э. Кодд в работе [7] указывает на избыточность данных и аномалии, которые возникают при выполнении операций над отношениями, представляет концепцию функциональной зависимости (ФЗ), демонстрирует возможность её использования для решения проблем проектирования БД, приводит определения второй нормальной формы (2НФ), транзитивной ФЗ и

третьей нормальной формы (3НФ). В научной литературе встречаются их различные интерпретации. В частности, отличия в определениях 3НФ объясняются различными уточнениями, накладываемыми на определение транзитивной зависимости [8, 9].

Открытие аксиом и правил вывода ФЗ из заданного множества ФЗ [10], а также построение аксиоматики Армстронга для ФЗ [11], дали возможность разработать алгоритмы вычисления канонического покрытия (в терминологии Мейера [9]) для заданного множества ФЗ и замыкания для множества атрибутов [12, 9, 13, 14].

Одним из способов приведения отношения к 3НФ является декомпозиция, обоснованием которой стала теорема Хеза (Heath) [15]. К «подводным камням» декомпозиции относится зависимость проекций (по терминологии Риссанена (Rissanen) [16]), которая может стать причиной аномалий. Данная проблема рассмотрена в статье А. Филлиповича [17], который исследовал взаимные ФЗ (ВФЗ) и их свойства, предложил алгоритм обнаружения ВФЗ, понятие взаимно-независимой нормальной формы, которую можно рассматривать как синоним ациклической БД, и способ приведения к ней.

Другой способ предложил Бернштейн (Bernstein), представив алгоритм синтеза полной схемы базы данных в 3НФ для заданного множества функциональных зависимостей [18].

Специфика различных подходов к задаче проектирования схемы реляционной БД вызвана отличиями в формальных определениях эквивалентности и критериях качества схемы [19]. Известными классическими алгоритмами приведения схемы отношения к 3НФ являются алгоритмы Ульмана [13], Делобеля-Гейси (Delobel-Gasey) [10], результатами которых не всегда есть схема в 3НФ, Берштейна [18], Ислура (Isloor) [20], Неклюдовой-Цаленко [21], который дает количественно оптимальную схему БД; Мейера, реализованный через построение кольцевых покрытий [9]. Преимущества и недостатки большинства из указанных алгоритмов, а также их соответствие различным определениям эквивалентности реляционных схем рассмотрены в монографии [8]. Поиск эффективных алгоритмов решения задачи синтеза оптимальной схемы БД в 3НФ продолжается и сегодня (например, [22, 23, 14]).

Недостатки 3НФ были рассмотрены и учтены в работе Хеза [15] при формулировании определения усиленной 3НФ и, позднее, в работе Кодда [24] (известной как – нормальная форма Бойса-Кодда (НФБК)). Одним из первых алгоритмов приведения «почти» к НФБК является алгоритм Берштейна [18], который позволяет избавляться от транзитивных зависимостей первичных атрибутов от ключей, не содержащих эти атрибуты. Более поздние алгоритмы представлены, например, в работах [25, 26].

С введением многозначных зависимостей (МЗЗ) [27] Р. Фагин (Fagin) исследует их свойства и определяет новую четвертую нормальную форму (4НФ) [28]. Строгий и полный набор правил вывода для МЗЗ, а также правила, которые связывают ФЗ и МЗЗ, представлены в статье [29], независимость построенной системы аксиом обсуждается в работе Мендельзона (Mendelzon) [30], а её полнота — в статье Бискапа (Biskap) [31].

Зависимости соединения (ЗС) и аномалии, которые они вызывают, были рассмотрены в статье Риссанена (Rissanen) [32] и исследованы в работах [33, 34]. Концепция пятой нормальной формы (5НФ) представлена Р. Фагиным в статье [35]. Из результата об отсутствии полного конечного множества правил вывода для ЗС, который представлен в работе С. Петрова [36], следует невозможность построения полной и корректной аксиоматики, отсюда — невозможность построения канонического покрытия.

Изложению результатов в теории зависимостей и теории нормализации для реляционных БД посвящены работы Д. Мейера [9], В. Дрибаса [8], М. Цаленко [37].

Кроме указанных 1-5НФ предлагались различные варианты улучшения классических нормальных форм, например, улучшенная 3НФ (An Improved Third Normal Form) [38], а также — введение других видов нормальных форм, например, нормальной формы с элементарным ключом (Elementary Key Normal Form), которая занимает промежуточное положение между 3НФ и НФБК [39], нормальной формы с полным ключом (Key-Complete Normal Form) [40], доменно-ключевой нормальной формы (ДКНФ) [41], кортеже-необходимой нормальной формы (Essential Tuple Normal Form) [42], которая определяется в терминах ФЗ и ЗС, и занимает промежуточное положение между 4НФ и 5НФ. Отдельным видом является шестая нормальная форма (6НФ), которая была введена в 2002 г. для хронологических БД [43].

Одним из современных направлений развития классической теории нормализации реляционных БД является распространение её принципов на нечеткие реляционные базы данных (НРБД) [44, 45].

Подведем итоги. Несмотря на весомые результаты, теория нормализации в реляционных БД носит фрагментарный характер и далека еще от удовлетворительного завершения. Авторы статьи также имеют некоторые наработки по данной теме, а именно: было предложено строгое математическое доказательство корректности и полноты аксиоматики Армстронга, выполненное в традициях математической логики путем установления совпадения семантического и синтаксического следований [46]; предложен критерий полноты аксиоматики Армстронга в терминах мощностей множества атрибутов и универсального домена [47]. Отметим, что нормализация, основной целью которой есть поддержка целостности данных, вступает в противоречие с эффективностью обработки операций в БД; именно поэтому сейчас уже можно говорить о начале создания теории денормализации и о естественном (точнее говоря, оптимальном по определенному критерию) синтезе нормализации и денормализации.

Литература

[1] *Codd E. F.* A Relational Model of Data for Large Shared Data Banks // Communications of the ACM. — 1970. — V. 13, № 6. — P. 377–387.

[2] *Дейт К. Дж.* Введение в системы баз данных. — М.: Наука, 1980. — 464 с.

[3] *Elmasri R., Navathe S. R.* Fundamentals of Database Systems. — Massachusetts: Addison–Wesley, 2000. — 893 p.

[4] *Дейт К. Дж.* Введение в системы баз данных. — М.: Вильямс, 2005. — 1328 с.

[5] *Jaeschke G., Schek H. J.* Remarks on the Algebra of Non First Normal Form Relations // Proceedings of the 1st ACM SIGACT-SIGMOD symposium on Principles of database systems. — Los Angeles, California, 1982. — P. 124–138.

[6] *Makinouchi A.* A Consideration of Normal Form on Not-necessarily Normalized Relations in the Relational Data Model // Proceedings of the Third International Conference on Very Large Data Bases. — Tokyo: IEEE Computer Society, 1977. — P. 447–453.

[7] *Codd E. F.* Further Normalization of the Data Base Relational Model // Data Base Systems: Courant Computer Science Symposia Series 6. — New York: Prentice-Hall, 1972.

[8] *Дрибас В. П.* Реляционные модели баз данных. — Минск: БГУ им. В. И. Ленина, 1982. — 192 с.

[9] *Мейер Д.* Теория реляционных баз данных. — М.: Мир, 1987. — 608 с.

[10] *Delobel C. Gasey R.* Decomposition of a database and the theory of Boolean switching function // IBM Journal of Research and Development. — 1973. — V. 17, № 5. — P. 374–386.

[11] *Armstrong W. W.* Dependency structures of data base relationships // Proc. IFIP '74. — Amsterdam: North Holland, 1974. — P. 580–583.

[12] *Beeri C., Bernstein P. A.* Computational problems related to the design of normal form relation schemas // ACM Transactions on Database Systems. — 1979. — V. 4, № 1. — P. 30–59.

[13] *Ульман Дж.* Основы систем баз данных. — М.: Финансы и статистика, 1983. — 334 с.

[14] *Григорьев Ю. А.* Алгоритм синтеза частично оптимальной схемы реляционной базы данных // Электронное научно-техническое издание «Наука и образование». — 2012. — No 1. — Режим доступа: http://technomag.edu.ru/doc/294486.html.

[15] *Heath I. J.* Unacceptable File Operations in Relational Database // ACM SIGFIDET Workshop on Data Description, Access, and Control. — San Diego, Calif., 1971. — P. 19–33.

[16] *Rissanen J.* Independent components of relations // ACM Transactions on Database Systems. — 1977. — V. 2, № 4. — P. 317–325.

[17] *Филлипович А.* Взаимные функциональные зависимости // Системный администратор. — 2002. — No 1. — С. 84–89.

[18] *Bernstein P. A.* Synthesizing Third Normal Form relations from functional dependencies // ACM Transactions on Database Systems. — 1976. — V. 1, № 4. — P. 277–298.

[19] *Beeri C., Bernstein P., Goodman N.* A sophisticate's introduction to database normalization theory // Proceedings of 4th International Conference on Very Large Data Bases. — West Berlin, 1978. — P. 113–124.

[20] *Isloor S. S.* An algorithm with logical simplicity for designing third normal form relations data base schema for functional dependencies // Proceedings of International Conference on DBMS (ICMOD 78). — Fast Milano, Italy, 1978. — P. 31–50.

[21] *Неклюдова Е. А., Цаленко М. Ш.* Синтез логической схемы реляционной базы данных // Программирование. — 1979. — No 6. — С. 58–68.

[22] *Зорин И.* Теоретико-графовое приведение реляционной базы данных к третьей нормальной форме Э. Кодда // Электронное научное издание «Устойчивое инновационное развитие: проектирование и управление». — 2009. — Т. 5. — С. 50–59.

[23] *Виноградова М. В., Игушев Э. Г.* Конструктор баз данных на основе сущностей и их реквизитов с возможностью нормализации // Электронное научно-техническое издание «Наука и образование». — 2011. — No 10.

[24] *Codd E. F.* Recent Investigations into Relational Data Base Systems // Proceedings of IFIP Congress 74. — Stockholm: North-Holland, 1974. — P. 1017–1021.

[25] *Lin W. Y.* Efficient algorithm for BCNF-decomposition // Information and Software Technology. — 1992. — V. 34, № 5. — P. 308–312.

[26] *Bahmani A., Naghibzadeh M., Bahmani B.* Automatic database normalization and primary key generation // CCECE/CCGEI. — Niagara Falls, Canada, 2008. — P. 11–16.

[27] *Zaniolo C.* Analysis and design of relational schemata for database systems: — Ph.D. dissertation, Tech. Rep. UCLA-Eng-7769, Dep. Computer Science, Univ. California at Los Angeles, July 1976.

[28] *Fagin R.* Multivalued Dependencies and a New Normal Form for Relational Databases // ACM Transactions on Database Systems. — 1977. — V. 2, № 1. — P. 262–278.

[29] *Beeri C., Fagin R., Howard J.* A complete axiomatization for functional and multivalued dependencies // Proc. ACM-SIGMOD Conf. (Toronto, Canada, Aug. 3-5). — ACM, New York, 1977. — P. 47–61.

[30] *Mendelzon A. O.* On axiomatizing multivalued dependencies in relational databases // ACM Transactions on Database Systems. — 1979. — V. 26, № 1. — P. 37–44.

[31] *Biskup J.* Inferences of multivalued dependencies in fixed and undetermined universes // Theoretical Computer Science. — 1980. — V. 10, № 1. — P. 93–105.

[32] *Rissanen J.* Independent components of relations // ACM Transactions on Database Systems. — 1977. — V. 2, № 4. — P. 317–325.

[33] *Aho A. V., Beeri C., Ullman J. D.* The theory of joins in relational databases // Proc. 18th Symp. on Foundations of Computer Science. — Providence, R.I., 1977. — P. 107–113.

[34] *Dayal U., Bernstein P. A.* The fragmentation problem: lossless decomposition of relations into files // Proceedings of the ACM SIGMOD international conference on Management of data, 1979. — P. 143–151.

[35] *Fagin R.* Normal Forms and Relational Database Operators // Proceedings of the ACM SIGMOD International Conference on Management of Data (Boston, Mass., May 30-June 1). — ACM, New York, 1979. — P. 153–160.

[36] *Петров С. В.* Об аксиоматизации зависимостей по соединению // Применение методов математической логики: Тезисы докл. IV Всес. конф. «Представление знаний и синтез программ». — Таллин: АН ЭССР, 1986. — С. 151–152.

[37] *Цаленко М. Ш.* Моделирование семантики в базах данных. — М.: Наука, 1989. — 287 с.

[38] *Ling T. W., Tompa F. W., Kameda T.* An Improved Third Normal Form for Relational Databases // ACM Transactions on Database Systems. — 1981. — V. 6, № 2. — P. 329–346.

[39] *Zaniolo C.* A New Normal Form for the Design of Relational Database Schemata // ACM Transactions on Database Systems. — 1982. — V. 7, № 3. — P. 489–499.

[40] *Vincent M. W.* Redundancy Elimination and a New Normal Form for Relational Database Design // In Semantics in Databases. — 1998. — V. 1358 of LNCS. — P. 247–264.

[41] *Fagin R.* A Normal Form for Relational Databases That Is Based on Domains and Keys // Communications of the ACM. — 1981. — V. 6. — P. 387–415.

[42] *Darwen H., Date C., Fagin R.* A Normal Form for Preventing Redundant Tuples in Relational Databases // Proceedings of the 15th International Conference on Database Theory – ICDT'2012. — Berlin, Germany, 2012. — P. 114–126.

[43] *Date C., Darwen H.* Temporal Data and the Relational Model. — Lorentzos: Morgan Kaufmann, 2002. — 422 p.

[44] *Chen G., Kerre E. E., Vandenbulcke J.* Normalization based on ffd in a fuzzy relational data model // Inform Syst. — 1996. — V. 21. — P. 299–310.

[45] *Bahar Ö., Yazici A.* Normalization and Lossless Join Decomposition of Similarity-Based Fuzzy Relational Databases // International Journal of Intelligent Systems. — 2004. — V. 19. — P. 885–917. Published online in Wiley InterScience (www.interscience.wiley.com).

[46] *Буй Д. Б., Пузикова А. В.* Полнота аксиоматики Армстронга // Вестник Киевского национального университета имени Тараса Шевченко, Серия: Физико-математические науки. — 2011. — No 3. — С. 103–108.

[47] *Буй Д. Б., Пузикова А. В.* Критерий полноты аксиоматики Армстронга // Материалы международной конференции «Теоретические и прикладные аспекты построения программных систем» – TAAPSD'2011. — Ялта, 2011. — С. 30–34.

Модели и методы обеспечения безопасности программных средств (обзор)

Д. Б. Буй, В. Г. Скобелев

dmitriybuy@mail.ru, skvb@iamm.ac.donetsk.ua

Киевский национальный университет им. Тараса Шевченко

Доклад представляет собой обзор (на основе анализа 92 источников) моделей и методов, предназначенных для обеспечения безопасности программных средств современных компьютерных систем (КС) на протяжении всего их жизненного цикла. Рассмотрены подходы к разработке технологий, обеспечивающих безопасность программных средств (ПС), и возникающие при этом сложности. Охарактеризована общая схема, применяемая для построения системы защиты ПС, а также модели и методы, применяемые в процессе валидации и верификации ПС. Указаны возможные направления соответствующих исследований.

Известно, что обеспечение надежности функционирования является одной из наиболее актуальных проблем для КС с критической областью применения. Требования, предъявляемые к безопасности КС, устанавливаются рядом международных стандартов. На их основе формируется технология обеспече-

ния безопасности КС на протяжении всего их жизненного цикла. Существенной составляющей такой технологии является технология создания ПС (программного обеспечения и баз данных), обеспечивающая безопасность функционирования ПС, т.е. функциональную, технологическую и эксплуатационную безопасности ПС [1, 2]. В настоящее время проблемы обеспечения функциональной и технологической безопасности ПС изучены недостаточно и на их решение направлены основные усилия, а проблема обеспечения эксплуатационной безопасности ПС достаточно полно проработана и постепенно отходит на второй план [3].

Значение технологий, обеспечивающих безопасность ПС, стимулировало появление соответствующих методологий:, например, Microsoft Solutions Framework (MSF) [4], Rational Unified Process (RUP) [5, 6, 7], EX-treme Program-ming (XP) [8]. В каждом из подходов большое значение имеет тестирование, представляющее собой управляемое исполнение программ, предназначенное для проверки их корректности. Выделяют следующие типы тестирования ПС [9]: модульное, интеграционное, системное, нагрузочное. Понятия верификации и валидации ПС последовательно уточнялись стандартами IEEE 610.12-1990 [10], IEEE 1012-2004 [11] и ISO/IEC 12207 [12]. Верификация ПС представляет собой проверку соответствия их характеристик заданным требованиям и стандартам. Отметим, что верификация ПС выполняется всегда, а методы ее осуществления существенно используют формальные модели [13]. Валидация ПС представляет собой проверку соответствия их характеристик потребностям пользователей и заказчиков ПС [14, 15]. Валидация является значительно менее формализованной областью чем верификация.

Основными этапами разработки системы защиты ПС являются [16]: выделение характеристик, влияющих на безопасность функционирования ПС; выбор принципов обеспечения безопасности функционирования ПС; определение категорий отказов, определяющих безопасность функционирования ПС; выделение уровней безопасности ПС в зависимости от ситуаций, возникающих при отказах; определение типов внешних и внутренних угроз безопасности функционирования ПС; распределение ресурсов, предназначенных для создания системы защиты ПС; выбор и реализация системы защиты ПС. В докладе эти этапы подробно рассмотрены. Выбор системы защиты ПС определяет способ его установки, а также механизмы и методы защиты ПС [17].

В настоящее время выделяют следующие группы методов верификации ПС: экспертиза, статический анализ, формальные методы, динамический анализ, синтетические методы [18]. В докладе дана подробная характеристика этим группам методов. Подробнее остановимся на формальных методах верификации. Основными методами построения доказательства того, что формальная модель удовлетворяет спецификации, являются проверка модели (model checking) [19] и логический вывод [20]. Проверка модели представляет собой автоматический анализ формальной модели, результатом которого в случае, когда модель не удовлетворяет спецификации, является «опровергающее вычисление», т.е. последовательность действий, на которых нарушается спецификация. Логический вывод состоит в построении и формальном дока-

зательстве утверждений (инвариантов), которые истинны в каждый момент времени и из конъюнкции которых вытекает спецификация модели.

В докладе рассмотрены модели и методы, применяемые для решения многогранной проблемы обеспечения безопасности функционирования современных ПС на протяжении всего их жизненного цикла. Существующие в настоящее время подходы к решению этой проблемы далеки от проработанной технологии. Отметим некоторые задачи, решение которых необходимо для создания указанной технологии. Во-первых, разработка таких эффективных методов валидации ПС, которые исключат неоправданное удлинение срока выполнения проекта. Во-вторых, создание эффективных автоматизированных средств верификации логико-алгебраических моделей и абстрактных машин. В-третьих, разработка эффективных средств, предназначенных для противодействия несанкционированному исследованию ПС. Достижение этой цели требует системного анализа соотношения между современными средствами противодействия несанкционированному исследованию ПС и методами их преодоления [21, 22, 23]. В частности, требуют внимания теоретические и прикладные исследования методов запутывания программ с помощью соответствующих преобразований (obfuscation transformations) [24, 25].

Литература

[1] *Гайдамакин Н. А.* Теоретические основы компьютерной безопасности. — Екатеринбург: Изд-во Уральского университета, 2008. — 212 с.

[2] *Щербаков А. Ю.* Современная компьютерная безопасность. Теоретические основы. Практические аспекты. — М.: Книжный мир, 2009. — 352 с.

[3] *Шаньгин В. Ф.* Информационная безопасность компьютерных систем и сетей. — М.: Форум, 2011. — 416 с.

[4] Microsoft solutions framework // [Электронный ресурс] Режим доступа: http://www.microsoft.com/Rus/Msdn/msf/Default.mspx.

[5] Unified Process. Методология и технология. Материалы компании Interface Ltd // [Электронный ресурс] Режим доступа: http://www.interface.ru/home.asp?artId=779.

[6] *Кратчен Ф.* Введение в Rational Unified Process. — М.: Вильямс, 2002. — 239 с.

[7] *Якобсон А., Буч Г., Рамбо Д.* Унифицированный процесс разработки программного обеспечения. — СПб.: Питер, 2002. — 496 с.

[8] *Бек К.* Экстремальное программирование. — СПб.: Питер, 2002. — 224 с.

[9] *Скобелев В. Г.* Безопасность IT-систем (обзор) // Радіоелектронні і комп'ютерні системи. — 2013. — № 5. — С. 352–361.

[10] IEEE 610.12-1990 Standard glossary of soft-ware engineering terminology, corrected edition. — IEEE, 1991.

[11] IEEE 1012-2004 Standard for verification and validation. — IEEE, 2005.

[12] ISO/IEC 12207 Systems and software engi-neering – software life cycle processes. — ISO, 2008.

[13] *Кулямин В. В.* Методы верификации программного обеспечения // [Электронный ресурс] Режим доступа: http://www.viva64.com/go.php?url=282.

[14] *Rakitin S. R.* Software verification and validation: a practitioner's guide. — Boston.: Artech House, 1997. — 271 p.

[15] *Чень М., Цинь К., Ку Х. М., Мишара П.* Валидация на системном уровне. Высокоуровневое моделирование и управление тестированием. — М.: Техносфера, 2014. — 296 с.

[16] *Липаев В. В.* Функциональная безопасность программных средств // Jet Info. — 2004. — № 8. — С. 3–28.

[17] *Середа С. А.* Оценка эффективности систем защиты программного обеспечения // [Электронный ресурс] Режим доступа: http://www.consumer.nm.ru/sps_eval.htm.

[18] *Синицин С. В., Налютин Н. Ю.* Верификация программного обеспечения. — М.: МИФИ, 2006. — 157 с.

[19] *Кларк Э., Грамберг О., Пелед Д.* Верификация моделей программ: model checking. — М.: МЦНМО, 2002. — 416 с.

[20] *Eriksson J., Back R. J.* Applying PVS background theories and proof strategies in invariant based programming // LNCS. — 2010. — V. 6447. — P. 24–39.

[21] *Расторгуев С. П., Дмитриевский Н. Н.* Искусство защиты и раздевания программ. — М.: Совмаркет, 1991. — 94 с.

[22] *Семьянов П. В., Зегжда Д. П.* Анализ средств противодействия исследованию программного обеспечения и методы их преодоления // [Электронный ресурс] Режим доступа: http://www.password-crackers.com/publications/research.txt.

[23] *Середа С. А.* Анализ средств преодоления систем защиты программного обеспечения // Радиоэлектроника и Телекоммуникации. — 2002. — № 4. — С. 11–16.

[24] *Чернов А. В.* Анализ запутывающих преобразований программ // [Электронный ресурс] Режим доступа: http://www.citrorum.ru/security/articles/analysis/.

[25] *Barak B., Goldreich O., Impagliazzo R., Rudich S., Sahai A., Vadhan S.* On the (Im)possibility of Obfuscating Programs // LNCS. — 2001. — V. 2139. — P. 1–18.

Интеллектуальный подход к извлечению знаний из данных с помощью алгоритмов индуктивного моделирования

А. С. Булгакова, В. В. Зосимов

sashabulgakova@list.ru, zosimovvv@bk.ru

Николаевский национальный университет им.В.А. Сухомлинского, Николаев

Введение

С ростом объемов накопленных информационных баз данных необходимы новые методы, алгоритмы и программные средства интеллектуального анализа данных для построения моделей сложных процессов и систем. Создание адекватных и относительно простых программ, которые будут «извлекать» необходимые знания из данных, значительно облегчит работу человека. На сегодняшний день, одним из эффективных методов решения данной проблемы является индуктивная самоорганизация моделей по экспериментальным

данным (индуктивное моделирование). В этом подходе к моделированию вместо традиционного дедуктивного пути «от общих закономерностей функционирования объекта - к конкретной математической модели» используется индуктивный подход «от конкретных данных наблюдений - к общей модели»: исследователь представляет выборку, выдвигает гипотезу о возможном классе моделей и задает критерий выбора лучшей модели в этом классе. Далее работает компьютер, соответственно появляется возможность минимизировать влияние субъективных факторов и получить модель как объективный результат [1].

Анализ данных с помощью индуктивной самоорганизации моделей

Анализ данных с помощью индуктивной самоорганизации моделей Задача анализа данных ставится следующим образом:

— имеется достаточно большая база данных (bigdata);
— предполагается, что в базе данных находятся некие «скрытые знания».

Необходимо разработать методы обнаружения знаний, скрытых в больших объёмах исходных данных и классифицировать их по определенным признакам. В текущих условиях глобальной конкуренции именно найденные закономерности (знания) могут быть источником дополнительного конкурентного преимущества.

Одним из таких методов могут выступать методы индуктивной самоорганизации моделей, а именно метод группового учета аргументов - МГУА, который позволяет автоматически находить ранее неизвестные функциональные зависимости, заложенные в выборке данных и, следовательно, открывать новые знания [2].

МГУА основан на принципах теории обучения и самоорганизации, в частности, на принципе массовой «селекции» или самоорганизующемся направленном переборе всевозможных вариантов построения решающего правила классификации [2]. Задача построения решающего правила в МГУА представляется как задача индуктивного построения модели, усложняющейся в процессе работы алгоритма.

В предлагаемой работе рассматривается класс задач моделирования, который содержит информацию о n измерениях, m входных переменных (признаков) $X[n x m]$ и одной выходной переменной $y[n x 1]$. Необходимо найти модель зависимости вход-выход или функцию принадлежности к определенным классам.

Постановка задачи. Задача построения модели с выбором ее структуры и оценки параметров сводится к формированию по выборке данных некоторого множества Ф моделей-кандидатов $f \in \Phi$ различной структуры функции (1):

$$\hat{y}_f = f(X, \hat{\theta}_f), \tag{1}$$

где $\hat{\theta}_f = \langle \theta_0, \theta_i, \theta_{ij}, \theta_{ijk}, ... \rangle$ - вектор коэффициентов, и поиску оптимальной мо-

дели из этого множества Φ как решения задачи дискретной оптимизации при условии минимума внешнего критерия селекции CR:

$$f^* = \arg\min_{f \in \Phi} CR(y, f(X, \hat{\theta}_f)), \tag{2}$$

В роли критерия селекции можно использовать критерий регулярности (3), который основан на разбиении выборки на обучающую (А) и проверочную (В):

$$AR_{B|A} = \|y_B - \hat{y}_{B|A}\|^2 = \|y_B - X_B \hat{\theta}_A\|^2. \tag{3}$$

На сегодня разработаны и исследованы много разновидностей алгоритмов МГУА переборного [2] и итерационного типов. Переборные алгоритмы эффективны как средство структурной идентификации, но только для ограниченного числа аргументов. Итерационные алгоритмы работоспособны при достаточно большом количестве аргументов. Наиболее эффективным итерационным алгоритмом в задачах анализа данных является обобщенный итерационный алгоритм, т.к. он обобщает все предыдущие архитектуры итерационных алгоритмов, объединяя в себе их сильные стороны [3].

Обобщенный итерационный алгоритм. Обобщенный итерационный алгоритм (ОИА) предназначен для работы с большими наборами данных. Формально в общем случае для ряда r определим ОИА следующим образом:

1) входной матрицей является $X_{r+1} = (y_1^r, ..., y_F^r, x_1, ..., x_m)$;

2) применяются операторы перехода вида (4) и (5) с квадратичным частным описанием:

$$y_l^{r+1} = f(y_i^r, y_j^r), l = 1, 2, ..., C_F^2, i, j = \overline{1, F}, \tag{4}$$

$$y_l^{r+1} = f(y_i^r, x_j), l = 1, 2, ..., F_m, i = \overline{1, F}, j = \overline{1, m}; \tag{5}$$

3) для каждого описания находится оптимальная структура (6):

$$f(u, v) = a_0 d_1 + a_1 d_2 u + a_2 d_3 v + a_3 d_4 uv + a_4 d_5 u^2 + a_5 d_6 v^2, \tag{6}$$

где $d_k =< 0, 1 >$, $d_{opt} = \arg\min_{l=\overline{1,q}} CR_l$, $q = 2^p - 1, f_{opt}(u, v) = f(u, v, d_{opt})$;

4) алгоритм останавливается при выполнении условия $CR_{min}^{r+1} \geqslant CR_{min}^r$, и оптимальная модель соответствует значению CR_{min}^r на r-м ряду.

Пути повышения быстродействия вычислений в ОИА. Поскольку оптимизация структуры частных моделей происходит с помощью комбинаторного алгоритма, т.е. полным перебором всех возможных комбинаций одночленов частных описания, это требует значительных затрат времени. Значительные затраты времени при этом идут на формирование матриц систем условных уравнений для каждого варианта структуры каждого частных описаний, по которым вычисляются соответствующие матрицы систем нормальных уравнений. Далее в результате решения этих уравнений получаются

оценки параметров и вычисляются значения критериев вариантов, после чего по минимальному из этих значений выбирается лучший вариант. Это и есть описание работы активного нейрона полиномиальной нейросети МГУА. Но формирование матриц условных уравнений для каждого частного описания с последующим вычислением соответствующих нормальных систем в комбинаторном алгоритме нецелесообразно, так как приводит к многократным вычислениям одних и тех же величин [4].Таким образом, для получения оценок коэффициентов всех возможных вариантов моделей каждого частных описания достаточно один раз построить полную матрицу и «вытягивать» из нее необходимые частные нормальные системы. Численные эксперименты по использованию данного подхода на искусственных данных описаны в [3-4].

Выводы

В статье рассмотрен интеллектуальный подход к анализу данных, основанный на индуктивной самоорганизации моделей по экспериментальным данным. Предложенный обобщенный итерационный алгоритм предназначен для работы с большими наборами данных и обобщает все предыдущие архитектуры итерационных алгоритмов, объединяя в себе их сильные стороны. Кроме того, описаны пути повышения быстродействия вычислений.

Литература

[1] *Степашко В. С.* Теоретические аспекты МГУА как метода индуктивного моделирования // УсиМ. — 2003. № 2. — С. 54–62.

[2] *Ivakhnenko A., Ivakhnenko G.,Muller J.* Self-Organization of Optimum Physical Clusteing of Data Sample for a Weakened Description and Forecasting of Fuzzy Objects // Pattern Recognition and Image Analysis. — 1993. — V. 3, № 4. — P. 415–422.

[3] *Степашко В. С., Булгакова А. С.* Обобщенный итерационній алгоритм метода группового учета аргументов // УсиМ. — 2013. № 2. — С. 5–18.

[4] *Зосимов В. В.,Булгакова А. С.* Использование GRID-систем для распределения процесса вычислений по данным в алгоритмах индуктивного моделирования // Матеріали міжнародної наукової конференції «Інтелектуальні системи прийняття рішень та проблеми обчисленого інтелекту, ISDMCI»— Євпаторія, 2013. — P. 416–418.

Анализ устойчивости векторной инвестиционной булевой задачи Марковица с критериями Вальда в метрике Гёльдера

С. Е. Бухтояров, В. А. Емеличев

emelichev@tut.by

Белорусский государственный университет, Минск

Рассматривается s-критериальный дискретный вариант известной инвестиционной задачи Марковица [1] с максиминными критериями Вальда и паретовским принципом оптимальности:

$$\max_{x\in X}\min_{i\in N_m}\sum_{j\in N_n} e_{ijk}x_j,\quad k\in N_s=\{1,2,\ldots,s\}. \tag{1}$$

Здесь n – количество альтернативных инвестиционных проектов (активов); m – количество прогнозных состояний финансового рынка, т.е. число вариантов сценариев развития; s – количесвто показателей экономической эффективности (доходности) инвестиционного проекта. Пусть $x_j = 1$, если j-й проект ($j \in N_n$) реализуется, и $x_j = 0$ в противном случае. Инвестиционным портфелем назовем булевый вектор $x = (x_1, x_2, \ldots, x_n)^T$. Через $X \subset \mathbf{E}^n$, где $\mathbf{E} = \{0, 1\}, |X| > 1$, будем обозначать множество всех допустимых инвестиционных портфелей, т. е. тех, реализация которых не превосходит начального капитала инвестора. Инвестиционный портфель x будем оценивать величиной $\sum_{j\in N_n} e_{ijk}x_j$, где e_{ijk} – ожидаемая оценка эффективности вида $k \in N_s$ инвестиционного проекта $j \in N_n$ в случае, когда рынок находится в состоянии $i \in N_m$ [2, 3]. В этом контексте исходными данными задачи является трехмерная матрица эффективности проектов E размером $m \times n \times s$ с элементами e_{ijk} из $\mathbf{R}$.

Множество Парето, состоящее из парето-оптимальных инвестиционных портфелей, обозначим через $P^s(E)$. Радиусом устойчивости задачи (1), как обычно [4, 5], будем называть число

$$\rho(m,n,s,p)=\begin{cases}\sup\Xi, & \text{если } \Xi\neq\varnothing,\\ 0 & \text{в противном случае,}\end{cases}$$

где

$$\Xi=\{\varepsilon>0\ :\ \forall E'\in\Omega_p(\varepsilon)\ \ (P^s(E+E')\subseteq P^s(E))\},$$
$$\Omega_p(\varepsilon)=\{E'\in\mathbf{R}^{m\times n\times s}\ :\ ||E'||_p<\varepsilon\},$$

$|E'||_p$ – норма Гёльдера $l_p, 1 \leqslant p \leqslant \infty$, матрицы E'. Ясно, что при выполнении равенства $P^s(E) = X$ радиус устойчивости равен бесконечности. Задачу, для которой $P^s(E) \neq X$, будем называть нетривиальной.

Используя классическое неравенство Гёльдера

$$a^Tb\leqslant||a||_p||b||_q,$$

где $a, b \in \mathbf{R}^n$, а величины p и q связаны соотношением $1/p+1/q = 1$ и $p \in [1, \infty]$, доказана следующая

Теорема 1. *При любых $m, n, s \in \mathbf{N}$ и $p \in [1, \infty]$ для радиуса устойчивости нетривиальной задачи (1) справедливы следующие оценки*

$$\min_{x\notin P^s(E)}\ \max_{x'\in P(x,E)}\frac{\gamma(x,x')}{||x+x'||_1^{1/q}}\leqslant\rho(m,n,s,p)\leqslant$$
$$\leqslant(mns)^{1/p}\min_{x\notin P^s(E)}\ \max_{x'\in P(x,E)}\frac{\gamma(x,x')}{||x-x'||_1} \tag{2}$$

где $P(x,E) = \{x' \in P^s(E) : f(x') \geqslant f(x)\ \&\ f(x') \neq f(x)\}$; $f(x) = (f_1(x), f_2(x), \ldots, f_s(x))$; $f_k(x) = \min\Big\{\sum_{j\in N_n} e_{ijk} : i \in N_m\Big\}$, $k \in N_s$; $\gamma(x,x') = \min\{f_k(x') - f_k(x) : k \in N_s\}$.

Отметим, что ранее в [6] аналогичные результаты были получены для радиуса устойчивости парето-оптимального портфеля многокритериальной инвестиционной задачи с критериями Вальда.

Из теоремы вытекает ранее известный результат.

Следствие 1. [5] *При любых* $m, n, s \in \mathbf{N}$ *справедливы оценки*

$$\min_{x \notin P^s(E)} \max_{x' \in P(x,E)} \frac{\gamma(x,x')}{||x+x'||_1} \leqslant \rho(m,n,s,\infty) \leqslant \min_{x \notin P^s(E)} \max_{x' \in P(x,E)} \frac{\gamma(x,x')}{||x-x'||_1}$$

Из этого следствия получим следующее утверждение, которое свидетельствует о достижимости оценок (2) в случае, когда $p = \infty$.

Следствие 2. *Если для всякой пары портфелей* $x \notin P^s(E)$ *и* $x' \in P(x,E)$ *множество* $\{j \in N_n \ : \ x_j = x'_j = 1\}$ *пусто, то при любых* $m, n, s \in \mathbf{N}$ *верна формула*

$$\rho(m,n,s,\infty) = \min_{x \notin P^s(E)} \max_{x' \in P(x,E)} \frac{\gamma(x,x')}{||x-x'||_1}$$

Работа выполнена при частичной финансовой поддержке Белорусского республиканского фонда фундаментальных исследований (проект Ф13К-078).

Литература

[1] *Markowitz H. M.* Portfolio selection: efficient diversification of investments. — Oxford: Wiley-Blackwell, 1991. — 402 p.

[2] *Виленский П. Л., Лившиц В. Н., Соляк С. А.* Оценки эффективности инвестиционных проектов: теория и практика. — М. 2008.

[3] *Царев В. В.* Оценки экономической эффективности инвестиций. — СПб. 2004.

[4] *Емеличев В. А., Коротков В. В.* Устойчивость векторной инвестиционной булевой задачи с критериями Вальда // Дискретная математика. — 2012. — Т. 24, № 3. — С. 3–16.

[5] *Емеличев В. А., Коротков В. В.* О радиусе устойчивости векторной инвестиционной задачи с критериями минимаксного риска Сэвиджа // Кибернетика и системный анализ. — 2012. — № 3. — С. 68–77.

[6] *Емеличев В. А., Коротков В. В.* Об устойчивости решений многокритериальной инвестиционной задачи в метрике Гёльдера // Весці НАН Беларусі. Сер. фіз.-мат. навук. — 2012. — № 4. — С. 42–48.

Квантовые коммуникационные вычисления на основе квантового хеширования

А. В. Васильев

alexander.ksu@gmail.com

Казанский федеральный университет, Казань

Квантовое хеширование

В работе [1] нами предложен метод криптографического квантового хеширования, позволяющий представлять классическую информацию в виде

квантовой суперпозиции специального вида. Предложенная функция является необратимой, т.е. обеспечивает невозможность извлечения ключа из его хеш-кода.

Пусть n – длина хешируемых сообщений (рассматриваемых как n-битные числа, $q = 2^n$, $B =\subset \{0, \ldots, q-1\}$. Квантовая хеш-функция $\psi_{q,B} : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes(\log|B|+1)}$ определяется следующим образом. Для сообщения $w \in \{0,1\}^n$ положим

$$|\psi_{q,B}(w)\rangle = \frac{1}{\sqrt{|B|}} \sum_{i=1}^{|B|} |i\rangle \left(\cos \frac{2\pi b_i w}{q} |0\rangle + \sin \frac{2\pi b_i w}{q} |1\rangle \right). \qquad (1)$$

Предложенный нами метод хеширования включает множество параметров B, от которых зависит вероятность коллизий (обозначим ее δ), и нами предложен ряд алгоритмов по вычислению этих параметров для заданного δ. Другими словами, в предложенном хешировании заложена возможность повышения устойчивости средствами самого метода. Отметим, что при увеличении устойчивости (уменьшении δ), например, в 1 000 000 раз, получаемый квантовый хеш увеличится лишь на 20 кубит.

Также отметим, что от мощности множества B зависит размер квантового хеш-кода, который по построению равен $\log|B| + 1$. Таким образом, выбор множества B является ключевым для построения устойчивого и компактного квантового хеширования. Нами был доказан следующий результат.

Теорема 1. *Для произвольного $\delta > 0$ существует такое множество $B \subset \mathbb{Z}_q$ такое, что $|B| = \lceil (2/\delta^2) \ln(2q) \rceil$ и функция $\psi_{q,B}$ является δ-устойчивой квантовой хеш-функцией, т.е. квантовые хеш-коды различных сообщений $w_1 \neq w_2$ близки к ортогональным:*

$$|\langle \psi_{q,B}(w_1) \,|\, \psi_{q,B}(w_2) \rangle| < \delta. \qquad (2)$$

Эффективные квантовые коммуникационные вычисления

В данной работе рассмотрена модель квантовых односторонних коммуникационных вычислений, в которой первый участник $\mathcal{A}$ проводит часть вычислений, передает сообщение (некоторое квантовое состояние) второму участнику $\mathcal{B}$, который завершает вычисления и выдает результат. Под сложностью вычислений в данной модели понимается количество передаваемых кубитов.

Нами предложен квантовый коммуникационный протокол для вычисления булевых функций, основанный на квантовом хешировании и представлении булевых функций характеристическими полиномами [2].

Теорема 2. *Пусть $f(x_1, \ldots, x_{n_1}, y_1, \ldots, y_{n_2})$ – булева функция от $n = n_1 + n_2$ переменных, а полиномы $g_1(x_1, \ldots, x_{n_1})$ и $g_2(y_1, \ldots, y_{n_2}\}$ над кольцом $\mathbb{Z}_q$ таковы, что их сумма $g = g_1 + g_2$ равна нулю тогда и только тогда, когда $f = 1$. Тогда для произвольного $\delta \in (0,1)$ f может быть вычислена квантовым односторонним коммуникационным протоколом сложности $O(\log\log q + \log(1/\delta))$.*

Схема доказательства.

Пусть вычислитель $\mathcal{A}$ получает на вход последовательность $\sigma = \sigma_1 \dots \sigma_{n_1}$ значений первых n_1 переменных, а $\mathcal{B}$ – последовательность $\gamma = \gamma_1 \dots \gamma_{n_2}$ значений оставшихся n_2 переменных. Опишем протокол вычисления данной функции в квантовой односторонней коммуникационной модели, основанный на квантовом хешировании.

1. Для булевой функции f зафиксируем некоторое $\delta \in (0,1)$. По определению из [2] полином $g = g_1 + g_2$ является характеристическим полиномом функции f над $\mathbb{Z}_q$.

2. На основе входного набора $\sigma = \sigma_1 \dots \sigma_{n_1}$ вычислитель $\mathcal{A}$ строит квантовый хеш-код для значения $g_1(\sigma)$

$$|\psi_{q,B}(g_1(\sigma))\rangle = \frac{1}{\sqrt{|B|}} \sum_{i=1}^{|B|} |i\rangle \left(\cos \frac{2\pi b_i g_1(\sigma)}{q} |0\rangle + \sin \frac{2\pi b_i g_1(\sigma)}{q} |1\rangle \right) \qquad (3)$$

3. Хеш-код $|\psi_{q,B}(g_1(\sigma))\rangle$ передается вычислителю $\mathcal{B}$, который на основе своей части входных данных $\gamma = \gamma_1 \dots \gamma_{n_2}$ создает хеш-код для значения $-g_2(\gamma)$:

$$|\psi_{q,B}(-g_2(\gamma))\rangle = \frac{1}{\sqrt{|B|}} \sum_{i=1}^{|B|} |i\rangle \left(\cos \frac{2\pi b_i (-g_2(\gamma))}{q} |0\rangle + \sin \frac{2\pi b_i (-g_2(\gamma))}{q} |1\rangle \right) \qquad (4)$$

и выполняет сравнение полученных хеш-кодов с помощью известной процедуры SWAP-test [3].

4. Если $f(\sigma,\gamma) = 1$, то по определению характеристического полинома $g(\sigma,\gamma) = g_1(\sigma) + g_2(\gamma) = 0$. Соответственно, $g_1(\sigma) = -g_2(\gamma)$, а их хеш-коды будут одинаковыми, т.е. $\langle \psi_{q,B}(g_1(\sigma)) | \psi_{q,B}(-g_2(\gamma)) \rangle = 1$, и вероятность правильного ответа будет равна 1. Если же $f(\sigma,\gamma) = 0$, то $|\langle \psi_{q,B}(g_1(\sigma)) | \psi_{q,B}(-g_2(\gamma)) \rangle| < \delta$, и вероятность неправильного ответа ограничена константой $1/2 + \delta^2/2$.

Таким образом, вычислитель $\mathcal{B}$ выдает результат 1 в качестве значения функции тогда и только тогда, когда в результате сравнения значения построенных хеш-кодов совпали. В случаях, когда $f(\sigma,\gamma) = 1$, данный протокол будет приводить к безошибочным результатам. Если же $f(\sigma,\gamma) = 0$, то, благодаря δ-устойчивости квантовой хеш-функции, вычислители $\mathcal{A}$ и $\mathcal{B}$ получат неправильный ответ с вероятностью, не превосходящей $1/2 + \delta^2/2$.

Сложность такого коммуникационного протокола равна $\log|B| + 1 = O(\log\log q + \log(1/\delta))$ при выборе множества B согласно теореме 1. □

Отметим, что единственным существенным ограничением при построении такого квантового коммуникационного протокола для булевых функций является наличие характеристического полинома, допускающего разложение на сумму двух полиномов, зависящих от непересекающихся наборов переменных. Простейшим вариантом таких характеристических полиномов являются линейные полиномы, а примерами таких функций могут служить: функция EQ_n – проверка равенства двух двоичных наборов длины n и $Palindrome_n$ – проверка симметричности двоичного набора, т.е. является ли входное слово палиндромом.

Отметим также, что можно уменьшить вероятность ошибки до δ, несколько модифицировав вычисления на стороне $\mathcal{B}$. А именно, вычислитель $\mathcal{B}$ будет строить свой хеш-код для $g_2(\gamma)$, "дописывая" его непосредственно к хеш-коду $|\psi_{q,B}(g_1(\sigma))\rangle$, получая таким образом хеш для $g_1(\sigma)+g_2(\gamma)=g(\sigma,\gamma)$:

$$|\psi_{q,B}(\sigma,\gamma)\rangle \quad = \quad \frac{1}{\sqrt{t}}\sum_{i=1}^{t}|i\rangle\left(\cos\frac{2\pi b_i g(\sigma,\gamma)}{m}|0\rangle+\sin\frac{2\pi b_i g(\sigma,\gamma)}{m}|1\rangle\right). \qquad (5)$$

Далее вычислитель $\mathcal{B}$ применяет преобразование Адамара ко всем кубитам, кроме последнего, и измеряет полученное состояние относительно стандартного вычислительного базиса. $\mathcal{B}$ выдает ответ 1 (принимает входной набор), только если результатом измерения состояния всех кубитов окажется $|0\rangle$. Вероятность такого исхода равна 1 при $f(\sigma,\gamma)=1$, либо не превосходит δ при $f(\sigma,\gamma)=0$.

Кроме того, данный подход позволяет вычислять булевы функции $f:\{0,1\}^{n_1}\times\{0,1\}^{n_2}\to\{0,1\}$, для которых не существует указанного выше разложения характеристического полинома на $g_1(x_1,\ldots,x_{n_1})$ и $g_2(y_1,\ldots,y_{n_2})$. В этом случае можно рассмотреть следующее разложение:

$$g(x_1,\ldots,x_{n_1},y_1,\ldots,y_{n_2})=g_1(x_1,\ldots,x_{n_1})+g_2(x_{i_1},\ldots,x_{i_k},y_1,\ldots,y_{n_2}). \qquad (6)$$

Такое разложение существует всегда, т.к. при $k=n_1$ и $g_1\equiv 0$ можно положить $g_2\equiv g$.

Для таких функций протокол можно дополнить пересылкой от $\mathcal{A}$ к $\mathcal{B}$ дополнительных кубит со значениями $x_{i_1},\ldots,x_{i_k}$. Соответственно, сложность протокола становится $O(k+\log\log q)$. Во всех рассмотренных нами примерах $q=2^{n^{O(1)}}$ (более того, полином над $\mathbb{Z}_{2^n}$ существует для любой булевой функции от n переменных [2]), и поэтому при $k=O(\log n)$ описанный выше протокол будет иметь сложность $O(\log n)$.

Работа выполнена при поддержке РФФИ, проект № 14-07-00878-а.

Литература

[1] F M Ablayev and A V Vasiliev. Cryptographic quantum hashing. *Laser Physics Letters*, 11(2):025202, 2014.

[2] Farid Ablayev and Alexander Vasiliev. Algorithms for quantum branching programs based on fingerprinting. *Electronic Proceedings in Theoretical Computer Science*, 9:1–11, 2009.

[3] Harry Buhrman, Richard Cleve, John Watrous, and Ronald de Wolf. Quantum fingerprinting. *Phys. Rev. Lett.*, 87(16):167902, Sep 2001.

О нижних оценках ненадежности схем в базисе $\{0, 1, \overline{x}, x_1\&x_2,\ x_1\&x_2\&x_3\}$

А. В. Васин

alvarvasin@mail.ru

Пензенский государственный университет, Пенза

Рассматривается реализация булевых функций схемами (см., например, в [1]) из ненадежных функциональных элементов в произвольном полном конечном базисе B. Предполагаем, что все элементы схемы независимо друг от друга с вероятностью $\varepsilon \in (0, 1/2)$ подвержены инверсным неисправностям на выходах. Эти неисправности характеризуются тем, что в исправном состоянии функциональный элемент реализует приписанную ему булеву функцию ψ, а в неисправном – функцию $\overline{\psi}$. Считаем, что схема S из ненадежных элементов реализует булеву функцию $f(x_1, x_2, ..., x_n)$, если при поступлении на входы схемы двоичного набора $\mathbf{a} = (a_1, a_2, ..., a_n)$ при отсутствии неисправностей на выходе схемы S появляется значение $f(\mathbf{a})$.

Ненадежностью $P(S)$ схемы S назовем максимальную вероятность ошибки на выходе схемы S при всевозможных входных наборах схемы. *Надежность* схемы S равна $1 - P(S)$. Пусть $P_\varepsilon(f) = \inf P(S)$, где инфимум берется по всем схемам S из ненадежных элементов, реализующим булеву функцию $f(x_1, x_2, ..., x_n)$. Схема A из ненадежных элементов, реализующая функцию f, называется *асимптотически оптимальной (асимптотически наилучшей) по надежности*, если $P(A) \sim P_\varepsilon(f)$ при $\varepsilon \to 0$.

Число k, будем называть *коэффициентом ненадежности* полного базиса, если все функции в этом базисе можно реализовать схемами с ненадежностью, асимптотически не больше $k\varepsilon$ (при $\varepsilon \to 0$), и найдется функция f, которую нельзя реализовать схемой с ненадежностью, асимптотически меньше чем $k\varepsilon$ (при $\varepsilon \to 0$).

В работе [2] обосновано, что $k \in \{1, 2, 3, 4, 5\}$. Эта статья посвящена полным базисам с коэффициентом ненадежности $k = 5$.

Пусть:

$$\Psi_1 = \{0, 1, \bar{x}_1\} \cup \bigcup_{k=2}^{\infty} \{\mathop{\&}_{i=1}^{k} x_i\},$$

$$\Psi_2 = \{0, 1\} \cup \bigcup_{k=2}^{\infty} \{\mathop{\&}_{i=1}^{k} x_i\} \cup \bigcup_{k=1}^{\infty} \bigcup_{j=1}^{k} \{\bar{x}_j \cdot \mathop{\&}_{i=1, i\neq j}^{k} x_i\},$$

$$\Psi_1^* = \{0, 1, \bar{x}_1\} \cup \bigcup_{k=2}^{\infty} \{\bigvee_{i=1}^{k} x_i\},$$

$$\Psi_2^* = \{0, 1\} \cup \bigcup_{k=2}^{\infty} \{\bigvee_{i=1}^{k} x_i\} \cup \bigcup_{k=1}^{\infty} \bigcup_{j=1}^{k} \{\bar{x}_j \vee \bigvee_{i=1, i\neq j}^{k} x_i\}.$$

С.И. Аксенов [3] сформулировал следующую теорему:

Теорема 1 ([3]). *Пусть B – полный базис и $B \not\subseteq \Psi_1, B \not\subseteq \Psi_2, B \not\subseteq \Psi_1^*, B \not\subseteq \Psi_2^*$. Тогда любую булеву функцию f можно реализовать схемой S над B с ненадежностью $P(S) \leqslant 4\varepsilon + c\varepsilon^2$ при $\varepsilon \in (0, \varepsilon_0]$, где константы $c > 0$, $\varepsilon_0 \in (0, 1/2)$ зависят от базиса.*

Из теоремы 1 следует: если полный базис B удовлетворяет условиям $B \not\subseteq \Psi_1, B \not\subseteq \Psi_2, B \not\subseteq \Psi_1^*, B \not\subseteq \Psi_2^*$, то его коэффициент ненадежности $k \in \{1, 2, 3, 4\}$.

Нижние оценки ненадежности в полных базисах $B \subseteq \Psi_1$, или $B \subseteq \Psi_2$, или $B \subseteq \Psi_1^*$, или $B \subseteq \Psi_2^*$ пока не доказаны. В этой работе рассмотрен базис $B_4 = \{0, 1, x1, \bar{x}_1, x_1\&x_2, x_1\&x_2\&x_3, x_1\&x_2\&x_3\&x_4\}$.

Теорема 1 является верхней оценкой ненадежности, которая не дает представления о коэффициенте ненадежности таких базисов B_4.

С.И. Аксеновым [4] получена верхняя оценка ненадежности схем в произвольном полном конечном базисе при инверсных неисправностях на выходах элементов. Он доказал, что существуют такие константы $\varepsilon_0 \in (0, 1/2)$ и $d > 0$, зависящие от базиса,что любую булеву функцию f можно реализовать такой схемой S, что $P(S) \leqslant 5\varepsilon + d\varepsilon^2$ при $\varepsilon \in (0; \varepsilon_0]$.

В работе [5] явно найдены константы d, ε_0 и доказана теорема 2.

Теорема 2 ([5]). *В произвольном полном конечном базисе B любую булеву функцию f можно реализовать схемой A с ненадежностью $P(A) \leqslant 5\varepsilon + 182\varepsilon^2$ при $\varepsilon \leqslant 1/960$.*

Теорема 2 справедлива и для базисов B_4.

Автором в [2] решена задача построения асимптотически оптимальных по надежности схем при инверсных неисправностях на выходах элементов в полных базисах из трехвходовых элементов. В [2] доказаны нижние оценки ненадежности для базисов $B \subset \{0, 1, \bar{x}_1, x_1\&x_2, x_1\&x_2\&x_3\}$ и показано, что коэффициент ненадежности указанного базиса равен 5. Не трудно видеть, что этот базис являются подмножествами множеств $B_4 \subset \Psi_1$.

Поэтому можно предположить, что для базиса B_4 коэффициент ненадежности базиса также равен 5. Для проверки справедливости последней гипотезы необходимо доказать нижние оценки ненадежности для этого базиса.

Обозначим $K(n)$ – множество булевых функций f, зависящих от переменных $x_1, x_2, \ldots, x_n$, не представимых в виде $(x_i^a \& g(\tilde{x}))^b$ $(i = 1, 2, ..., n,\ a, b \in \{0, 1\})$ и сформулируем теорему о нижних оценках ненадежности для базиса B_4.

Тогда справедлива следующая теорема:

Теорема 3. *Пусть $B_4 = \{0, 1, x1, \bar{x}_1, x_1\&x_2, x_1\&x_2\&x_3, x_1\&x_2\&x_3\&x_4\}$ – полный конечный базис. Пусть функция $f(\tilde{x}) \in K(n)$, и S – любая схема, реализующая функцию f. Тогда $P(S) \geqslant 5\varepsilon(1-\varepsilon)^4$ при $\varepsilon \in (0, 1/960]$.*

Из теоремы 3 следует, что:

Следствие 1. *Пусть $B \subset B_4$ – полный конечный базис. Пусть функция $f(\tilde{x}) \in K(n)$, и S – любая схема, реализующая функцию f. Тогда $P(S) \geqslant 5\varepsilon(1-\varepsilon)^4$ при $\varepsilon \in (0, 1/960]$.*

Введем обозначение $B_4^* = \{0, 1, x1, \bar{x}_1, x_1 \vee x_2, x_1 \vee x_2 \vee x_3, x_1 \vee x_2 \vee x_3 \vee x_4\}$. Так как, ненадежность схем в двойственных базисах совпадает, то из следствия 1 следует:

Следствие 2. *Пусть $B \subset B_4^*$ – полный конечный базис. Пусть функция $f(\tilde{x}) \in K(n)$, и S – любая схема, реализующая функцию f. Тогда $P(S) \geqslant 5\varepsilon(1-\varepsilon)^4$ при $\varepsilon \in (0, 1/960]$.*

Из теоремы 2 и следствий 1 и 2 следует, что в любом из полных базисов B таких, что $B \subset B_4$ или $B \subset B_4^*$ для почти всех функций асимптотически оптимальные по надежности схемы функционируют с ненадежностью, асимптотически равной 5ε при $\varepsilon \to 0$. Следовательно верна следующая теорема:

Теорема 4. *Пусть B – полный базис, $B \subset B_4$ и $B \subset B_4^*$. Тогда коэффициент базиса B равен 5.*

Работа поддержана грантом РФФИ, проект № 14-01-31360 и № 14-01-00273.

Литература

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

[2] *Васин А. В.* Асимптотически оптимальные по надежности схемы в полных базисах из трехвходовых элементов // Дисс. ... канд. физико-математических наук. Пенза, 2010. – 100 с.

[3] *Аксенов С. И.* О надежности схем в широком классе полных базисов // Материалы IX Международного семинара "Дискретная математика и ее приложения посвященного 75-летию со дня рождения академика О.Б. Лупанова (Москва, МГУ, 18-23 июня 2007г.) / Под редакцией О.М. Касим-Заде. – М: Изд-во механико-математического факультета МГУ, 2007. С. 55 – 56.

[4] *Аксенов С. И.* О надежности схем над произвольной полной системой функций при инверсных неисправностях на выходах элементов // Известия высших учебных заведений. Поволжский регион. Естественные науки. № 6(21) 2005. С. 42–55.

[5] *Алехина М. А., Васин А. В.* О надежности схем в базисах, содержащих функции не более чем трех переменных // «Ученые записки Казанского государственного университета. Серия Физико-математические науки». Изд-во Казанского университета, 2009, Т. 151, кн. 2, С. 25–35.

Асимптотическое перечисление помеченных эйлеровых кактусов

В. А. Воблый, А К. Мелешко

vitvobl@yandex.ru, konstantin_meleshko@rambler.ru

МГТУ им. Н.Э. Баумана, Москва

Кактусом называется связный граф, в котором нет ребер, лежащих более чем на одном простом цикле [1, с. 93]. Все блоки кактуса – ребра или простые циклы (многоугольники). Форд и Уленбек перечислили помеченные кактусы с заданным распределением числа вершин по многоугольникам [2]. В работе [3] перечислены помеченные эйлеровы кактусы с заданным числом вершин.

Теорема 1. *Для числа D_n помеченных эйлеровых кактусов с n вершинами при $n \to \infty$ верна асимптотическая формула*

$$D_n \sim cn^{-5/2}a^n n!\,,$$

где $c \approx 0.1079436709$, $a \approx 2.5424753735$.

Доказательство. В работе [3] получена формула

$$D_n = \frac{(n-1)!}{n}[z^{-1}] \exp\left(\frac{nz^2}{2(1-z)}\right) z^{-n},$$

где $[z^{-1}]$ – оператор формального вычета.

Используем теорему Флажоле и Седжвика [4, Теорема VIII.8]:

Обозначим $F(N,n) = [z^N]\{a(z)(b(z))^n\} = \frac{1}{2\pi i} \oint a(z)(b(z))^n \frac{dz}{z^{N+1}}$
Пусть функции $a(z)$ *и* $b(z)$ *удовлетворяют следующим условиям.*
1. Функции $a(z) = \sum\limits_{j \geqslant 0} a_j z^j$ *и* $b(z) = \sum\limits_{j \geqslant 0} b_j z^j$ *аналитические в точке* $z = 0$ *и имеют неотрицательные коэффициенты, кроме того* $b(0) \neq 0$.
2. $НОД\{j \mid b_j > 0\} = 1$.
3. Если $R \leqslant \infty$ *радиус сходимости* $b(z)$, *то радиус сходимости* $a(z)$ *не меньше* R.
Через T *обозначим величину* $T = \lim\limits_{x \to R-0} \frac{xb'(x)}{b(x)}$. *Пусть* λ *положительное число такое, что* $0 < \lambda < T$, *и пусть* r – *единственный положительный корень уравнения* $r\frac{b'(r)}{b(r)} = \lambda$. *Обозначим* $\sigma = \frac{d^2}{dr^2}(\ln b(r) - \lambda \ln r)$.

Тогда для $N = \lambda n$ *целого при* $n \to \infty$ *и* $N \to \infty$ *верно асимптотическое равенство*

$$F(N,n) \sim a(r) \frac{(b(r))^n}{r^{N+1}\sqrt{2\pi n\sigma}} .$$

Формула для D_n может быть представлена в виде

$$D_n = \frac{(n-1)!}{n}[z^n]\Big\{z\Big(\exp\Big(\frac{z^2}{2(1-z)}\Big)^n\Big\} = \frac{(n-1)!}{n}F(N,n),$$

где $N = n,\ \lambda = 1,\ a(z) = z,\ b(z) = \exp\Big(\frac{z^2}{2(1-z)}\Big)$.

Так как ряд для $\bar{B}(z)$ сходится при $|z| < 1$, оператор формального вычета является контурным интегралом.
Очевидно, функции $a(z)$ и $b(z)$ аналитические в точке $z = 0$ и $b(0) = 1$. Функция $b(z)$ имеет положительные коэффициенты, так как $b(z) = \exp(\bar{B}(z))$ и $\bar{B}(z)$ – производящая функция для числа помеченных блоков частного вида. Поскольку $b_2 > 0$, $b_3 > 0$, имеем $НОД\{j \mid b_j > 0\} = 1$. Так как $z = 1$ – ближайшая к началу координат особая точка $b(z)$, радиус сходимости R функции $b(z)$ равен 1. Очевидно, $a(z)$ имеет бесконечный радиус сходимости. Таким образом, условия 1-3 теоремы Флажоле-Седжвика выполнены.

Найдем $T = \lim\limits_{x \to 1-0} \frac{xb'(x)}{b(x)} = \lim\limits_{x \to 1-0} \frac{x(2x - x^2)}{2(1-x)^2} = +\infty$,

$0 < \lambda < T$. В нашем случае уравнение $r\frac{b'(r)}{b(r)} = \lambda$ имеет вид

$r\frac{2r - r^2}{2(1-r)^2} = 1$. Решая это уравнение с помощью Maple, видим, что его единственным положительным корнем является число $r \approx 0.5391888728$. Вычисляя

величину,

$$\sigma = \Big(\frac{b'(r)}{b(r)}\Big)' + \frac{\lambda}{r^2} = \Big(\frac{r(2r-r^2)}{2(1-r)^2}\Big)' + \frac{1}{r^2} = \frac{2-2r}{2(1-r)^2} + \frac{2r-r^2}{(1-r)^3} + \frac{1}{r^2}\ .$$

получим $\sigma \approx 13.6592157423$. Также с помощью Maple вычислим

$$c = \frac{a(r)}{r\sqrt{2\pi\sigma}} = \frac{1}{\sqrt{2\pi\sigma}} \approx 0.1079436709\ ,\ a = \frac{b(r)}{r} \approx 2.5424753735\ .$$

Окончательно при $n \to \infty$ имеем асимптотику

$$Ca_n = \frac{(n-1)!}{n} F(N,n) \sim \frac{(n-1)!}{n} \frac{1}{\sqrt{2\pi\sigma}} n^{-1/2} \left(\frac{b(r)}{r}\right)^n \sim n! c n^{-5/2} a^n\ .$$

Литература

[1] *Харари Ф., Палмер Э.* Перечисление графов.— М.: Мир, 1977. — 326 с.

[2] *Ford G.W., Uhlenbeck G.E.* Combinatorial problems in theory graphs/ III // Proc. Nat. Acad. Sci. U.S.A. — 1956. — v. 42, — p. 529–535.

[3] *Воблый В. А.* Перечисление помеченных эйлеровых кактусов. // XI Международ. семинар "Дискретная математика и ее приложения". — М.: Изд. МГУ, 2012. — С. 275–277.

[4] *Flajolet Ph. Sedgewick R.* Analytic combinatorics.— Cambridge University Press, 2009. — 826 p.

Перечисление помеченных тетрациклических эйлеровых блоков

В. А. Воблый, А. К. Мелешко

vitvobl@yandex.ru, konstantin_meleshko@rambler.ru

МГТУ им. Н.Э. Баумана, Москва

Граф называется четным, если каждая его вершина имеет четную степень. Эйлеров граф – это связный четный граф. Рид [1] перечислил помеченные четные и эйлеровы графы. Он получил выражение для производящей функции таких графов с заданными числами вершин и ребер. Тазава [2] перечислил помеченные эйлеровы блоки с заданным числом вершин, им найдено нелинейное функциональное уравнение для соответствущей производящей функции. Автор [3] получил точные и асимптотические формулы для числа помеченных бициклических и трициклических эйлеровых графов.

В данной работе получена явная формула для числа помеченных эйлеровых тетрациклических блоков с заданным числом вершин и найдена асимптотика для числа таких графов с большим числом вершин. В дальнейшем, тетрациклический граф – это связный граф с цикломатическим числом равным 4.

Теорема 1. *Пусть B_n - число помеченных тетрациклических эйлеровых блоков с n вершинами, тогда при $n \geqslant 6$ верна формула*

$$B_n = \frac{n!}{5760}(n-2)(n-4)(n-5)(n^2+11n+18)\ .$$

Доказательство. Из 17 гомеоморфных типов тетрациклических блоков только один – эйлеров [4]. Он имеет вид треугольника с двойными ребрами.

Пусть H – гомеоморфный тип с a вершинами, b ребрами, b_0 петлями, b_i – число пучков ребер кратности i , $A(H)$ – порядок группы автоморфизмов графа H. Тогда число помеченных графов с n вершинами и гомеоморфным типом H равно [6, лемма 2]:

$$\frac{n!}{2^{b_0}A(H)}\mathrm{Coef}_{x^{n-a}}\frac{x^{b+b_0-\sum_{i=1}^{b}b_i}\prod_{i=1}^{b}(x+i(1-x))^{b_i}}{(1-x)^b}\ .$$

Так как в нашем случае $a=3, b=6, b_0=b_1=b_3=b_4=b_5=b_6=0$, $b_2=3, A(H)=48$, имеем

$$B_n = \frac{n!}{48}\mathrm{Coef}_{x^{n-3}}\frac{x^3(x+2(1-x)^3)}{(1-x)^6}\ .$$

С помощью известного разложения [5, с. 709]

$$(1-w)^{-m-1} = \sum_{n=0}^{\infty}\binom{m+n}{m}w^n$$

получим

$$B_n = \frac{n!}{48}\mathrm{Coef}_{x^{n-3}}\Big(\frac{x^3}{(1-x)^3}+3\frac{x^3}{(1-x)^4}+3\frac{x^3}{(1-x)^5}+\frac{x^3}{(1-x)^6}\Big) =$$

$$\frac{n!}{48}\mathrm{Coef}_{x^{n-3}}\Big(\sum_{k=0}^{\infty}\binom{k+2}{2}x^{k+3}+3\sum_{k=0}^{\infty}\binom{k+3}{3}x^{k+3}+3\sum_{k=0}^{\infty}\binom{k+4}{4}x^{k+3}+$$

$$\sum_{k=0}^{\infty}\binom{k+5}{5}x^{k+3}\Big) = \frac{n!}{48}\Big(\binom{n-4}{2}+3\binom{n-3}{3}+3\binom{n-2}{4}+\binom{n-1}{5}\Big) =$$

$$= \frac{n!}{5760}(n-2)(n-4)(n-5)(n^2+11n+18)\ .$$

Доказательство закончено. ∎

Из теоремы очевидным образом вытекает следствие.

Следствие. При $n \to \infty$ верно асимптотическое равенство

$$B_n \sim \frac{n^5}{5760}n!\ .$$

Литература

[1] *Read R.C.* Euler graphs on labelled nodes // Canad. J. Math. Название журнала. — 1962. — v. 14, — p. 482–486.

[2] *Tazawa S.* Enumeration of labelled 2-connected Euler graphs // J. Combinatorics, Information and System Sciences. — 1998. — V. 23, № 1-4. — P. 407–414.

[3] *Воблый В. А.* Перечисление помеченных бициклических и трициклических эйлеровых графов // Матем. заметки. — 2012. — Т. 92, № 5. — С. 678–683.

[4] *Неар B. R.* Enumeration homeomorphically irreducible star graphs // J. Math. Phys. 7(1966), No. 7, 1582-1587.

[5] *Прудников А. П. и др.* Интегралы и ряды. т. 1 — М.: Наука, ГРФМЛ, 1981. — 800 с.

[6] *Степанов В. Е.* О некоторых особенностях строения случайного графа вблизи критической точки // Теория вероятн. и ее примен.. — 1987. — Т. 32, № 4. — С. 633–657.

Правильные скобочные автоматы

А. А. Вылиток, М. А. Зубова

vylitok@cs.msu.su, nai_999801@mail.ru

Московский государственный университет, Москва, Тольяттинский государственный университет, Тольятти

Ранее в [1] авторами был предложен новый формализм для описания КС-языков. С одной стороны, он является расширением класса недетерминированных конечных автоматов (НКА). С другой, его можно рассматривать как НКА над специальным алфавитом, и, следовательно, применять различные алгоритмы эквивалентного преобразования НКА (построение минимальных автоматов, универсального автомата и др.) [2]– [3], получая более приемлемые по некоторым характеристикам (меньшее число вершин, дуг и др.) объекты в предлагаемом формализме.

В этой работе введены понятия правильных скобочных автоматов и правильных магазинных автоматов и доказана теорема о их связи.

Пусть

$$K = (\, Q, \Sigma, \gamma, S, F \,) \tag{1}$$

недетерминированный конечный автомат Рабина–Скотта, определяющий язык, обозначаемый $\mathcal{L}(K)$, Q — множество состояний, S и F — подмножества множества Q, называемые множеством стартовых и множеством финальных состояний соответственно. Функция переходов γ для автомата (1) будет иметь вид $\gamma : Q \times Q \to \mathcal{P}(\Sigma \cup \{\varepsilon\})$, где $\mathcal{P}(\Sigma \cup \{\varepsilon\})$ — множество всех подмножеств множества $\Sigma \cup \{\varepsilon\}$.

Специальные автоматы для КС-языков

Для каждого n из множества $\mathbb{N}_0$, рассмотрим множества $\mathbb{N}_{(n)} = \{\, 1, 2, \ldots, n{-}1, n \,\}$ и $\mathbb{Z}_{(n)} = \{\, -n, -(n{-}1), \ldots, -2, -1, 0, 1, 2, \ldots, n{-}1, n \,\}$. (Каждый элемент i из множества $\mathbb{N}_{(n)}$ символизирует i-ю пару скобок. Также i символизирует i-ю открывающую скобку, а $-i$ — соответствующую закрывающую скобку.) Иногда мы будем считать множество $\mathbb{Z}_{(n)}$ *алфавитом*, содержащим $2n{+}1$ символов, и, следовательно, рассматривать слова и языки над $\mathbb{Z}_{(n)}$.

Определение 1. Для заданного $n \geqslant 0$ определим язык $[\mathbb{Z}_{(n)}{}^*]$. Будем называть его языком, согласованным по скобкам (а каждое его слово — словом, согласованным по скобкам).

Рекурсивно определим слово, согласованное по скобкам:

— ε и 0 — согласованные по скобкам слова;
— если w и v — согласованные слова, то $u = vw$ — тоже согласованное по скобкам слово;
— если w— согласованное слово, $i \in \mathbb{N}_{(n)}$, то $u = iw{-}i$ — тоже согласованное по скобкам слово;
— никакое другое слово не является согласованным по скобкам.

Определение 2. Определим скобочный автомат B следующим образом:

$$B = (\, Q, \Sigma, \text{ч}, S, F, n\,), \tag{2}$$

где Q – множество состояний, Σ – заданный алфавит, S и F – множества стартовых и финальных состояний соответственно, n из $\mathbb{N}_0$ определяет множество скобок $\mathbb{Z}_{(n)}$, ч – функция переходов вида $\text{ч} : Q \times Q \to \mathcal{P}\left((\Sigma \cup \{\varepsilon\}) \times \mathbb{Z}_{(n)}\right)$. Будем считать, что мы одновременно определяем функции ч_γ и ч_ζ вида

$$\text{ч}_\gamma : Q \times Q \to \mathcal{P}\left(\Sigma \cup \{\varepsilon\}\right)\ ,\ \text{ч}_\zeta : Q \times Q \to \mathcal{P}\left(\mathbb{Z}_{(n)}\right)\ .$$

При этом, если для состояний q', q'' из множества Q выполняется условие $\text{ч}(q', q'') \ni (a, i)$, то будем считать, что выполнены условия $\text{ч}_\gamma(q', q'') \ni a$ и $\text{ч}_\zeta(q', q'') \ni i$. Никаких иных значений функции ч_γ и ч_ζ не содержат.

Правильные скобочные автоматы

Определение 3. Для каждого скобочного автомата (2) определим скобочно-зеркальный автомат

$$(\, Q, \Sigma, \text{ч}^{(R)}, F, S, n\,), \text{где:}$$

— условие $\text{ч}^{(R)}_\gamma(q', q'') \ni a$, где $a \in \Sigma \cup \Sigma \cup \{\varepsilon\}$, выполнено тогда и только тогда, когда $\text{ч}_\gamma(q'', q') \ni a$;
— условие $\text{ч}^{(R)}_\zeta(q', q'') \ni i$, где $i \in \mathbb{Z}_{(n)}$, выполнено тогда и только тогда, когда $\text{ч}_\zeta(q'', q') \ni -i$.

Будем обозначать автомат, скобочно-зеркальный к заданному B, записью $B^{(R)}$.

Определение 4. Скобочный автомат (2) называется правильным, если:

— для любой последовательности состояний $q_0\,,\, q_1\,,\, \ldots,\, q_j\, \in Q$, такой что

$$q_0 \in S, \quad \text{ч}(q_k, q_{k+1}) \ni (a_k, i_k), \qquad \text{для } k \in \{0, \ldots, j{-}1\},$$

образующей согласованный префикс $v = i_1 i_2 i_3 \ldots i_j \in \mathbb{Z}_{(n)}{}^*$, существует путь

$$q_j\,,\, q_{j+1}\,,\, \ldots,\, q_m\, \in Q, \quad q_m \in F,$$
$$\text{ч}(q_k, q_{k+1}) \ni (a_k, i_k), \text{ для } k \in \{j, \ldots, m{-}1\},$$

который определяет слово $w = i_1 i_2 i_3 \ldots i_j i_{j+1} \ldots i_m$, причём $w \in [\mathbb{Z}_{(n)}{}^*]$.

— аналогичное условие выполняется для соответствующего скобочно-зеркального автомата.

Правильные магазинные автоматы

Определение 5. Для слова $v \in \mathbb{Z}^*_{(n)}$, где $v = a_1 a_2 \dots a_m$, обратным словом назовем $\bar{v} = b_m b_{m-1} \dots b_1$, где $b_k = -a_k$ для любого $k \in \{1, \dots, m\}$.

Очевидно, что если слово $v \in \mathbb{Z}^*_{(n)}$ согласовано по скобкам, то $\bar{v}$ также согласовано по скобкам.

Определение 6. Для магазинного[1] автомата $P = (Q, \Sigma, \Gamma, \delta, s, z_0, \{f\})$[2] определим зеркальный автомат

$$P^R = (Q, \Sigma, \Gamma, \delta^R, f, z_0, \{s\}), \text{ где:}$$

— условие $\delta^R(q', a, z) \ni (q'', z), a \in \Sigma, z \in \Gamma, q', q'' \in Q$ выполнено тогда и только тогда, когда $\delta(q'', a, z) \ni (q', z)$;
— условие $\delta^R(q', \varepsilon, z) \ni (q'', zz'), z, z' \in \Gamma, q', q'' \in Q$ выполнено тогда и только тогда, когда $\delta(q'', \varepsilon, z) \ni (q', z(-z'))$;
— условие $\delta^R(q', \varepsilon, zz') \ni (q'', z), z, z' \in \Gamma, q', q'' \in Q$ выполнено тогда и только тогда, когда $\delta(q'', \varepsilon, zz') \ni (q', z)$.

Определение 7. Магазинный автомат называется правильным, если для него и для его зеркального автомата выполняется условие:

— для любой конфигурации $(q', vw, \alpha) \in Q \times (\Sigma \times \mathbb{Z}_{(n)})^* \times \Gamma^*$, $vw \in (\Sigma \times \mathbb{Z}_{(n)})^*$, $\alpha \neq \varepsilon$ найдется такая конфигурация $(q'', w, \varepsilon) \in Q \times (\Sigma \times \mathbb{Z}_{(n)})^* \times \Gamma^*$, что $(q', vw, \alpha) \Rightarrow^+ (q'', w, \varepsilon)$.

Теорема 1. *Магазинный автомат, построенный на основе правильного скобочного автомата (такое построение описано в [1]), является правильным.*

Доказательство. Предположим противное — т.е. пусть в магазинном автомате, построенном на основе правильного скобочного автомата, найдется конфигурация (q', vw, α), для которой *не существует* конфигурации (q'', w, ε), такой что $(q', vw, \alpha) \Rightarrow^+ (q'', w, \varepsilon)$. Это значит, что после того, как построенный МП-автомат считает все символы из входной цепочки, его магазин не будет пуст. Из этого следует, что исходный скобочный автомат допускает слово с лишними открывающими скобками — что невозможно вследствие сделанного предположения (правильности скобочного автомата). Значит, наше предположение неверно, т.е. магазинный автомат, построенный на основе правильного скобочного автомата, также является правильным. ∎

Литература

[1] *Вылиток А. А., Зубова М. А., Мельников Б. Ф.* Об одном расширении класса конечных автоматов для задания контекстно-свободных языков // Вестник Московского университета, **15**. — 2013. — № 1. — С. 39–45.

[1] Про магазинные автомата можно узнать здесь [].

[2] Мы здесь требуем, чтобы магазинный автомат имел единственное финальное состояние, хотя это непринципиально.

[2] *Melnikov B.* A new algorithm of the state-minimization for the nondeterministic finite automata // The Korean J. of Comp. and Appl. Math. A new algorithm of the state-minimization for the nondeterministic finite automata // The Korean J. of Comp. and Appl. Math., **6**. — 1999. — № 2. — P. 277–290.

[3] *Melnikov B.* Once more on the edge-minimization of nondeterministic finite automata and the connected problems // Fundamenta Informaticae, **104**. — 2010. — № 3. — P. 267–283.

[4] *Ахо А., Ульман Дж.* Теория синтаксического анализа, перевода и компиляции. Т.1 — М.: Мир, 1978. — 612 с.

О сравнительных характеристиках моделей квантовых алгоритмов Гровера — алгоритма точного и алгоритма с ошибками

Б. Н. Габбасов

bgabbasov@gmail.com

Казанский (Приволжский) Федеральный Университет, Казань

Рассматривается алгоритм квантового поиска Гровера. Пусть $f(x)$ - булева функция от n переменных. Согласно условию алгоритма Гровера, функция f задана в виде черного ящика, или оракула, O. Тогда итерацию алгоритма гровера можно записать в виде оператора:

$$H^{\otimes n}(2\langle 0||0\rangle - I)H^{\otimes n}O = (2\langle\psi||\psi\rangle - I)O,$$

где H - преобразование Адамара, $\langle\varphi| = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}\langle x|$ - суперпозиция взятых с равными весами состояний [1, стр. 315].

Обозначим $T = \{x|f(x) = 1\}$ — множество входов x, где $f(x) = 1$, $F = \{x|f(x) = 0\}$ — множество входов x, где $f(x) = 0$, $N_T = |T|$ - количество входов, где функция f истинна, $N_F = |F|$ - количество входов, где функция f ложна, $N = N_T + N_F = 2^n$ - количество всех возможных входов.

Предположим, что имеющийся черный ящик, или оракул, $\widehat{O}$ не является идеальным квантовым вычислительным устройством и на некотором входе e вместо истинного результата может выдать, с некоторой вероятностью $p_{err} > 0$, ложный результат, то есть

$$\widehat{O}\langle x|\langle q| = \begin{cases} \langle x|\langle q|, & x \in F \\ \langle x|\langle q \oplus 1|, & x \in T \setminus \{e\} \\ \langle x|\langle q \oplus 1 \oplus \xi|, & x = e \end{cases} \qquad (1)$$

где ξ - булева случайная величина со значенями из $\{0, 1\}$.

Работа алгоритма Гровера может быть представлена геометрически на комплексной плоскости. Поставим в соответствие вектору $x\langle f| + y\langle t|$ точку на комплексной плоскости $x + iy$. Тогда итерация алгоритма Гровера может быть записана в виде умножения на число $e^{i\theta}$, где $\cos(\theta/2) = \sqrt{N_F/N}$.

Заметим, что оракул с ошибкой может быть представлен в виде последовательного выполнения оператора ошибки и идеального оракула:

$$\widehat{O} = O(I - \xi 2\langle e||e\rangle)$$

Рассмотрим оператор $I-2\langle e||e\rangle$. В терминах комплексных чисел его можно записать следующим образом:

$$g(h) = \mathrm{Re}(h) + i\left(\cos\lambda\,\mathrm{Im}(h) + \sin\lambda\sqrt{1-|h|^2}\right) \qquad (2)$$

где $\cos\lambda = (N_T - 2)/N_T$.

Теорема 1. *Пусть $\widehat{O}$ - оракул, определенный в (1). Тогда вероятность успешного завершения алгоритма Гровера при использовании оракула $\widehat{O}$ увеличится не более, чем на $(1-\cos\lambda)^2/4$, где $\cos\lambda = (N_T-2)/N_T$, по сравнению со случаем, когда используется идеальный оракул O, вычисляющий функцию $f(x)$ без ошибок.*

Лемма 2. *Существует множество $A(\gamma)$, зависящее от параметра γ, $0 \leqslant \gamma \leqslant \pi/2$, удовлетворяющее следующим условиям:*

1. *Точка $e^{i\gamma} \in A(\gamma)$, $0 \leqslant \gamma \leqslant \pi/2$*
2. *Если точка $h \in A(\gamma)$, то $e^{i\delta}h \in A(\gamma+\delta)$*
3. *Если точка $h \in A(\gamma)$, то $g(h) \in A(\gamma)$, где $g(h)$ определен как в (2)*

Литература

[1] *Нильсен М., Чанг И.* Квантовые вычисления и квантовая информация. — М.: Мир, 2006 г. — 824 с.

Вычислительные возможности квантовых и классических OBDD

А. Ф. Гайнутдинова

aida.ksu@gmail.com

Казанский федеральный университет, г. Казань

Определения и предварительные сведения

Сравнительный анализ сложности решения задач на классических моделях и их квантовых аналогах – актуальное направление исследований теоретической информатики. Ветвящиеся программы – известная модель для вычисления булевых функций [1]. Упорядоченная ветвящаяся диаграмма решений (OBDD) – это ветвящаяся программа, в которой на каждом вычислительном пути переменные считываются не более одного раза в одном и том же порядке.

Вероятностная OBDD (POBDD) над множеством переменных $X = \{x_1, \ldots, x_n\}$ ширины d определяется следующим образом.

$$P_n = (v_0, T, Accept).$$

Здесь v_0 – d-мерный стохастический вектор-столбец, $T = \{\langle j_i, A_i(0), A_i(1)\rangle\}_{i=1}^{n}$ – последовательность d-мерных стохастических преобразований, где $A_i(0)$, $A_i(1)$ – стохастические по столбцам $(d \times d)$-матрицы, $Accept \subseteq \{1, \ldots, d\}$ – множество принимающих вершин. На i-ом шаге $(i = 1, \ldots, n)$ программа P_n считывает значение входной переменной $x_{j_i} = \sigma_{j_i}$ и преобразует текущий вектор v_{i-1} в вектор $v_i = A_i(\sigma_{j_i})$. После считывания входного набора $\sigma = \sigma_1, \ldots, \sigma_n$ финальный вектор $v_n(\sigma) = (p_1, \ldots, p_d) = A(\sigma)v_0$, где $A(\sigma) = A(\sigma_n) \cdots A(\sigma_1)$. Программа P_n принимает входной набор с вероятностью

$$Pr_{accept}^{P_n}(\sigma) = \sum_{i \in Accept} p_i.$$

Детерминированная OBDD (DOBDD) может быть определена как частный случай POBDD, где начальный вектор распределения вероятностей и матрицы преобразований содержат только нули и единицы.

Для OBDD P_n длина всегда не превосходит n, поэтому естественной мерой сложности является ширина $Width(P)$.

Недетерминированная OBDD (NOBDD) позволяет на каждом шаге при считывании переменной переходить из текущей вершины в более чем одну последующую вершину, поэтому для входного набора σ могут существовать несколько вычислительных путей. NOBDD P_n принимает входной набор σ тогда и только тогда, когда существует вычислительный путь, соответствующий σ, завершающийся в принимающей вершине.

Перед определением квантовой OBDD, кратко приведем некоторые понятия квантовых вычислений (см., например [2]). Квантовая система (QS) с d устойчивыми состояниями может быть описана при помощи d-мерного комплекснозначного Гильбертова пространства $(\mathcal{H}^d)$. Чистое состояние QS (обозначается $|\psi\rangle$)– это элемент пространства $\mathcal{H}^d$, вектор с нормой 1 (унитарный вектор): $\sqrt{\langle\psi|\psi\rangle} = 1$. Унитарная эволюция – это изменение состояния QS за определенный период времени, которое описывается d-мерной унитарной матрицей U.

Квантовая OBDD (QOBDD) на множестве переменных $X = \{x_1, \ldots, x_n\}$, определенная на QS с d устойчивыми состояниями

$$Q_n = \big(|\psi_0\rangle, T, Accept\big).$$

Здесь $|\psi_0\rangle$ – начальный унитарный вектор, $T = \{\langle j_i, U_i(0), U_i(1)\rangle\}_{i=1}^{n}$ последовательность унитарных преобразований, где $U_i(0)$ or $U_i(1)$– $(d \times d)$-унитарные матрицы, $Accept \subset \{1, \ldots, d\}$ множество принимающих состояний. Процесс вычисления Q_n на входе $\sigma = \sigma_1, \ldots, \sigma_n$ аналогичен вероятностному случаю. Исключение составляет определение вероятности принятия входа. Пусть $|\psi(\sigma)\rangle = (z_1, \ldots, z_d)$ – финальный вектор распределения амплитуд состояний после считывания σ.

$$Pr_{accept}^{Q_n}(\sigma) = \sum_{i \in Accept} |z_i|^2.$$

OBDD называется *стабильной*, если преобразования, применяемые на каждом шаге, не зависят от номера шага.

Пусть P – вероятностная (квантовая) OBDD над множеством переменных $X = \{x_1, \ldots, x_n\}$. Будем говорить, что P вычисляет функцию $f(x_1, \ldots, x_n)$ с ограниченной ошибкой, если существует константа $\varepsilon \in (0, 1/2]$ такая, что $\Pr^P_{accept}(\sigma) \geqslant 1/2 + \varepsilon$, если $f(\sigma) = 1$ и $\Pr^P_{accept}(\sigma) \leqslant 1/2 - \varepsilon$, если $f(\sigma) = 0$. В случае, когда $\varepsilon = 1/2$, будем говорить, что P вычисляет f без ошибки.

В работе [3] рассматривается симметрическая булева функция MOD_p, принимающая значение 1 только на тех входных наборах, в которых число единиц кратно p (p – простое число), для которой показано:

Теорема 1. *[3] Функция MOD_p, вычислима с ограниченной ошибкой квантовой стабильной OBDD ширины $O(\log p)$.*

Теорема 2. *[3] Любая детерминированная OBDD, вычисляющая MOD_p, имеет ширину не менее p. Любая стабильная вероятностная OBDD, вычисляющая MOD_p с ограниченной ошибкой имеет ширину не менее p.*

Известно, что экспоненциальное премущество квантовых OBDD перед классическими – максимально возможное для всюду определенных функций:

Теорема 3. *[4]. Пусть функция f вычислима один раз читающей квантовой ветвящейся программой Q. Тогда $Width(Q) = \Omega(\log Width(P))$, где P – детерминированная OBDD минимальной ширины, вычисляющая f.*

Основные результаты

В данной работе рассматриваются OBDD, вычисляющие частичные функции. Показывается, что в этом случае экспоненциальный разрыв в сложности между квантовыми и классическими OBDD может быть усилен.

В работе [5] рассмотрено семейство унарных проблем отделимости (promise problems). Показано, что такое семейство может распознаваться без ошибки квантовым автоматом с 2 состояниями, в то время как детерминированный автомат требует не менее 2^{k+1} состояний.

На основе данного семейства определим частичную булеву функцию:

$$\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}(\sigma) = \begin{cases} 1 & , \text{ if } \#_1(\sigma) = 0 \pmod{2^{k+1}} \\ 0 & , \text{ if } \#_1(\sigma) = 2^k \pmod{2^{k+1}} \\ * & , \text{ в противном случае} \end{cases},$$

где $\sigma \in \{0,1\}^n$ и функция не определена на наборах, отображающихся в *.

Теорема 4. *Для любого $k \geqslant 0$ существует стабильная QOBDD ширины 2, вычисляющая частичную функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ без ошибки.*

Доказательство. Для построения стабильной QOBDD Q_n, вычисляющей функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ без ошибки, используются идеи работы [5].

$$Q_n = \big(|\psi_0\rangle, T, Accept\big),$$

где $|\psi_0\rangle = (1,0)$, $T = \{\langle i_j, U(0), U(1)\rangle\}_{j=1}^{l}$, $U(1) = \begin{pmatrix} \cos(\frac{\pi}{2*2^k}) & -\sin(\frac{\pi}{2*2^k}) \\ \sin(\frac{\pi}{2*2^k}) & \cos(\frac{\pi}{2*2^k}) \end{pmatrix}$, $U(0) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $Accept = \{1\}$. Если во входном наборе σ число еди-

ниц $\#_1(\sigma) = 0 \pmod{2^{k+1}}$, программа Q_n завершит работу в состоянии $|\psi_n(\sigma)\rangle = (\pm 1, 0)$ и примет набор σ с вероятностью 1. На входных наборах σ с числом единиц $\#_1(\sigma) = 2^k \pmod{2^{k+1}}$ финальное состояние $|\psi_n(\sigma)\rangle = (0, \pm 1)$. Вероятность принятия таких наборов программой Q_n равна 0. Следовательно, Q_n вычисляется функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ без ошибки. ∎

Теорема 5. *Пусть P_n – DOBDD, вычисляющая частичную функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$. Тогда ширина P_n не меньше 2^{k+1}.*

Теорема 6. *Пусть P_n – стабильная POBDD, вычисляющая частичную функцию $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ с ограниченной ошибкой. Тогда ширина P_n не меньше 2^{k+1}.*

Не сложно построить стабильную DOBDD ширины 2^{k+1}, вычисляющую $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$. Нижняя оценка $k+1$ ширины NOBDD для функции $\mathtt{PartialMOD}^{\mathtt{k}}_{\mathtt{n}}$ следует из известного соотношения между детерминированной и недетерминированной пространственной сложностью.

Работа выполнена при поддержке РФФИ, проект № 14-07-00878.

Литература

[1] *Wegener I.* Branching Programs and Binary Decision Diagrams. — SIAM, 2000.

[2] *Nielsen M. A., Chuang I. L.* Quantum Computation and Quantum Information. — Cambridge University Press, 2000.

[3] *Гайнутдинова А. Ф.* О сравнительной сложности квантовых и классических бинарных программ // Дискретная математика. — 2002. — Т. 14, № 3. — С. 109–121.

[4] *Ablayev F., Gainutdinova A., M. Karpinski M., Moore C., Pollette C.* On the computational power of probabilistic and quantum branching program // Information and Computation. — 2005. — Т. 203, № 2. — С. 145–162.

[5] *Ambainis A., Yakaryılmaz A.* Superiority of exact quantum automata for promise problems // Information Processing Letters. — 2012. — Т. 112, № 7. — P. 289–291.

Об одной задаче оптимального выбора пропускных способностей каналов транспортных сетей

М. В. Гостев, Р. Ф. Хабибуллин

mailtogmv@gmail.com, Rustem.Khabibullin@ksu.ru

Казанский (Приволжский) федеральный университет, Казань

Одной из важнейших проблем проектирования транспортных сетей, наряду с выбором топологической структуры сети и распределением транспортных потоков по сети, является проблема выбора пропускных способностей каналов транспортировки. Увеличение пропускных способностей каналов транспортной сети способствует уменьшению суммарного времени транспортировки по сети и повышению общей пропускной способности сети. При этом, чем меньше суммарное время транспортировки по всем каналам сети, тем быстрее окупаются затраты на ее создание. Однако увеличение пропускных способностей

каналов транспортировки приводит к увеличению стоимости всей сети. Задача состоит в выборе таких пропускных способностей каналов, что суммарное время транспортировки по всем каналам сети было бы минимальным, а суммарная стоимость таких каналов не превышала бы заданной величины.

Пусть топологическая структура транспортной сети задана в виде графа $G = (V, U)$, где множество вершин $V = (v_1, \ldots, v_n)$ графа представляет множество транспортных узлов, а множество ребер $U = (u_1, \ldots, u_m)$ — множество каналов транспортировки [1,2]. Каждое ребро графа транспортной сети представляется двумя дугами, соответствующими двум противоположным направлениям движения между двумя смежными узлами.

Для вычисления времени транспортировки по каналам транспортной сети используются различные модели (см., например, [3,4]). Наиболее используемой из них является модель Бюро общественных дорог [5]. Используя эту модель, и учитывая, что пропускные способности дуг каждого ребра (в обоих направлениях) должны быть одинаковыми, получаем следующее выражение для суммарного времени транспортировки по всем каналам сети:

$$T(c) = \sum_{k=1}^{m} \left[t'_{0k} \left(1 + \alpha \left(\frac{f'_k}{c_k} \right)^{\beta} \right) + t''_{0k} \left(1 + \alpha \left(\frac{f''_k}{c_k} \right)^{\beta} \right) \right],$$

где c_k — пропускная способность каждой дуги ребра $u_k \in U$, f'_k и f''_k — потоки, проходящие по дугам ребра $u_k \in U$ в его противоположных направлениях, t'_{0k} и t''_{0k} — время транспортировки без задержек и помех (так называемое время пути в свободном потоке [3]) по соответствующим дугам ребра $u_k \in U$, $k = 1, \ldots, m$.

Задача заключается в выборе таких значений пропускных способностей ребер $c = (c_1, \ldots, c_m)$, которые минимизируют функцию $T(c)$ и удовлетворяют условиям:

$$\sum_{k=1}^{m} (a_k c_k + b_k) \leqslant S, \quad c_k > \bar{f}_k, \quad k = 1, \ldots, m,$$

где S — ограничение на стоимость сети, $\bar{f}_k = max\{f'_k, f''_k\}$, $k = 1, \ldots, m$.

Считаем, не ограничивая общности, что все $\bar{f}_k > 0$, $k = 1, \ldots, m$. Определим $S_0 = \sum_{k=1}^{m} (a_k \bar{f}_k + b_k)$. Можно показать, что для $S > S_0$ решение задачи существует и единственно. С помощью правила множителей Лагранжа решение поставленной задачи оптимизации сводится к решению следующей системы уравнений:

$$\alpha\beta \, \frac{t'_{0k} {f'_k}^{\beta} + t''_{0k} {f''_k}^{\beta}}{{c_k}^{\beta+1}} - \lambda a_k = 0, \quad k = 1, \ldots, m, \tag{1}$$

$$\sum_{k=1}^{m} (a_k c_k + b_k) - S = 0, \tag{2}$$

где λ — множитель Лагранжа.

Очевидно, что каждое уравнение (1) при произвольном фиксированном значении $\lambda > 0$ имеет единственное решение. Обозначим через $\hat{c}_k(\lambda)$ корень

k-го уравнения при фиксированном значении $\lambda > 0$, $k = 1, \ldots, m$, и

$$\bar{c}_k(\lambda) = \sqrt[\beta+1]{\frac{\alpha\beta \bar{f}_k^{\beta}(t'_{0k} + t''_{0k})}{\lambda a_k}}, \quad k = 1, \ldots, m.$$

Справедливо следующее утверждение.

Утверждение 1. *Для произвольного $\lambda > 0$ выполняются неравенства*

$$\hat{c}_k(\lambda) \leqslant \bar{c}_k(\lambda), k = 1, \ldots, m.$$

Обозначим через λ^* оптимальное значение λ, и

$$\bar{\lambda} = \frac{\alpha\beta \sum_{k=1}^{m} a_k^{\beta+1} \bar{f}_k^{\beta}(t'_{0k} + t''_{0k})}{(\sum_{k=1}^{m} a_k \bar{f}_k - S_0 + S)^{\beta+1} \sum_{k=1}^{m} a_k}.$$

Утверждение 2. *Справедливо неравенство*

$$\lambda^* \leqslant \bar{\lambda}.$$

Таким образом, для любого $\lambda > 0$ корни $\hat{c}_k(\lambda)$ системы уравнений (1) находятся на соответствующих полуинтервалах $(\bar{f}_k, \bar{c}_k(\lambda)]$, а оптимальное значение $\lambda^* \in (0, \bar{\lambda}]$. Для решения системы уравнений (1)-(2), а, следовательно, и исходной задачи с любой заданной точностью, предложен эффективный численный метод, основанный на использовании полученных оценок и применении дихотомии.

Рассмотрим следующее выражение для выбора пропускных способностей каналов транспортировки:

$$c_k = \frac{S - S_0 + \sum_{k=1}^{m} a_k f_k}{\sum_{k=1}^{m} a_k} \sqrt[\beta+1]{\frac{\sum_{k=1}^{m} a_k {f_k}^{\beta} t_{0k}}{a_k {f_k}^{\beta} t_{0k}}}, \quad k = 1, \ldots, m. \tag{3}$$

Рассмотрим также три способа задания значений f_k в этом выражении:

A1) $f_k = max\{f'_k, f''_k\}$, $k = 1, \ldots, m$;

A2) $f_k = \frac{1}{2}(f'_k + f''_k)$, $k = 1, \ldots, m$;

A3) $f_k = \sqrt{f'_k f''_k}$, если $min\{f'_k, f''_k\} > 0$ для всех $k = 1, \ldots, m$.

Нетрудно убедиться, что получающиеся по формуле (3) значения пропускных способностей каналов $c = (c_1, \ldots, c_m)$ при каждом способе A1, A2 или A3 удовлетворяют всем ограничениям задачи. Отметим также, что в случае симметричных потоков на каналах, т.е. когда $f'_k = f''_k$, $k = 1, \ldots, m$, все три способа задания f_k в (3) дают одно и то же оптимальное решение задачи. Таким образом, в общем случае решения, полученные по формуле (3) с использованием A1, A2 или A3, можно использовать в качестве приближенных решений задачи.

Для оценки относительных погрешностей $\delta_1(S)$, $\delta_2(S)$, $\delta_3(S)$ соответствующих приближенных решений были проведены вычислительные эксперименты с тестовыми примерами. Случайным образом генерировалась топология транспортной сети и варианты распределения потоков f'_k, f''_k, $k = 1, \ldots, m$ по

каналам сети. Для каждого такого варианта при различных значениях $S > S_0$ находилось оптимальное решение с необходимой точностью с помощью разработанного численного метода и вычислялись приближенные решения по формуле (3) с использованием А1, А2 и А3.

Проведенные расчеты показали следующее. На всех тестовых примерах значения относительных погрешностей строго уменьшались с увеличением значения S, и для любого $S > S_0$ имело место $\delta_1(S) > \delta_2(S) > \delta_3(S)$, т.е. наилучшие результаты показало использование способа А3. При этом $\delta_2(S)$ незначительно, в пределах долей процента, больше, чем $\delta_3(S)$, в то время как $\delta_1(S)$ значительно больше. Наиболее существенное влияние на значения погрешностей оказывает величина S. При использовании способа А3, для $S > 1,1S_0$ погрешность не превышала пяти процентов. Для $S > 1,25S_0$ погрешность была уже меньше трех процентов. Для $S > 1,75S_0$ погрешность в подавляющем большинстве случаев была менее одного процента.

Таким образом, когда возможно использование приближенных решений, можно получить решение вычислением по простой формуле (3) с использованием А2 или А3, вместо численного решения задачи.

Литература

[1] *Форд Л. Р., Фалкерсон Д. Р.* Потоки в сетях. — М.: Мир, 1963. — 276 с.

[2] *Гольштейн Е. Г., Юдин Д. Б.* Задачи линейного программирования транспортного типа. — М.: Наука, 1969. — 382 с.

[3] *Bell M. G. H., Iida Y.* Transportation Network Analysis. — New York: Wiley, 1997. — 226 p.

[4] *Branston D.* Link capacity functions: a review // Transportation Research. — 1976. — V. 10, № 4. — P. 223–236.

[5] *Bureau of Public Roads.* Traffic Assignment Manual. — Washington, D.C.: U.S. Bureau of Public Roads, 1964.

О спектральных свойствах тонких языков

П. С. Дергач

dergachpes@mail.ru

Кафедра МаТИС, г. Москва

Введение

Работа является продолжением исследований автора в области теории кодирования регулярных текстов. Основной задачей этой теории является проблема декодирования при алфавитном кодировании (сокращенно ДПАК). Ее смысл в том, чтобы по произвольной паре, состоящей из регулярного языка $P \subseteq A^*$ и функции алфавитного кодирования $f : A- > B^*$, определить, является ли f однозначно декодируемой на P. Здесь A и B - произвольные фиксированные конечные алфавиты. О понятии регулярных языков и алфавитного кодирования можно прочитать в [1] и [2] соответственно. Данная проблема

была успешна разрешена Ал. А. Марковым в [3]. В свою очередь, в [4] автором было получено альтернативное решение проблемы ДПАК, сводящее ее к конечному перебору слов из P длины не более $L(R, f)$, где $L(R, f)$ - функция, явным образом зависящая от количества состояний задающего регулярный язык автомата и от длины схемы кодирования. Положительное решение проблемы ДПАК позволило поставить вопрос об эффективности функций алфавитного кодирования и ввести на них отношение строгого частичного порядка. А именно, говорим, что функция f_1 богаче, чем функция f_2, если класс регулярных языков, доставляющих положительное решение проблемы ДПАК для функции f_1, шире соответствующего класса для функции f_2. При помощи этого отношения на множестве функций алфавитного кодирования можно построить решетку. В данной работе изучается один из минимальных классов этой решетки. В дальнейшем он будет называться классом 1-тонких языков и обозначаться через T_1. Соответствующая T_1 функция кодирования имеет вид $\tilde{f}(a_i) = b$, где $i = 1, \ldots, |A|$ и $b \in B$. Выбор этого класса обусловлен, во-первых, тем, что $\tilde{f}$, в отличие от функций других минимальных классов, будет инвариантна относительно перестановки букв алфавита A. Это позволяет дать классу T_1 альтернативное определение, не использующее таких понятий, как кодирование и решетка. А именно, класс T_1 можно получить, если наложить на регулярные языки ограничение, запрещающее им содержать различные слова одинаковой длины. Во-вторых, алгоритм решения проблемы ДПАК, изложенный в [4], работает для T_1 за линейное от перебираемой длины время. Эти соображения позволяют сделать предположение о принципиальной роли класса 1-тонких языков в исследуемой модели. В работе описывается структура элементов класса T_1, а также приводится их каноническое универсальное представление в терминах регулярных выражений. О понятии регулярных выражений можно прочитать в [1]. Следует понимать, что T_1 является минимальным классом и использовать его для описания свойств регулярных языков общего вида затруднительно. Поэтому в работе делается обобщение класса T_1, для которого, с одной стороны, все еще можно быстро решать проблему ДПАК, и, с другой стороны, с помощью которого аппроксимируется уже произвольный регулярный язык. Это делается следующим образом. Для каждого натурального значения s рассматривается класс регулярных языков, в которых максимальное количество несовпадающих слов одинаковой длины меньше бесконечности и равно s. Такие языки называются s-тонкими а класс всех s-тонких языков обозначается через T_s. В качестве обобщения класса T_1 берется класс $\bigcup_{i=1}^{\infty} T_s$. Он называется классом тонких языков и обозначается через T. Для класса T также удается описать его регулярную структуру и построить универсальное представление его элементов.

Основные понятия

Пусть A - непустое конечное множество и пусть $P \subseteq A^*$ - произвольное множество слов в этом алфавите. *Спектром* этого множества называем множество $\{l(\alpha) | \alpha \in P\}$ и обозначаем его через $Sp(P)$. Здесь через $l(\alpha)$ обозначена длина слова α. Пусть $P_1, P_2 \subseteq A^*$. Будем говорить, что эти множества *спек-*

трально независимы, если их спектры не пересекаются. Пусть $P_1, \ldots, P_r \subseteq A^*$, $r \geqslant 1$. Будем говорить, что эти множества *спектрально независимы в совокупности*, если любые два из них спектрально независимы. В противном случае говорим, что эти множества *спектрально зависимы в совокупности*.

Пусть β - непустое слово в алфавите A. Если существует слово α в алфавите A такое, что $\beta = \alpha l^k$ для некоторого $k > 1$, то говорим, что β *измельчимо*. Иначе говорим, что β *неизмельчимо*.

Пусть $(\alpha, \beta, \gamma, k, m) \in (A^*)^3 \times \mathbb{N} \times (\mathbb{N} \cup \{0\})$. Говорим, что $(\alpha, \beta, \gamma, m, n)$ - *порождающий след*, если выполнено одно из двух условий:

1. $\beta = \gamma = \lambda$, $k = 1, m = 0$;

2. $\beta \neq \lambda$, у α и β нет одинаковых непустых окончаний, β неизмельчимо и не является началом γ.

Говорим, что множество $P \subseteq A^*$ является *прогрессивным*, если оно представимо с помощью регулярного выражения $\alpha \cdot (\beta^k)^* \cdot \beta^m \cdot \gamma$ для некоторого порождающего следа $(\alpha, \beta, \gamma, k, m)$. В этом случае говорим также, что множество P *имеет порождающий след* $(\alpha, \beta, \gamma, k, m)$. Упорядоченную тройку (α, β, γ) называем *основанием* множества P. Называем множество $P \subseteq A^*$ *общепрогрессивным*, если оно является конечным объединением прогрессивных множеств с одинаковым основанием.

Пусть A - непустой конечный алфавит и $s \in \mathbb{N}$. Введем понятие s-тонкого множества в алфавите A. Регулярное множество P, $P \subseteq A^*$, называем *s-тонким в алфавите* A, если

1) существуют s слов $\beta_1, \beta_2, \ldots, \beta_s \in P$ таких, что $l(\beta_1) = l(\beta_2) = \ldots = l(\beta_s)$ и для них не существуют $i, j \in \mathbb{N}$ такие, что $1 \leqslant i < j \leqslant s$ и $\beta_i = \beta_j$;

2) для любых $s+1$ слов $\alpha_1, \alpha_2, \ldots, \alpha_{s+1} \in P$ таких, что $l(\alpha_1) = l(\alpha_2) = \ldots = l(\alpha_{s+1})$ существуют $i, j \in \mathbb{N}$ такие, что $1 \leqslant i < j \leqslant s+1$ и $\alpha_i = \alpha_j$.

Другими словами, в P должно быть s несовпадающих слов одинаковой длины, но не должно быть $s+1$ несовпадающих слов одинаковой длины.

Для всех $s \in \mathbb{N}$ обозначаем через T_s множество всех s-тонких множеств в алфавите A. Через T обозначаем множество $\bigcup_{i=1}^{\infty} T_s$. Называем это множество *классом тонких множеств*, а его элементы - *тонкими множествами*.

Формулировка результатов

Теорема 1. *Любое конечное объединение спектрально независимых в совокупности общепрогрессивных множеств является 1-тонким множеством.*

Теорема 2. *Любое 1-тонкое множество представимо в виде конечного объединения спектрально независимых в совокупности общепрогрессивных множеств.*

Теорема 3. *Любое конечное объединение непересекающихся прогрессивных множеств является тонким множеством.*

Теорема 4. *Любое тонкое множество представимо в виде конечного непересекающегося объединения прогрессивных множеств.*

Литература

[1] *Кудрявцев В. Б., Алешин С. В, Подколзин А. С.* Введение в теорию автоматов. — М.: Наука, 1985.

[2] *Яблонский С. В.* Введение в дискретную математику. — М.: Наука, 1986.

[3] *Марков Ал. А.* Введение в теорию кодирования. — М.: Наука, 1982.

[4] *Дергач П. С.* Об однозначности алфавитного декодирования // Интеллектуальные системы. — 2011. — Т. 15, № 1-4. — С. 349–361.

О порождающих системах в классах монотонных функций многозначной логики

О. С. Дудакова

olga.dudakova@gmail.com

МГУ им. М. В. Ломоносова, Москва

Известно, что при $k \leqslant 7$ все предполные классы функций k-значной логики являются конечно-порожденными [1], а начиная с $k = 8$ существуют предполные классы монотонных функций, не имеющие конечного базиса [2]; полного описания конечно-порожденных предполных классов монотонных функций к настоящему времени не получено. В работах автора [3, 4, 5, 6] получен критерий конечной порожденности для предполных классов функций, монотонных относительно частично упорядоченных множеств ширины два, а также условия существования конечных порождающих систем для ряда других семейств классов монотонных функций. В данной работе продолжены исследования в этом направлении.

Пусть $\preccurlyeq$ — частичный порядок на множестве $E_k = \{1, 2, \ldots, k\}$. Положим $\mathcal{P} = (E_k, \preccurlyeq)$. Будем считать, что множество $\mathcal{P}$ имеет наименьший и наибольший элементы. Через $\mathcal{M}_\mathcal{P}$ будем обозначать класс всех монотонных функций над $\mathcal{P}$ (отметим, что класс $\mathcal{M}_\mathcal{P}$ является предполным [7]).

Функцию $\lambda(x_0, x_1, \ldots, x_k)$ будем называть *функцией выбора*, если для каждого набора $(i, a_1, \ldots, a_k) \in \mathcal{P}^{k+1}$ выполняется равенство

$$\lambda(i, a_1, \ldots, a_k) = a_i.$$

Легко видеть, что если замкнутый класс функций k-значной логики содержит все константы $1, 2, \ldots, k$ и функцию выбора, то он является конечно-порожденным. Отметим также, что если $\mathcal{P}$ — частично упорядоченное множество, содержащее хотя бы одну цепь длины 2, то $\lambda(x_0, x_1, \ldots, x_k) \notin \mathcal{M}_\mathcal{P}$.

Положим

$$\mathcal{P}_\lambda = \{(a, b_1, \ldots, b_k) \in \mathcal{P}^{k+1} \mid \text{ если } i \preccurlyeq j, \text{ то } b_i \preccurlyeq b_j\}.$$

Легко видеть, что функция λ монотонна на множестве $\mathcal{P}_\lambda$. Назовем *монотонной функцией выбора* функцию $\nu(x_0, x_1, \ldots, x_k)$ из $\mathcal{M}_\mathcal{P}$, совпадающую на множестве $\mathcal{P}_\lambda$ с функцией $\lambda(x_0, x_1, \ldots, x_k)$. Нетрудно показать, что если

класс $\mathcal{M}_{\mathcal{P}}$ содержит монотонную функцию выбора, то он является конечно-порожденным.

Пусть a_1 и a_2 — элементы множества $\mathcal{P}$, не сравнимые относительно частичного порядка $\preccurlyeq$. Элемент $b \in \mathcal{P}$ называется *верхней гранью* элементов a_1 и a_2, если выполняется неравенство $a_1, a_2 \preccurlyeq b$. Верхняя грань b элементов a_1 и a_2 называется *минимальной верхней гранью* этих элементов, если не существует такой верхней грани c элементов a_1 и a_2, что $c \neq b$ и $c \preccurlyeq b$. Верхняя грань b элементов a_1 и a_2 называется *точной верхней гранью* этих элементов ($\sup(a_1, a_2)$), если для любой верхней грани c элементов a_1 и a_2 выполняется неравенство $b \preccurlyeq c$ Аналогичным образом определяется *нижняя, максимальная нижняя* и *точная нижняя грань* элементов a_1 и a_2 (точная нижняя грань обозначается через $\inf(a_1, a_2)$). Через $|\mathcal{P}|$ будем обозначать число элементов множества $\mathcal{P}$. Положим $w_{\mathcal{P}} = \max|J|$, где максимум берется по всем антицепям J множества $\mathcal{P}$; величину $w_{\mathcal{P}}$ будем называть *шириной* множества $\mathcal{P}$. Положим $h_{\mathcal{P}} = \max|I|$, где максимум берется по всем цепям I множества $\mathcal{P}$; величину $h_{\mathcal{P}}$ будем называть *высотой* множества $\mathcal{P}$.

Рассмотрим два семейства частично упорядоченных множеств. Семейство $\mathbb{S}_1$ состоит из всех множеств $\mathcal{P}$, таких, что для любых двух элементов $a, b \in \mathcal{P}$ в $\mathcal{P}$ существует по крайней мере один из элементов $\sup(a, b)$ и $\inf(a, b)$. Семейство $\mathbb{S}_2$ состоит из всех множеств $\mathcal{P}$, для которых выполняется следующее условие: для любой пары несравнимых элементов a_1 и a_2, не имеющих в $\mathcal{P}$ точной верхней грани, и для любой верхней грани c элементов a_1 и a_2, не сравнимой с некоторой минимальной верхней гранью b этих элементов, в $\mathcal{P}$ существует $\sup(b, c)$. Отметим, что в классе множеств ширины два выполняется равенство $\mathbb{S}_1 = \mathbb{S}_2$, в классе множеств высоты не более 5 также выполняется равенство $\mathbb{S}_1 = \mathbb{S}_2$, а в общем случае имеет место включение $\mathbb{S}_1 \subset \mathbb{S}_2$.

В работе [8] был получен следующий результат.

Теорема 1. *Пусть $\mathcal{P}$ — частично упорядоченное множество ширины два. Класс $\mathcal{M}_{\mathcal{P}}$ содержит монотонную функцию выбора тогда и только тогда, когда $\mathcal{P} \in \mathbb{S}_1$.*

В работе [9] получено частичное обобщение необходимого условия теоремы 1 на случай множеств произвольной ширины:

Теорема 2. *Пусть $\mathcal{P}$ — произвольное частично упорядоченное множество. Если класс $\mathcal{M}_{\mathcal{P}}$ содержит монотонную функцию выбора, то $\mathcal{P} \in \mathbb{S}_2$.*

Основным результатом настоящей работы является обобщение достаточного условия теоремы 1 на случай множеств высоты 5 (произвольной ширины):

Теорема 3. *Пусть $\mathcal{P}$ — произвольное частично упорядоченное множество, $h(\mathcal{P}) \leqslant 5$. Если $\mathcal{P} \in \mathbb{S}_1$, то в классе $\mathcal{M}_{\mathcal{P}}$ содержится монотонная функция выбора.*

Следствие 1. *Если $\mathcal{P}$ — частично упорядоченное множество, такое, что $h(\mathcal{P}) \leqslant 5$ и $\mathcal{P} \in \mathbb{S}_1$, то класс $\mathcal{M}_{\mathcal{P}}$ является конечно-порожденным.*

Работа выполнена при поддержке РФФИ, проект № 14-01-00598.

Литература

[1] *Lau D.* Bestimmung der Ordnung maximaler Klassen von Funktionen der k-wertigen Logik // Z. math Log. und Grundl. Math. — 1978. — 24. — S. 79–96.

[2] *Tardos G.* A not finitely generated maximal clone of monotone operations // Order. — 1986. — 3. — P. 211–218.

[3] *Дудакова О. С.* О классах функций k-значной логики, монотонных относительно множеств ширины два // Вестн. Моск. ун-та. Серия 1. Математика. Механика. — 2008. — № 1. — С. 31–37.

[4] *Дудакова О. С.* О конечной порожденности замкнутых классов монотонных функций в P_k // Учен. зап. Казан. ун-та. Серия Физ.-матем. науки. — 2009. — Т. 151, кн. 2. — С. 65–71.

[5] *Дудакова О. С.* О конечной порожденности предполных классов монотонных функций девятизначной логики // Мат-лы XVIII Междунар. школы-семинара "Синтез и сложность управляющих систем" (Пенза, 28 сентября – 3 октября 2009 г.). — М.: Изд-во мех.-матем. ф-та МГУ, 2009. — С. 38–41.

[6] *Дудакова О. С.* О классах функций k-значной логики, монотонных относительно множеств ширины три // Мат-лы X Междунар. семинара "Дискретная математика и ее приложения" (Москва, МГУ, 1–6 февраля 2010 г.). — М.: Изд-во мех.-матем. ф-та МГУ, 2010. — С. 178–180.

[7] *Мартынюк В. В.* Исследование некоторых классов в многозначных логиках // Проблемы кибернетики. — М.: Наука. — 1960. — Т. 3. — С. 49–60.

[8] *Дудакова О. С.* О порождающих системах специального вида для предполных классов монотонных функций k-значной логики // Мат-лы XVI Междунар. конф. "Проблемы теоретической кибернетики" (Нижний Новгород, 20–25 июня 2011 г.). — Нижний Новгород: Изд-во Нижегородского гос. ун-та, 2011. — С. 145–147.

[9] *Дудакова О. С.* О существовании порождающих систем специального вида в классах монотонных функций k-значной логики // Мат-лы VIII молодежной научной школы по дискретной математике и ее приложениям (Москва, 24–29 октября 2011 г.). Часть I. — 2011. — С. 27–29.

О графическом разнообразии шаров

А. А. Евдокимов, Т. И. Федоряева

evdok@math.nsc.ru, fti@math.nsc.ru

Институт математики им.С.Л.Соболева, Новосибирск

Пусть $\tau_i(G)$ — число всех различных шаров радиуса i в метрическом пространстве обыкновенного связного графа G с обычным расстоянием между вершинами (т.е. длиной кратчайшей цепи, соединяющей эти вершины).

Определение 1 [1]. Вектор $\tau(G) = (\tau_0(G), \tau_1(G), \ldots, \tau_d(G))$, где $d = d(G)$ — диаметр графа G, называется *вектором разнообразия шаров* графа G.

Векторы такого вида впервые рассмотрены в [2], где предложено изучать строение графов как дискретных метрических пространств через разнообразие и пересекаемость метрических шаров, содержащихся в графе. При таком подхо-

де естественно формулируется задача характеризации векторов разнообразия шаров графов.

Определение 2. Вектор $\tau = (\tau_0, \tau_1, \ldots, \tau_d)$, составленный из целых неотрицательных чисел, называется *графическим разнообразием шаров*, если существует граф, вектор разнообразия шаров которого совпадает с τ. Этот граф называется *графической реализацией* вектора τ.

Задача описания векторов разнообразия шаров решена в [1] для деревьев, а в общем случае для графов остается открытой. В [3] детально исследовались компоненты вектора разнообразия шаров и соотношения между ними в графах (деревьях) с дополнительными ограничениями на число вершин и диаметр. В частности, получены необходимые и достаточные условия реализуемости целочисленных векторов специального вида графическим разнообразием шаров. В [4] найден богатый класс целочисленных векторов, являющихся графическим разнообразием шаров. В [3, 5] установлены точные верхние оценки и точные нижние оценки числа различных шаров радиуса i в n-вершинных графах диаметра d. Эти оценки дают широкий класс целочисленных векторов, не являющихся графическим разнообразием шаров. Кроме того, для графического разнообразия шаров $(\tau_0, \tau_1, \ldots, \tau_d)$ результаты из [6] показывают нетривиальные взаимосвязи его компонент τ_i, принимающих наибольшие возможные значения. В настоящей работе описываются векторы разнообразия шаров для графов малого диаметра.

Нам потребуются точные верхние оценки $\overline{\tau_i}$ и точные нижние оценки $\underline{\tau_i}$ числа различных шаров радиуса i в n-вершинных графах диаметра d ($n \geqslant d+1 \geqslant 2$ или $n = d+1 = 1$).

Теорема 1 [3, 5]. *Для произвольного n-вершинного графа G диаметра d справедливы следующие неравенства: $\underline{\tau_i} \leqslant \tau_i(G) \leqslant \overline{\tau_i}$, где*

$$\underline{\tau_i} = \begin{cases} n, & \text{если } i = 0, \\ 2(d-i)+1, & \text{если } 1 \leqslant i < d, \\ 1, & \text{если } i \geqslant d; \end{cases}$$

$$\overline{\tau_i} = \begin{cases} n, & \text{если } 0 \leqslant i < d \text{ и } i \leqslant \max\{\lfloor d/2 \rfloor, s\}, \\ 3(d-i)+1, & \text{если } \lfloor d/2 \rfloor < s < i < d, \\ n+d+\lfloor d/2 \rfloor - 3i, & \text{если } s \leqslant \lfloor d/2 \rfloor < i \leqslant \lfloor d/2 \rfloor + s \text{ и } i < d, \\ 2(d-i)+1, & \text{если } \lfloor d/2 \rfloor + s < i < d, \\ 1, & \text{если } i \geqslant d, \end{cases}$$

$$s = n - d - 1.$$

Лемма 1 [1]. *Пусть P — простая цепь длины d. Тогда вектор разнообразия шаров цепи P равен $\Delta_d = (\Delta_0^d, \Delta_1^d, \ldots, \Delta_d^d)$, где*

$$\Delta_i^d = \begin{cases} d+1, & \text{если } 0 \leqslant i \leqslant \lfloor d/2 \rfloor, \\ 2(d-i)+1, & \text{если } \lfloor d/2 \rfloor < i < d, \\ 1, & \text{если } i \geqslant d, \end{cases}$$

В работе будем использовать графы $H_{n,d,t}$, построенные в [3].

Теорема 2 [3, 6]. *Пусть* $n \geqslant d+1+t$ *и* $0 < t < d$. *Тогда* $\tau(H_{n,d,t}) = (n, \ldots, n, \Delta_{t+1}^d, \Delta_{t+2}^d, \ldots, \Delta_d^d)$.

Для обыкновенного связного графа G через $B_i^G(x)$ обозначим шар радиуса i с центром в вершине x относительно обычной метрики ρ_G.

Лемма 2. *Справедливы следующие утверждения:*

(i) *Если вектор* $(\tau_0, \tau_1, \ldots, \tau_d)$ *является графическим разнообразием шаров, то выполняется система неравенств* $\tau_0 \geqslant \ldots \geqslant \tau_i \geqslant \tau_{i+1} \geqslant \ldots \geqslant \tau_d = 1$.

(ii) *Если* $B_i^G(u) = B_i^G(v)$, *то* $B_{i+1}^G(u) = B_{i+1}^G(v)$.

Лемма 3. *Пусть* $d \geqslant 2$ *и* $\tau_0 \geqslant \tau_1$. *Тогда вектор* $(\tau_0, \tau_1, \tau_2, \ldots, \tau_d)$ *является графическим разнообразием шаров тогда и только тогда, когда* $(\tau_1, \tau_1, \tau_2, \ldots, \tau_d)$ *есть графическое разнообразие шаров.*

Пусть граф G имеет вершину v степени 2. Рассмотрим различные вершины v_1, v_2, смежные с v. Определим граф G_v следующим образом. Пусть $u \notin V(G)$. Полагаем $V(G_v) = V(G) \cup \{u\}$, $E(G_v) = E(G) \cup \{v_1u, v_2u\}$. В дальнейшем треугольник графа, имеющий две вершины степени 2, будем называть *висячим треугольником.*

Лемма 4. *Пусть граф* G *имеет вершину* v *степени* 2, $\tau(G) = (\tau_0, \tau_1, \tau_2, \ldots, \tau_d)$ *и* $d \geqslant 2$. *Тогда*

(i) $\tau(G_v) = (\tau_0 + 1, \tau_1 + 1, \tau_2, \ldots, \tau_d)$, *если граф* G *не имеет висячего треугольника, содержащего вершину* v;

(ii) $\tau(G_v) = (\tau_0 + 1, \tau_1 + 2, \tau_2, \ldots, \tau_d)$, *если в графе* G *существует висячий треугольник с вершиной* v.

Теорема 3 (критерий графичности разнообразия шаров). *Целочисленный вектор* $(\tau_0, \tau_1, \ldots, \tau_d)$ *при* $d \leqslant 3$ *есть графическое разнообразие шаров тогда и только тогда, когда* $\tau_0 \geqslant \ldots \geqslant \tau_i \geqslant \tau_{i+1} \geqslant \ldots \geqslant \tau_d = 1$ *и*

(i) $\tau_0 \geqslant 2$ *при* $d = 1$;

(ii) $\tau_1 \geqslant 3$ *при* $d = 2$;

(iii) *при* $d = 3$ *справедливо неравенство* $\tau_2 \geqslant 3$, *а соотношения* $3 \leqslant \tau_1 = \tau_2 \leqslant 5$ *и* $\tau_1 = \tau_2 + 1 = 5$ *не выполняются.*

Доказательство. Пусть $(\tau_0, \tau_1, \ldots, \tau_d)$ — графическое разнообразие шаров и граф G — его графическая реализация. Тогда выполняется система неравенств, приведенная в лемме 2(i).

При $d = 0$ из леммы 2(i) имеем $\tau_0 = 1$. Кроме того, $\tau(K_1) = (1)$.

Пусть $d = 1$. Тогда $d(G) = 1$. Следовательно, граф G изоморфен K_{τ_0}, причём $\tau_0 \geqslant 2$. Обратно, при $\tau_0 \geqslant 2$ и $\tau_1 = 1$ имеем $\tau(K_{\tau_0}) = (\tau_0, \tau_1)$.

Докажем утверждение при $d = 2$. В силу леммы 3 достаточно ограничиться случаем $\tau_0 = \tau_1$. Тогда по теореме 1 получаем $\tau_1 \geqslant \Delta_1^d = 3$. Обратно, при $\tau_1 = 3$ по лемме 1 имеем $\tau(P_3) = (3, 3, 1)$, а при $\tau_1 \geqslant 4$ в силу теоремы 2 получаем $\tau(H_{\tau_1,2,1}) = (\tau_1, \tau_1, 1)$.

Пусть $d = 3$. Не уменьшая общности, в силу леммы 3 будем считать, что $\tau_0 = \tau_1$. Из теоремы 1 получаем $\tau_1 \geqslant \Delta_1^d = 4$ и $\tau_2 \geqslant \Delta_2^d = 3$, причём если $\tau_1 \leqslant 5$, то $\tau_2 \leqslant 3$. Таким образом, справедливы требуемые соотношения из (iii). Теперь построим графы, реализующие все такие векторы $(\tau_1, \tau_1, \tau_2, \tau_3)$.

Если $\tau_2 = 3$, то $\tau(P_4) = (4, 4, 3, 1)$ по лемме 1 и $\tau(H_{\tau_1,3,1}) = (\tau_1, \tau_1, 3, 1)$ при любом $\tau_1 \geqslant 5$ по теореме 2. Пусть теперь $\tau_2 \geqslant 4$. Тогда $\tau_1 \geqslant 6$.

Случай 1: $\tau_2 = 4$. Нетрудно построить графы G_1 и G_2 такие, что $\tau(G_1) = (6, 6, 4, 1)$ и $\tau(G_2) = (7, 7, 4, 1)$, причем граф G_2 имеет вершину степени 2, не входящую в треугольник. По лемме 4 вектор $(\tau_1, \tau_1, 4, 1)$ является графическим разнообразием шаров при любом $\tau_1 \geqslant 6$.

Случай 2: $\tau_2 = 5$. Аналогично случаю 1 строится граф G_3, имеющий вершину степени 2, не входящую в треугольник, с вектором разнообразия шаров $\tau(G_3) = (6, 6, 5, 1)$. По лемме 4 вектор $(\tau_1, \tau_1, 5, 1)$ — графическое разнообразие шаров для любого $\tau_1 \geqslant 6$.

Случай 3: $\tau_2 \geqslant 6$. По теореме 2 получаем $\tau(H_{\tau_2, 3, 2}) = (\tau_2, \tau_2, \tau_2, 1)$. Граф $H_{\tau_2, 3, 2}$ имеет вершину степени 2, не входящую в треугольник. Тогда для любого $\tau_1 \geqslant \tau_2$ по лемме 4 вектор $(\tau_1, \tau_1, \tau_2, 1)$ является графическим разнообразием шаров. ∎

Работа выполнена при поддержке РФФИ, проект № 14-01-00507.

Литература

[1] *Федоряева Т. И.* Разнообразие шаров в метрических пространствах деревьев // Дискрет. анализ и исслед. операций. Сер. 1. — 2005. — Т. 12, № 3. — С. 74–84.

[2] *Евдокимов А. А.* Локально изометрические вложения графов и свойство продолжения метрики // Сиб. журн. исслед. операций. — 1994. — Т. 1, № 1. — С. 5–12.

[3] *Федоряева Т. И.* Векторы разнообразие шаров для графов и оценки их компонент // Дискрет. анализ и исслед. операций. — 2007. — Т. 14, № 2. — С. 47–67.

[4] *Рычков К. Л.* О достаточных условиях существования графа с заданным разнообразием шаров // Дискрет. анализ и исслед. операций. Сер. 1. — 2006. — Т. 13, № 1. — С. 99–108.

[5] *Федоряева Т. И.* Точные верхние оценки числа различных шаров заданного радиуса в графах с фиксированными числом вершин и диаметром // Дискрет. анализ и исслед. операций. — 2009. — Т. 16, № 6. — С. 74–92.

[6] *Федоряева Т. И.* Мажоранты и миноранты класса графов с фиксированными диаметром и числом вершин // Дискрет. анализ и исслед. операций. — 2013. — Т. 20, № 1. — С. 58–76.

О сильной разрешимости и сильной допустимости нечетких линейных систем неравенств

О. А. Емец, А. О. Емец

yemetsli@mail.ru,yemets2008@ukr.net

Полтавский университет экономики и торговли, Полтава

В работе [1] исследована сильная разрешимость и сильная допустимость нечетких линейных систем уравнений. В докладе в этом аспекте рассматриваются нечеткие линейные системы неравенств.

Нечеткое число - это множество A, состоящее из пар $a|\mu(a)$, где $a \in R^1$, $\mu(a) \in [0; 1]$, т.е. $A = \{a|\mu(a); a \in [a_L, a_R] \subset R^1, \mu \in [0; 1]\}$. *Носителем* нечеткого числа называют множество чисел $a \in R^1$ в множестве пар A, образующих

это нечеткое число, для которых задается $\mu(a)$. Соответствие, ставящее числу $a \in [a_L, a_R]$ число $\mu(a) \in [0;1]$, называют *функцией принадлежности* нечеткого числа A.

Нечеткое число $A = \{a_1|\mu(a_1), ..., a_n|\mu(a_n)\}$ называют *дискретным* (или *нечетким числом с дискретным носителем*), если функция принадлежности на множестве $\{a\} \subset [a_L, a_R]$, которое является конечным $\{a\} = \{a_i\}_{i=1}^n$, $\mu(a_i) \in (0;1]$ $\forall i = 2, 3, ..., n-1$. Пусть $a_L = a_1 < a_2 < ... < a_{n-1} < a_n = a_R$. Нечеткое число $A = \{a|\mu(a); \forall a \in [a_L, a_R] \subset R^1\}$ называют *континуальным* (или *нечетким числом с континуальным носителем*), если функция принадлежности $\mu(a)$ определена для всех $a \in [a_L, a_R]$: $\mu(a_L) = \mu(a_R) = 0$; $0 < \mu(a) < 1$ $\forall a \in (a_L, a_R)$.

Пиковыми точками нечеткого числа A назовем точки a носителя этого нечеткого числа, в которых $\mu(a) = 1$. Дискретное нечеткое число $A = \{a_1|\mu_1, ..., a_n|\mu_n\}$, где $a_1 < a_2 < ... < a_n$, назовем *однопиковым*, если все пиковые точки этого числа идут подряд: $\alpha_i = a_{i+1}, a_{i+2}, ..., a_{i+p} = \alpha_R$. Точки $a_{i+1}, ..., a_{i+p}$ при этом будем называть *пиком* дискретного нечеткого числа A. При $p = 1$ пик дискретного нечеткого числа A назовем *острым*, в противном случае ($p > 1$) - *не острым*. Континуальное нечеткое число $A = \{a|\mu(a); \forall a \in [a_L, a_R]\}$ назовем *однопиковым*, если все пиковые точки этого числа образуют отрезок $[\alpha_L, \alpha_R] \subset (a_L, a_R)$. Этот отрезок $[\alpha_L, \alpha_R]$ назовем *пиком* континуального нечеткого числа A. Если концы этого пика совпадают: $\alpha_L = \alpha_R$, то такой пик континуального нечеткого числа назовем *острым*, в противном случае ($\alpha_L \neq \alpha_R$) - *не острым*.

Нечеткое число A назовем *нормальным*, если его функция принадлежности есть неубывающей на $[a_L, \alpha_L]$ и невозрастающей на $[\alpha_R, a_R]$. Нормальное однопиковое число назовем *стандартным (континуальным или дискретным). Стандартизированным* нечетким числом назовем дискретное нечеткое число вида $A = \{a_{L_0}|0;\ a_{L_1}|0,25;\ a_{L_2}|0,5;\ a_{L_3}|0,75;\ a_{L_4}|1;\ a_{R_4}|1;\ a_{R_3}|0,75;\ a_{R_2}|0,5;\ a_{R_1}|0,25;\ a_{R_0}|0\}$, где $a_{L_0} < a_{L_1} < a_{L_2} < a_{L_3} < a_{L_4} < a_{R_4} < a_{R_3} < a_{R_2} < a_{R_1} < a_{R_0}$. Это число можно задавать упорядоченной десяткой $A = (a_{L_0}, a_{L_1}, a_{L_2}, a_{L_3}, a_{L_4}, a_{R_4}, a_{R_3}, a_{R_2}, a_{R_1}, a_{R_0}) = (\underline{a_0}, \underline{a_1}, \underline{a_2}, \underline{a_3}, \underline{a_4}, \overline{a_4}, \overline{a_3}, \overline{a_2}, \overline{a_1}, \overline{a_0})$. Если пик острый, то, очевидно, $\underline{a_4} = \overline{a_4}$.

Далее идет речь только о стандартизированных нечетких числах.

Используем терминологию из [2] относительно интервальных матриц и линейных систем неравенств.

Нечеткой матрицей A^f назовем 5-ти слойную таблицу (матрицу, массив), на каждом слое состоящую из интервальных матриц, где на слое t матрица $I_A^t = (a_{ijt})$ при $t = const$ является интервальной матрицей $I_A^t = [\underline{A}^t, \bar{A}^t]$. Здесь $\underline{A}^t = (\underline{a}_{ijt}) \in R^{m \times n}$, $\bar{A}^t = (\overline{a}_{ijt}) \in R^{m \times n}$, $t \in \{0, 1, 2, 3, 4\}$; a_{ijt} обозначим элементы матрицы A^f $a_{ijt} = [\underline{a}_{ijt}, \overline{a}_{ijt}]$, $\underline{a}_{ijt}$, $\overline{a}_{ijt}$ - параметры нечеткого стандартизированного числа $a_{ij} = (\underline{a}_{ij0}, \underline{a}_{ij1}, \underline{a}_{ij2}, \underline{a}_{ij3}, \underline{a}_{ij4}, \overline{a}_{ij4}, \overline{a}_{ij3}, \overline{a}_{ij2}, \overline{a}_{ij1}, \overline{a}_{ij0})$, i - номер строки, j - номер столбца. Число t назовем *номером слоя* матрицы A^f; матрицу I_A^t - слоем t матрицы A^f.

В случае, когда $n = 1$ матрицу $A^f \in R^{m \times 1 \times 5}$ назовем *нечетким вектором (столбцом)* с m нечеткими координатами и обозначим b^f. Вектор I_b^t - назовем t слоем вектора b^f, а вектор b^f - пятислойным; $t = 0, 1, 2, 3, 4$.

Напомним определение интервальной линейной системы неравенств [2].
Интервальной линейной системой неравенств

$$I_A x \leqslant I_b \tag{1}$$

называют семейство всех систем линейных неравенств

$$Ax \leqslant b, \tag{2}$$

где

$$A \in I_A; b \in I_b. \tag{3}$$

Введем понятие нечеткой линейной системы неравенств.
Нечеткой линейной системой неравенств

$$A^f x \leqslant b^f, \tag{4}$$

называются совокупность пяти интервальных линейных систем неравенств

$$I_A^4 x \leqslant I_b^4; I_A^3 x \leqslant I_b^3; I_A^2 x \leqslant I_b^2; I_A^1 x \leqslant I_b^1; I_A^0 x \leqslant I_b^0. \tag{5}$$

где нечеткая матрица A^f и интервальная матрица I_A^t, нечеткий вектор b^f и интервальный I_b^t соотносятся соответственно между собой в соответствии с определением нечеткой матрицы , а $t \in \{0,\ 1,\ 2,\ 3,\ 4\}$.

Поставим в соответствие линейной интервальной системе неравенств (1) $I_A^t x \leqslant I_b$, $t \in \{0,\ 1,\ 2,\ 3,\ 4\}$, семейство с номером t систем линейных неравенств вида (2) с данными вида (3) соответственно:

$$A^t x \leqslant b^t, \tag{6}$$

$$A^t \in I_A^t; b^t \in I_b^t. \tag{7}$$

Нечеткая линейная система неравенств (4) называется *сильно разрешимой (сильно допустимой) в смысле* t ($t \in \{0,\ 1,\ 2,\ 3,\ 4\}$), если при $t = const$ каждая из систем (6) с данными (7) разрешима (допустима). Сильную разрешимость (допустимость) в смысле t будем еще называть сильной разрешимостью (допустимостью) типа t.

Теорема 1. *Нечеткая линейная система неравенств вида (4)* $A^f x \leqslant b^f$ *сильно разрешима в смысле* t *тогда и только тогда, когда допустима система* $\bar{A}^t x^1 - \underline{A}^t x^2 \leqslant \underline{b}^t, t \in \{0,\ 1,\ 2,\ 3,\ 4\}$.

Сильным решением в смысле t ($t \in \{0,\ 1,\ 2,\ 3,\ 4\}$) нечеткой линейной системы неравенств (4) назовем вектор x^t, удовлетворяющий системе $A^t x \leqslant b^t$ $\forall A^t \in I_A^t$, $\forall b^t \in I_b^t$.

Теорема 2. *Если нечеткая линейная система неравенств (4)* $A^f x \leqslant b^f$ *сильно разрешима в смысле* t, *то она имеет сильное в смысле* t *решение* x^t.

Обозначим $e = (1, ..., 1)^T \in R^m$ - единичный вектор, здесь T - символ транспонирования. Размерность вектора e, если она явно не указана, легко определяется из контекста. Пусть $I_A^t = [A_c^t - \Delta^t; A_c^t + \Delta^t]$ - заданная интервальная $m \times n$ матрица, определяемая (5), а вектор $y \in Y_m \subset R^m$, где $Y_m = \{y \in R^m \mid |y| = e\}$. Обозначим D_z для вектора $z \subset R^m$ квадратную матрицу из $R^{m \times m}$, в которой его элементы стоят на главной диагонали, а остальные элементы - нули.

Введем в рассмотрение также матрицы: $A_{yz}^t = A_c^t - D_y \Delta^t D_z$, где $y \in Y_m$, $z \in Y_m$. Если y или z - единичный вектор e, то D_y, D_z соответственно в этом случае - единичные матрицы. Поэтому $A_{ye}^t = A_c^t - D_y \Delta^t$; $A_{ez}^t = A_c^t - \Delta^t D_z$, а $A_{-ez}^t = A_c^t + \Delta^t D_z$. Отметим, что $A_{ye}^t \in I_A^t$.

Определим $\forall x \in R^m$ вектор sgn x, i-ая координата которого такова $(\text{sgn } x)_i = \begin{cases} 1, & x_i \geqslant 0, \\ -1, & x_i < 0, \end{cases} \quad \forall i = 1, 2, ..., m.$

Теорема 3. *Следующие три утверждения эквиваленты: 1) x^t - сильное в смысле t решение нечеткой линейной системы неравенств $A^f x \leqslant b^f$; 2) x^t удовлетворяет условию $A_c^t x - b_c^t \leqslant -\Delta^t |x| - \delta^t$; 3) вектор $x^t = \bar{x}^t - \bar{\bar{x}}^t$, где для $\bar{x}^t, \bar{\bar{x}}^t$ справедливы условия $\bar{A}^t x^t - \underline{A}^t \bar{\bar{x}}^t \leqslant \underline{b}^t$; $\bar{x}^t \geqslant 0$; $\bar{\bar{x}}^t \geqslant 0$.*

Теорема 4. *Нечеткая линейная система неравенств вида (4) $A^f x \leqslant b^f$ сильно допустима в смысле t ($t \in \{0, 1, 2, 3, 4\}$) тогда и только тогда, когда допустима система $\bar{A}^t x \leqslant \underline{b}^t$.*

Теорема 5. *Если вектор x^t является сильным решением типа t, $t \in \{ 0, 1, 2, 3 \}$ нечеткой линейной системы неравенств $A^f x \leqslant b^f$, то он является сильным решением типа $t+1$ этой системы.*

Теорема 6. *Если нечеткая линейная система неравенств (4) $A^f x \leqslant b^f$ является сильно разрешимой типа t, $t \in \{ 0, 1, 2, 3 \}$, то она является сильно разрешимой типа $t+1$.*

Литература

[1] *Сергиенко И. В., Емец О. А., Емец А. О.* Системы линейных уравнений с данными в виде нечетких множеств: слабая разрешимость и слабая допустимость // Кибернетика и системный анализ. — 2014. (в печати).

[2] *Фидлер М., Недома Й., Рамик Я., Рон И., Циммерманн К.* Задачи линейной оптимизации с неточными данными. — М.-Ижевск: НИЦ "Регулярная и хаотическая динамика Ин-т компьютерных исследований, 2008. — 288 с.

Применение вейвлет-преобразования для определения средних диаметров объектов на изображении

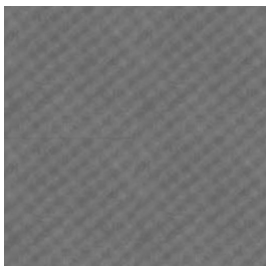
П. В. Желтов, В. И. Семенов, А. К. Шурбин

chnk@mail.ru, syundyukovo@yandex.ru, shurti@mail.ru

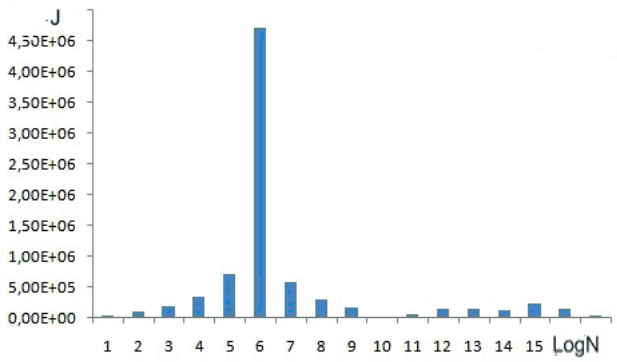
ФГБОУ ВПО «Чувашский государственный университет им. И.Н. Ульянова», Чебоксары

При разработке новых материалов важнейшая роль принадлежит микроскопической структуре, а следовательно, методам ее наблюдения и оценки. Количественные данные о геометрических параметрах микроструктуры позволяют получать достоверные зависимости между свойствами и структурой. Используя такие зависимости можно выбрать оптимальный состав, наилучшую технологию получения и обработки материалов. Решение задачи создания материалов с определенными свойствами, необходимыми для нормального функционирования изделий в процессе эксплуатации, является насущной необходимостью.

В данной работе для определения средних размеров микрообъектов на изображении, полученных электронным микроскопом, используется непрерывное быстрое вейвлет-преобразование. Для вычисления вейвлет-спектра применяется дискретная версия непрерывного вейвлет-преобразования в частотной области, то есть применяется быстрое преобразование Фурье, что позволяет не несколько порядков увеличить скорость вычисления вейвлет-преобразования изображений [1,2]. Для того, чтобы определить средний размер микрообъектов на изображении достаточно нескольких минут. Для этого горизонтальной и вертикальной разверткой, считывается интенсивность каждого пикселя изображения в bmp – формате, вычисляется вейвлет-преобразование изображения с разными масштабными коэффициентами. Изображение разлагается на 18 уровней и строится гистограмма распределения суммарной интенсивности J, приходящейся на каждый уровень разложения, от логарифма N по основанию два, где N – масштабное число. На рис. 1 а) представлено изображение тестовой сетки, полученное электронным микроскопом.[1]

а) 

[1] Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 14–07–00143 а.



б)

Рис. 1. Изображение сетки
и распределение интенсивности от уровня разложения

На рис. 1 б) представлено распределение интенсивности от уровня разложения. Средний диаметр микрообъектов D вычисляется по формуле:

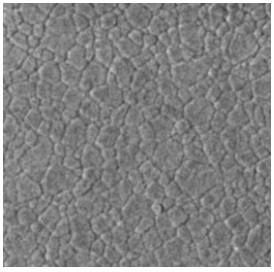
$$D = \frac{\sum\limits_{i=1}^{9} J_i * i}{\sum\limits_{i=1}^{9} J_i}, \tag{1}$$

где i – номер разложения,

J_i - суммарная интенсивность i – го разложения.

На рис.2 а) представлено изображение зерен керамики, полученное электронным микроскопом. На рис.2 б) представлено распределение интенсивности от уровня разложения для керамики.

а)



б)

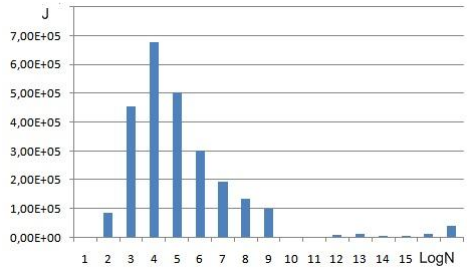


Рис. 2. Изображение зерен керамики
и распределение интенсивности от уровня разложениия

Вычислив по формуле (1) мы получаем средний диаметр D микрообъектов в логарифмическом масштабе. Средний диаметр микрообъектов в масштабе изображения вычисляется по формуле

$$D_{cp} = 2^D, \tag{2}$$

Средний диаметр микрообъектов, вычисленный по формуле (2) совпадает со средним диаметром, измеренным классическим способом. Например, таким

образом, легко установить зависимость среднего диаметра микрообъектов от температуры образца.

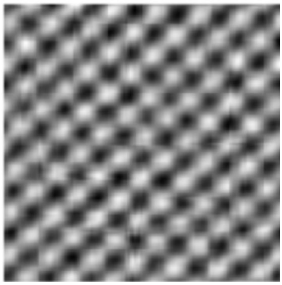
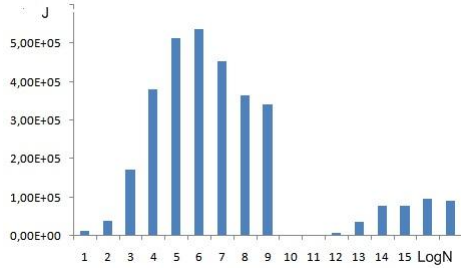
а)  б) 

Рис. 3. Изображение сетки и распределение интенсивности от уровня разложения для металла

Так как при суммировании всех уровней разложения получается исходное изображение, то мы можем посмотреть, какая картина получается для каждого уровня разложения или для нескольких уровней разложения. На рис.3 а) представлено изображение, полученное при суммировании трех уровней разложения и последующего увеличения контраста, для изображения на рис. 1. На рис. 3 б) представлено распределение интенсивности от уровня разложения для материала из металла.

Литература

[1] *Желтов П. В., Семенов В. И.* Вейвлет-преобразование акустического сигнала: Монография. — Казань: Изд-во казан. гос. техн. ун-та, 2009. — 93 с.

[2] *Желтов П. В., Семенов В. И., Шурбин А. К.* Применение непрерывного быстрого вейвлет-преобразования для обработки изображений // Materialy VIII miedzunarodowej naukowi praktycznej konferencji. — Przemysl: Nauka studia, 2012. — С. 65–69.

О верхней оценке длины слабопрефиксного кода для одного семейства КС-языков

Л. П. Жильцова, И. И. Крылова

larzhil@rambler.ru, krylova.irina239@gmail.com

ННГУ, Нижний Новгород

В работе рассматривается задача экономного алфавитного кодирования для одного семейства контекстно-свободных языков (КС-языков), содержащего языки вида xx^R, где x - произвольное слово в некотором алфавите, а x^R - обращение слова x.

Пусть $B = \{b_1, b_2, \ldots, b_m\}$ - алфавит, B^* - множество всех слов в алфавите B, B^+ - множество всех непустых слов в B. Для заданного алфавита B двоичное алфавитное кодирование можно задать схемой $f : b_i \to v_i\,(i = 1, \ldots, m)$, $v_i \in \{0,1\}^*$.

Набор $V = (v_1, v_2, \ldots, v_m)$ называется кодом, v_i - элементарными кодами, и набор $D = (|v_1|, |v_2| \ldots, |v_m|)$ - спектром длин элементарных кодов (здесь $|x|$ - длина слова x).

Процесс кодирования слова $\alpha = b_{i_1} b_{i_2} \ldots b_{i_k}$ состоит в замене символов из B их элементарными кодами: $f_V(\alpha) = v_{i_1} v_{i_2} \ldots v_{i_k}$.

Под языком L в алфавите B понимается произвольное множество слов $L \subseteq B^*$. Схема алфавитного кодирования f задает отображение

$$f_V : L \to \{0,1\}^+;$$

f_V должно обладать свойством инъективности, позволяющим однозначно расшифровывать закодированные слова. Кодирование, обладающее свойством инъективности, будем называть взаимно-однозначным.

Пусть $D^{(all)}(L)$ - множество всех двоичных кодов, для которых f_V взаимно однозначно на языке L, и $D(L)$ - некоторый его подкласс: $D(L) \subseteq D^{(all)}(L)$. Частотной характеристикой слова α будем называть вектор частот $P_\alpha = (p_1(\alpha), \ldots, p_m(\alpha))$, где $p_i(\alpha) = \frac{|\alpha|_i}{|\alpha|}$ - отношение числа вхождений буквы b_i в α к длине α. Величину

$$C(V, P_\alpha) = \frac{|f_V(\alpha)|}{|\alpha|} = \sum_{i=1}^{m} p_i(\alpha) \cdot |v_i|$$

назовем стоимостью кодирования слова α отображением f_V; стоимость оптимального кодирования для $\alpha \in L$ в классе кодов $D(L)$ определим как

$$C(L, \alpha) = \inf_{V \in D(L)} \{C(V, P_\alpha)\}.$$

Код V назовем оптимальным для α в классе кодов $D(L)$, если $C(V, P_\alpha) = C(L, \alpha)$.

Пусть $D^0(L)$ - множество всех кодов из $D(L)$, которые могут оказаться оптимальными хотя бы для одного сообщения $\alpha \in L$:

$$D^0(L) = \{V | V \in D(L), \exists \alpha (C(V, P_\alpha) = C(L, \alpha))\}.$$

Известно [1], что для любого языка L множество $D^0(L)$ конечно, поэтому оптимальное кодирование для любого слова в любом языке существует.

Матрицей оптимального кодирования для языка L в классе кодов $D(L)$ назовем матрицу $M(L)$, строками которой являются все различные спектры длин кодов из $D^0(L)$. В силу конечности $D^0(L)$ матрица $M(L)$ также конечна.

Если матрица оптимального кодирования известна, можно найти спектр оптимального кода путем минимизации линейной формы на конечном множестве спектров, образующих матрицу. Разрешимость проблемы распознавания взаимной однозначности кодирования дает возможность построить код $V \in D^0(L)$ по найденному спектру.

Для языка всех сообщений $L = B^*$ и класса кодов $D(L) = D^{(all)}(L)$ задачи описания матрицы оптимального кодирования и построения эффективного алгоритма оптимального кодирования решены: неравенство Мак-Миллана

дает аналитическое описание матрицы $M(B^*)$, и алгоритм Хаффмана позволяет минимизировать неотрицательную линейную форму на множестве целочисленных решений неравенства Мак-Миллана, причем при решении задачи оптимального кодирования достаточно ограничиться классом префиксных кодов.

Учет структуры языка сообщений может существенно влиять на эффективность кодирования [1]. Однако всё усложняется при переходе к более сложным, чем B^*, языкам, когда $L \subset B^*$. Например, в классе регулярных языков при $D(L) = D^{(all)}(L)$ задача построения матрицы оптимального кодирования является до сих пор открытой. Она решена Ал.А. Марковым для подкласса регулярных языков - языков, порождаемых связными источниками [1].

Известно, что для класса кодов $D^{(all)}(L)$ задача построения матрицы оптимального кодирования в классе КС-языков, и даже в более узком классе детерминированных КС-языков, алгоритмически неразрешима [2], причем причина неразрешимости лежит в алгоритмической неразрешимости проблемы распознавания взаимной однозначности кодирования в классе $D^{(all)}(L)$. Поэтому представляет интерес изучение рассматриваемой задачи для более узких подклассов КС-языков, при сужении классов взаимно-однозначных кодов.

Сложность расшифровки матрицы оптимального кодирования характеризует значность $k(L) = \max\{d_{ij} \in M(L)\}$ этой матрицы (максимум значений её элементов). Знание оценки сверху для $k(L)$ позволяет ограничить класс векторов, которые могут входить в $M(L)$.

Рассмотрим семейство КС-языков $\{L_m\}$. Язык L_m порождается КС - грамматикой $G_m = \langle B_m, S, R, S\rangle$, где

$B_m = \{b_1, b_2, \ldots, b_m\}$ -терминальный алфавит (алфавит языка),

S - нетерминальный алфавит, состоящий из единственного символа, являющегося аксиомой грамматики,

R - множество правил: $R = \{S \to b_i S b_i, S \to b_i b_i\,(i = 1..m)\}$.

Грамматика G_m порождает множество всех слов в m-буквенном алфавите, являющихся палиндромами четной длины. Так, например, слово $\alpha = \alpha_1\alpha_2\alpha_3\alpha_3\alpha_2\alpha_1 \in L_m$, где $\alpha_i \in B_m$, $(m = 1, 2, 3)$.

Будем говорить, что схема алфавитного кодирования $f : b_i \to v_i$ обладает свойством слабого префикса, если для любых $i, j (1 \leqslant i, j \leqslant m, i \neq j)$ слово v_i не является одновременно префиксом и суффиксом слова v_j; код V, задаваемый схемой с таким свойством, будем называть слабопрефиксным. Наличие у схемы f свойства префикса влечет и наличие свойства слабого префикса, обратное утверждение неверно.

Рассмотрим следующий пример. Пусть $B = \{b_1, b_2, b_3, b_4\}$. Схема f : $b_1 \to 0$, $b_2 \to 1$, $b_3 \to 01$, $b_1 \to 10$ обладает свойством слабого префикса.

Лемма 1. *Если схема f обладает свойством слабого префикса, то алфавитное кодирование, задаваемое схемой f, взаимно однозначно на L_m.*

Класс слабопрефиксных кодов обозначим $D^{(sp)}(L_m)$. Опишем процесс построения одного слабопрефиксного кода.

Все слова в алфавите $\{0,1\}$ расположим в лексикографическом порядке:

$$0,1,00,01,10,11,000,001,010,011,100,101,110,111,0000,\ldots \qquad (1)$$

Положим $v_1 = 0$. Двигаясь по (1) слева направо, в качестве v_i $(i = 2,3,\ldots\ldots,m)$ будем выбирать первое из слов, которое образует слабопрефиксный код с множеством выбранных ранее элементарных кодов. Выбор очередного элементарного кода всегда можно сделать, так как одним из претендентов на включение в код всегда является, например, слово 0^j1 при некотором j. В результате построения будет получаться следующая последовательность элементарных кодов:

$$(0,1,01,10,001,011,100,110,0001,0011,0111,1000,1100,1110,00001,\ldots)\,.$$

Первые m слов из нее образуют слабопрефиксный код для языка L_m. Обозначим этот код через V_m^*.

Лемма 2. *При $D(L_m) = D^{(sp)}(L_m)$ спектр кода V_m^* принадлежит матрице оптимального кодирования $M(L_m)$.*

Лемма 3. *Значность $k(L_m)$ матрицы оптимального слабопрефиксного кодирования равна* $\max\{|v_i|\,|v_i \in V_m^*\}$.

Теорема 4. $k(L_m) \leqslant \lfloor \log_2(4m+1)+1 \rfloor$.

Следствие 1. *Для семейства языков $\{L_m\}$ задача оптимального алфавитного кодирования в классе слабопрефиксных кодов алгоритмически разрешима.*

Заметим, что в классе хорошо известных префиксных кодов для языка всех сообщений B^* максимальная значность матрицы оптимального кодирования равна $m-1$. Заметим также, что для языка B^* слабопрефиксные коды не являются взаимно-однозначными.

Литература

[1] *Марков Ал. А.* Введение в теорию кодирования. — М.: Наука, 1982. — 192 с.

[2] *Жильцова Л. П.* Об алгоритмической сложности задач оптимального алфавитного кодирования для контекстно-свободных языков // Дискретная математика. — 1989. — Т. 1, № 2. — С. 38–51.

О замкнутых классах функций, содержащих функцию почти единогласия

Д. Н. Жук

zhuk@intsys.msu.ru

МГУ им.М.В.Ломоносова, Москва

Известно, что замкнутые классы, содержащие функцию почти единогласия обладают многими важными свойствами, например, все такие классы

конечно-порождены, а также могут быть заданы предикатами арности менее, чем арность функции почти единогласия [1, 2]. В связи с этим встают два вопроса: как определить, что замкнутый класс содержит функцию почти единогласия; каким может быть порядок (минимальная арность функций в базисе) замкнутого класса, содержащего функцию почти единогласия. Этим двум вопросам и посвящена данная работа.

Работа разбита на 4 части. В первой части даются основные определения, необходимые для формулировки результатов. Во второй части приводятся оценки на минимальную арность функции почти единогласия в замкнутом классе, порождённом множеством функций арности не более n. В третьей части приводятся оценки для классов, заданных множеством предикатов арности не более n. Эти оценки дают простой алгоритм проверки существования функции почти единогласия в замкнутом классе, хотя изначально было не понятно, существует ли алгоритм. Последняя часть посвящена порядку замкнутых классов, содержащих функцию голосования (функцию почти единогласия арности 3). До последнего времени было непонятно, каким он может быть в трех- и четырехзначном случае. Автору, совместо с Sebastian Kerkhoff, удалось решить задачу для этих случаев и, таким образом, полностью закрыть вопрос о порядке замкнутых классов, содержащих функцию голосования.

Введение

Пусть $\mathbb{N} = \{1, 2, 3, \ldots\}$ — множество всех натуральных чисел, $\mathbb{N}_0 = N \cup \{0\}$, $E_k = \{0, 1, 2, \ldots, k-1\}$, для $n \geqslant 1$ положим $P_k^n = \{f^n | f^n : E_k^n \to E_k\}$, $P_k = \bigcup\limits_{n \geqslant 1} P_k^n$. Элементы P_k будем называть функциями k-значной логики. На множестве P_k обычным образом определяем оператор замыкания $[\,]$ относительно операций суперпозиции. Множество $M \subseteq P_k$ называем замкнутым, если $[M] = M$.

Отображение $E_k^h \to \{0, 1\}$ будем называть предикатом арности h. Пусть

$$R_k^h = \{\rho | \rho : E_k^h \to \{0, 1\}\}, R_k = \bigcup_{h \geqslant 1} R_k^h.$$

Обычным образом определим понятие *функция сохраняет предикат*. Будем писать, что $\begin{pmatrix} b_1 \\ b_2 \\ \ldots \\ b_h \end{pmatrix} \in \rho$, если $\rho(b_1, b_2, \ldots, b_h) = 1$.

Будем говорить, что функция $f \in P_k^m$ сохраняет предикат ρ, если

$$f \begin{pmatrix} a_{1,1} & a_{2,1} & \ldots & a_{m,1} \\ a_{1,2} & a_{2,2} & \ldots & a_{m,2} \\ \ldots & \ldots & \ldots & \ldots \\ a_{1,h} & a_{2,h} & \ldots & a_{m,h} \end{pmatrix} := \begin{pmatrix} f(a_{1,1}, a_{2,1}, \ldots, a_{m,1}) \\ f(a_{1,2}, a_{2,2}, \ldots, a_{m,2}) \\ \ldots \\ f(a_{1,h}, a_{2,h}, \ldots, a_{m,h}) \end{pmatrix} \in \rho$$

для любых

$$\begin{pmatrix} a_{1,1} \\ a_{1,2} \\ \dots \\ a_{1,h} \end{pmatrix}, \begin{pmatrix} a_{2,1} \\ a_{2,2} \\ \dots \\ a_{2,h} \end{pmatrix}, \dots, \begin{pmatrix} a_{m,1} \\ a_{m,2} \\ \dots \\ a_{m,h} \end{pmatrix} \in \rho.$$

Функция f называется функцией почти единогласия (ФПЕ), если выполняется соотношение

$$f(x, y, y, \dots, y) = f(y, x, y, \dots, y) = \dots = f(y, y, \dots, y, x) = y.$$

Минимальная арность функции почти единогласия в замкнутом классе, порождённом конечным множеством функций.

Для $M \subseteq P_k$ через $nuf_k(M)$ обозначим минимальную арность ФПЕ, принадлежащую $[M]$. Если $[M]$ не содержит функцию почти единогласия, то положим $nuf_k(M) = \infty$. Через ID_k обозначим множество всех идемпотентных функций из P_k, то есть функций, удовлетворяющих тождеству $f(x, x, \dots, x) = x$. Положим

$$nuf_k(m) = max\{nuf_k(M) \mid M \subseteq P_k^m, nuf_k(M) < \infty\},$$

$$nuf_k^{(id)}(m) = max\{nuf_k(M) \mid M \subseteq ID_k^m, nuf_k(M) < \infty\}.$$

Теорема 1. *$nuf_k^{(id)}(m) \leqslant m \cdot k^3$ для любых $k \geqslant 2$, $m \geqslant 2$.*

Теорема 2. *$nuf_k(m) \leqslant k^2 \cdot (km)^{(3k)^k}$ для любых $k \geqslant 2$, $m \geqslant 2$.*

Следует отметить, что Maroti доказал в [3], что проблема существования функции почти единогласия в замкнутом классе, заданном конечным множеством функций, алгоритмически разрешима, но не смог привести никакой оценки на $nuf_k(m)$ и $nuf_k^{(id)}(m)$.

Минимальная арность функции почти единогласия в замкнутом классе, заданном конечным множеством предикатов.

Для $C \subseteq R_k$ через $\widetilde{nuf}_k(C)$ обозначим минимальную арность ФПЕ, сохраняющей каждый предикат из C. Если множество C не сохраняется функцией почти единогласия, то положим $\widetilde{nuf}_k(C) = \infty$. Пусть

$$\widetilde{nuf}_k(m) = max\{\widetilde{nuf}_k(C) \mid C \subseteq R_k^m, \widetilde{nuf}_k(C) < \infty\}.$$

В работе [4] были доказаны следующие теоремы.

Теорема 3. *$\widetilde{nuf}_k(m) \leqslant ((k-1)(m-1))^{3^k+1}$ для любых $k \geqslant 2$, $m \geqslant 2$.*

Теорема 4. *$\widetilde{nuf}_k(m) \geqslant (m-1)^{2^{k-1}}$ для любых $k \geqslant 2$, $m \geqslant 2$.*

Теорема 5. *$\widetilde{nuf}_k(m) \geqslant 2^{2^{k-3}}$ для любых $k \geqslant 4$.*

Последние две теоремы показывают, что Теорема 3 не может быть существенно улучшена, и от двойной экспоненты по k нельзя избавиться.

Следует отметить, что одновременно с автором L.Barto показал в [5], что $\widetilde{nuf}_k(m) \leqslant 4^{8^{k^m}}$.

Подрядок замкнутого класса, содержащего функцию голосования.

Порядком замкнутого класса M называется минимальное n, такое что M порождается функциями арности n, и обозначается через $ord(M)$. Через MAJ_k обозначим все замкнутые классы k-значной логики, содержащие функцию голосования. Положим

$$\lambda_k(2) := \max\{ord(C) \mid C \in MAJ_k\}.$$

Теорема 6. *[6, Lakser]* $\lambda_k(2) = k \cdot (k-2)$ *для* $k \geqslant 5$

Нам удалось найти $\lambda_3(2)$ и $\lambda_4(2)$.

Теорема 7. *[7, Kerkhoff, Zhuk]* $\lambda_3(2) = 5$, $\lambda_4(2) = 8$.

Работа выполнена при поддержке РФФИ, проект № 13-01-00684-а.

Литература

[1] *K. A. Baker, A. F. Pixley.* Polynomial interpolation and the Chinese remainder theorem for algebraic systems. Math. Z. 1975. Bd 143, N 3. 165-174.

[2] *Benoit Larose.* Near-Unanimity Functions and CSP MathsCSP, Oxford, March 2006.

[3] *Miklós Maróti.* The existence of a near-unanimity term in a finite algebra is decidable The Journal of Symbolic Logic Volume 74, Number 3, Sept. 2009.

[4] *D. Zhuk.* The existence of a near-unanimity function is decidable. Algebra universalis, February 2014, Volume 71, Issue 1, pp 31-54.

[5] *L. Barto.* Finitely related algebras in congruence distributive varieties have near unanimity terms. Canadian Journal of Mathematics, published electronically on December 24, 2011, doi:10.4153/CJM-2011-087-3.

[6] *H. Lakser* Finitely generated clones of terms. Algebra universalis, 1989, Volume 26, pp 78-89.

[7] *S. Kerkhoff, D. Zhuk* The generation of clones with majority operations. Algebra universalis, Accepted.

Метод отсечений с аппроксимацией надграфика и оценка точности решения

И. Я. Заботин, Р. С. Яруллин

IYaZabotin@mail.ru, YarullinRS@gmail.com

Казанский (Приволжский) федеральный университет, Казань

Предлагается метод решения задачи выпуклого программирования, который относится к классу методов отсечений (напр., [1–8]). Метод использует

последовательное погружение надграфика целевой функции в многогранные множества и основан на идеях известного метода уровней [5]. Главное отличие предлагаемого метода от алгоритма [5] заключается в том, что в нем не требуется вложение каждого из аппроксимирующих множеств в предыдущее. Эта особенность метода дает возможность периодического отбрасывания получаемых в процессе решения дополнительных ограничений, формирующих аппроксимирующие множества.

Пусть $f(x)$ – выпуклая в n-мерном евклидовом пространстве R_n функция, D – выпуклое ограниченное замкнутое множество из R_n. Решается задача

$$\min\{f(x) : x \in D\}. \tag{1}$$

Пусть $f^* = \min\{f(x) : x \in D\}$, $X^* = \{x \in D : f(x) = f^*\}$, $x^* \in X^*$, $\partial f(x)$ – субдифференциал функции $f(x)$ в точке $x \in R_n$. Положим $K = \{0, 1, \ldots\}$, $epi(f, D) = \{(x, \gamma) \in R_{n+1} : x \in D,\ \gamma \geqslant f(x)\}$.

Выбирается выпуклое замкнутое множество $M_0 \subset R_{n+1}$ такое, что $epi(f, R_n) \subset M_0$. Задаются числа ε_0, α, β_{-1}, удовлетворяющие условиям $\varepsilon_0 > 0, \quad \alpha \leqslant f^* \leqslant \beta_{-1}$. Полагается $\delta_0 = +\infty$, $i = 0$, $k = 0$.

1. Отыскивается решение (y_i, γ_i), где $y_i \in R_n$, $\gamma_i \in R_1$, следующей задачи:

$$\min\{\gamma : (x, \gamma) \in M_i,\ x \in D,\ \gamma \geqslant \alpha\}. \tag{2}$$

Если $f(y_i) = \gamma_i$, то y_i – решение задачи (1), и процесс завершается.

2. Полагается

$$l_i = (1 - \lambda_i)\gamma_i + \lambda_i \beta_i,$$

где $\lambda_i \in (0, \bar{\lambda}]$, $\bar{\lambda} < 1$, $\beta_i = \min\{\beta_{i-1}, \delta_i\}$. Строится множество $U_i \subset D$ следующим образом. В U_i включается каждая точка $x \in D$, для которой найдется такое число γ_x, что $\gamma_x \leqslant l_i$ и $(x, \gamma_x) \in M_i$.

3. Выбирается точка $x_i \in U_i$.

4. Если выполняется неравенство

$$f(x_i) - \gamma_i > \varepsilon_k,$$

то полагается $Q_i = M_i$, и следует переход к п. 5. В противном случае полагается $i_k = i$, $z_k = x_{i_k}$, $\sigma_k = \gamma_{i_k}$, выбирается выпуклое замкнутое множество $Q_i \subset R_{n+1}$ такое, что

$$epi(f, R_n) \subset Q_i, \tag{3}$$

задается $\varepsilon_{k+1} > 0$, значение k увеличивается на единицу.

5. Выбирается $a_i \in \partial f(x_i)$ и полагается

$$M_{i+1} = Q_i \bigcap \{(x, \gamma) \in R_{n+1} : f(x_i) + \langle a_i, x - x_i \rangle \leqslant \gamma\}. \tag{4}$$

6. Полагается $\delta_{i+1} = f(x_i)$. Значение i увеличивается на единицу, и следует переход к п. 1.

Нетрудно доказать, что $(x^*, f^*) \in M_i$ для всех $i \in K$, т. е. задача (2) разрешима. На основе неравенства $\gamma_i \leqslant f^*$, $i \in K$, легко обосновывается критерий оптимальности точки y_i, заложенный в п. 1 метода.

Заметим, что условия выбора значений l_i, λ_i, β_i гарантируют непустоту множеств U_i при всех $i \in K$.

Ясно, что задача (2) для всех $i \in K$ будет являтся задачей линейного программирования, если D – многогранник, а M_0, Q_i, $i \in K$, выбраны так, что M_{i+1} вида (4) задается с помощью линейных неравенств.

Множество M_0 можно выбирать разными способами. Прежде всего отметим, что можно положить $M_0 = R_{n+1}$. В таком случае пара (y_0, γ_0), где y_0 – любая точка из D, а $\gamma_0 = \alpha$, может быть принята за решение задачи (2) при $i = 0$. Удобно задать множество M_0 с помощью одного линейного неравенства $\langle c, x\rangle - \gamma \leqslant \langle c, u\rangle - f(u)$, где $u \in R_n$, $c \in \partial f(u)$, или группой подобных неравенств. Если $f(x) = \max\limits_{j\in J} f_j(x)$, где J – конечное множество номеров, $f_j(x)$, $j \in J$, – выпуклые в R_n функции, то допустимо положить $M_0 = epi(f_{j_0}, R_n)$, где $j_0 \in J$.

Приведем теперь принципиальное замечание относительно задания множеств Q_i, на основе которого выявится возможность периодического обновления аппроксимирующих множеств за счет отбрасывания любого числа отсекающих плоскостей.

Для тех пар номеров i, k, при которых выполняются неравенства

$$f(x_i) - \gamma_i \leqslant \varepsilon_k, \tag{5}$$

условие (3) дает большие возможности в выборе множеств $Q_i = Q_{i_k}$, а значит, и в задании множеств M_{i+1}. В случае (5) можно положить, например,

$$Q_i = Q_{i_k} = R_{n+1}. \tag{6}$$

Тогда M_{i+1} задается лишь неравенством $f(x_i) + \langle a_i, x - x_i\rangle \leqslant \gamma$. То есть в случае (6) все полученные к шагу $i = i_k$ отсекающие плоскости в построении M_{i+1} не участвуют. Далее, при выполнении условия (5) множества $Q_i = Q_{i_k}$ можно задавать также в виде

$$Q_i = M_{r_i}, \tag{7}$$

где $0 \leqslant r_i \leqslant i = i_k$, поскольку при всех $r_i = 0, \ldots, i$ ввиду (3) выполняется включение $epi(f, R_n) \subset M_{r_i}$. Согласно (6), (7) при каждом $i = i_k$ можно отбрасывать при формировании M_{i_k+1} любое количество любых накопленных к шагу i_k отсекающих плоскостей.

Лемма 1. *Пусть предложенным методом построена последовательность $\{(x_i, \gamma_i)\}$, $i \in K$. Тогда для каждого $k \in K$ найдется номер $i = i_k$, для которой выполнится неравенство (5).*

Отметим, что лемма 1 гарантирует, во-первых, существование последовательностей $\{z_k\}$, $\{\sigma_k\}$, $k \in K$, и, во-вторых, гарантирует возможность обновления множеств M_{i_k+1} на итерациях с номерами $i = i_k$ за счет выбора Q_{i_k}.

Теорема 1. *Пусть в методе числа ε_k, $k \in K$, выбраны с условием, что $\varepsilon_k \to 0$, $k \to \infty$. Тогда для последовательностей $\{z_k\}$, $\{\sigma_k\}$, $k \in K$, построенных методом, справедливы равенства*

$$\lim_{k\in K} f(z_k) = f^*, \quad \lim_{k\in K} \sigma_k = f^*.$$

В заключение обсудим оценки точности решения задачи для последовательности $\{z_k\}$, $k \in K$.

Отметим, что для каждого $k \in K$ на основе соотношений $\sigma_k \leqslant f^* \leqslant f(z_k)$ и условия (5) при $i = i_k$ получается оценка $f(z_k) - f^* \leqslant \varepsilon_k$. Кроме того, при дополнительных условиях на целевую функцию задачи (1) для построенных методом точек z_k можно оценить и величины $||z_k - x^*||$, $k \in K$. А именно, справедлива следующая

Теорема 2. *Пусть функция $f(x)$ сильно выпукла в R_n с константой сильной выпуклости μ. Тогда для каждой точки z_k, $k \in K$, построенной предложенным методом, справедлива следующая оценка:*

$$||z_k - x^*|| \leqslant \sqrt{\frac{\varepsilon_k}{\mu}}.$$

Литература

[1] *Булатов В. П.* Методы погружения в задачах оптимизации. — Новосибирск: Наука, 1977. — 161 с.

[2] *Заботин И. Я.* О некоторых алгоритмах погружений-отсечений для задачи математического программирования // Изв. Иркутского гос. ун-та. Сер. "Математика". — 2011. — Т. 4, № 2. — С. 91–101.

[3] *Левитин Е. С., Поляк Б. Т.* Методы минимизации при наличии ограничений // Ж. вычисл. матем. и матем. физ.. — 1966. — Т. 6, № 5. — С. 787–823.

[4] *Колоколов А. А.* Регулярные разбиения и отсечения в целочисленном программировании // Сиб. журн. исслед. операций. — 1994. — Т. 1, № 2. — С. 18–39.

[5] *Нестеров Ю. Е.* Введение в выпуклую оптимизацию. — Москва: МЦНМО, 2010. — 274 с.

[6] *Нурминский Е. А.* Метод отделяющих плоскостей с ограниченной памятью для решения задач выпуклой негладкой оптимизации // Вычисл. методы и программирование— 2006. — Т. 7. — С. 133–137.

[7] *Kelley J. E.* The cutting-plane method for solving convex programs // SIAMJ. — 1960. — V. 8, № 4. — P. 703–712.

[8] *Lemarechal C., Nemirovskii A., Nesterov Yu.* New variants of bundle methods // Mathematical Programming. — 1995. — V. 69. — P. 111–148.

О мощности и структуре разрешающих множеств k-пороговых функций

Е. М. Замараева

elena.zamaraeva@gmail.com

ННГУ им. Лобачевского, Нижний Новгород

Введение

Рассматриваются функции $f : E_n^d = \{0, 1, \ldots, n-1\}^d \to \{0, 1\}, n \geqslant 2, d \geqslant 1$. Обозначим

$$M_\nu(f) = \{x \in E_n^d : f(x) = \nu\} \qquad (\nu = 0, 1).$$

Функция $f : E_n^d \to \{0, 1\}$ называется *пороговой*, если существуют вещественные числа $a_0, a_1, \ldots, a_d$ такие, что

$$M_1(f) = \left\{ x \in E_n^d : \sum_{j=1}^{d} a_j x_j \leqslant a_0 \right\},$$

при этом неравенство $\sum\limits_{j=1}^{d} a_j x_j \leqslant a_0$ называется *пороговым*.

Обозначим через $\mathfrak{T}(d, n)$ множество всех пороговых функций, заданных на множестве E_n^d.

Функция $f : E_n^d \to \{0, 1\}$ называется *k-пороговой*, $k \geqslant 1$, если существуют вещественные числа $a_{10}, a_{11}, \ldots, a_{kd}$ такие, что

$$M_1(f) = \left\{ x \in E_n^d : \sum_{j=1}^{d} a_{ij} x_j \leqslant a_{i0} \text{ для } i = 1, \ldots, k \right\}, \tag{1}$$

при этом неравенства $\sum\limits_{j=1}^{d} a_{ij} x_j \leqslant a_{i0}$ для $i = 1, \ldots, k$ называются *пороговыми*. Если имеет место (1), то будем говорить, что k-пороговая функция *задана* этой системой линейных неравенств.

Если f – k-пороговая функция над E_n^d, то существуют пороговые функции $f_1, \ldots, f_k$ над E_n^d такие, что

$$f(x) = f_1(x) \& \ldots \& f_k(x).$$

Будем говорить, что f *задана* набором функций $f_1, \ldots, f_k$.

Обозначим через $\mathfrak{T}(d, n, k)$ множество всех k-пороговых функций, заданных на множестве E_n^d.

Разрешающим множеством функции f из заданного класса C называется множество точек T такое, что если для некоторой функции $g \in C$, $f(x) = g(x)$ для всех $x \in T$, то f и g тождественны. Разрешающее множество T называется *тупиковым*, если никакое его собственное подмножество не является разрешающим. Разрешающее множество T функции f называется *наименьшим*,

если оно имеет минимальную мощность среди всех разрешающих множеств для функции f.

Точка x для некоторой функции f из класса C называется *существенной*, если существует некая функция $h \in C$, совпадающая с f на всей области определения за исключением точки $x : f(x) \neq h(x)$. Множество существенных точек функции обозначается через $S(f)$. Обозначим

$$S_\nu(f) = \{x \in S(f) : f(x) = \nu\}.$$

Известно, что тупиковое разрешающее множество пороговой функции представлет собой множество всех ее существенных точек. Заметим, что в общем случае множество существенных точек не является разрешающим для k-пороговой функции. Кроме того, в общем случае тупиковое разрешающее множество не единственно для k-пороговой функции.

Пусть $\sigma(f, C)$ – мощность наименьшего разрешающего множества для $f \in C$ относительно класса C. *Длиной обучения* в классе функций C называется

$$\sigma(C) = \max_{f \in C} \sigma(f, C).$$

Исходя из [1] и [2], известна оценка длины обучения в классе пороговых функций при фиксированном d:

$$\sigma(\mathfrak{T}(n, d)) = \Theta(\log_2^{d-2} n).$$

В работе предлагаются оценки мощности разрешающего множества и количества тупиковых разрешающих множеств k-пороговых функций. В ряде случаев при $d = 2$ описывается структура разрешающих множеств пороговых функций.

Разрешающие множества k-пороговых функций

Утверждение 1. *Для любого набора пороговых функций $f_1, \ldots, f_k$, задающих k-пороговую функцию f, выполнено следущее включение:*

$$S_1(f_i) \cap M_1(f) \subset S_1(f), i = 1, \ldots, k.$$

Утверждение 2. *Для любого набора пороговых функций $f_1, \ldots, f_k$, задающих k-пороговую функцию f, справедливо:*

$$x' \in S_0(f_i), x' \in \bigcap_{j \neq i} M_1(f_j) \Longrightarrow x' \in S_0(f).$$

Утверждение 3. $\sigma(\mathfrak{T}(d, n, k)) = n^d$ *при* $k > 1$.

Например, для функций $f \in \mathfrak{T}(d, n, k), k > 1, f(x) \equiv 0$, тупиковым разрешающим является все множество E_n^d.

Утверждение 4. *Пусть $f \in \mathfrak{T}(d, n, k), k > 1$, и $M_1(f) = \{x'\}$. Тогда тупиковое разрешающее множество единственно и представлено следующим образом:*

$$T(f) = \{x'\} \cup \{x \in E_n^d : НОД(|x_1 - x'_1|, \ldots, |x_d - x'_d|) = 1\}.$$

Введем обозначения:

$$M_0^i(f) = \{x \in M_0(f) : \sum_{j=1}^{d} a_{ij}x_j > a_{i0}\},$$

$$D(E_n^d) = \{x \in E_n^d : \exists i = 1, \ldots, d : x_i = 0 \vee x_i = n-1\}.$$

Утверждение 5. *Пусть для некоторой функции $f \in \mathfrak{T}(2,n,2)$ выполняется разбиение $E_n^d = M_1(f) \cup M_0^1(f) \cup M_0^2(f)$, справедливое для любой пары пороговых неравенств, задающих функцию f, и $D(E_n^d) \cap M_1(f) \neq \varnothing$. Тогда*

$$\sigma(f, \mathfrak{T}(2,n,2)) \leqslant 9.$$

Обозначим через $J(f,C)$ количество всех тупиковых разрешающих множеств для функции f в классе C. На вопрос о единственности тупикового разрешающего множества для k-пороговых функций при $k = 2, d = 2$ отвечает следующее

Утверждение 6.

$$\max_{f \in \mathfrak{T}(2,n,2)} J(f, \mathfrak{T}(2,n,2)) = \Omega(n^2).$$

Пусть

$$P(f) = \mathrm{Conv}(M_1(f)),$$

$$P'(f) = \{x : a_{i1}x_1 + a_{i2}x_2 \leqslant a_{i0} + 1\},$$

где $a_{i1}x_1 + a_{i2}x_2 \leqslant a_{i0}, {a_{i1}}^2 + {a_{i2}}^2 = 1, i = 1, \ldots, |\mathrm{Vert}(P(f))|$, является системой неравенств, задающей $P(f)$, и каждое неравенство соответствует уравнению грани $P(f)$.

Тогда имеет место следующее

Утверждение 7. *Пусть $f \in \mathfrak{T}(2,n,k)$ и $|M_1(f)| > 1$. Тогда $(P' \setminus P) \cap M_0(f) \cup \mathrm{Vert}(P(f)) = T$ является разрешающим множеством функции f.*

Утверждение 8. *Пусть $f \in \mathfrak{T}(2,n,2)$, $|M_1(f)| > 1$ и $D(E_n^2) \cap M_1(f) = \varnothing$. Тогда существует разрешающее множество T такое, что:*

$$|T| = O(n-l),$$

где l - расстояние между наиболее удаленными друг от друга точками из $M_1(f)$.

Литература

[1] *Шевченко В. Н.* О нижней оценке сложности расшифровки пороговых функций k-значной логики // Журн. вычисл. матем. и матем. физики. — 1999. — Т. 39, № 2. — С. 346–352.

[2] *Золотых Н. Ю., Чирков А. Ю.* Сложность расшифровки пороговых функций многозначной логики // Материалы XI Международного семинара «Дискретная математика и ее приложения», посвященного 80-летию со дня рождения академика О. Б. Лупанова (18–23 июня 2012 г.) — Москва: Изд-во механико-матем. факультета МГУ., 2012. — С. 63–67.

О базисах клона всех ультрафункций ранга 2

С. В. Замарацкая

swetlana_zam@mail.ru

ФГБОУ ВПО «Восточно-Сибирская государственная академия образования», Иркутск

Пусть $E = \{0, 1\}$, $F = \{\{0\}, \{1\}, \{0, 1\}\}$. *Ультрафункцией ранга 2* называется отображение $f : E^n \to F$. Множество всех ультрафункций обозначим через $P_2^{\sim}$.

Пусть $f(x_1, ..., x_m), f_1(x_1, ..., x_n), ..., f_m(x_1, ..., x_n) \in P_2^{\sim}$.

Операция суперпозиции $f(f_1(x_1, ..., x_n), ..., f_m(x_1, ..., x_n))$ порождает ультрафункцию $h(x_1, ..., x_n)$ следующим образом: *для каждого набора значений переменных* $(a_1, ..., a_n) \in E^n$

$$h(a_1, ..., a_n) = \begin{cases} \bigcap\limits_{b_i \in f_i(a_1, ..., a_n)} f(b_1, ..., b_m), \text{ если это пересечение не пусто;} \\ \\ \bigcup\limits_{b_i \in f_i(a_1, ..., a_n)} f(b_1, ..., b_m), \text{ иначе.} \end{cases}$$

Суперпозиция позволяет определять значения ультрафункции f на наборах элементов из множества F.

Клоном называется множество ультрафункций, содержащее все проекции, а также замкнутое относительно операции суперпозиции.

Клон N называется *максимальным*, если для любого клона N_1 из $N \subseteq N_1 \subseteq P_2^{\sim}$ следует, что $N_1 = N$ или $N_1 = P_2^{\sim}$.

В [1] доказано, что в $P_2^{\sim}$ всего 11 максимальных клонов: $\mathbb{K}_1$, ..., $\mathbb{K}_{11}$.

Будем говорить, что две произвольные ультрафункции f и g связаны *отношением* ρ на множестве $P_2^{\sim}$, если выполняется $f \in \mathbb{K}_i \Leftrightarrow g \in \mathbb{K}_i$ $(i \in \{1, ..., 11\})$. Очевидно, что ρ – отношение эквивалентности.

В зависимости от принадлежности максимальным клонам множество всех ультрафункций можно разбить на классы эквивалентности, характеризуемые функцией $\varphi(f) = (a_1 ... a_{11})$, значение которой

$$a_i = \begin{cases} 0, \text{ если } f \in K_i \\ 1, \text{ если } f \notin K_i \end{cases}, \text{ где } i \in \{1, ..., 11\}.$$

Теорема 1. *Число различных классов эквивалентности в* $P_2^{\sim}$ *равно 45.*

Система ультрафункций *полна* в $P_2^{\sim}$ тогда и только тогда, когда она целиком не содержится ни в одном из 11 максимальных клонов [1].

Множество ультрафункций $B = \{ f_1, , ..., f_m \}$ называется *базисом*, если:
1) наименьший клон, содержащий B, совпадает с $P_2^{\sim}$;
2) при удалении из B любой функции условие «1» нарушается.

Классы эквивалентности можно объединять таким образом, чтобы получающиеся множества ультрафункций представляли собой базис. Тогда имеем возможность перечислить и подсчитать количество всех возможных базисов

ультрафункций по отдельным мощностям, с точностью до принадлежности ультрафункций классам эквивалентности. Соответственно наибольшая полученная при этом мощность будет являться максимальной мощностью базиса.

Теорема 2. *В $P_2^{\sim}$ существует ровно 895 типов базисов: 1 базис мощности 1, 180 базисов мощности 2, 686 базисов мощности 3 и 28 базисов мощности 4.*

Аналогичные результаты для функций k-значной логики можно посмотреть в [2].

Литература

[1] *Пантелеев В. И.* Критерий полноты для доопределяемых булевых функций // Вестник СамГУ. — Естественнонаучная серия — 2005. — № 2(36). — 20с.

[2] *Rosenberg I. G., Miyakawa M., Stojmenovic I., Lau D.* Classifications and basis enumerations in many-valued logics // IEEE. — 1987. — P. 152–160.

Об эквивалентности ограниченно недетерминированных автоматов-преобразователей над полугруппами

В. А. Захаров

zakh@cs.msu.su

Московский государственный университет имени М.В. Ломоносова, Москва

Пусть заданы два конечных множества $\mathcal{A}$ (*входной алфавит*) и $\mathcal{B}$ (*выходной алфавит*). Обозначим записями $\mathcal{A}^*$ и $\mathcal{B}^*$ множество всех конечных слов входного и выходного алфавитов, включая пустое слово. Для произвольной пары конечных слов w_1 и w_2 запись w_1w_2 будет обозначать сцепление (конкатенацию) этих слов.

Конечным автоматом-преобразователем над парой алфавитов $\mathcal{A}, \mathcal{B}$ называется модель вычислений, которая описывается системой переходов $\pi = \langle Q, q_s, F, T\rangle$, компонентами которой является конечнное множество *состояний* Q, *начальное состояние* q_s, $q_s \in Q$, множество *финальных состояний* F, $F \subseteq Q$, и конечное *отношение переходов* T: $T \subseteq Q \times \mathcal{A} \times Q \times \mathcal{B}^*$.

Для заданного слова $\alpha = a_1a_2\dots a_n$ входного алфавита *вычислением* автомата-преобразователя π на слове α называется последовательность четверок

$$(q_1, a_1, q_2, \beta_1), (q_2, a_2, q_3, \beta_3), \dots, (q_i, a_i, q_{i+1}, \beta_i), \dots, (q_n, a_n, q_{n+1}, \beta_n),$$

удовлетворяющая следующим двум требованиям:

1. $q_1 = q_s$ — начальное состояние,
2. $(q_i, a_i, q_{i+1}, \beta_i) \in T$ для любого i, $1 \leqslant i, \leqslant n$.

Если $q_{n+1} \in F$, то указанное вычисление называется *успешным*, а слово $\beta = \beta_1\beta_2\dots\beta_n$ называется *результатом* этого вычисления. Обозначим записью $C(\pi, \alpha)$ множество всех результатов успешных вычислений автомата-преобразователя π на слове α.

Два автомата-преобразователя π_1 и π_2 над алфавитами $\mathcal{A}$ и $\mathcal{B}$ считаются эквивалентными, если равенство $C(\pi_1, \alpha) = C(\pi_1, \alpha)$ выполняется для любого конечного входного слова α.

Автомат-преобразователь $\pi = \langle Q, q_s, F, T\rangle$ называется *детерминированным*, если для любой буквы a, $a \in \mathcal{A}$, и любого состояния q, $q \in Q$, отношение переходов T содержит не более одной четверки вида (q, a, q', β). Для всякого натурального числа k автомат-преобразователь $\pi = \langle Q, q_s, F, T\rangle$ называется *k-недетерминированным*, если для любого конечного входного слова α множество $C(\pi, \alpha)$ содержит не более k выходных слов.

В статье [1] было установлено, что проблема эквивалентности для недетерминированных конечных автоматов-преобразователей неразрешима. В то же время, как было показано в статье [2], проблема эквивалентности для детерминированных конечных автоматов-преобразователей разрешима за полиномиальное время. Такое большое различие в сложности проблемы эквивалентности для детерминированных и недетерминированных автоматов побудило исследователей обратиться к промежуточным классам ограниченно недетерминированных автоматов-преоразователей. В статье [3] было установлено, что для любого заданного k можно за полиномиальное время проверить, является ли заданный автомат-преобразователь k-недетерминированным, а в статье [4] удалось показать, что за полиномиальное время можно проверить существование такого натурального k, для которого заданный автомат-преобразователь является k-недетерминированным. Более простое решение проблемы ограниченоой недетерминированности автоматов-преобразователей было предложено в статье [5]. Далее в статьях [6, 7] двумя разными способами была доказана разрешимость проблемы эквивалентности для ограниченно недетерминированных автоматов-преобразователей; сложность предложенных разрешающих алгоритмов ограничена двойной экспонентой. В статьях [8, 9] удалось построить более простые алгоритмы экспоненциальной сложности проверки эквивалентности для указанного класса автоматов-преобразователей.

В последние годы интерес к последовательным автоматам-преобразователям стали проявлять исследователи в области системного программирования, поскольку некоторые обобщения этого класса автоматов оказались подходящей моделью вычислений для задач верификации системных программ. В статьях [10, 11] рассматривались автоматы-преобразователи над бесконечным входным алфавитом, множества символов которого для каждого отдельного перехода описываются формулами тех идли иных логико-математических теорий. Для этого класса автоматов-преобразователей были разработаны алгоритмы проверки эквивалентности с использованием SMT-решателей — разрешающих процедур проверки выполнимости формул в логико-математических теориях. В статьях [12, 13] автоматы-преобразователи рассматривались как реагирующие программы, вычисляющие по заданному потоку событий (слову из $\mathcal{A}^*$) последовательность действий над данными (слово из $\mathcal{B}^*$). В частности, в модели реагирующей программы, предложенной в статье [13], допускалась возможность того, что разные последовательности действий могут осуществлять одно и то же преобразование данных. Таким образом, слова выходного алфавита представляют собой выражения, определяющие элементы

некоторой полугруппы. В этой же работе был предложен полиномиальный по времени алгоритм проверки эквивалентности конечных детерминированных автоматов-преобразователей над полугруппами слов выходного алфавита, вложимыми в группы с разрешимой проблемой тождеств. В настоящей заметке аннонсируется основной результат дальнейших исследований проблемы эквивалентности автоматов-преобразователей, рассматриваемых как реагирующие программы над полугруппами операторов.

Пусть $\mathcal{B}$ — множество образующих полугруппы S; элемент полугруппы S, соответствующий слову β, $\beta \in \mathcal{B}^*$, условимся обозначать записью $[\beta]$. Конечный автомат-преобразователь π над входным алфавитом $\mathcal{A}$ и полугруппой S с множеством образующих элементов $\mathcal{B}$ называется k-ограниченно недетерминированным, если для любого конечного входного слова α множество $[C(\pi,\alpha)] = \{[\beta] \ : \beta \in C(\pi,\alpha)\}$ содержит не более k элементов полугруппы S. Два автомата-преобразователя π_1 и π_2 над входным алфавитом $\mathcal{A}$ и полугруппой S считаются S-эквивалентными, если для любого конечного входного слова α множества элементов $[C(\pi_1,\alpha)]$ и $[C(\pi_2,\alpha)]$ совпадают.

Теорема 1. *Если полугруппа S вложима в группу с разрешимой проблемой тождеств, то для любого натурального k проблема S-эквивалентности конечных k-ограниченно недетерминированных автоматов-преобразователей разрешима за экспоненциальное время.*

Для построения разрешающего алгоритма был использован один из вариантов метода совместных вычислений, ранее использованный в работе [14].

Работа выполнена при поддержке гранта РФФИ 12-01-00706.

Литература

[1] *Griffiths T. V.* The unsolvability of the equivalence problem for Λ-free nondeterministic generalized machines // Journal of the Assocoation for Computing Machinary. — 1968. — v. 15 — N 2 — p. 109-113.

[2] *Blattner M., Head T.* The decidability of equivalence for deterministic finite transducers // Journal of Computer and System Science. — 1979 — v. 19 — N 1. — p. 45-49.

[3] *Guarari E., Ibarra O.* A note on finite-valued and finitely ambiguous transducers // Mathematical Systsems Theory. — 1983 — v. 16 — p. 61-66.

[4] *Weber M.* On the valuedness of finite transducers // Acta Informatica. — 1989 — v. 27 — N 8. — p. 749-780.

[5] *Sakarovitch J., de Souza R.* On the decidability of bounded valuedness for transducers // Lecture Notes in Computer Science. — 2008. — v. 5516. — p. 588-600.

[6] *Culik K. II, Karhumaki J.* The equivalence of finite valued transducers (On HDT0L languages) is decidable // Theoretical Computer Science. — 1986. — v. 47. — p. 71-84.

[7] *Weber M.* Decomposing finite-velued transducers and deciding their equivalence // SIAM Journal on Computing. — 1993 — v. 22 — N 1. — p. 175-202.

[8] *de Souza R.* On the decidability of the equivalence for k-valued transducers // Lecture Notes in Computer Science. — 2008. — v. 5257. — p. 252-263.

[9] *de Souza R.* On the decidability of the equivalence for a certain class of transducers // Lecture Notes in Computer Science. — 2009. — v. 5583. — p. 478-489.

[10] *Bjorner N., Veanes M.* Symbolic transducers // Technical Report MSR-TR-2011-3, Microsoft Research, January. — 2011.

[11] *Veanes M., Molnar D., Livshits B.* Decision procedures for composition and equivalence of symbolic finite state transducers // Technical Report MSR-TR-2011-32, Microsoft Research, March. — 2011.

[12] *Alur R., Deshmukh J. V.* Nondeterministic streaming string transducers // Lecture Notes in Computer Science. — 2011. — v. 6756. — p. 1-20.

[13] *Захаров В. А.* Об эквивалентности потоковых программ // Материалы XI Международного семинара «Дискретная математика и ее приложения», (Москва, МГУ, 18-23 июня 2012 г.), Изд-во механико-математического ф-та МГУ Москва. — 2012. — с. 119-121. . — 2005. — v. 3317. — p. 293-305.

[14] *Zakharov V. A., Zakharyaschev I. M.* On the equivalence checking problem for a model of programs related with muti-tape automata // Lecture Notes in Computer Science. — 2005. — v. 3317. — p. 293-305.

Задача редактирования для симметрических линейных пространств графов

Д. В. Захарова

dvzakh@rambler.ru

Нижегородский государственный университет им. Н.И. Лобачевского

В задаче редактирования относительно класса графов $\mathbf{X}$ требуется заданный граф превратить в граф из $\mathbf{X}$, изменяя (удаляя и добавляя) наименьшее число ребер. Для класса $\mathbf{X}$, состоящего из графов, у которых каждая компонента связности является кликой (такие графы называют кластерными), она известна также как задача кластеризации и является NP-полной [1]. В [1] доказана также ее NP-полнота при фиксированном числе кластеров $p \geqslant 2$, а для $p = 2$ (двукластерные графы) предложен приближенный алгоритм с константной оценкой мультипликативной точности. Рассмотрим следующий эвристический алгоритм для задачи двукластерного редактирования.

Алгоритм А. Множество вершин графа произвольным образом разбивается на два подмножества и вычисляется расстояние от данного графа до двукластерного графа, в котором эти подмножества являются кликами. Затем рассматриваются все графы, получаемые переносом одной вершины из одного множества в другое, и для каждого из них тоже вычисляется соответствующее расстояние. Если для какой-то вершины оно оказывается меньше исходного расстояния, то эта вершина действительно переносится. Это повторяется, пока переносы не прекратятся.

Алгоритм А нетрудно реализовать так, чтобы время его работы оценивалось как $O(n^3)$.

Теорема 1. *Если граф G отличается от двукластерного не более чем на $\lfloor n/2 \rfloor - 1$ ребер, то алгоритм А найдет ближайший к G двукластерный граф.*

Дополнительным к двукластерному графу является полный двудольный граф. Это позволяет распространить алгоритм А и теорему 1 на полные дву-

дольные графы. Множество всех полных двудольных графов с фиксированным множеством вершин образует симметрическое линейное пространство графов (СЛПГ). Это означает, что оно замкнуто относительно изоморфизма и симметрической разности графов. При любом фиксированном множестве вершин имеется не более 14 СЛПГ, все они описаны в [2]. Для некоторых из них задача редактирования тривиальна, для других эффективно решается, для третьих NP-трудна. К последней категории относятся следующие СЛПГ:

— $\mathbf{L_1}$ - все полные двудольные графы и все двукластерные графы,
— $\mathbf{L_2}$ - графы из $\mathbf{L_1}$ с четными степенями всех вершин,
— $\mathbf{L_3}$ - полные двудольные графы с четными степенями и двукластерные графы с нечетными степенями всех вершин,
— $\mathbf{L_4}$ - все полные двудольные графы,
— $\mathbf{L_5}$ - графы из $\mathbf{L_4}$ с четными степенями всех вершин.

Для этих пяти классов на базе алгоритма A можно предложить полиномиальные по времени алгоритмы редактирования, дающие точные решения, когда входной граф достаточно близок к целевому классу. Пусть d - наименьшее различие между графами в данном классе. Тогда эти алгоритмы дают оптимальное решение во всех случаях, когда входной граф находится на расстоянии не более $\lfloor (d-1)/2 \rfloor$ от целевого класса. Это аналогично декодированию помехоустойчивых кодов, исправляющему все ошибки, исправление которых гарантируется кодовым расстоянием.

Для всех остальных СЛПГ задача редактирования решается за линейное время.

Литература

[1] *Shamir R., Sharan R., Tsur D.* Cluster graph modification problems // Discrete Appl. Math. — 2004. — V. 144, — P. 173–182.

[2] *Захарова Д. В.* Симметрические линейные пространства графов // Дискретная математика. — 2011. — Т. 23, № 2. — С. 103–1067.

О некоторых видах композиции квантовых хэш–генераторов

М. Т. Зиятдинов

gltronred@gmail.com

КФУ, Казань

Введение

В статье [1] вводятся *δ–устойчивые $(K, d+\ell)$ квантовые генераторы*, позволяющие построить $\delta+\varepsilon$–устойчивую квантовую хэш–функцию на основе ε–универсального семейства хэш–функций:

Определение 1 (Квантовый хэш–генератор). *Пусть $K = |\mathbb{X}|$ и $G = \{g_1, \ldots, g_D\}$ является семейством функций $g_j : \mathbb{X} \to \mathbb{F}_q$. Пусть $\ell \geqslant 1$. Для*

$g \in G$ через ψ_g обозначим классическую–квантовую функцию $\psi_g : \mathbb{X} \to (\mathcal{H}^2)^\ell$, определяемую правилом

$$\psi_g : w \mapsto |\psi_g(w)\rangle = \sum_{i=1}^{\ell} \alpha_i(g(w))|i\rangle. \tag{1}$$

Пусть $d = \log D$. Определим классическую–квантовую функцию $\psi_G : \mathbb{X} \to (\mathcal{H}^2)^{\otimes(d+\ell)}$ по правилу

$$\psi_G : w \mapsto |\psi_G(w)\rangle = \frac{1}{\sqrt{D}} \sum_{j=1}^{D} |j\rangle|\psi_{g_j}(w)\rangle.$$

Будем говорить, что G генерирует δ–устойчивую $(K; d+\ell)$ квантовую хэш–функцию ψ_G и будем называть G δ–устойчивым $(K; d+\ell)$ квантовым хэш–генератором, если ψ_G является δ–устойчивой $(K; d+\ell)$ квантовой хэш–функцией.

Композиция квантовых хэш–генераторов

На основе двух квантовых хэш–генераторов с соответствующими параметрами можно построить новый хэш–генератор, обладающий большей устойчивостью. Доказательству этого посвящены следующие два свойства.

Утверждение 1. *Пусть $G_1 = \{g_{11}, \ldots, g_{1D}\}$ и $G_2 = \{g_{21}, \ldots, g_{2D}\}$ являются δ_1–устойчивым $(K, d+\ell)$ и δ_2–устойчивым $(K, d+\ell)$ квантовыми хэш–генераторами соответственно.*

Пусть, кроме того, в генераторах в правилах (1) используются одни и те же функции α_i, являющиеся гомоморфизмами $\mathbb{F}_q$ в $\mathcal{H}^2$

Тогда $G = \{g_1, \ldots, g_D\}$ является δ–устойчивым $(K, d+\ell)$ квантовым хэш–генератором, где $g_j(x) = g_{1j}(x) + g_{2j}(x)$ и $\delta = D\delta_1\delta_2$

Доказательство. Пусть w и w' являются различными словами. Тогда

$$\begin{aligned} |\langle\psi(w)|\psi(w')\rangle| &= \frac{1}{D}\left|\sum_{j=1}^{D}\langle j|j\rangle \sum_{i=1}^{\ell} \alpha_i^{-1}(g_j(w))\alpha_i(g_j(w'))\langle i|i\rangle\right| = \\ &= \frac{1}{D}\left|\sum_{j=1}^{D}\sum_{i=1}^{\ell} \alpha_i\big(g_j(w') - g_j(w)\big)\right|, \end{aligned}$$

Подставив определения g_j и произведя простые преобразования, получаем:

$$|\langle\psi(w)|\psi(w')\rangle| = \frac{1}{D}\left|\sum_{i=1}^{\ell}\sum_{j=1}^{D} \alpha_i\big(g_{1j}(w') - g_{1j}(w)\big) \times \alpha_i\big(g_{2j}(w') - g_{2j}(w)\big)\right|$$

Применяем неравенство Коши–Буняковского и замечаем, что в правой части находится произведение соответствующих оценок для генераторов G_1 и G_2, поэтому $|\langle\psi(w)|\psi(w')\rangle| \leqslant D\delta_1\delta_2$

∎

Утверждение 2. *Пусть* $G_1 = \{g_{11}, \ldots, g_{1D_1}\}$ *и* $G_2 = \{g_{21}, \ldots, g_{2D_2}\}$ *являются* δ_1*–устойчивым* $(K, d_1 + \ell_1)$ *и* δ_2*–устойчивым* $(K, d_2 + \ell_2)$ *квантовыми хэш–генераторами, соответственно.*

Тогда $G = \{g_1, \ldots, g_D\}$ *является* δ*–устойчивым* $(K, d_1 + d_2 + \ell_1 + \ell_2)$ *квантовым хэш–генератором, где* $g_j(x) = g_{1j_1}(x)q + g_{2j_2}(x)$, $\alpha_i(xq+y) = \alpha_{i1}(x) + \alpha_{i2}(y)$, *все операции выполняются в поле* $\mathbb{F}_{q^2}$, $D = D_1 D_2$ *и* $\delta = \delta_1\delta_2$

Доказательство. Пусть w и w' — различные слова. Тогда

$$\begin{aligned} |\langle\psi(w)|\psi(w')\rangle| &= \frac{1}{D}\left|\sum_{j=1}^{D}\langle j|j\rangle\sum_{i=1}^{\ell}\alpha_i^{-1}(g_j(w))\alpha_i(g_j(w'))\langle i|i\rangle\right| = \\ &= \frac{1}{D}\left|\sum_{j=1}^{D}\sum_{i=1}^{\ell}\alpha_i\big(g_j(w') - g_j(w)\big)\right|. \end{aligned}$$

Заменив g_j и α_i определениями и преобразуя, получаем, что $|\langle\psi(w)|\psi(w')\rangle|$ равно

$$\frac{1}{D_1 D_2}\left|\sum_{j_1,j_2=1}^{D_1,D_2}\sum_{i_1,i_2=1}^{\ell_1,\ell_2}\alpha_{i1}\big(g_{1j_1}(w') - g_{1j_1}(w)\big)\times\alpha_{i2}\big(g_{2j_2}(w') - g_{2j_2}(w)\big)\right|.$$

Применяем неравенство Коши–Буняковского и получаем требуемое неравенство $|\langle\psi(w)|\psi(w')\rangle| \leqslant \delta_1\delta_2$ ∎

Литература

[1] *Ablayev F. M., Vasiliev A. V.* Cryptographic Quantum Hashing // Laser Physics Letters. — 2014. — V. 11, № 2. — P. .202–205

Об одном способе квантового хэширования. Групповой подход

М. Т. Зиятдинов

gltronred@gmail.com

КФУ, Казань

Общее описание

Рассмотрим схему хэширования из [1] и обобщим её на случай произвольных групп, что позволит применять её в большем количестве случаев и с различными соотношениями параметров.

Пусть у нас имеется некоторая (произвольная конечная) группа G с групповой операцией $\circ$ и единицей e. Пусть у нас имеется также гомоморфизм $f : G \to [(\mathcal{H}^2)^{\otimes m} \to (\mathcal{H}^2)^{\otimes m}]$

Определение 1. *По аналогии с [1] назовем «хорошим» набор элементов K_{good} множества автоморфизмов $\mathrm{Aut}(G)$, если для любого неединичного элемента g группы G выполняется:*

$$\forall g \in G, g \neq e : \frac{1}{|K_{\text{good}}|^2} \left| \sum_{k \in K_{\text{good}}} \langle \psi_0 | f(k\{g\}) | \psi_0 \rangle \right|^2 < \varepsilon \tag{1}$$

В [2] доказано, что для группы $\mathbb{Z}_q$ «хорошее» множество автоморфизмов существует.

Определение 2. *Определим квантовую хэш–функцию на основе классической хэш–функции h из X^n в группу G, «хорошего» набора автоморфизмов $K = \{k_0, \ldots, k_{t-1}\}$ и гомоморфизма f в пространство $[(\mathcal{H}^2)^{\otimes m} \to (\mathcal{H}^2)^{\otimes m}]$ как*

$$|\Psi_{h,G,K,f,m}(x)\rangle = \frac{1}{\sqrt{t}} \sum_{j=0}^{t-1} \Big(|j\rangle \otimes f\big(k_j\{h(x)\}\big) |\psi_0\rangle \Big). \tag{2}$$

Для практического применения квантовой хэш–функции необходимо, чтобы она обладала квантовым аналогом криптографических свойств классической хэш–функции — δ–устойчивостью — определенной в [1]. Покажем, что справедливо

Утверждение 1. *Введённая хэш–функция является δ–устойчивой.*

Доказательство. Рассмотрим квадрат скалярного произведения значений квантовой хэш–функции на разных входных наборах:

$$|\langle \Psi_{h,G,K,f,m}(x) | \Psi_{h,G,K,f,m}(x') \rangle|^2 = \left| \frac{1}{t} \sum_{j=0}^{t-1} \langle \psi_0 | f(k_j\{h^{-1}(x) \circ h(x')\}) | \psi_0 \rangle \right|^2$$

В случае, если коллизии хэш–функции h нет, и $h(x) \neq h(x')$, произведение $h(x') \circ h(x)^{-1}$ будет равно какому–то элементу $g \neq e$ группы G, и по свойству «хорошего» подмножества K_{good} квадрат скалярного произведения будет равен ε. В противном случае, он будет равен единице.

∎

Пример: группа $\mathbb{Z}_q$. Схему хэширования, предложенную в [1], можно рассматривать как частный случай описываемой схемы. Опишем этот факт в виде

Лемма 2. *Существует квантовая хэш–функция вида (2) для группы $\mathbb{Z}_q$.*

Пример: группа $G_1 \times G_2$. Элементами группы $G_1 \times G_2$ являются пары элементов соответствующих групп (g_1, g_2), с операцией, заданной покомпонентно. Единицей группы является пара (e_1, e_2), состоящая из единиц соответствующих групп.

Лемма 3. *Если существует квантовая хэш–функция для групп G_1, G_2 в пространствах $(\mathcal{H}^2)^{\otimes m_1}$ и $(\mathcal{H}^2)^{\otimes m_2}$ с гомоморфизмами f_1 и f_2 и «хорошими» множествами автоморфизмов $\mathbb{K}_1$ и $\mathbb{K}_2$, соответственно, то мы можем построить квантовую хэш–функцию для $G_1 \times G_2$ в пространстве $(\mathcal{H}^2)^{\otimes (m_1+m_2)}$.*

Доказательство. В качестве гомоморфизма выберем покомпонентное отображение, а в качестве автоморфизмов — всевозможные покомпонентные комбинации автоморфизмов исходных групп. ∎

Пример: произвольные абелевы группы.

Теорема 4. *Для произвольной абелевой группы G существует квантовая хэш–функция вида (2).*

Доказательство. Поскольку произвольная абелева группа G по аналогии с китайской теоремой об остатках для целых чисел может быть представлена в виде

$$G = \mathbb{Z}_{p_1^{\sigma_1}} \otimes \cdots \otimes \mathbb{Z}_{p_t^{\sigma_t}},$$

мы можем применить лемму 2 для того, чтобы построить квантовые хэш–функции в каждой из групп $\mathbb{Z}_{p_1^{\sigma_1}}, \ldots, \mathbb{Z}_{p_t^{\sigma_t}}$, а затем лемму 3, чтобы объединить эти хэш–функции в хэш–функцию для группы G. ∎

О построении «хороших» подмножеств автоморфизмов

Пусть выбраны группа G и гомоморфизм $f : G \to [(\mathcal{H}^2)^{\otimes m} \to (\mathcal{H}^2)^{\otimes m}]$.
Тогда справедлива

Теорема 5. *Если существует множество автоморфизмов $\mathbb{K}$ из группы всех автоморфизмов $\mathrm{Aut}(G)$: $\mathbb{K} \subseteq \mathrm{Aut}(G)$, такое, что для любого элемента группы G среднее значение модуля гомоморфного образа элемента по всем выбранным автоморфизмам равно нулю:*

$$\forall g \in G : \mathbf{E}_{k \in \mathbb{K}} \Big[\langle \psi_0 | f(k\{g\}) | \psi_0 \rangle \Big] = 0,$$

то «хорошее» множество K мощности $|K| \geqslant \frac{2}{\varepsilon} \ln |G|$ можно построить с вероятностью $1/|G|$, выбирая элементы из $\mathbb{K}$ случайным образом.

Доказательство. Мы хотим показать, что для случайного подмножества K (каждый элемент выбирается случайным образом из $\mathbb{K}$) вероятность получить «плохое» для какого–то g множество в смысле уравнения (1) мало.

Рассмотрим случайные величины $X_i = \langle \psi_0 | f(k_i\{g\}) | \psi_0 \rangle$, где $K = \{k_1, k_2, \ldots, k_t\}$, и $Y_t = \sum_{i=1}^{t} X_i$.

Покажем, что последовательность случайных величин $Y_0 = 0, Y_1, Y_2, \ldots, Y_{|K|}$ является мартингалом с ограниченными разностями.

Применим теперь неравенство Азумы:

$$\mathsf{P}\left(|Y_{|K|} - Y_0| > \lambda\right) = \mathsf{P}\left(\Big|\sum_{i=1}^{|K|} X_i\Big| > \lambda\right) \leqslant 2 \exp\left\{-\frac{\lambda^2}{2|K|}\right\} \tag{3}$$

Положим $\lambda = \sqrt{\varepsilon}|K|$, тогда неравенство Азумы (3) примет вид:

$$\mathsf{P}\left(\Big|\sum_{i=1}^{|K|} X_i\Big| > \sqrt{\varepsilon}|K|\right) \leqslant 2 \exp\left\{-\frac{\varepsilon |K|^2}{2}\right\} \leqslant \frac{1}{|G|}$$

при $|K| \geqslant \frac{2}{\varepsilon} \ln |G|$.

Это неравенство означает, что множество K не является «хорошим» для *некоторого* $g \neq e$. Поэтому с вероятностью $1/|G|$ существует «хорошее» множество автоморфизмов K. ∎

Для построения множества $|K|$ можно также применить различные эвристические методы, такие как генетические алгоритмы, метод отжига и т.п. Для построения «хороших» множеств автоморфизмов группы $\mathbb{Z}_q$ подобные методы были применены в [2].

Ограничения на размерность подпространств и группу в квантовой хэш–функции

В [3] описана классификация конечных унитарных групп, генерируемых отражениями (у.г.г.о.)

Пусть задана квантовая хэш–функция $|\Psi_{h,G,K,f,m}(x)\rangle$. В силу классификации Шепарда–Тодда справедлива

Теорема 6. *В произвольной квантовой хэш–функции* $|\Psi_{h,G,K,f,m}(x)\rangle$ *группа* G *является подгруппой некоторой у.г.г.о. в унитарном пространстве* $(\mathcal{H}^2)^{\otimes m}$; *и напротив, размерность пространства* m *не может быть меньше, чем размерность пространства, в котором возможна у.г.г.о., подгруппой которой является* G.

Литература

[1] *Ablayev F. M., Vasiliev A. V.* Quantum Hashing // `http://arxiv.org/abs/1310.4922`

[2] *Ablayev F. M., Vasiliev A. V.* Algorithms for quantum branching programs based on fingerprinting // Electronic Proceedings in Theoretical Computer Science. — 2009. — V. 9. — P. 1–11.

[3] *Shephard G. C., Todd J. A.* Finite unitary reflection groups // Canadian J. Math. — 1954. — V. 6, — P. 274–304.

Сложность расшифровки пороговой функции

Н. Ю. Золотых, А. Ю. Чирков

`zolotykh@vmk.unn.ru, chirkov@vmk.unn.ru`

Нижегородский государственный университет им. Н. И. Лобачевского

Пусть $E_k = \{0, 1, \ldots, k-1\}$, $k \geqslant 2$. Функция $f : E_k^n \to \{0, 1\}$ называется *пороговой*, если существуют числа $a_0, a_1, \ldots, a_n$, такие, что

$$f^{-1}(1) = \left\{ x = (x_1, x_2, \ldots, x_n) \in E_k^n : \sum_{j=1}^{n} a_j x_j \leqslant a_0 \right\}. \qquad (1)$$

Обозначим $\mathcal{T}(n, k)$ множество всех пороговых функций, заданных на E_k^n. Множество $T \subseteq E_k^n$ называется *разрешающим* для функции $f \in \mathcal{T}(n, k)$, если для

любой функции $g \in \mathcal{T}(n,k) \setminus \{f\}$ найдется точка $z \in T$, такая, что $f(z) \neq g(z)$. Разрешающее множество функции f называется *минимальным*, или *тупиковым*, если никакое его собственное подмножество не является разрешающим для функции f. Известно, что для любой пороговой функции f минимальное разрешающее множество единственно. Минимальное разрешающее множество функции f обозначим $T(f)$. *Длиной обучения* (в классе пороговых функций) называется величина

$$\sigma(n,k) = \max_{f \in \mathcal{T}(n,k)} |T(f)|.$$

Заметим, что $\sigma(n,k)$ зависит от n экспоненциально, в частности, $\sigma(n,2) = 2^n$, поэтому представляет интерес поиск оценок для $\sigma(n,k)$, когда n фиксировано. В [1, 2] доказано, что $\sigma(2,k) = 4$. В [3, 4] получена верхняя оценка $\sigma(n,k) = O(\log^{n-2} k)$ при $n \geqslant 2$. В [5, 1] установлена нижняя оценка $\sigma(n,k) = \Omega(\log^{n-2} k)$. Таким образом, при любом фиксированном $n \geqslant 2$

$$\sigma(n,k) = \Theta(\log^{n-2} k) \tag{2}$$

(здесь и далее все асимптотические оценки даны в предположении, что n фиксировано, а $k \to \infty$).

Пусть с каждой функцией $f \in \mathcal{T}(n,k)$ связан оракул, позволяющий по произвольной точке $x \in E_k^n$ получить значение $f(x)$. Под *расшифровкой* заранее не известной пороговой функции f понимается восстановление коэффициентов $a_0, a_1, \ldots, a_n$, для которых выполнено (1). Пусть $\mathcal{A}$ — алгоритм расшифровки, $\tau(\mathcal{A}, f)$ — число обращений к оракулу при расшифровке алгоритмом $\mathcal{A}$ функции f. *Сложностью алгоритма* $\mathcal{A}$ назовем

$$\tau(\mathcal{A}) = \max_{f \in \mathcal{T}(n,k)} \tau(\mathcal{A}, f).$$

Сложностью задачи расшифровки пороговой функции назовем

$$\tau(n,k) = \min_{\mathcal{A}} \tau(\mathcal{A}),$$

где минимум берется по всем алгоритмам $\mathcal{A}$ расшифровки функций в классе $\mathcal{T}(n,k)$.

Легко видеть, что

$$\sigma(n,k) \leqslant \tau(n,k).$$

Учитывая (2), при $n \geqslant 2$ получаем

$$\tau(n,k) = \Omega(\log^{n-2} k). \tag{3}$$

В [6, 7] предлагается алгоритм расшифровки $\mathcal{A}'$ в классе $\mathcal{T}(n,k)$, для которого

$$\frac{\tau(\mathcal{A}')}{\sigma(n,k)} = O(\log k). \tag{4}$$

В [1, 8] предлагается алгоритм расшифровки $\mathcal{A}''$ в классе $\mathcal{T}(2,k)$, для которого $\tau(\mathcal{A}'') = 6\log(k-1)+4$. Учитывая (2), при $n \geqslant 2$ из (4) получаем $\tau(\mathcal{A}') = O(\log^{n-1} k)$. Таким образом, при $n \geqslant 2$

$$\tau(n,k) = O(\log^{n-1} k).$$

Эту оценку удалось улучшить.

Теорема 1. *Существует алгоритм $\mathcal{A}$ расшифровки функций из класса $\mathcal{T}(n,k)$, для которого при любом фиксированном $n \geqslant 3$*

$$\tau(\mathcal{A}) = O(\log^{n-2} k).$$

Учитывая нижнюю оценку (3), получаем при любом фиксированном $n \geqslant 3$

$$\tau(n,k) = \Theta(\log^{n-2} k)$$

Литература

[1] *Шевченко В. Н., Золотых Н. Ю.* О сложности расшифровки пороговых функций k-значной логики // Доклады Академии наук. — 1998. — Т. 362, № 5. — С. 606–608.

[2] *Золотых Н. Ю.* О сложности расшифровки пороговых функций, зависящих от двух переменных // Материалы XI Межгосударственной школы-семинара «Синтез и сложность управляющих систем». Часть I. — М.: Изд-во Центра прикладных исследований при механико-математическом ф-те МГУ, 2001. — С. 74–79.

[3] *Золотых Н. Ю., Чирков А. Ю.* Сложность расшифровки пороговых функций многозначной логики // Материалы XI Международного семинара «Дискретная математика и ее приложения», посвященного 80-летию со дня рождения академика О. Б. Лупанова (18–23 июня 2012 г.) / Под редакцией О.М. Касим-Заде. — М.: Изд-во механико-матем. факультета МГУ. — 2012. — С. 63–77.

[4] *Chirkov A. Yu., Zolotykh N. Yu.* On the number of irreducible points in polyhedra. arXiv:1306.4289 — 2013.

[5] *Золотых Н. Ю., Шевченко В. Н.* О нижней оценке расшифровки пороговых функций k-значной логики // Журнал вычислительной математики и математической физики. — 1999. — Т. 39, № 2. — С. 346–352.

[6] *Hegedüs T.* Generalized teaching dimensions and the query complexity of learning // Proc. 8th Ann. ACM Conf. on Computational Learning Theory (COLT'95). — New York: ACM Press, 1995. — P. 108–117.

[7] *Золотых Н. Ю., Шевченко В. Н.* Расшифровка пороговых функций k-значной логики // Дискретный анализ и исследование операций. — 1995. — Т. 2, № 3. — С. 18–23.

[8] *Золотых Н. Ю.* Пороговые функции, зависящие от двух переменных: сложность расшифровки и мощность разрешающего множества // Материалы 4-й молодежной научной школы по дискретной математике и ее приложениям. — М.: Изд-во мех.-мат. фак. МГУ, 2000. — С. 48–54.

Кибернетическая модель циклического управления конфликтными потоками с последействием

А. В. Зорин

zoav1602@gmail.com

Нижегородский государственный университет им. Н. И. Лобачевского — национальный исследовательский университет, Нижний Новгород

Пусть в систему массового обслуживания поступают конфликтные независимые входные потоки Π_1, Π_2, ..., Π_m, $m < \infty$. Требования каждого потока могут быть одного из двух типов: «быстрые» и «медленные». «Быстрые» требования могут образовывать группы, следующие за «медленными» требованиями. Поэтому в реальных потоках интервалы между требованиями, как правило, зависимы и разнораспределены. При этом, статистический анализ реальных транспортных потоков [1] позволяет предположить, что «медленные» требования образуют рекуррентный поток, а размеры групп независимы и одинаково распределены. Пусть $a_j(t)$, $t \geqslant 0$ — общая плотность распределения интервалов между «медленными» требованиями, $f_j(b)$ — вероятность поступления в группе $b = 1, 2, \ldots$ требований по потоку Π_j, $j = 1, 2, \ldots, m$. Требования потока Π_j помещаются в накопитель O_j неограниченного объема. Обслуживание требований осуществляется единственным прибором с $2m$ состояниями $\Gamma^{(1)}$, $\Gamma^{(2)}$, ..., $\Gamma^{(2m)}$. В состоянии $\Gamma^{(2j-1)}$ обслуживаются требования только из очереди O_j. В состоянии $\Gamma^{(2j)}$ требования не обслуживаются и осуществляется акт переналадки с целью разрешения конфликтности. Длительность пребывания в состоянии $\Gamma^{(r)}$ равна T_r, $r = 1, 2, \ldots, 2m$. Обслуживание конфликтных потоков осуществляется в классе циклических алгоритмов, то есть смена состояния обслуживающего устройства происходит по схеме $\ldots \to \Gamma^{(1)} \to \Gamma^{(2)} \to \ldots \to \Gamma^{(2m)} \to \Gamma^{(1)} \to \ldots$. Длительности обслуживания требований могут быть зависимыми величинами с различными законами распределения. Поэтому для задания процесса обслуживания используем потоки насыщения $\Pi_1^{\text{нас}}$, $\Pi_2^{\text{нас}}$, ..., $\Pi_m^{\text{нас}}$. При состоянии прибора $\Gamma^{(2j-1)}$ поток насыщения $\Pi_j^{\text{нас}}$ содержит ℓ_j требований, а при прочих состояниях — 0 требований. Требования из очереди O_j обслуживаются в порядке поступления. Для построения и анализа математической модели данной конфликтной системы массового обслуживания будет применен кибернетический подход [2, 3].

Определим схему конфликтной управляющей системы массового обслуживания, выявим информацию, функциональные и статистические связи между блоками. Схема системы содержит следующие блоки: 1) внешняя среда с одним состоянием; 2) входные полюса первого типа — входные потоки Π_1, Π_2, ..., Π_m; 3) входные полюса второго типа — потоки насыщения $\Pi_1^{\text{нас}}$, $\Pi_2^{\text{нас}}$, ..., $\Pi_m^{\text{нас}}$; 4) внешняя память — накопители O_1, O_2, ..., O_m; 5) устройства по переработке информации внешней памяти — устройства по организации дисциплины очередей; 6) внутренняя память — обслуживающее устройство; 7) устройство переработки информации во внутренней памяти — граф сме-

ны состояния обслуживающего устройства; 8) выходные полюса — выходные потоки $\Pi_1^{\text{вых}}, \Pi_2^{\text{вых}}, \ldots, \Pi_m^{\text{вых}}$.

В качестве дискретной временно́й шкалы моментов наблюдения за системой выберем моменты $\tau_0 = 0$, τ_1, τ_2, ... смены состояний обслуживающего устройства. Пусть $\Gamma_0 \in \{\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(2m)}\} = \Gamma$ есть состояние обслуживающего устройства в момент 0, $\Gamma_i \in \Gamma$ обозначает состояние обслуживающего устройства на промежутке $(\tau_{i-1}, \tau_i]$. Положим $r \oplus 1 = r + 1$ при $r < 2m$, $2m \oplus 1 = 1$. Определим отображение $v(\cdot)\colon \Gamma \to R$ равенством $v(\Gamma^{(r)}) = T_{r\oplus 1}$. Тогда последовательность моментов наблюдения порождается рекуррентным соотношением $\tau_{i+1} = \tau_i + v(\Gamma_i)$, $i = 0, 1, \ldots$. Пусть $\zeta_{j,i}$ — оставшееся время до поступления следующей после τ_i группы требований потока Π_j. Выберем $\nu_i = (\Gamma_i, \zeta_{j,i})$ в качестве метки требований потока Π_j, поступающих на промежутке $(\tau_i, \tau_{i+1}]$. Пусть $\eta_{j,i}$ — число поступивших за промежуток $(\tau_i, \tau_{i+1}]$ требований потока Π_j, $\varphi_j(b, y; \gamma, t)$ — условная вероятность события $\eta_{j,i} = b$, $\zeta_{j,i+1} < y$ при условии $\Gamma_i = \gamma$, $\zeta_{j,i} = t$. Нелокальное описание входного полюса Π_j первого типа зададим условными распределениями $\{\varphi_j(b, \infty; \gamma, t); b = 0, 1, \ldots\}$, $\gamma \in \Gamma$, $t \geqslant 0$ выделенной дискретной компоненты $\{\eta_{j,i}; i = 0, 1, \ldots\}$ маркированного точечного процесса $\{(\tau_i, \eta_{j,i}, \nu_i); i = 0, 1, \ldots\}$. Из предположения о вероятностной структуре входного потока Π_j следует, что интегральное преобразование $\sum_{b=0}^{\infty} \int_0^{\infty} z^b e^{-sy}\, d_y \varphi_j(b, y; \Gamma^{(r)}, t)$ принимает вид $e^{-s(t - T_{r\oplus 1})}$, при $t > T_{r\oplus 1}$, и принимает вид $\int_0^{\infty} \sum_{b=0}^{\infty} (f_j(z))^{b+1} e^{-sy}\, d_y G_b(T_{r\oplus 1} - t, y)$ при $t \geqslant T_{r\oplus 1}$, где $a_j^{*1}(t) = a_j(t)$, $a_j^{*(b+1)}(t) = \int_0^t a^{*b}(s) a_j(t-s)\, ds$, $b = 1, 2, \ldots$, и $G_b(t, y)$ принимает значение $\int_t^{t+y} a_j(s)\, ds$ при $b = 0$, и значение $\int_0^t a_j^{*b}(s) \big(\int_{t-s}^{t+y-s} a_j(s_1)\, ds_1 \big)\, ds$ при $b = 1, 2, \ldots$. Для нелокального описания входного полюса $\Pi_j^{\text{нас}}$ второго типа выберем $\nu_i' = \Gamma_i$ в качестве метки требований потока насыщения $\Pi_j^{\text{нас}}$ на промежутке $(\tau_i, \tau_{i+1}]$. Пусть $\xi_{j,i}$ — число требований потока насыщения $\Pi_j^{\text{нас}}$ на промежутке $(\tau_i, \tau_{i+1}]$. Условное распределение $\bar{\varphi}_j(\cdot; \gamma)$ величины $\xi_{j,i}$ при фиксированном значении метки $\nu_i' = \gamma$ задается равенствами $\bar{\varphi}_j(\ell_j; \Gamma^{(2j-2)}) = \bar{\varphi}_j(0; \Gamma^{(r)}) = 1$ при $r \neq 2j - 2$, $\bar{\varphi}_j(b; \gamma) = 0$ в остальных случаях. Введем отображение $u(\cdot)\colon \Gamma \to \Gamma$ равенствами $u(\Gamma^{(r)}) = \Gamma^{(r\oplus 1)}$. Тогда динамика обслуживающего устройства описывается рекуррентным по $i = 0, 1, \ldots$ соотношением $\Gamma_{i+1} = u(\Gamma_i)$. Пусть $\varkappa_{j,i}$ задает состояние блока внешней памяти и число требований в очереди O_j. При экстремальной стратегии обслуживания изменение состояния внешней памяти происходит по закону $\varkappa_{j,i+1} = \max\{0, \varkappa_{j,i} + \eta_{j,i} - \xi_{j,i}\}$. Выберем в качестве основной анализируемой характеристики состояние обслуживающего устройства (внутреннюю память), длины очередей (внешнюю память) и метки входных полюсов. Введем множества $R_+^m = \{(t_1, t_2, \ldots, t_m)\colon t_1 \geqslant 0, t_2 \geqslant 0, \ldots, t_m \geqslant 0\}$, $S = \Gamma \times X \times R_+^m$. Пусть $\mathfrak{S}$ — наименьшая σ-алгебра, содержащая множества вида $\{(\gamma, x, y)\colon y < y^1\}$, $\gamma \in \Gamma$, $x \in X$, $y^1 \in R_+^m$. Ниже $\bar{0} \in X$ — нулевой вектор из X. Используемые ниже при формулировании основных результатов работы математические понятия из теории общих цепей Маркова определены в [4].

Теорема 1. *Пусть $P_0(\cdot)$ — распределение вероятностей на $(S, \mathfrak{S})$. Существует вероятностное пространство $(\Omega, \mathfrak{F}, \mathbf{P})$, на котором определена случайная последовательность $\{(\zeta_i(\omega), \varkappa_i(\omega), \eta_i(\omega), \xi_i(\omega), \Gamma_i(\omega); i = 0, 1, \ldots\}$, с век-*

торами $\zeta_i(\omega) = (\zeta_{1,i}(\omega), \ldots, \zeta_{m,i}(\omega))$, $\varkappa_i(\omega) = (\varkappa_{1,i}(\omega), \ldots, \varkappa_{m,i}(\omega))$, $\eta_i(\omega) = (\eta_{1,i}(\omega), \ldots, \eta_{m,i}(\omega))$, $\xi_i(\omega) = (\xi_{1,i}(\omega), \ldots, \xi_{m,i}(\omega))$, *так что: 1) для* σ*-алгебры* $\widehat{\mathfrak{F}}_i$, *порожденной набором случайных элементов* $\Gamma_0(\omega)$, $\varkappa_0(\omega)$, $\eta_t(\omega)$, $\xi_t(\omega)$, $\zeta_t(\omega)$, $t = 0, 1, \ldots, i-1$ *и* $\zeta_i(\omega)$ *вариант условной вероятности* $\mathsf{P}(\{\omega\colon \eta_i(\omega) = b^1, \xi_i(\omega) = b^2, \zeta_{i+1}(\omega) < y\}|\widehat{\mathfrak{F}}_i)$ *имеет вид* $\prod_{j=1}^m \bar{\varphi}_j(b_j^2; \Gamma_i(\omega)) \times \varphi_j(b_j^1, y_j; \Gamma_i(\omega), \zeta_i(\omega))$; *2) выполняются рекуррентные соотношения между случайными элементами* $\varkappa_{j,i}(\omega)$, $\eta_{j,i}(\omega)$, $\xi_{j,i}(\omega)$ *и* $\Gamma_{i+1}(\omega)$, $\varkappa_{j,i+1}(\omega)$; *3) вектор* $(\Gamma_0(\omega), \varkappa_0(\omega), \zeta_0(\omega))$ *имеет распределение* $P_0(\cdot)$. *Последовательность*

$$\{(\Gamma_i, \varkappa_i, \zeta_i); i = 0, 1, \ldots\}$$

является однородной общей цепью Маркова с переходной вероятностью $\tilde{P}(\cdot,\cdot)\colon S \times \mathfrak{S} \to [0,1]$, *определяемой на паре* $(\Gamma^{(r)}, x, y^0) \in S$, $\{(\Gamma^{(s)}, w, y)\colon y < y^1\}) \in \mathfrak{S}$ *формулой* $\sum_{b=0}^{\ell_j - x} \varphi_j(b, y_j^1; r, y_j^0) \prod_{l \neq j} \varphi_l(w_l - x_l, y_l^1; r, y_l^0)$ *для* $r = 2j-2$, $s = r \oplus 1$, $w = \bar{0}$; *формулой* $\varphi_j(w_j + \ell_j - x_j, y_j^1; r, y_j^0) \prod_{l \neq j} \varphi_l(w_l - x_l, y_l^1; r, y_l^0)$ *для* $r = 2j-2$, $s = r \oplus 1$, $w \neq \bar{0}$; *формулой* $\prod_{l=1}^m \varphi_l(w_j - x_j, y_l^1; r, y_l^0)$ *для* $r \neq 2j-2$, $s = r \oplus 1$; *значением 0 в остальных случаях. В пространстве состояний* S *содержится* $2m$*-цикл* $\{E_r\colon r = 0, 1, \ldots, 2m-1\}$ *с* $E_r = \{(\Gamma^{(r+1)}, x, y)\colon x \in X; y \in R_+^m\}$.

Теорема 2. *Пусть для каждого* $j = 1, 2, \ldots, m$ *существует* t_j^0, *такое что* $a_j(t) = 0$ *при* $t < t_j^0$ *и* $a_j(t) > 0$ *при* $t \geqslant t_j^0$. *Определим меру* $\vartheta(\cdot)$ *на* $(S, \mathfrak{S})$ *соотношением* $\vartheta\{(\Gamma^{(s)}, w, y)\colon y < y^1\} = y_1^1 \times y_2^1 \times \ldots \times y_m^1$ *при* $s = 1$, $w = \bar{0}, y^1 \in R_+^m$, *и значением 0 в остальных случаях. Тогда стохастическое ядро* $\tilde{P}$ ϑ*-неприводимо.*

Теорема 3. *Пусть для каждого* $j = 1, 2, \ldots, m$ *число* t_j^0 *и плотность* $a_j(t)$ *удовлетворяет предположениям теоремы 2 и* $a_j(t)$ *— непрерывная функция для всех* $t > t_j^0$. *Каждое множество вида* $\{(\Gamma^{(r)}, x, y)\colon \bar{0} \leqslant y^1 \leqslant y < y^2\}$, $\Gamma^{(r)} \in \Gamma$, $x = 0, 1, \ldots,$ *с достаточно малым* $\max\{y_1^2 - y_1^1, y_2^2 - y_2^1, \ldots, y_m^2 - y_m^1\}$ *является минорантным для стохастического ядра* $\tilde{P}$.

Работа выполнена в рамках фундаментальной НИР «Математическое моделирование и анализ стохастических эволюционных систем и процессов принятия решений» (№ госрегистрации 01201456585).

Литература

[1] *Fedotkin M. A., Rachinskaya M. A.* Parameters estimator of the probabilistic model of batches traffic flow with the non-intensive movement // Distributed computer and communication networks: control, computation, communication. — М.: Техносфера, 2013. — С. 357–364.

[2] *Ляпунов А. А., Яблонский С. В.* Теоретические проблемы кибернетики // Проблемы кибернетики. — М.: Физматгиз, 1963. Вып. 9. — С. 5–22.

[3] *Зорин А. В.* Кибернетический подход к построению и анализу математической модели тандема двух перекрестков // Проблемы теоретической кибернетики. Материалы XVI Международной конференции (Нижний Новгород, 20-25 июня 2011 г.) / Под ред. Ю.И. Журавлева. — Нижний Новгород: Изд-во Нижегородского университета, 2011. — С.179–183.

[4] *Meyn S. P., Tweedie R. L.* Markov chains and stochastic stability. — London: Springer-Verlag, 1993. — 566 p.

Избыточность конструктивных описаний эйлеровых графов

М. А. Иорданский

iordanski@mail.ru

Нижегородский государственный педагогический университет им. К.Минина, Нижний Новгород

Введение

Рассматривается конструктор графов, включающий вместе с каждым графом его изоморфные копии, к которым применяются бинарные (*операции склейки*), при выполнении которых производится отождествление изоморфных подграфов $G'_1 \subseteq G_1$ и $G'_2 \subseteq G_2$ графов-операндов G_1 и G_2. Подграф $\tilde{G}$, полученный в результате отождествления G'_1 и G'_2, образует *подграф склейки* результирующего графа G.

Граф G называется *суперпозицией* графов из $\Im$, если $G \in \Im$ или G можно получить путем последовательного применения операций склейки к графам из $\Im$ и к графам, полученным из $\Im$ с помощью операций склейки.

Операции склейки вносят избыточность в задание информации о графах, позволяя формулировать условия наследования различных характеристических свойств графов в виде ограничений на вид отождествляемых подграфов, их выбор в подграфах-операндах и способ отождествления [1]. Избыточность конструктивного описания можно разбить на две компоненты: вершинную и реберную. Вершинная избыточность оценивается по формуле

$$I_v^s(G) = \frac{\sum_{i=0}^{q} |V(\tilde{G}_i)|}{|V(G)|}. \qquad (1)$$

где q - число операций склейки в суперпозиции s, реализующей граф G, $\tilde{G}_i$-подграф склейки i-ой операции.

Для оценки реберной избыточности используется формула

$$I_e^s(G) = \frac{\sum_{i=0}^{q} |E(\tilde{G}_i)|}{|E(G)|}. \qquad (2)$$

В [2] получена оценка реберной избыточности (2)для конструктивных описаний гамильтоновых планарных графов. В данной работе оценивается величина вершинной избыточности для эйлеровых графов.

Оценки вершинной избыточности

Пусть $\Im_n$ множество n - вершинных обыкновенных эйлеровых графов; S - множество всех суперпозиций, реализующих граф $G \in \Im_n$. Каждый эйлеров граф может быть построен из простых циклов путем их склейки по пустым подграфам. Поэтому конструктивные описания эйлеровых графов могут обладать лишь вершинной избыточностью (1). Определим функцию $I_v(\Im_n)$ шенноновского типа :

$$\min_{s\in S} I_v^s(G) = I_v(G); \max_{G\in\Im_n} I_v(G) = I_v(\Im_n).$$

Справедлива

Теорема 1.

$$I_v(\Im_n) = \frac{n-3}{2}.$$

Доказательство. Оценка сверху. Пусть $s(v_j)$ степень вершины v_j произвольного эйлерова графа $G, |V(G)| = n, |E(G)| = m$. При вычислении $I_v(G)$ справедливо равенство

$$\sum_{i=1}^{q} |V(\tilde{G}_i)| = \sum_{j=1}^{n} (\frac{s(v_j)}{2} - 1).$$

Поскольку

$$\sum_{j=1}^{n} s(v_j) = 2m,$$

то $I_v(G) = (m-n)/n$. Отсюда, учитывая что $m \leqslant n(n-1)/2$, получаем, что $I_v(\Im_n) \leqslant (n-3)/2$.

Оценка снизу. Каждый полный граф K_n с нечетным n, являющийся эйлеровым, можно построить с помощью $(n-1)/2-1$ операций склейки по O_n циклов C_n. При этом $I_v(\Im_n) \geqslant (n-3)/2$. ∎

Для множества $\Im_n^p$ планарных n - вершинных обыкновенных эйлеровых графов получаем

Следствие 1.

$$I_v(\Im_n^p) = 2 - \frac{6}{n}.$$

Литература

[1] *Иорданский М. А.* Конструктивные описания графов // Дискретный анализ и исследование операций. — 1996. — Т. 3, № 4. — С. 35–63.

[2] *Иорданский М. А.* Избыточность конструктивных описаний гамильтоновых планарных графов // Материалы XI Международного семинара "Дискретная математика и её приложения"(Москва, МГУ, 18-22 июня 2012 г.) — М: Изд-во механико-математического факультета МГУ, 2012. — С. 285–288.

Гамильтоновы циклы матроида

А. Н. Исаченко, Я. А. Исаченко

isachenkoan@mail.ru, yarais@mail.ru

Белорусский государственный университет, Минск, ООО «Микротест», Москва

Гамильтонов цикл матроида это цикл с числом элементов на единицу большим ранга матроида. В статье вводится понятие степени элемента матроида

и указывается необходимое условие гамильтоновости цикла в терминах степеней элементов.

Введение

Матроид это пара $M = (S, F)$, где S конечное множество элементов, а $F \subseteq 2^S$ — семейство, для которого выполняются следующие условия (аксиомы независимости) [1]:

(i1) $\varnothing \in F$;

(i2) если $A \subseteq B$ и $B \in F$, то $A \in F$;

(i3) если $A, B \in F$ и $|A| = |B|+1$, то найдется $a \in A \setminus B$, такое что $B \cup a \in F$. Подмножества из F называются независимыми, из $2^S \setminus F$ — зависимыми. Ранг $\rho(A)$ подмножества $A \in 2^S$ в матроиде $M = (S, F)$ определяется как мощность максимального независимого множества, содержащегося в A. $\rho(S)$ есть ранг матроида. Максимальное по включению независимое множество называется базой, минимальное по включению зависимое множество — циклом матроида.

Эквивалентно, в терминах циклов, матроид определяется как пара $M = (S, \Sigma)$, где S конечное множество, а $\Sigma \subseteq 2^S$ — семейство, для которого выполняются следующие условия (аксиомы циклов):

(c1) если $C_1, C_2 \in \Sigma$, $C_1 \neq C_2$, то $C_1 \not\subset C_2$;

(c2) если $C_1, C_2 \in \Sigma$, $C_1 \neq C_2$, и $e \in C_1 \setminus C_2$, то для любого $x \in C_1 \cap C_2$ существует $C_3 \in \Sigma$ такое, что $e \in C_3 \subseteq (C_1 \cup C_2) \setminus x$.

Пусть $M = (S, F)$ и $A \subseteq S$. Ограничением матроида M на множество A называют матроид $M_{S \setminus A} = (A, F_{S \setminus A})$, с $F_{S \setminus A} = \{B | B \subseteq A, B \in F\}$.

Элементы $a, b \in S$ в матроиде $M = (S, F)$ называют связными, если существует цикл C такой, что $a, b \in C$. Матроид $M = (S, F)$ называют связным, если любые два элемента множества S являются связными.

Пусть $M = (S, F)$ — матроид ранга $\rho(S) = k$, $k < |S|$. Цикл C матроида M назовём гамильтоновым, если $|C| = k + 1$. Соответственно базу B матроида назовём гамильтоновой, если существует содержащий её гамильтонов цикл. Матроид, содержащий гамильтонов цикл, так же будем называть гамильтоновым.

Понятие гамильтонова цикла, гамильтоновой базы и гамильтонова матроида введено в работах [2, 3]. В них же указан ряд свойств, касающихся сложности распознавания гамильтонова цикла матроида. В частности показано, что относительно оракула «H-периметр» задача распознавания гамильтонова цикла является полиноминально разрешимой. Сведения о двадцати двух матроидных оракулах и их полиномиальной сводимости можно найти в работе [4]. В работе [5] доказана следующая теорема.

Теорема 1. *Гамильтонов матроид является связным.*

Необходимое условие гамильтоновости цикла

Пусть $M = (S, F)$ — матроид. Степенью $d(e)$ элемента $e \in S$ назовём количество циклов матроида M, содержащих e.

В частности, если $d(e) = 0$, то элемент e входит в любую базу матроида M.

Теорема 2. *Пусть $M = (S, F)$ — матроид на множестве S с n элементами и рангом $\rho(S) = k$, $0 < k < n$. Если матроид $M = (S, F)$ гамильтонов, то $d(e) \geqslant n - k$ для любого $e \in S$.*

Доказательство. Проведём доказательство индукцией по $n - k$. Пусть $n = k + 1$. Тогда, если M гамильтонов, то S является циклом, причём единственным и, следовательно, $d(e) = 1$, для любого $e \in S$.

Пусть теорема справедлива для $n \geqslant k + 1$. Рассмотрим матроид M на $n + 1$ элементе ранга k. Если M гамильтонов, то пусть C его гамильтонов цикл. Имеем $|S \setminus C| \geqslant 2$, следовательно, существуют элементы e, $u \in S \setminus C$. Рассмотрим матроид $M_e = (S \setminus e, F_e)$, являющийся ограничением M на множество элементов $S \setminus e$. Матроид M_e содержит цикл C, ранг $\rho_e(S \setminus e)$ матроида M_e равен k. Следовательно, M_e является гамильтоновым и по индуктивному предположению степень $d_e(v)$ любого элемента $v \in S \setminus e$ в матроиде M_e удовлетворяет неравенству $d_e(v) \geqslant n - k$. Так как матроид M по теореме 1 является связным, то для любого элемента $v \in S \setminus e$ существует цикл C_{ev}, содержащий элементы e и v. Поскольку любой цикл матроида M_e является циклом матроида M, а цикл C_{ev} не входит в множество циклов матроида M_e, то для степени элемента $v \in S \setminus e$ в матроиде M получим $d(v) \geqslant d_e(v) + 1 \geqslant n - k + 1$. Если рассмотреть матроид $M_u = (S \setminus u, F_u)$, то аналогично получим $d(e) \geqslant n - k + 1$. ∎

Литература

[1] *Айгнер М.* Комбинаторная теория. — М.: Мир, 1982. — 558 с.

[2] *Исаченко А. Н* Приложения теории матроидов и гамильтоновы матроиды // Третья международная научная конференция «Математическое моделирование и дифференциальные уравнения»: тезисы докладов; Брест, 17-22 сентября 2012 г. / Брест. Гос. Ун-т имени А.С. Пушкина — Брест: БрГУ, 2012. — С. 29–30.

[3] *Исаченко А. Н, Исаченко Я. А* Периметр матроида и задача коммивояжера на матроиде // XI Белорусская математическая конференция: Тез. докл. Междунар. науч. конф. Минск, 5-10 ноября 2012 г. — Часть 4. — Мн.: Институт математики НАН Беларуси, 2012. — С. 87–88.

[4] *Исаченко А. Н, Исаченко Я. А, Ревякин А. М.* О периметрах и окружениях матроида // Вестник МГАДА: серия экономика. — 2013. — № 1. — С. 63–67.

[5] *Исаченко А. Н, Исаченко Я. А* Свойства гамильтоновых матроидов // Международный конгресс по информатике: информационные системы и технологии — International Congress on Computer Science: Information Systems and Technologies : материалы междунар. науч. конгресса, Республика Беларусь, Минск, 4-7 нояб. 2013 г. — Минск: БГУ, 2013. — С. 538–541.

О базисах клона всех гиперфункций ранга 2

А. С. Казимиров, В. И. Пантелеев, Л. В. Токарева

a.kazimirov@gmail.com, vl.panteleyev@gmail.com, lidia.t@mail.com

ВСГАО, Иркутск

Пусть $|A|$ — мощность множества A, 2^A — множество всех подмножеств A и $E_2 = \{0, 1\}$. Определим следующее множество функций:

$$P^-_{2,n} = \{f \mid f : E_2^n \to 2^{E_2} \setminus \{\varnothing\}\}, P^-_2 = \bigcup_n P^-_{2,n}$$

Функции из P^-_2 называются гиперфункциями. Суперпозиция

$$f(f_1(x_1, \ldots, x_m), \ldots, f_n(x_1, \ldots, x_m))$$

с внешней гиперфункцией $f(x_1, ..., x_n)$ и внутренними гиперфункциями $f_1(x_1, ..., x_m), ..., f_n(x_1, ..., x_m)$ определяет гиперфункцию $g(x_1, ..., x_m)$ следующим образом:

$$g(\alpha_1, \ldots, \alpha_m) = \bigcup_{\beta_i \in f_i(\alpha_1, \ldots, \alpha_m)} f(\beta_1, \ldots, \beta_n)$$

для любого набора $(\alpha_1, \ldots, \alpha_m) \in E_2^m$.

Множества, содержащие все функции-проекции и замкнутые относительно суперпозиции, называются клонами. Клон называется максимальным, если единственным клоном, его содержащим и не совпадающим с ним, является клон всех гиперфункций.

Известно [1], что для рассматриваемого множества гиперфункций число максимальных клонов равно 9.

Для каждой гиперфункции однозначным образом определим вектор принадлежности максимальным клонам.

На множестве всех гиперфункций определим отношение эквивалентности следующим образом: эквивалентными будут гиперфункции, у которых совпадают векторы принадлежности максимальным клонам.

Теорема 1. *Множество всех гиперфункций порождает 119 классов эквивалентности.*

Множество гиперфункций называется полным, если оно содержится только в клоне всех гиперфункций. Множество гиперфункций называется базисом, если оно является полным множеством, но при удалении хотя бы одной гиперфункции это свойство нарушается.

С использованием разбиения на классы эквивалентности найдены мощности всех возможных базисов, и число различных типов базисов одинаковой мощности. При этом два базиса считаются разными по типу, если хотя бы для одной гиперфункции некоторого базиса не найдется эквивалентной в другом базисе. Базисы клона всех гиперфункций ранга 2 имеют мощности от 1 до 7, для мощности 1 существует только один тип базиса, для мощности 2 существует 581 тип базиса, для мощности 3 — 19 299, для мощности 4 — 58 974,

для мощности 5 — 27 857, для мощности 6 — 2316, и для мощности 7 — 35 различных типов базиса.

Известно [2, 3], что для булевых функций число аналогичных классов эквивалентности равно 15, имеется один тип базиса мощности 1, 17 типов базиса мощности 2, 22 типа базиса мощности 3, 2 типа базиса мощности 4, базисов большей мощности не существует.

Работа выполнена при поддержке РФФИ: проекты № 12-01-00351, № 13-01-00621.

Литература

[1] *Тарасов В. В.* Критерий полноты для не всюду определенных функций алгебры логики// Проблемы кибернетики. – М. : Наука, 1975. – Вып. 30. – С. 319–325.

[2] *Яблонский С. В.* О суперпозициях функций алгебры логики // Мат. сб. – 1952. – Т. 30,№ 2(72), С. 329–348.

[3] *Krnić L.* Types of bases in the algebra of logic // Glasnik matematicko-fizicki i astronomski. Ser 2. – 1965. – Vol. 20. – P. 23–32.

Оценка алгоритмов распределения ресурсов в когнитивных системах связи с зональными аукционами

А. Ю. Кашуба

leksser@rambler.ru

ООО «АСТ Поволжье», Казань

В настоящей работе численно исследуется эффективность алгоритмов распределения ресурсов в когнитивных системах связи с зональными аукционами, которые были предложены в работе [1].

Рассматривается двухуровневая система. На верхнем уровне менеджер распределяет ресурсы между первичными пользователями, которые распределяют избыточный ресурс среди вторичных пользователей своих зон с помощью аукциона.

Согласно [1], задача менеджера сети имеет следующий вид:

$$\min \to \sum_{k=1}^{n} (\theta_k f_k(u_k) + \omega_k(u_k)),$$

$$\sum_{k=1}^{n} u_k \leqslant R, \tag{1}$$

$$u_k \geqslant \alpha_k, k = \overline{1,n},$$

где

— n — количество зон, на которое поделена сеть; количество первичных пользователей, однозначно сопоставленных зонам;

— R — количество ресурсов (спектра);

— I_k — множество индексов вторичных пользователей в k-й зоне($k = \overline{1,n}$);
— u_k — неизвестное количество спектра, выделяемое первичному пользователю k-й зоны;
— $\omega_k(u_k) = \omega_{2,k}(u_k) - \omega_{1,k}(u_k)$ ($\omega_{1,k}(u_k)$ — доход менеджера сети от k-й зоны, $\omega_{1,k}(u_k) = \rho_k u_k, \rho_k > 0$; $\omega_{2,k}(u_k)$ — стоимость выделенного спектра (затраты менеджера сети) для k-й зоны, $\omega_{2,k}(u_k) = \sigma_{1,k}u_k + \sigma_{2,k}$, либо $\omega_{2,k}(u_k) = \delta_{3,k}u_k^2 + \delta_{2,k}u_k + \delta_{1,k}$);
— θ_k ($\theta_k > 0$) — доля k-й зоны;
— $f_k(u_k)$ — доход от аукциона k-й зоны, является оптимальным значением функции (2).

$$\min \rightarrow (\mu_k(x_k) - \sum_{i \in I_k} \eta_i(y_i)), \tag{2}$$

где

— x_k — неизвестное количество спектра, предложенное первичным пользователем вторичным пользователям в k-й зоне; $g_k(x_k)$ — его цена за данное количество спектра ($g_k(x_k) = \xi_k x_k^{\nu_k}, \xi_k, \nu_k > 0$), α_k ($\alpha_k > 0$) — минимальное количество спектра, используемое самим первичным пользователем (т.е. не предлагаемое вторичным пользователям);
— y_i — неизвестное количество спектра, полученное i-ым вторичным пользователем, β_i ($\beta_i > 0$) — максимальное для него количество спектра, $h_i(y_i)$ — цена i-го пользователя ($i \in I_k, k = \overline{1,n}$) за данное количество ($h_i(y_i) = r_i S_i$, r_i — доход за бит достигаемой частоты передачи, S_i — спектральная эффективность передачи i-го вторичного пользователя);
— $\mu_k'(x_k) = g_k(x_k)$, $\eta_i'(y_i) = h_i(y_i)$.

В [1] предложено решать задачу (1) двойственным методом:

$$\max_{\lambda \geqslant 0} \rightarrow \psi(\lambda),$$

$$\psi(\lambda) = \min_{u \in U} L(u, \lambda) = \sum_{k=1}^{n} \min_{u_k \geqslant \alpha_k} (\theta_k f_k(u_k) + \omega_k(u_k) + \lambda u_k) - \lambda R, \tag{3}$$

$$U = \{u \in R^n \mid u_k \geqslant \alpha_k, k = \overline{1,n}\}.$$

Решение полученной задачи можно найти с помощью алгоритмов решения одномерных задач оптимизации (к примеру, методом золотого сечения [2]) и алгоритма нахождения аукционной цены, описанного в [1]. Алгоритм нахождения аукционной цены включает в себя упорядочивание (сортировку) вторичных пользователей по убыванию цены за приобретение единицы спектра. После этого тем вторичным пользователям, у которых цена не меньше цены продажи, запрашиваемой первичным пользователем, распределяется имеющийся спектр в порядке начиная с вторичного пользователя с максимальной ценой, заканчивая – с минимальной ценой.

Оценка количества итераций и тестирование

Пусть ε_λ и ε_{u_k} — заданные точности нахождения λ и u_k, соответственно. Пусть $[a,\ b]$ и $[\alpha_k,\ B_k]$ — начальные отрезки для λ и для u_k, соответственно.

Введем обозначения:

— m — количество итераций (шагов), затрачиваемых на поиск решения;
— n — количество зон (первичных пользователей);
— m_λ — количество шагов, затрачиваемых на нахождение λ. Для алгоритма золотого сечения:

$$m_\lambda \geqslant -\frac{\ln \varepsilon_\lambda - \ln(b-a)}{0,69};$$

— m_{u_k} — количество шагов, затрачиваемых на нахождение u_k. Для алгоритма золотого сечения:

$$m_{u_k} \geqslant -\frac{\ln \varepsilon_{u_k} - \ln(B_k - \alpha_k)}{0,69};$$

— m_{Ik} — количество шагов нахождения аукционной цены, $m_{Ik} \leqslant \mid I_k \mid$;
— m_{Sk} — количество шагов, затрачиваемых на сортировку.

С учетом фиксированных цен $h_i(y_i)$ $(i \in I_k, k = \overline{1,n})$ имеем:

$$m \leqslant m_\lambda(\sum_{k=1}^{n} m_{u_k}(m_{Ik}+1)) + \sum_{k=1}^{n} m_{Sk}.$$

Было проведено тестирование алгоритма решения задачи (3).

Рассматривались статические случаи, когда вторичные пользователи закреплены за зонами. Распределялись вторичные пользователей по зонам случайно согласно либо равномерному, либо нормальному законам распределения. Тип распределения на результаты, приведенные ниже, не влиял.

При использовании [0,1000] и $[\alpha_k, R]$ в качестве начальных отрезков для нахождения λ и u_k $(k = \overline{1,n})$, соответственно, количества зон от 10 до 100 и количества вторичных пользователей от 10 до 1010 количество итераций поиска решения на верхнем уровне задачи не превышало 50, а при фиксированных ε_λ и ε_{u_k} $(k = \overline{1,n})$, равных 10^{-4}, было равным 34. Количество обращений к поиску решения k-й задачи нижнего уровня не превышало 1300.

Было получено, что с увеличением размерности задачи как по количеству зон (первичных пользователей), так и по количеству вторичных пользователей затрачиваемые время и количество итераций, необходимые для поиска решения распределения спектра, увеличивались не более чем линейно.

Была рассмотрена адаптивная стратегия относительно точности задач нижнего уровня (уточнение ε_{u_k} $(k = \overline{1,n})$ в зависимости от приближения к решению задачи верхнего уровня), что позволило ускорить процесс поиска решения приблизительно на 25-30%.

Рассматривались динамические случаи, когда вторичные пользователи могут перемещаться из зоны в зону. Данные случаи сводились к статическим, в которых каждый вторичный пользователь принадлежал той зоне, в которой он находился больше всего за определенный промежуток времени. При таком подходе результаты тестирования совпадали с результатами выше.

Литература

[1] *Konnov I. V.* Optimization of Network Resources with Zonal Auctions // Internet Decision Technology (accepted). — 2013.

[2] *Васильев Ф. П.* Численные методы решения экстремальных задач. — М.: Наука, 1988. — 552 с.

Об уравнениях в словах с тремя неизвестными

Л. Г. Киселева

kiseleva_lg@sandy.ru

ННГУ им. Н.И. Лобачевского, Нижний Новгород

Уравнение в словах с тремя неизвестными задается равенством пары различных слов в алфавите неизвестных $\{x, y, z\}$:

$$F(x, y, z) = G(x, y, z) \tag{1}$$

Уравнение называется однородным, если каждое неизвестное входит одинаковое число раз в левую и правую часть, в противном случае уравнение называется неоднородным.

Тройка слов (x_0, y_0, z_0) в алфавите A называется решением, если при подстановке их вместо неизвестных получается графическое тождество.

Решения (x_0, y_0, z_0) и (x'_0, y'_0, z'_0) называются изоморфными, если они удовлетворяют одному и тому же множеству уравнений.

Решение (x_0, y_0, z_0) называется тривиальным, если $x_0 = u^i, y_0 = u^j, z_0 = u^k$, или $x_0 = u^i, y_0 = u^j, z_0 = v$, или $x_0 = u, y_0 = v, z_0 = \varphi(u, v)$, где $\varphi(u, v)$ – пустое слово или некоторое произведение слов u, v. Остальные решения называются нетривиальными.

Нахождение всех тривиальных решений уравнения (1) сводится к решению некоторого диофантова уравнения.

Решение (x_0, y_0, z_0) называется минимальным, если для любого решения (x_1, y_1, z_1) выполнена система неравенств: $|x_0| \leqslant |x_1|, |y_0| \leqslant |y_1|, |z_0| \leqslant |z_1|$.

В работе [1] доказано, что уравнения с двумя неизвестными имеют только тривиальные решения вида $x_0 = u^i, y_0 = u^j$.

В [2] доказано, что неоднородное уравнение с тремя неизвестными имеет не более одного нетривиального решения, с точностью до изоморфизма. Для уравнения с четырьмя неизвестными это утверждение неверно.

Существуют однородные уравнения с тремя неизвестными, которые имеют бесконечное множество неизоморфных нетривиальных решений, например, уравнения

$$xyz = zxy, \ xzxy = yxzx.$$

Длиной уравнения (1) называется

$$|F(x, y, z)| + |G(x, y, z)| = l(F, G),$$

т.е. сумма длин левой и правой части.

В [3] доказано, что любое однородное уравнение (1) длины не более 10 либо не имеет нетривиального решения либо имеет бесконечное множество неизоморфных нетривиальных решений. Найден пример однородного уравнения

длины 12, которое имеет единственное нетривиальное решение с точностью до изоморфизма:

$$x^2zy^2z = y^2z^2x^2.$$

Доказано, что почти все уравнения неоднородны. Отсюда следует

Теорема 1. *Почти все уравнения с тремя неизвестными имеют не более одного нетривиального решения с точностью до изоморфизма.*

В работе изучены свойства минимальных решений.

Теорема 2. *Если (x_0, y_0, z_0) - минимальное нетривиальное решение уравнения (1), то $min\{|x_0|, |y_0|, |z_0|\} = 1$, $|A| = 2$, т.е. алфавит решений состоит из двух букв.*

Литература

[1] *Blum E. K.* A note on free subsemigroups with two generators // Bull. Amer. Math. Soc. — 1965. — Т. 71, № 4. — С. 678–679.

[2] *Киселева Л. Г.* Алгебраическое исследование простейших кодов и бескоэффициентных уравнений в словах // Матем. Сб. — 1979. — Т. 108(150), № 4. — С. 529–550.

[3] *Киселева Л. Г.* О числе решений некоторых уравнений в словах с тремя неизвестными // Проблемы теоретической кибернетики, тезисы докладов XV международной конференции.— Казань, 2008. — С. 48.

Ещё одна биекция в перечислительной комбинаторике

Л. М. Коганов

Научный центр нелинейной волновой механики и технологии РАН, Москва

1. Будем понимать под {3}-деревьями свободные (некорневые, никуда не вложенные) деревья с помеченными начальным отрезком натурального ряда $[n] = \{1, \ldots, n\}$ концевыми (= висячими = листьями) вершинами, все внутренние вершины которых имеют исключительно степень 3 и не имеют никаких пометок.

Аналогично определяются $\{2,3\}$-, $\{3,4\}$-, ... и прочие { }-деревья. При этом в фигурных скобках указываются *допустимые и реализуемые* степени вершин, а наименьшее натуральное число 1 опускается согласно общеизвестной теореме о нетривиальных деревьях, содержащих не менее двух — предельный случай простых цепей — висячих вершин.

{3}-деревья фигурируют как основные объекты исследования в междисциплинарном научном направлении, именуемом биоинформатикой [1, гл. 7, с. 240–251; 278–280].

Там же рассматривается задача перечисления, причём перечисляющим параметром является как правило число n висячих вершин, через которое с помощью, допустим, *леммы о рукопожатиях* выражаются остальные параметры, как то: число внутренних степени 3 вершин, общее число рёбер и т.п.

2. *Алгоритм* (предварительная версия) последовательного посистемного наращения вплоть до получения требуемого {3}-дерева или же *системы* всех без исключения требуемых {3}-деревьев с заданным наперёд значением перечисляющего параметра n таков.

1′. Процесс начинается с однорёберного двухвершинного графа-дерева, концы которого помечены сответственно числами 1 и 2, а само ребро-спэйс (space, термин взят из [2]) — римской цифрой I.

Далее предлагается последовательное формирование по наращению стеблей (полуребро со стрелкой на конце — термин В.А. Лисковца).

2′. При встраивании в условную «середину» (любая внутренняя точка гомеоморфного образа открытого интервала по У.Т. Татту) указанного спэйса-ребра стре́лки добавляемого стебля мы:

a) метим свободный от стре́лки конец стебля первым незанятым натуральным числом (на 1-м шаге это будет 3).

b) новое, образованное стеблем ребро-спэйс, обозначим римской нумерацией числа $2n-3$ (см. конец предыдущего раздела; так, при $n=3$ это будет III).

Далее, так как каждое перед подразбиением (*subdivision* operation) остриём стебля ребро является перешейком (или мостом = a bridge) в смысле Ф. Харари, то мы имеем после подразбиения 2 ветви, доставляемые основанием встроенного стебля (= точкой подразбиения = точкой крепления), а именно: ветвь, содержащую висячую вершину 1 (мы отождествляем, как сейчас принято, вершины и их метки) и ветвь, дополнительную до указанной *в исходном до шага дереве* с подразбитым встраиванием стебля ребром *после шага алгоритма*.

Тогда:

c) в ветви, содержащей вершину 1, по всем рёбрам (кроме нового встроенного ребра-стебля — см. предыдущий пункт b) и кроме ребра-стебля, встроенного на непосредственно предыдущем шаге — см. ниже), включая части подразбитого ребра (имеется в виду его часть, принадлежащая указанной ветви) все без исключения метки рёбер-спэйсов оставляем без изменений:

$$X := X;$$

в ветви же дополнительной к рассмотренной выше, образованной указанным подразбиением без добавляемого стебля, сдвигаем нумерацию всех без исключения рёбер-спэйсов на I (римское):

$$Y := Y + \mathrm{I}.$$

При этом *такой же сдвиг* римской нумерации осуществляется и во встроенном на непосредственно предыдущем шаге стебле, ныне ребре-спэйсе, причём вне зависимости от нахождения в ветвях — содержащей ли вершину 1, или же дополнительной по отношению к точке подразбиения.

После чего продолжаем всевозможными допустимыми способами процесс посистемного наращения [3] путём подразбиения новых рёбер и их указанной пошаговой (пере-) нумерации вплоть до образования требуемого 3-дерева. Что

доставляет естественный процесс кодирования (римской нумерации) всех без исключения рёбер вкупе с явной формулой перечисления мощности системы указанных деревьев (см. ниже).

3. **Теорема 1.** *При $n \geqslant 3$ существует универсальная посистемная биекция по последовательному наращению спэйсов (и их указанной выше перенумерации) системы 3-деревьев с перечисляющим параметром — числом висячих вершин n и диаграммами связей с числом* $\mathrm{rank} = n - 2$ *полуокружностей [3].*

Следствие. Мощность обеих указанных систем при $n \geqslant 3$ есть согласно [3] (см. также [4])

$$[2 \cdot (n-2) - 1]!! = (2n-5)!!,$$

и при этом выводе мы обходимся без трудоёмкого кодирования по Прюферу ([1], — см. также статью А. Мовшовича – Ф. Харари, указанную в списке литературы в [4]).

4. **Замечание 1.** Работа алгоритма из раздела 2 настоящего текста была проверена вручную автором вплоть до $n = 5$, возможно, он, этот алгоритм, *потребует некоторых видоизменений и уточнений.* Однако в силу рассуждений, представленных в [1] и, ранее и независимо, в [4], теорема предыдущего раздела *верна вне рассмотренного алгоритма* последовательной пошаговой римской пометки-нумерации рёбер-спэйсов.

Замечание 2. Аналогично перечисляются $\{2,3\}$-деревья с двумя параметрами — числом висячих (помеченных) вершин и числом внутренних (непомеченных) точек подразбиения в смысле Ф. Харари. При этом мы трактуем $\{2,3\}$-дерево как оснащённое *размещением с повторениями* $\{3\}$-дерево, где роль *типов* играют рёбра-спэйсы с данной указанным выше алгоритмом — раздел 2 настоящего текста — римской нумерацией, а роль *неразличимых внутри типов* дробинок-элементов (обычно маленьких кружочков на диаграммах) — играют внутренние вершины подразбиений (= степени 2).

Отметим, что ранее перечисление $\{3\}$- и $\{2,3\}$-деревьев осуществлялось совершенно иными методами в программе Лесли Ричарда Фулдса – Роберта У. Робинсона: см. особенно в [5, разд. 1 и ссылки, данные там].

5. Автор признателен И.М. Паку (университет Лос-Анджелеса) за обсуждение, состоявшееся в Москве в МЦНМО 1 октября 2004 года. Автор также сердечно признателен Д.С. Романову (МГУ, кафедра математической кибернетики ВМиК) за дружескую помощь в подготовке текста.

Литература

[1] *Бородовский М., Екишева С.* Задачи и решения по анализу биологических последовательностей (сер. «Биоинформатика и молекулярная биология»). — М. – Ижевск: НИЦ «Регулярная и хаотическая динамика», Институт компьютерных исследований, 2008. — 440 с.

[2] *Gessel I. M., Stanley R. P.* Stirling polynomials // J. Combinatorial Theory. — 1978. — V. A24, № 1. — P. 24–33.

[3] *Коганов Л. М.* Универсальная биекция между перестановками Гесселя – Стенли и диаграммами связей соответствующих рангов // УМН. — 1996. — Т. 51, вып. 2 (308). — С. 165–166.

[4] *Коганов Л. М.* Универсальная биекция между частично помеченными бинарными свободными деревьями и плоскими монотонными деревьями // Формальные степенные ряды и алгебраическая комбинаторика. 12-я Международная конференция, FPSAC'00: дополнительные тезисы. — М.: МАКС Пресс, 2000. — С. 40–41.

[5] *Foulds L. R., Robinson R. W.* Enumerating phylogenetic trees with multiple labels // Discrete Mathematics. — 1988. — V. 72. — P. 129–139.

Нелинейное управление на многообразиях с компенсацией неизвестных возмущений

С. И. Колесникова

skolesnikova@yandex.ru

Томский государственный университет, Томск

Методы нелинейной адаптации на многообразиях являются одним из наиболее перспективных инструментов нелинейного синтеза систем управления [1, 2, 3, 4]). Одно из ранних успешных применений формализма инвариантов для построения системы управления осуществлено Г.В.Щипановым (напр., [5]), впервые поставившим задачу синтеза «регулятора с полной компенсацией».

Постановка задачи. В докладе рассматривается задача конструирования управления сложным объектом с плохо формализуемой правой частью в его описании:

$$\begin{aligned} \dot{x_j}(t) &= f_j(x_1, x_2, ..., x_n; \Theta; u_j), j = \overline{1, m}; \\ \dot{x_j}(t) &= f_j(x_1, x_2, ..., x_n; \Theta), j = \overline{m+1, n}, \end{aligned} \tag{1}$$

где $x = (x_1, x_2, ..., x_n) \in R^n$ – вектор состояний, $\Theta \in R^k$ – вектор параметров, $u \in R^m, m < n$ – вектор управления, $f \in R^n$ – непрерывная (нелинейная) вектор-функция; компоненты вектора $f_1, f_2, ..., f_m$ неизвестны. Для объекта (1) ставится задача нахождения закона управления $u(x)$, обеспечивающего перевод объекта управления (1) из произвольного начального состояния $x(0)$ в некоторой области фазового пространства в заданное состояние и его стабилизацию в некоторой окрестности целевого многообразия $\Psi(x) = 0$, где $\Psi(x)$ – специальным образом определенная макропеременная (функция, зависящая от нескольких координат объекта).

Решение задачи. Будем строить систему управления для объекта (1), робастную к неизвестным составляющим описания (1).

Изложим кратко идейные стороны подходов к управлению на многообразиях [2, 3, 4], реализованных в алгоритмах 1–3 на примере системы 2-го порядка, как преамбулу к алгоритму 4 решения задачи:

$$\begin{aligned} \dot{x_1}(t) &= f_1(x); \\ \dot{x_2}(t) &= f_2(x) + u, \end{aligned} \tag{2}$$

где f_1 – известная функция, f_2 – неизвестная нелинейная функция;

Алгоритм 1 скользящего управления на многообразиях.

1. Нахождение мажоранты: $\lambda(x) : |c_1 f_1(x) + f_2(x)| \leqslant \lambda(x) \forall x \in R^2$.
2. Определение целевого многообразия: $\Psi(x) = 0, \Psi(x) = c_1 x_1 + x_2$.
3. Выбор функции Ляпунова: $V(t) = 0.5\Psi^2(x)$.
4. Вывод робастного закона управления по отношению к неопределенности f_2 согласно условию Ляпунова $V(t) < 0$:

   $u(x) = -\beta(x) sign(\Psi(x)), \beta(x) : \beta(x) \geqslant \lambda(x) + \beta_0$, где $\beta_0 = const > 0$.

К нежелательным особенностям алгоритма можно отнести следующие положения: а) высокая чувствительность к присутствию шумов, к параметру c_1 ; б) обязательное знание мажоранты $\lambda(x)$; в) управление $u(x)$ разрывное.

Алгоритм 2 аналитического конструирования агрегированных регуляторов (АКАР).

1. Расширение фазового пространства введением дополнительной переменной $x_3 := f_2(x)$, при этом указывается закон изменения неопределенности: $\dot{x_3} = g(x(t))$.
2. Определение целевого многообразия (для удобства сравнения выберем цель управления в таком же виде, полагая $c_1 = 1$): $\Psi(x) = 0, \Psi(x) = x_1 + x_2$.
3. Постановка задача формирования управляющего воздействия, переводящего объект (2) из заданного начального состояния $x(0)$ в окрестность многообразия $\Psi(x) = 0$ и доставляющего минимум определенному оптимизирующему функционалу вида: $J = \int\limits_0^\infty (\varphi^2(\Psi(x)) + \omega^2 \dot{\Psi}^2(t))\, dt$, где функция $\varphi(\Psi(x))$ обладает свойствами: а) однозначная, непрерывная, дифференцируемая функция для всех $\Psi(x)$; б) $\Psi(0) = 0$; в) $\varphi(\Psi(x)) > 0$.
4. Применение вариационного принципа, используемого в задачах на условный экстремум: выделение устойчивого подсемейства экстремалей, удовлетворяющих функциональным уравнениям Эйлера-Лагранжа вида: $\omega\dot{\Psi}(x(t)) + \Psi(x(t)) = 0$ и доставляющих безусловный минимум функционалу J.
5. Вывод закона управления: $u(x) = -\omega^{-1}(\Psi(x) + \omega(f_1(x) + x_3))$

Алгоритм 3 гарантирующего регулятора на многообразиях с компенсацией интервальных помех. Логику алгоритма 3 гарантирующего регулятора на примере (2) можно описать следующим образом:

$$\begin{aligned} \dot{x_1}(t) &= f_1(x); \\ \dot{x_2}(t) &= f_2(x) + u. \Rightarrow \end{aligned} \qquad \begin{aligned} \dot{x_1}(t) &= f_1(x); \\ \dot{x_2}(t) &= f_2(x) + u + z(x); \\ \dot{z}(t) &= \eta(x_2 - x_{20}). \end{aligned}$$

Алгоритм 4 синтезирует идеи всех трех выше описанных алгоритмов 1–3 и является теоретическим обоснованием гарантирующих регуляторов.

Алгоритм 4 управления на многообразиях с неполным описанием.

1. Применение алгоритма 2 (метод АКАР) для формирования структурной схемы системы управления:

$$u(x) = -\omega^{-1}(\Psi(x) + \omega(f_1(x) + f_2(x))).$$

2. Применение алгоритма 3: включение дополнительной переменной в уравнение, содержащее переменную управления:

$$\dot{x_1}(t) = f_1(x);$$
$$\dot{x_2}(t) = f_2(x) + u(x) + \upsilon(x);$$
$$\dot{\upsilon}(t) = g(x),$$

где f_1 – известная функция, f_2 – неизвестная функция, $g(x)$ – неизвестная функция, подлежащая далее определению; $u(x)$ – АКАР – управление; $\upsilon(x)$ – составляющая АКАР-управления для компенсации возмущения.
3. Применение алгоритма 1 для определения закона $g(x)$ с функцией Ляпунова в виде $V(t) = 0.5(\Psi^2(x) + \upsilon^2(x))$.
4. Вывод управления $\upsilon(x))$ из условия отрицательности производной функции Ляпунова: $\dot{\upsilon}(t) = -\eta\Psi(x)$.

Итоговая система управления имеет вид:

$$\dot{x_1}(t) = f_1(x);$$
$$\dot{x_2}(t) = -\omega^{-1}\Psi(x) - f_1(t) + \upsilon(x);$$
$$\dot{\upsilon}(t) = -\eta\Psi(x).$$

Перечислим особенности применения и достоинства алгоритма 4.

1. Не требуется знание границы неопределенности f_2.
2. Алгоритм 4 – обоснование метода синтеза гарантирующих регуляторов.
3. Алгоритм 4 содержит все достоинства базовых алгоритмов 1-3 и является теоретическим обоснованием алгоритма построения нечеткого регулятора (Коломейцева А.Б., Хо Д.Л.), основанного на методе АКАР и идее множественного управления [1].

Заключение. В докладе сделан сравнительный обзор базовых алгоритмов управлений на многообразиях. Приведено теоретическое обоснование гарантирующего регулятора, компенсирующего возмущения многомерного нелинейного объекта, причем без обременительных ограничений на характер возмущения.

Благодарности. Автор выражает глубокую признательность А.А. Колесникову за постановку задачи о необходимости сравнения управления на многообразиях плохоформализуемым объектом [6] и метода синтеза гарантирующих регуляторов при наихудших возмущениях.

Работа выполнена при поддержке РФФИ, проект № 13-08-01015-а.

Литература

[1] *Красовский А. А.* Математическая и прикладная теория. Избранные труды. — М.: Наука, 2002. — 362 с.

[2] *Колесников А. А.* Синергетика и проблемы теории управления: сборник научных трудов. — М.: ФИЗМАТЛИТ, 2004. — 504 с.

[3] *Халил Х. К.* Нелинейные системы: монография. — М.: Институт компьютерных исследований, 2009. — 812 с.

[4] *Astolfi A., Karagiannis D., Ortega R.* Nonlinear and Adaptive Control with Applications // Springer. — 2008. — p. 290.

[5] *Щипанов Г. В.* Теория, расчет и методы проектирования автоматических регуляторов // Автоматика и телемеханика. — 1939. — № 1. — С. 49–66.

[6] *Колесникова С. И.* Использование апостериорной информации для управления плохо формализуемым динамическим объектом // Автометрия. — 2010. — Т. 46, — № 6. — С. 78–89.

О числе максимальных повторов и субпериодичностей в формальных словах

Р. М. Колпаков

foroman@mail.ru

МГУ им. М.В.Ломоносова, Москва

Рассматривается задача оценки максимального возможного числа максимальных повторов с разрывом в произвольных словах фиксированной длины. Под повтором с разрывом в слове понимается фактор вида uvu, где u и v — некоторые непустые слова (фактором слова $a_1a_2\dots a_n$ называется произвольный непустой фрагмент $a_ia_{i+1}\dots a_j$ этого слова). Слова u называются соответственно левой и правой копиями повтора, а слово v называется разрывом повтора. Периодом повтора называется суммарная длина копии и разрыва повтора. Для любого $\alpha > 1$ повтор с разрывом называется α-повтором если отношение его периода к длине его копии не превосходит α. Повтор с разрывом называется максимальным, если выполняются следующие два условия:

1. если повтор не является префиксом слова, то символ, предшествующий в слове левой копии повтора, отличен от символа, предшествующего его правой копии;
2. если повтор не является суффиксом слова, то символ, следующий в слове за правой копией повтора, отличен от символа, следующего за его левой копией.

Другими словами, повтор с разрывом является максимальным, если он не содержится в более длинном повторе с разрывом, имеющем тот же период. Отметим, что один и тот же фактор в слове может рассматриваться в качестве различных повторов с разрывом, имеющих разные периоды, т.е. повторы с разрывом, имеющие разные периоды, могут иметь одновременно одинаковые начальные и конечные позиции в слове. В данной работе мы считаем такие повторы различными.

Повторы с разрывами являются естественным обобщением факторов вида uu, где u — некоторое непустое слово. Такие факторы называются квадратами. Периодом квадрата uu называется длина слова u. Квадраты являются классическим объектом для исследований в словарной комбинаторике. В частности, вопросы, связанные с максимальным возможным числом квадратом и их эффективным поиском в формальных словах, исследованы в [1, 2, 3]. Нетрудно заметить, что любой повтор с разрывом однозначным образом расширяется с

сохранением периода либо до квадрата, либо до максимального повтора в разрывом. Поэтому, принимая во внимание ранее разработанные эффективные алгоритмы поиска квадратов в словах, задача эффективного поиска повторов в разрывом в словах сводится к задаче эффективного поиска максимальных повторов с разрывом. Задача эффективного поиска повторов в разрывом исследовалась ранее в ряде работ. В частности, в [4] предложен алгоритм поиска максимальных повторов с разрывом в слове длины n за время $O(n \log n + S)$ где S — размер выходных данных. в [5] предложен алгоритм поиска в словах повторов с разрывом фиксированной длины за время $O(n \log d + S)$, где n — длина слова, d — длина разрыва и S — размер выходных данных. Для размера S выходных данных в этих работах не было получено каких-либо нетривиальных оценок. Недавно в работе [6] получена оценка $O(\alpha^2 n)$ для возможного числа максимальных α-повторов с разрывом в слове длины n и предложен алгоритм поиска всех таких повторов за время $O(\alpha^2 n)$ в случае константного алфавита. В данной работе мы усиливаем оценки, полученные в [6], следующим образом.

Теорема 1. *Для любого $\alpha > 1$ в слове длины n имеется не более, чем $O(\alpha n)$, максимальных α-повторов с разрывом.*

Отметим, что, с другой стороны, для любого α нетрудно построить слово длины n, содержащее $\Omega(\alpha n)$ максимальных α-повторов с разрывом. Поэтому оценка, полученная в теореме 1, является оптимальной по порядку.

Другим естественным обобщением квадратов в словах являются периодичности. Под периодичностью понимается фактор слова, порядок которого не меньше 2 (порядком фактора называется отношение его длины к его минимальному периоду). Периодичности имеют фундаментальное значение для словарной комбинаторики [7], а также для различных приложений, таких как комбинаторные алгоритмы на строках [8, 2], молекулярная биология [9], сжатие текстовых данных [10]. Периодичность называется максимальной, если она не содержится в более длинной периодичности с тем же минимальный периодом. Нетрудно заметить, что любая периодичность в слове однозначным образом расширяется до максимальной периодичности. Поэтому максимальные периодичности представляют собой удобный и экономный способ задания любых других периодичностей в слове. В [11] доказано, что в слове длины n содержится не более $O(n)$ максимальных периодичностей, и предложен алгоритм поиска всех максимальных периодичностей в слове за время $O(n)$ в случае константного алфавита. Оценка для числа максимальных периодичностей, полученная в [11], неоднократно уточнялась в последующих работах. Наилучшая к настоящему времени оценка для этого числа получена в [12].

Наряду с периодичностями можно естественным образом рассматривать в слове факторы с порядком меньшим, чем 2. Мы называем такие факторы субпериодичностями. Субпериодичность называется δ-субпериодичностью, где $0 < \delta < 1$, если ее порядок не меньше $1 + \delta$. Аналогично понятию максимальной периодичности, субпериодичность называется максимальной, если она не содержится в более длинной периодичности или субпериодичности с тем же минимальным периодом. Алгоритмы поиска всех максимальных δ-субпериодичностей в слове предложены в [6]. Нетрудно заметить, что каждая

максимальная δ-субпериодичность в слове однозначным образом представляется в виде некоторого максимального $1/\delta$-повтора с разрывом, период которого совпадает с минимальным периодом субпериодичности. Тем самым в слове существует взаимно однозначное соответствие между всеми максимальными δ-субпериодичностями и максимальными $1/\delta$-повторами с разрывом, представляющими эти субпериодичности. Поэтому в любом слове число максимальных δ-субпериодичностей не превосходит числа максимальных $1/\delta$-повторов с разрывом. Таким образом, из теоремы 1 непосредственным образом вытекает

Теорема 2. *Для любого δ, где $0 < \delta < 1$, в слове длины n имеется не более, чем $O(n/\delta)$, максимальных δ-субпериодичностей.*

С другой стороны, для любого δ нетрудно построить слово длины n, содержащее $\Omega(n/\delta)$ максимальных δ-субпериодичностей. Поэтому оценка, полученная в теореме 2, также является оптимальной по порядку.

Работа выполнена при поддержке РФФИ, проект № 14-01-00598.

Литература

[1] *Crochemore M.* An optimal algorithm for computing the repetitions in a word // Information Processing Letters. — 1981. — V. 12. — P. 244–250.

[2] *Crochemore M., Rytter W.* Squares, cubes, and time-space efficient string searching // Algorithmica. — 1995. — V. 13. — P. 405–425.

[3] *Gusfield D., Stoye J.* Linear time algorithms for finding and representing all the tandem repeats in a string // Journal of Computer and System Sciences. — 2004. — V. 69, № 4. — P. 525–546.

[4] *Brodal G., Lyngso R., Pedersen C., Stoye J.* Finding Maximal Pairs with Bounded Gap // Journal of Discrete Algorithms. — 2000. — V. 1, № 1. — P. 77–104.

[5] *Kolpakov R., Kucherov G.* Finding Repeats with Fixed Gap // Proceedings of 7th International Symposium on String Processing and Information Retrieval (SPIRE'00). — 2000. — P. 162–168.

[6] *Kolpakov R., Podolskiy M., Posypkin M., Khrapov N.* Searching of gapped repeats and subrepetitions in a word // Lecture Notes in Computer Science. — 2014. — V. 8486. — P. 212–221.

[7] *Lothaire M.* Combinatorics on Words. — volume 17 of Encyclopedia of Mathematics and Its Applications, Addison Wesley, 1983.

[8] *Galil Z., Seiferas J.* Time-space optimal string matching // Journal of Computer and System Sciences. — 1983. — V. 26, № 3. — P. 280–294.

[9] *Gusfield D.* Algorithms on Strings, Trees, and Sequences. — Cambridge University Press, 1997.

[10] *Storer J.* Data compression: methods and theory. — Computer Science Press, Rockville, MD, 1988.

[11] *Kolpakov R., Kucherov G.* On Maximal Repetitions in Words // Journal of Discrete Algorithms. — 2000. — V. 1, № 1. — P. 159–186.

[12] *Crochemore M., Ilie L., Tinta L.* Towards a solution to the "runs"conjecture // Lecture Notes in Computer Science. — 2008. — V. 5029. — P. 290–302.

Применение вероятностного метода к изучению разбиений целого числа на слагаемые

А. В. Колчин

andrei.kolchin@gmail.com

Москва

Мы покажем, как с использованием хорошо известной в теории вероятностей схемы размещения частиц по ячейкам можно изучать асимптотическое поведение числа разбиений целого числа на слагаемые. Нашей целью является не получение наилучшей возможной оценки, а демонстрация возможностей вероятностного метода.

Напомним, что всякое представление натурального числа в виде суммы натуральных чисел (частей) называется разбиением числа. Это понятие прежде всего комбинаторного и теоретико-числового характера. Задачи о разбиениях сыграли важную роль для всей математики.

Как писал Ж.-К. Рота в предисловии к монографии [1], теория разбиений — это одна из весьма немногих ветвей математики, которая может быть воспринята всяким, кто проявит немногим более чем простую любознательность к этому предмету. Приложения ее обнаруживаются всюду, где подсчитываются либо классифицируются дискретные объекты, будь то молекулярное или атомное строение вещества, теория чисел или комбинаторные задачи самого разного происхождения.

Разбиения изучаются в комбинаторике и теории чисел; к классическим комбинаторным относятся задачи подсчета и перечисления разбиений, в теории чисел решаются проблемы об аддитивных представлениях чисел с арифметическими ограничениями на слагаемые (таковы, например, известные проблемы Гольдбаха и Варинга). При решении задач о разбиениях возникают серьезные трудности, их преодоление потребовало большой изобретательности и повлекло создание специальных методов теории разбиений (см., например, [1]). Исторически первым и общим для всей теории разбиений явился метод производящих функций. Разработанный Л. Эйлером в том числе и для нужд теории разбиений, этот аналитический метод оказался эффективным инструментом и для комбинаторики, и для теории чисел; он был развит до таких тонких форм, как метод производящих функций Дирихле, метод тригонометрических сумм, метод характеристических функций — методов, применяемых не только в комбинаторике и теории чисел. Другие методы теории разбиений пока еще не столь универсальны.

Именно Эйлер заложил основы теории разбиений числа. В дальнейшей разработке этой теории принимали активное участие такие крупные математики, как Гаусс, Кэли, Лагранж, Лежандр, Литлвуд, Радемахер, Рамануджан, Сильвестр, Харди, Шур, Якоби.

При рассмотрении приложений теории разбиений к различным областям математики выявляется взаимосвязь комбинаторных и асимптотических методов.

Рассмотрим задачу о числе разбиений целого положительного числа n на s целых положительных слагаемых, не превосходящих целого числа r (разбиения, отличающиеся лишь порядком слагаемых, считаются за одно). Обозначим это число разбиений через $C_{n,s,r}$. Для этого числа известен ряд как точных, так и асимптотических формул (см., например, [1]).

Оказывается, что можно легко и быстро получить компактное асимптотическое выражение для указанного числа разбиений с использованием вероятностных рассуждений (см., например, [2, 3]). Действительно, нетрудно увидеть, что разбиение числа на части описывается классической схемой равновероятного размещения n неразличимых дробинок по s неразличимым ящикам, при условии, что каждый ящик может содержать не более r дробинок.

Пусть $\xi_1, \xi_2, \ldots, \xi_s$ — независимые случайные величины, принимающие значения $1, 2, \ldots, r$ с равными вероятностями, которые могут быть интерпретированы как заполнения соответствующих ящиков в вышеописанной схеме размещения. Нетрудно видеть, что

$$\begin{aligned}\mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_s = n\} &= \sum_{\substack{k_1+k_2+\cdots+k_s=n \\ k_1,k_2,\ldots,k_s \leqslant r}} \mathbf{P}\{\xi_1 = k_1, \xi_2 = k_2, \ldots, \xi_s = k_s\} \\ &= C_{n,s,r}\frac{1}{r^s}.\end{aligned} \tag{1}$$

Кроме того, справедливы равенства

$$\mathbf{M}\xi_1 = \frac{r+1}{2} = m, \qquad \mathbf{D}\xi_1 = \frac{r^2-1}{12} = \sigma^2.$$

Поэтому для изучения асимптотического поведения $C_{n,s,r}$ можно использовать хорошо развитый аппарат локальных предельных теорем теории вероятностей (см., например, [4, 5, 6]).

Вначале изучим поведение числа разбиений в так называемой «центральной» области изменения параметров.

Очевидно, что верно соотношение

$$\mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_s = n\} = \mathbf{P}\left\{\frac{\xi_1 + \xi_2 + \cdots + \xi_s - sm}{\sigma} = \frac{n - sm}{\sigma}\right\}.$$

Положим $x = (n - sm)/\sigma$. Используя приведенные выше соотношения для математического ожидания m и дисперсии σ^2, находим, что

$$x = \frac{2n - s(r+1)}{\sqrt{(r^2-1)/3}}.$$

Рассмотрим простейший случай. Будем считать, что число r фиксировано. Из соотношения (1) нетрудно получить, с использованием локальной сходимости к стандартному нормальному закону, что при фиксированном r, при параметрах n и s, стремящихся к бесконечности так, что отношение n к s остается внутри некоторого конечного интервала, соотношение

$$\frac{1}{r^s}C_{n,s,r} = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}(1 + o(1))$$

имеет место равномерно по всем x из произвольного фиксированного конечного интервала. Таким образом, верно следующее утверждение.

Теорема 1. *Пусть r — фиксированное целое число. Если параметры n и s стремятся к бесконечности так, что отношение n к s остается лежащим внутри некоторого конечного интервала, то для числа $C_{n,s,r}$ разбиений целого положительного числа n на s целых положительных слагаемых, не превосходящих r, справедливо равенство*

$$C_{n,s,r} = \frac{r^s}{\sqrt{2\pi}} e^{-x^2/2}(1 + o(1)),$$

равномерно для всех

$$x = \frac{2n - s(r+1)}{\sqrt{(r^2-1)/3}},$$

лежащих внутри произвольного конечного интервала.

Изучению поведения указанного числа разбиений в остальных областях изменения параметров n, s, r автору предполагается целесообразным посвятить отдельное исследование.

Литература

[1] Г. Эндрюс, *Теория разбиений*. Наука, Москва, 1982.

[2] J. Spencer, P. Erdős, *Probabilistic Methods in Combinatorics*. Akadémiai Kiadó, Budapest, 1974.

[3] N. Alon, J. Spencer, *The Probabilistic Method*. Wiley, New York, 1992.

[4] Б. В. Гнеденко, А. Н. Колмогоров, *Предельные распределения для сумм независимых случайных величин*. ГТТИ, Москва–Ленинград, 1949.

[5] В. В. Петров, *Предельные теоремы для сумм независимых случайных величин*. Наука, Москва, 1987.

[6] В. Ф. Колчин, *Асимптотические методы теории вероятностей*. МИЭМ, Москва, 1988.

Аукционный принцип распределения сетевых ресурсов

И. В. Коннов

konn-igor@ya.ru

Казанский федеральный университет, Казань

Недавнее развитие информационных и телекоммуникационных технологий выдвигает много новых задач по эффективному управлению такими сложными системами, связанных с распределением сетевых ресурсов. Несмотря на значительное повышение мощности устройств обработки и передачи информации, детерминированные правила распределения ресурсов (полосы передачи, буферной памяти, емкости батарей) при существующем неравномерном спросе многочисленных пользователей приводят к неэффективной загрузке сети;

напр. [1]. Поэтому создание гибких и простых адаптивных механизмов распределения сетевых ресурсов является весьма актуальным. Одним из таких механизмов является *аукцион* с делимыми товарами, в котором предъявляются минимальные требования к информированности пользователей, в отличие от игровых моделей. Новый подход к построению моделей аукциона на основе вариационных неравенств был предложен в [2]. В работе [3] была предложена сепарабельная модель многопродуктового аукциона. В настоящей работе строится модель многопродуктового аукциона при наличии связывающих ограничений, которая может быть применена для различных задач распределения ресурсов телекоммуникационных систем.

Рассмотрим модель аукциона n товаров с m продавцами ($I = \{1, \ldots, m\}$) и l покупателями ($J = \{1, \ldots, l\}$), при этом i-й продавец заявляет свою вектор-функцию цен предложения $g^i(x^{(i)})$, $x^{(i)} \in X_i \subset R^n$, а j-й покупатель заявляет свою функцию цены спроса $h^j(y^{(j)})$, $y^{(j)} \in Y_j \subset R^n$. Допустим также, что, помимо явных участников аукциона, могут быть еще и внешние (неявные) участники, которые представлены отображением избыточного спроса $E(p)$, где $p \in P \subset R^n$ есть вектор цен. Скажем, что векторы $\bar{x}^{(i)} \in X_i$ для $i \in I$ и $\bar{y}^{(j)} \in Y_j$ для $j \in J$ и $\bar{p} \in P$ определяют состояние равновесия, если

$$\langle g^i(\bar{x}^{(i)}) - \bar{p}, x^{(i)} - \bar{x}^{(i)} \rangle \geqslant 0 \quad \forall x^{(i)} \in X_i \qquad \text{для } i \in I, \tag{1}$$

$$\langle h^j(\bar{y}^{(j)}) - \bar{p}, y^{(j)} - \bar{y}^{(j)} \rangle \leqslant 0 \quad \forall y^{(j)} \in Y_j \qquad \text{для } j \in J, \tag{2}$$

а также

$$\left\langle \sum_{i \in I} \bar{x}^{(i)} - \sum_{j \in J} \bar{y}^{(j)} - E(\bar{p}), p - \bar{p} \right\rangle \geqslant 0 \quad \forall p \in P. \tag{3}$$

Отметим, что равновесная задача (1)–(3) эквивалентна прямо - двойственной системе вариационных неравенств.

Примерами приложений этой модели являются задачи распределения спектра широкополосных линий в когнитивных радиосистемах, где первичные пользователи распределяют через аукцион избыточные ресурсы обычным пользователям, а также задачи мобильной выгрузки данных сетевыми операторами (покупателями) через точки доступа (продавцов) вспомогательных сетей (например, WiFi); см. [4], [5]. Для упрощения описания метода предположим, что $\mu_i'(x^{(i)}) = g^i(x^{(i)})$, $\eta_j'(y^{(j)}) = h^j(y^{(j)})$ и $\tau'(p) = E(p)$, где $\mu_i : X_i \to R$, $i \in I$, $-\eta_j : Y_j \to R$, $j \in J$ и $-\tau : P \to R$ – выпуклые дифференцируемые функции. Тогда система (1)–(3) может быть заменена задачей о поиске седловой точки $(\bar{x}, \bar{y}, \bar{p}) \in X \times Y \times P$, такой что

$$M(\bar{x}, \bar{y}, p) \leqslant M(\bar{x}, \bar{y}, \bar{p}) \leqslant M(x, y, \bar{p}) \quad \forall (x, y, p) \in X \times Y \times P;$$

где $x = (x^{(i)})_{i \in I}$, $y = (y^{(j)})_{j \in J}$, $X = \prod\limits_{i \in I} X_i$, $Y = \prod\limits_{j \in J} X_j$,

$$M(x, y, p) = \sum_{i \in I} \mu_i(x^{(i)}) - \sum_{j \in J} \eta_j(y^{(j)}) + \tau(p) - \left\langle p, \sum_{i \in I} x^{(i)} - \sum_{j \in J} y^{(j)} \right\rangle.$$

Двойственный метод. Выбираем начальный вектор цен $p^0 \in P$. На k-й итерации, $k = 0, 1, \ldots$, регулятор объявляет текущий вектор цен $p^k \in P$. Для каждого $i \in I$, i-й продавец независимо определяет свои объемы предложения $x^{k,(i)}$, решая задачу

$$\min_{x^{(i)} \in X_i} \rightarrow \quad \left\{ \mu_i(x^{(i)}) - \langle p^k, x^{(i)} \rangle \right\};$$

для каждого $j \in J$, j-й покупатель независимо определяет свои объемы спроса $y^{k,(j)}$, решая задачу

$$\max_{y^{(j)} \in Y_j} \rightarrow \quad \left\{ \eta_j(y^{(j)}) - \langle p^k, y^{(j)} \rangle \right\}.$$

Затем регулятор вычисляет дисбаланс

$$F(p^k) = E(p^k) - \sum_{i \in I} x^{k,(i)} + \sum_{j \in J} y^{k,(j)}$$

и корректирует вектор цен:

$$p^{k+1} = \pi_P[p^k + \theta_k F(p^k)], \theta_k > 0,$$

где $\pi_P[\cdot]$ обозначает проекцию на P.

Подходящий выбор длины шага позволяет, как известно, обеспечить сходимость этого процесса к решению; см. [6, гл.2]. Таким образом, получаем простую процедуру распределения ресурсов с минимальными требованиями к информации о поведении и интересах участников.

Работа выполнена при поддержке РФФИ, проект № 13-01-00029, а также Академии Финляндии, проект № 276064.

Литература

[1] Stańczak S., Wiczanowski M., Boche H. Resource allocation in wireless networks. Theory and Algorithms. — Berlin, Springer, 2006.

[2] Коннов И. В. О моделировании рынка аукционного типа // Исслед. по информатике. — Казань, 2006. — Вып. 10. — С. 73–76.

[3] Konnov I. V. On variational inequalities for auction market problems // Optim. Lett. — 2007. - V. 1, № 2. - P. 155–162.

[4] Iosifidis G., Koutsopoulos I. Double auction mechanisms for resource allocation in autonomous networks // IEEE J. Sel. Areas in Commun. — 2010. — V. 28, № 1. — P. 95–102.

[5] Iosifidis G., Gao L., Huang J., Tassiulas L.,(2013) An iterative double auction mechanism for mobile data offloading // IEEE Xplore: WiOpt 2013. — 2013. — P. 154–161.

[6] Гольштейн Е. Г., Третьяков Н. В. Модифицированные функции Лагранжа. — М.: Наука, 1989.

Некоторые особенности задачи синтеза булевых формул в полных базисах с прямыми и итеративными переменными

В. А. Коноводов

vkonovodov@gmail.com

Московский государственный университет им. М. В. Ломоносова,
Факультет вычислительной математики и кибернетики

Пусть $X = \{x_1, x_2, \ldots\}$ и $Y = \{y_1, y_2, \ldots\}$ — счетные множества булевых переменных, причем переменные из множества X (из Y) будем называть *прямыми* (соответственно *итеративными*). Для каждого множества переменных Z обозначим через $P_2(Z)$ множество всех функций алгебры логики (в дальнейшем — просто функций), зависящих от переменных из Z. Функции, не имеющие общих существенных переменных, будем называть *независимыми*.

На множестве $P_2(X \cup Y)$, согласно [1], определим следующие операции суперпозиции:

1. переименование (с отождествлением) прямых переменных,
2. подстановка констант 0, 1 вместо переменных,
3. переименование (без отождествления) итеративных переменных,
4. подстановка одной из двух независимых функций вместо итеративной переменной другой функции,
5. замена итеративных переменных прямыми переменными,
6. отождествление итеративных переменных.

Пусть $A \subset P_2(X \cup Y)$ — некоторое конечное множество базисных функций. В соответствии с введенными операциями суперпозиции будем рассматривать одновыходные схемы из функциональных элементов над базисом A (см., например, [2]), в которых:

1. прямые входы любого элемента либо присоединяются к входам схемы, либо являются константными входами (вход называется константным, если вместо него в базисный элемент подставлена константа 0 или 1);
2. итеративные входы любого элемента либо присоединяются к выходам других элементов, либо присоединяются к входам схемы, либо являются константными входами;
3. неконстантным входам схемы сопоставлены некоторые переменные из множества X.

Как обычно, формулами считаются те схемы, которые не содержат ветвлений выходов элементов.

Заметим, что с точки зрения рекурсивного определения формулы как символьной записи [3], указанные выше операции 1–6 дают возможность проводить суперпозицию только по итеративным переменным базисных функций.

Систему функций A будем называть *полной*, если для любой функции f, $f \in P_2(X)$, существует формула над A указанного вида, реализующая функцию f. Критерий полноты произвольной системы функций был получен в работе [1]. Везде далее рассматриваются только полные системы функций.

Сложностью $L(\mathcal{F})$ формулы $\mathcal{F}$ будем называть число функциональных элементов в ней. Функцией Шеннона $L_A(n)$ для сложности формул в базисе A, как обычно, называется максимальное значение $L_A(f)$ среди всех функций f, $f \in P_2(\{x_1, \ldots, x_n\})$, где $L_A(f)$ — минимальная сложность формулы из рассматриваемого класса, реализующей функцию f.

Функция Шеннона $L_A^C(n)$ для сложности схем из функциональных элементов указанного вида в базисе A определяется аналогично.

Пусть $A \subseteq P_2(X \cup Y)$. Множество тех функций, которые можно получить из функций системы A в результате применения операций суперпозиции с номерами из множества T', $T' \subseteq T = \{1, 2, 3, 4, 5, 6\}$, обозначим через $[A]_{T'}$, и пусть $[A]_T = [A]$.

В работе [1] вводится множество $\delta(A) = \left[[A]_{\{2\}} \cap P_2(Y)\right]_{\{3,4,6\}}$, которое будем называть *итеративным замыканием* базиса A. Заметим, что множество $\delta(A)$ является «обычным» замкнутым классом (см., например, [3]) в $P_2(Y)$, содержащим все константы, и поэтому совпадает с одним из классов системы

$$\Delta = \{B, I, O, D, K, L, M, P_2(Y)\},$$

где $B = \{0, 1\}$, $I = Y \cup B$, $O = I \cup \{\bar{y} : y \in Y\}$, класс D (класс K) содержит константы и дизъюнкции (соответственно, конъюнкции) переменных Y, а классы L и M состоят из линейных и монотонных функций от переменных Y соответственно. Далее за B^k обозначается k-я декартова степень множества B, то есть множество всех упорядоченных наборов длины k из элементов B.

Утверждение 1. *Для любой системы функций A, $A \subseteq P_2(X \cup Y)$, справедливо равенство*

$$\delta(A) = [A] \cap P_2(Y).$$

Таким образом, $\delta(A)$ определяет все те функции от итеративных переменных, которые можно получить из базисных функций рассматриваемыми операциями суперпозиции. Введение оператора δ позволяет классифицировать все системы функций от прямых и итеративных переменных по их итеративным замыканиям. Эта классификация имеет прямое отношение к исследованию сложности формул в соответствующих базисах.

В [4] установлено, что функция Шеннона $L_A^C(n)$ для сложности схем в произвольном базисе A, $A \subseteq P_2(X \cup Y)$, имеет «стандартный» порядок роста $2^n/n$. Кроме того, в [4] указано, что в случае формул порядок роста $L_A(n)$ не более, чем 2^n, и не менее, чем $2^n/\log n$.

Известно [5], что если $\delta(A) \in \{M, P_2(Y)\}$, то функция Шеннона для сложности формул при растущем значении натурального аргумента n ведет себя как[1] $\Theta(2^n/\log n)$. Кроме того, в [5] приведены примеры семейств базисов A, для которых $\delta(A) = I$ и порядок роста функции Шеннона составляет 2^n.

Утверждение 2. *В семействе базисов с итеративным замыканием вида D или K существуют такие базисы A, для которых $L_A(n) = \Theta(2^n)$.*

[1] Для числовых функций $g(n)$ и $h(n)$ натурального аргумента n запись $h(n) = O(g(n))$ означает, как обычно, что отношение $|h(n)/g(n)|$ ограничено сверху. Запись $h(n) = \Theta(g(n))$ означает, что $h(n) = O(g(n))$ и $g(n) = O(h(n))$.

Доказательство этого утверждения основано на рассмотрении сложности линейной функции $l_n = x_1 \oplus \ldots \oplus x_n$ в различных базисах. В ряде базисов, таких как $\{x_1^{\sigma_1} \ldots x_k^{\sigma_k} y_1 \,|\, (\sigma_1, \ldots, \sigma_k) \in B^k\} \cup \{y_1 \vee y_2\}$, где $k = \text{const}$, эта функция имеет сложность $\Theta(2^n)$. В базисах, приведенных в работе [5] для $\delta(A) = I$, линейная функция также является самой сложной. Однако, как показывает следующий результат, при переходе от базиса к базису сложность функции l_n может кардинально изменяться в рамках одного и того же семейства.

Утверждение 3. *В базисе* $A = \{(x_1 \oplus x_2)y_1, (x_1 \oplus x_2 \oplus 1)y_1, y_1 \vee y_2\}$ *сложность линейной функции* l_n *удовлетворяет соотношению*

$$L_A(l_n) = O(2^{n/2}).$$

С учетом мощностных нижних оценок это означает, что линейная функция не всегда является самой сложной для семейства базисов A с $\delta(A) = D$.

Работа выполнена при поддержке РФФИ, проект № 12-01-00964-а.

Литература

[1] *Ложкин С. А.* О полноте и замкнутых классах функций алгебры логики с прямыми и итеративными переменными // Вестн. Моск. ун-та, сер. 15: Вычислит. матем. и киберн. — 1999. — № 3. — С. 35–41.

[2] *Лупанов О. Б.* Асимптотические оценки сложности управляющих систем. — М.: Издательство МГУ, 1984. — 136 с.

[3] *Яблонский С. В.* Введение в дискретную математику. — М.:, 1986. — 384 с.

[4] *Ложкин С. А.* О сложности реализации функций алгебры логики схемами и формулами, построенными из функциональных элементов с прямыми и итеративными переменными // Труды III муждународной конференции «Дискретные модели в теории управляющих систем» Красновидово'98 (22–27 июня 1998 г.) — М: Диалог-МГУ, 1998. — С. 72–73.

[5] *Коноводов В. А.* О сложности булевых формул в базисах из элементов с прямыми и итеративными входами // Материалы IX молодежной научной школы по дискретной математике и ее приложениям (Москва, 16–21 сентября 2013 г.) — М: Изд-во ИПМ РАН, 2013. — С. 57–60.

Об одном типе локальных преобразований автономных автоматов

О. М. Копытова

omkop@list.ru

Донецкий национальный технический университет, Донецк

Введение

В работе продолжается изучение автоматов, устойчивых по поведению к локальной операции – перебросе дуг. Такое преобразование можно понимать как проявление "неисправности" функции переходов автомата при условии,

что вход-выходные отметки дуг при этом остаются неизменными. В [1] показано, что переброска в точности одной дуги в приведенном автомате всегда вызывает изменение его поведения – автомат становится не эквивалентным исходному автомату. Интерес представляет задача изучения устойчивости поведения приведенного автомата при переброске более, чем одной дуги.

В [2] найдены достаточные условия сохранения поведения автомата при переброске двух дуг и показано, что для любого натурального $k > 1$ существует приведенный автомат, в котором найдется подмножество из k дуг, одновременная переброска которых приводит к изоморфному автомату. В [3] найдены необходимые условия, при которых возможна переброска дуг в общем случае, и доказано, что при переброске двух дуг вход-выходные отметки на этих дугах совпадают. Найденные условия сформулированы в терминах идентификаторов автомата. Там же приведен пример не сильно связного автомата, допускающего переброску $k > 2$ дуг, и высказано предположение о том, что состояния, из которых перебрасываются дуги, не достижимы из тех состояний, куда направлена переброска. В [4] приведен пример сильно связного автомата, допускающего переброску четырёх дуг, который опровергает это предположение, причём вход-выходные отметки на перебрасываемых дугах различны – в отличие от случая переброски двух дуг.

В настоящей работе исследуется влияние переброски дуг на циклическую структуру графа переходов автомата. Заметим, что каждому автомату с числом входных символов $m > 1$ соответствует набор из m его автономных компонент. Возникает вопрос о том, как влияет переброска на множество автономных компонент автомата и их структуру. Ясно, что переброска, нарушающая изоморфизм автономных компонент, тем более нарушает и изоморфизм исходного и преобразованного автоматов. В [4] показано, что существует такая переброска одной дуги в приведенном автомате, которая сохраняет множество его автономных компонент, но при этом приводит к неизоморфному автомату. Рассмотрим влияние перебросок на циклическую структуру автономных автоматов и возможность сохранения изоморфизма исходного и преобразованного автоматов.

Основные результаты

Под автоматом будем понимать конечный приведенный автомат Мили $A = (S, X, Y, \delta, \lambda)$, где S, X, Y— множества состояний, входных и выходных символов соответственно, а δ, λ — функции переходов и выходов. Если $X = \{x\}$, т.е. x — единственный входной символ, то автомат называется автономным. Автономные автоматы вида $A_i = (A, x_i, Y, \delta, \lambda)$ назовем автономными компонентами автомата A, где $x_i \in X$, $i = 1, 2, ..., |X|$. В дальнейшем, если не оговорено противное, под автоматом понимается автономный автомат $A = (A, \{x\}, Y, \delta, \lambda)$. Пусть $e = (s, x, y, s_1)$ — дуга в графе переходов автомата A. Переброской дуги e в состояние s_2, отличное от s_1, называем замену этой дуги на дугу (s_1, x, y, s_2). Пусть автомат A' получен из автомата A переброской некоторого множества дуг. Если при этом автомат A' остается изоморфным исходному автомату A, то говорим, что переброска сохраняет изоморфизм, а автомат A допускает данную переброску. Задача заключается в том, что-

бы, исследуя структуру циклов в автономном автомате, найти условия, при которых существуют переброски, сохраняющие изоморфизм.

Неопределяемые понятия можно найти в [5].

Следующие утверждения основаны на том факте, что в общем случае граф переходов автономного автомата представляет собой набор компонент сильной связности (КСС) с входящими в них деревьями, причём каждая такая компонента является простым циклом. В частном случае цикл может быть петлёй. Если автомат сильно связен, то его граф переходов представляет собой единственный простой цикл. Если граф переходов автомата представляет собой один или несколько простых циклов, то автомат можно рассматривать как автономный групповой автомат, у которого $\{(s,x)|s \in S\} = S$. В формулировке следующих утверждений факт приведенности автомата будем оговаривать отдельно. Если не сказано противное, то под автоматом понимается неприведенный автомат.

Утверждение 1. *Переброска одной дуги в автономном сильно связном автомате всегда приводит к неизоморфному автомату.*

Утверждение 2. *Существует автономный неприведенный автомат, допускающий переброску одной дуги.*

Утверждение 3. *Существует автономный приведенный не сильно связный автомат, допускающий переброску двух дуг.*

Утверждение 4. *Существует автономный приведенный сильно связный автомат, допускающий переброску k дуг для всех $3 \leqslant k \leqslant n$, где n — число его состояний.*

Лемма 5. *Автономные приведенные автоматы A и B изоморфны тогда и только тогда, когда: 1) множества их КСС совпадают; 2) если s и t – эквивалентные циклические состояния соответственно автоматов A и B, то входящие в них деревья изоморфны.*

Теорема 6. *Пусть A — автономный сильно связный приведенный автомат и A' получен из A переброской двух дуг. Тогда A и A' не изоморфны.*

Теорема 7. *Пусть A — автономный приведенный групповой автомат и A' получен из A переброской двух дуг. Тогда A и A' не изоморфны.*

Другими словами, теоремы 6 и 7 утверждают, что в автономном приведенном автомате, граф переходов которого представляет собой один или группу циклов, произвольная переброска двух дуг всегда приводит к автомату, не изоморфному исходному.

Следствие 1. *Пусть $A = (S, X, Y, \delta, \lambda)$ — автомат, в котором $|X| = m > 1$ и пусть A_{x_i} — его автономные компоненты, где $x_i \in X$, $i = 1, 2, ..., m$. Если каждая автономная компонента является групповым автоматом, то для A не существует переброски 2-х дуг, сохраняющей изоморфизм.*

Рассмотрим следующий приведенный сильно связный автомат A:
$S = \{s_1, s_2, s_3, s_4, s_5, s_6\}$; $X = Y = \{0, 1\}$;
$\delta(s_1, 0) = s_2$, $\lambda(s_1, 0) = 0$; $\delta(s_1, 1) = s_3$, $\lambda(s_1, 1) = 0$;
$\delta(s_2, 0) = s_1$, $\lambda(s_2, 0) = 0$; $\delta(s_2, 1) = s_4$, $\lambda(s_2, 1) = 0$;
$\delta(s_3, 0) = s_4$, $\lambda(s_3, 0) = 0$; $\delta(s_3, 1) = s_5$, $\lambda(s_3, 1) = 0$;

$\delta(s_4,0)=s_3$, $\lambda(s_4,0)=0$; $\delta(s_4,1)=s_6$, $\lambda(s_4,1)=0$;
$\delta(s_5,0)=s_6$, $\lambda(s_5,0)=0$; $\delta(s_5,1)=s_1$, $\lambda(s_5,1)=0$;
$\delta(s_6,0)=s_5$, $\lambda(s_6,0)=0$; $\delta(s_6,1)=s_2$, $\lambda(s_6,1)=1$.

Поскольку автономная компонента A_0 состоит из трёх циклов длины 2, а A_1 — из двух циклов длины 3, и, значит, обе они суть групповые автоматы, то в соответствии со следствием 1, переброска любых двух дуг в этом автомате всегда нарушает изоморфизм. С другой стороны, легко проверить, что автомат A допускает следующую переброску четырёх дуг: дуга $(s_3,1,0,s_5)$ заменяется на $(s_3,1,0,s_6)$, $(s_4,1,0,s_6)$ на $(s_4,1,0,s_5)$, $(s_5,1,0,s_1)$ на $(s_5,1,0,s_2)$, $(s_6,1,1,s_2)$ на $(s_6,1,1,s_1)$.

Данный пример показывает, что для сильно связного приведенного автомата существуют переброски, сохраняющие его изоморфизм, однако пока не удалось построить автомат с двумя такими перебросками.

Литература

[1] *Грунский И. С., Копытова О. М.* О структуре контрольного эксперимента для определенно-диагностируемого автомата // Теория управляющих систем. — Киев: Наукова думка, 1987. — С. 40–54.

[2] *Копытова О. М.* О структуре автоматов, сохраняющих поведение при перебросках дуг // Труды VIII Международной конференции "Дискретные модели в теории управляющих систем". — М: Макс-Пресс, 2009. — С. 155–159.

[3] *Копытова О. М.* Устойчивость автоматов к неисправностям их функции переходов // Труды ИПММ АН Украины. — 2010. — № 21. — С. 57–66.

[4] *Бурлаева Е. И., Копытова О. М.* Об одном типе локальных преобразований конечного автомата // Материалы IV всеукраинской научно-технической конференции студентов, аспирантов и молодых ученых "Информационные управляющие системы и компьютерный мониторинг (ИУС КМ 2013)". — Донецк: ДонНТУ, 2013. — С. 479–484.

[5] *Грунский И. С., Козловский В. А.* Синтез и идентификация автоматов. — К.: Наукова думка, 2004. — 245 с.

Максимальные префиксные коды и проблема равенства в разных классах языков

С. Ю. Корабельщикова, Б. Ф. Мельников

kmv@atnet.ru, bormel@rambler.ru

САФУ, Архангельск, СамГУ, Тольятти

Максимальные префиксные коды рассматриваются в нескольких разных разделах широко известной монографии [1]. Мы произведем подсчет числа максимальных префиксных кодов ограниченной длины, а также опишем их связь с некоторыми вопросами теории формальных языков, в частности – с бесконечными итерациями языков.

Оценка числа максимальных префиксных кодов ограниченной длины

Для некоторых конечных алфавитов $\Delta = \{a_1, a_2, \ldots, a_n\}$ и $\Sigma = \{b_1, b_2, \ldots, b_q\}$ рассмотрим морфизм

$$h : \Delta^* \to \Sigma^*, \text{ где } h(a_1) = u_1\,,\ h(a_2) = u_2\,,\ \ldots,\ h(a_n) = u_n\,.$$

Слова $u_1, u_2, \ldots, u_n$ в алфавите Σ будем называть элементарными кодами. Если ни один элементарный код не является префиксом (началом) никакого другого элементарного кода, то говорим, что код обладает свойством префикса. Префиксный алфавитный код, задаваемый морфизмом h, является однозначно декодируемым (см. [1, стр. 134], [2, стр. 260]).

Префиксному коду $\{u_1, u_2, \ldots, u_n\}$ можно сопоставить корневое дерево, в котором каждой цепи от корня до концевой вершины соответствует элементарный код, и наоборот. Префиксный код называется максимальным, если порядок ветвления всех не концевых вершин кодового дерева равен q - мощности кодирующего алфавита Σ.

Нами произведен подсчет количества различных максимальных префиксных кодов над кодирующим алфавитом мощности q с ограничением t на длину слов. Эта задача эквивалентна задаче подсчёта всех корневых деревьев с порядком ветвления не концевых вершин q, имеющих не более t ярусов. Доказана следующая теорема:

Теорема 1. *Пусть N_t — число максимальных префиксных кодов с длиной слов не более, чем t в q-буквенном кодирующем алфавите. Тогда число максимальных префиксных кодов с длиной слов не более $t+1$ в q-буквенном кодирующем алфавите равно $(N_t + 1)^q$.*

Заметим, что всегда $N_1 = 1$, то есть имеется ровно 1 максимальный префиксный код с ограничением длины t=1, он состоит из всех однобуквенных слов кодирующего алфавита. В частности, в двоичном случае получим: $N_1 = 1$, $N_2 = 4$, $N_3 = 25$, $N_4 = 676$ и так далее.

Бесконечные итерации конечных языков

Как уже отмечалось, бесконечные итерации конечных языков были впервые рассмотрены одним из авторов настоящей статьи в [3]. Иными словами, рассматривались ω-языки вида A^ω, где A – конечный язык над заданным алфавитом Σ. (Мы предполагаем, что $|\Sigma| \geqslant 2$, а также – если специально не сказано иного – что $A \not\ni \varepsilon$.) Для двух конечных языков A и B мы рассматривали их равенство $A^\omega = B^\omega$, которое равносильно выполнению специального отношения эквивалентности:

$A \equiv B$ тогда и только тогда, когда

$$\begin{cases} (\forall\, u \in A^*)\,(\exists, v \in B^*)\,(u \in Pref(v)) \\ (\forall\, v \in B^*)\,(\exists, u \in A^*)\,(v \in Pref(u)) \end{cases} \tag{1}$$

Будем рассматривать равенство ω-языков и определенную с помощью (1) эквивалентность только для ω-языков типа A^ω и языков типа A^* .

В статье [4] приведено необходимое условие коммутирования в глобальном надмоноиде свободного моноида: если

$$A \cdot B = B \cdot A, \tag{2}$$

то $A \equiv B$.

Основной доказанный в [3] факт можно эквивалентно сформулировать следующим образом в терминах максимальных префиксных кодов:

Теорема 2. *Критерий эквивалентности $A \equiv B$.*
Для некоторой пары конечных языков A и B, таких что $A \equiv B$, существует конечный язык D (пусть $D = \{u_1, \ldots, u_n\}$), для которого выполнено следующее условие. Для некоторого нового алфавита $\Delta = \{c_1, \ldots, c_n\}$ существуют два языка $A', B' \subseteq \Delta^*$, такие что:

— *оба языка (A' и B') в качестве подмножеств содержат максимальные префиксные коды над алфавитом Δ;*
— *для морфизма*

$$h : \Delta^* \to \Sigma^*, \text{где } h(c_1) = u_1\,,\ \ldots,\ h(c_n) = u_n\,,$$

выполнены условия $h(A') = A$ и $h(B') = B$.

Сформулированный критерий эквивалентности $A \equiv B$ также является для (2) необходимым условием.

Частные случаи выполнения равенства коммутирования

Рассмотрим простые частные случаи выполнения равенства (2). На множества A и B не будем накладывать требование префиксности, однако одно из множеств (всюду пусть A) должно содержать не более 2 элементов.

В случае $|A| = 1$ приведённое выше необходимое условие равенства (2) является необходимым и достаточным.

Теперь пусть $|A| = 2$.

Несложно показать, что согласно приведённому выше критерию эквивалентности $A \equiv B$, возможны только следующие два варианта (и только они):

— $|\Sigma| = 1$, $A = \{0^k, 0^l\}$ при некоторых различных $k, l \geqslant 0$;
— $|\Sigma| = 2$, $A = \Sigma$.

Очевидно, что в первом случае условие

является, как и при $|A| = 1$, не только необходимым, но и достаточным; оно для однобуквенного алфавита Σ уже было рассмотрено. Для второго варианта справедлива теорема.

Теорема 3. *Если $A = \Sigma = \{0, 1\}$ и $A \cdot B = B \cdot A$, то*

$$B = \bigcup_{i \in I} \Sigma^i$$

для некоторого $I \subseteq N_0$.

Заметим, что мы не пользовались какими-либо ограничениями на $|B|$, поэтому в формулировке теоремы множество индексов $I \subseteq N_0$ может быть и бесконечным.

Литература

[1] *Лаллеман Ж.* Полугруппы и комбинаторные приложения. — М.: Мир, 1985. — 314 с.

[2] *Яблонский С. В.* Введение в дискретную математику. — М.: Наука, 2002. — 384 с.

[3] *Melnikov B.* The equality condition for infinite catenations of two sets of finite words // Int.J.of Found. of Comp. Sci. — 1993. — V. 4, № 3. — P. 267–274.

[4] *Алексеева А., Мельников Б.* Итерации конечных и бесконечных языков и недетерминированные конечные автоматы // Вектор науки Тольяттинского государственного университета. — 2011. — № 3. — С. 30–33.

Об одном классе задач, имеющих многоэтапный характер

А. Г. Коротченко, В. М. Сморякова

koangr@yandex.ru, smorykov@mail.ru

Нижегородский государственный университет им. Н.И. Лобачевского, г. Нижний Новгород

Рассматривается задача:

$$\sum_{j=1}^{m} Q_j(X(j)) \Rightarrow \max, \tag{1}$$

$$X(1) \in D_1(X(0)), X(0) \in D_0 \subseteq R^n,$$

$$X(j) \in D_j(X(j-1)), j = 2, ..., m,$$

$$X(j) = (x_1(j), ..., x_n(j)), D_j(X(j-1)) \subseteq R^n, j = 1, ..., m.$$

Здесь $X(0)$ - фиксированный элемент множества D_0, m - произвольное натуральное число. Наряду с задачей (1) мы также будем рассматривать следующие задачи:

$$Q_1(X(1)) \Rightarrow \max, \tag{2}$$

$$X(1) \in D_1(X(0)), X(0) \in D_0,$$

$$Q_j(X(j)) \Rightarrow \max, \tag{3}$$

$$X(j) \in D_j(X(j-1)), X(j-1) \in D_{j-1}(X(j-2)), j = 2, ..., m.$$

Пусть $X'(1)$ - решение задачи (2) для фиксированного $X(0) \in D_0$ (при условии его существования). Пусть также $X'(j)$ - решение задачи (3) при $X(j-1) = X'(j-1)$, $j = 2, ..., m$ (при условии его существования).

Пусть $A = \{a_1, ..., a_p, ...\}$ - бесконечная последовательность натуральных чисел такая, что $1 \leqslant a_1 < ... < a_p < ...$, а A_m её конечная подпоследовательность с последним элементом a_m, $1 \leqslant a_1 < ... < a_m$.

Назовём задачу (1) M_A-задачей, если для любого $a_m \in A$ и каждого $i \in A_m$, её решением является набор $< X'(1), ..., X'(i) >$. Такую задачу будем также называть многоэтапной задачей.

M_A-задачу будем называть $M(a_1, d)$-задачей, если последовательность A совпадает с арифметической прогрессией с первым членом a_1 и разностью d, где d - натуральное число.

Задачу $M(1, 1)$ для которой последовательность A совпадает с множеством всех натуральных чисел будем называть M-задачей.

Рассматриваемые задачи можно сформулировать на языке перевода некоторой многошаговой системы из указанного начального состояния в априори неизвестное конечное состояние, определяемое в процессе её функционирования. При этом множество возможных состояний системы на каждом шаге можно задавать точечно-множественным отображением. Качество же перевода системы из одного состояния в другое характеризуется локальным критерием, определённым на множестве возможных состояний системы. Локальность критерия понимается здесь так, что он связан с задачей конкретного шага или этапа. Наряду с локальным критерием система характеризуется и интегральным критерием, оценивающим поведение системы в целом.

Рассмотрим следующую задачу:

$$Q_m(x_1, ..., x_m) = \sum_{i=1}^{m} \psi_i(x_i) \Rightarrow \max, \tag{4}$$

$$X(m) = (x_1, ..., x_m) \in D_m,$$

$$D_m = \{X(m) \in P_m \subseteq R^m \mid f_i(x_{i-1}, x_i) \leqslant 0,\ i = 1, ..., m\},$$

где значение m не предполагается известным, а определяется в процессе решения задачи.

При этом $\psi_i(x_i) = c_i x_i$ локальный критерий, задаваемый на множестве, определяемом условиями $f_i(x_{i-1}, x_i) \leqslant 0$, $i = 1, ..., m$, а $Q_m(x_1, ..., x_m) = \sum\limits_{i=1}^{m} \psi_i(x_i)$ - интегральный критерий, определённый на множестве D_m.

Пусть $X'(k) = (x'_1, ..., x'_k)$ - решение задачи (4) при $m = k$, а $X''(k+1) = (x''_1, ..., x''_{k+1})$ - решение задачи (4) при $m = k + 1$ (при условии, что они существуют).

Тогда условие того, что задача (4) является M-задачей, можно записать в виде:

$$x'_i = x''_i, i = 1, ..., k, \tag{5}$$

для любого $k = 1, ..., m$, $m = 1, 2...$.

Рассмотрим задачу (4), когда $c_i > 0$, а $f_i(x_{i-1}, x_i)$ дифференцируемы на R^2_+, $i = 1, ..., m$. Обозначим производную функции $f_i(x_{i-1}, x_i)$ по x_i через $\varphi_i(x_{i-1}, x_i)$, а через $g_i(x_{i-1}, x_i)$ производную функции $f_i(x_{i-1}, x_i)$ по x_{i-1}, $i = 1, ..., m$.

Пусть функции$f_i(x_{i-1}, x_i)$, $i = 1, ..., m$, удовлетворяют следующим условиям:

Условие 1. *Для любого фиксированного* $x_{i-1} \in [0, \bar{x}_{i-1}]$ *уравнение*

$$f_i(x_{i-1}, x_i) = 0 \tag{6}$$

имеет единственный положительный корень $\bar{x}_i$, *а* $f_i(x_{i-1}, 0) < 0$ *при* $x_{i-1} > 0$ *и* $f_i(x_{i-1}, 0) \geqslant 0$ *при* $x_i \geqslant \bar{x}_i$, $i = 1, ..., m$.

Условие 2. *Для всех* $x_{i-1} \geqslant 0$, $x_i > 0$, *удовлетворяющих равенству (6), справедливы соотношения*

$$g_i(x_{i-1}, x_i) > 0, i = 1, ..., m,$$

$$0 < 1 - \frac{c_i}{c_{i-1}} \cdot \frac{\varphi_i(x_{i-1}, x_i)}{g_i(x_{i-1}, x_i)} \leqslant 1, i = 2, ..., m.$$

Пусть $\bar{x}_i$ - единственный положительный корень уравнения (6), существующий в силу условия 1, $i = 1, ..., m$ и пусть $\bar{X}(m) = (\bar{x}_1, ..., \bar{x}_m)$.

Теорема 1. *Если выполнены условия 1 и 2, то вектор* $\bar{X}(m) = (\bar{x}_1, ..., \bar{x}_m)$ *является решением задачи (4), когда* $c_i > 0$, *при* $\bar{x}_0 > 0$ *и имеют место соотношения (5), где* $f_i(\bar{x}_{i-1}, \bar{x}_i) = 0$, $i = 1, ..., m$.

Пусть теперь функции $f_i(x_{i-1}, x_i)$, $i = 1, ..., m$, $m = 1, 2...$, удовлетворяют условию 1 и условию 3.

Условие 3. *Для всех* $x_{i-1} \geqslant 0$, $x_i > 0$, *удовлетворяющих равенству (6), справедливы соотношения*

$$g_i(x_{i-1},\, x_i) > 0, i = 1, ..., m,$$

$$0 < 1 - \frac{c_i}{c_{i-1}} \cdot \frac{\varphi_i(x_{i-1}, x_i)}{g_i(x_{i-1}, x_i)} \leqslant 1,$$

$$0 < 1 - \frac{c_{i-1}}{c_{i-2}} \cdot \frac{\varphi_{i-1}(x_{i-2}, x_{i-1})}{g_{i-1}(x_{i-2}, x_{i-1})} \left(1 - \frac{c_i}{c_{i-1}} \cdot \frac{\varphi_i(x_{i-1}, x_i)}{g_i(x_{i-1}, x_i)}\right) \leqslant 1,$$

при $i = 3, 5, ..., m$, $m = 2p + 1$, $p = 1, 2, ...$

Тогда справедлива следующая теорема.

Теорема 2. *Если условия 1 и 3 выполняются, то задача (4), когда* $c_i > 0$, *является* $M(1, 2)$*-задачей.*

К рассмотренным задачам могут быть сведены задачи конструирования формул численного интегрирования систем обыкновенных дифференциальных уравнений. Указанные классы характеризуются тем, что число этапов (шагов) в каждом из рассматриваемых классов задач априори неизвестно, а определяется в процессе решения задачи. Используя описанный подход к решению многоэтапных задач, были построены процедуры, минимизирующие число вычислений правых частей систем обыкновенных дифференциальных уравнений с учётом выполнения ограничений, определяемых точностью вычислений. Приведены результаты вычислительного эксперимента по использованию построенных процедур численного интегрирования.

Литература

[1] *Коротченко А. Г.* О задачах математического программирования, имеющих многоэтапный характер // Вестник нижегородского государственного университета им. Н.И. Лобачевского. — 2011. — Т. 1, — С. 183–187.

[2] *Korotchenko A. G., Smoryakova V. M.* Multistage mathematical programming problems // Proceedings of the international conference "NUMERICAL COMPUTATIONS: THEORY AND ALGORITHMS.". — 2013. — p. 88.

[3] *Коротченко А. Г., , Лапин А. В.* Об одном алгоритме численного интегрирования с оптимальным выбором шага // Вестник нижегородского государственного университета им. Н.И. Лобачевского. — 2001. № 2(24). — С. 270-278.

[4] *Коротченко А. Г., Лапин А. В.* О построении приближенно оптимального алгоритма численного интегрирования // Вестник нижегородского государственного университета им. Н.И. Лобачевского — 2003. № 1(26). — С. 189-195.

Вложения формализмов знаний

К. И. Костенко

kostenko@kubsu.ru

Кубанский государственный университет, Краснодар

Формальные системы, реализующие логико-математическое описание содержания областей знаний называются формализмами знаний (кратко - формализмами) Существующие формализмы составляют индуктивно развиваемое семейство подходов, основанных на разнообразных схемах задания знаний [1]. Востребованность универсальной логико-математической модели связана с необходимостью унификации такого многообразия, обеспечения возможности совместного исследования или практического использования отличающихся форматов представления знаний. Предлагаемое уточнение понятия формализма знаний основно на общих алгебраических и алгоритмических инвариантах существующих и возможных формализмов. При этом инвариант понимается как универсальная категория для сущностей всякого формализма знаний.

Формализмы знаний

Уточнение понятия формализма знаний составляют конструкты, позволяющие рассматривать отдельные формализмы как дедуктивные реализации универсальной абстрактной модели. Такую модель составляют механизмы построения структурированных объектов, являющихся представлениями абстрактных знаний, и механизмы сравнения объектов. Полезность универсальной модели связана с унификацией представлений о природе формализованных знаний, развивающим понимание существующих моделей знаний, возможностей их практического использования.

Определение. Формализм знаний это - четвёрка $\Im = (M_\Im, D_{M_\Im}, \circ, \subseteq)$, где $M_\Im$ и $D_{M_\Im}$ - разрешимые множества, а $\circ : D_{M_\Im} \times D_{M_\Im} \to D_{M_\Im}$ - вычислимая операция композиции и $\subseteq$ - разрешимое отношение вложения на $D_{M_\Im}$.

Здесь множество $M_\Im$ (представлений знаний) является разрешимым подмножеством множества $D_{M_\Im}$ (фрагментов знаний) и содержит специальный пустой элемент Λ.

Конструкты универсальной модели формализма знаний являются очень общими. Это влечёт обширность класса конкретных формализмов. Полезность приведённого понятия зависит от глубины отражения концептуальных представлений о природе формализованных знаний, обеспечивающего возможность рассмотрения существующих моделей знаний как дедуктивных следствий универсальной формализации. Рассмотрим применение приведённого определения к задаче сравнения конкретных подходов к формализованному представлению знаний.

Определение. Формализм $\Re$ алгебраически вкладывается в формализм $\Im$ ($\Re \sqsubseteq_A \Im$), если существует такое вычислимое инъективное отображение $\xi : D_{M_\Re} \to D_{M_\Im}$, что

$$\forall C_1, C_2 \in D_{M_\Re}(\xi(C_1 \circ C_2) = \xi(C_1) \circ \xi(C_2)).$$

Определение. Формализм $\Re$ семантически вкладывается в формализм $\Im$ ($\Re \sqsubseteq_S \Im$), если существует такое вычислимое инъективное отображение $\xi : D_{M_\Re} \to D_{M_\Im}$, что

$$\forall C_1, C_2 \in D_{M_\Re}(C_1 \subseteq C_2 \to \xi(C_1) \subseteq \xi(C_2)).$$

Определение. Формализм $\Re$ вкладывается в формализм $\Im$ ($\Re \sqsubseteq \Im$), если существует такое вычислимое инъективное отображение $\xi : D_{M_\Re} \to D_{M_\Im}$, для которого $\Re \sqsubseteq_A \Im$ и $\Re \sqsubseteq_S \Im$.

Произвольные формализмы $\Im_1$ и $\Im_2$ эквивалентные, если $\Im_1 \sqsubseteq \Im_2$ и $\Im_2 \sqsubseteq \Im_1$. Если элементами $D_{M_\Im}$ являются множества, а $\circ$ и $\subseteq$ определяются как объединение и вложение множеств, то $\Im$ называется теоретико-множественным формализмом. Для таких формализмов понятия алгебраического и семантического вложения равносильны. Примеры формализмов, для которых имеет место только алгебраическое или семантическое вложение, приведены в [2].

Сравнение существующих моделей знаний

Определим формализмы для: атомарных продукционных систем (APS), образовательных пространств (LS) [3], баз знаний в дескрипционной логике (ALC) [2], абстрактных пространств знаний (AKS) [5], иерархических семантических сетей (SN) [6]. Сложность развёртывания системы определений связана с необходимостью уточнения инвариантов фрагмента знаний, семантической и алгебраической структуры для нескольких разных подходов к представлению знаний. Концепты фрагмента, композиции и вложения в обозначенных и других известных формализмах нередко представлены неявно и, в общем случае, допускают отличающиеся уточнения.

Теорема 1. *Для формализмов APS, LS, ALC, AKS и SN справедливы только такие вложения, которые представлены в последовательности*

$$APS \sqsubseteq LS \sqsubseteq ALC \sqsubseteq AKS \sqsubseteq SN.$$

Соотношения формализмов, представляющих рассмотренные модели представления знаний, не сохраняются, если ограничиться только алгебраическим или семантическим вложением.

Обозначим как K множество всех формализмов знаний. Наибольший в отношении вложения формализм называется универсальным. Существование такого формализма обосновывается в следующей теореме.

Теорема 2. $\exists \Im_U \in K \, \forall \Im \in K (\Im \sqsubseteq \Im_U)$.

Доказательство последней теоремы состоит в построении примера формализма, в котором операция композиции и отношение вложения моделируются на основе пересчёта множества всех вычислимых отображений вида $f : N \times N \to N$ и отношений $\rho \subseteq N \times N$, реализующих теоретико-рекурсивное представление операций композиции и отношений вложения на произвольных бесконечных разрешимых множествах.

Формализм дескрипционных логик ALC является базовым для семейства таких логик, являющихся расширениями указанного формализма. Они получаются развитием систем конструктов ALC, используемых в формулах языка. Примерами таких расширений являются $SHIF$ и $SHOIN$. Для этих расширений выполняются те же сравнения с рассмотренными выше формализмами, что и для ALC.

Теорема 3. $\Im_U \overline{\sqsubseteq} SN$.

Иерархические семантические сети не являются примером универсального формализма знаний. Это свойство связано с тем, что аксиоматика формализмов допускает возможность композиции одного и того же фрагмента знания бесконечным многообразием способов. Такое свойство не имеет места для любого из формализмов, представленных в теореме 1.

Вложения формализмов семантических сетей

Формализм иерархических семантических сетей представляет теоретический и практический интерес как развитие формализмов дескрипционных логик и абстрактных пространств знаний. Иерархические семантические сети удобны в качестве формата представления в связном виде реализаций универсальной логико-математической модели области знаний, основанных на алгебраических и алгоритмических инвариантах, и слабоформализованных моделей областей знаний, основанных на теоретико-множественных конструкциях и алгоритмах их обработки. Полное семантическое представление иерархической семантической сети составляют сети, каждая из которых размещается в некотором ярусе. Всякая неэлементарная вершина полного представления связана с соответствующей ей семантической сетью. Последняя сеть размещается в следующем ярусе иерархии. Глубина сети определяется числом уровней полного представления. Обозначим как SN_i - класс сетей глубины $i, i = 1, 2, ...$

Теорема 4. $\forall i \in N (SN_i \equiv_S SN_{i+1})$.

Теорема 5. $\forall i \in N (SN_i \equiv_S SN)$.

Работа выполнена при поддержке РФФИ, проект № 13-01-96513.

Литература

[1] *Sloman A.* Why We Need Many Knowledge Representation Formalisms // Proceedings BCS Expert Systems Conf. — Cambridge University Press, 1985, — С. 163–183.

[2] *Костенко К. И.* Вложения семантических сетей // Экологический вестник научных центров Черноморского экономического сотрудничества. — 2013. № 2. — С. 58–66.

[3] *Falmagne J.-Cl., Doignon J.-P.* Learning Spaces. — Berlin Heidelberg, 2011. — 417 p.

[4] *Schmidt-Schau S. M., Smolka G.* Attributive concept descriptions with complements // Artificial Intelligence. — 1991. — V. 48, — P. 1–26.

[5] *Костенко К. И.* Компоненты и операции абстрактных пространств знаний // Всероссийская конференция ЗОНТ09. — Новосибирск: ИМ СО РАН, 2009, — Т. 2, — С. 36–40.

[6] *Кузнецов И. П.* Семантические представления. — М.:Наука, 1986. — 304 с.

Вычислительное исследование дискретных моделей конформного поведения

С. Е. Кочемазов, А. А. Семенов, Д. Л. Фисенко

veinamond@gmail.com, biclop.rambler@yandex.ru, fisenk-dima@mail.ru

ИДСТУ СО РАН, ИМЭИ ИГУ, Иркутск

Конформным [1] называется поведение мультиагентных систем, при котором, принимая решение о выполнении некоторого действия, агент ориентируется на аналогичные решения других агентов, прямо или косвенно на него влияющих. Более детально, поведение агента называется конформным, если он принимает решение «действовать» лишь тогда, когда не менее определенного процента влияющих на него (как правило, напрямую) агентов принимает решение «действовать». В противном случае агент бездействует. Данный тип поведения весьма распространен в реальной жизни. Также интересно, своего рода, противоположное поведение: агент бездействует, если число действующих соседей превосходит некоторый порог, и действует в противном случае. Такое поведение в [1] названо антиконформным. В [1] предложена теоретико-игровая модель этих типов поведения. В [2] была введена автоматная (или дискретно-автоматная) модель конформного поведения, по смыслу аналогичная изученным ранее в [3], [4], [5] дискретно-автоматным моделям генных сетей с пороговыми весовыми функциями. По сути, рассмотренная в [2] модель является сетью Кауффмана [6] с пороговыми весовыми функциями, реализующими принцип конформности.

В [2] в модель были введены агенты, называемые агитаторами и лоялистами. Агитаторы действуют всегда без учета мнения своих соседей. Лоялисты, напротив, всегда бездействуют. Можно сказать, что агитаторы и лоялисты не проявляют конформного поведения. Агенты, проявляющие конформное по-

ведение, называются простыми. В [2] решались задачи размещения на графе сети относительно небольшого числа агитаторов, которые бы за малое число контактов переводили всех простых агентов в состояние действия. Также рассматривалась задача размещения относительно небольшого числа лоялистов с целью блокирования агитаторов. В качестве конкретных сетей использовались графы на 100 вершинах, сгенерированные в соответствии с моделью Эрдеша-Реньи.

В настоящей работе мы развиваем результаты [2] в следующих направлениях. Во-первых, мы рассматриваем более сложную модель влияния и полагаем, что на принятие агентом решения «действовать» его непосредственные соседи влияют с разными весами. Такая постановка эквивалентна рассмотрению модели, в которой ближайшее окружение агента, т.е. вершины, связанные с ним дугами, влияет на его мнение в наибольшей степени; вершины, отстоящие от него на расстояние (в графе), равное 2 – в меньшей степени, и т.д.. Второе отличие от результатов [2] состоит в том, что, помимо графов Эрдеша-Реньи (G_{np}-графы) [7], мы рассмотрели графы Уоттса-Строгатца (WS-графы) [8]. Для сгенерированных сетей решались задачи размещения агитаторов и лоялистов, в целом аналогичные задачам, рассмотренным в [2]. Для поиска соответствующих размещений использовался SAT-подход [9]. Сведение рассматриваемых задач к SAT осуществлялось в соответствии с принципами, подробно описанными в [2]. Ниже мы кратко останавливаемся на основных отличиях рассмотренных моделей от изученных в [2] и приводим результаты вычислительных экспериментов.

Рассматривается ориентированный граф G на n вершинах. Предполагаем, что произвольная вершина v_i, $i \in \{1, \ldots, n\}$, интерпретирует некоторого агента. В каждый момент времени $t \in \{0, 1, \ldots\}$ произвольной вершине v_i приписано число $x_i(t) \in \{0, 1\}$, называемое весом v_i в момент t. При переходе от момента t к моменту $t+1$ осуществляется синхронный пересчет весов всех вершин. Вершине v_i, $i \in \{1, \ldots, n\}$, сопоставляется множество $V_i = \{v_{s_1}, \ldots, v_{s_l}\}$, $\{s_1, \ldots, s_l\} \subset \{1, \ldots, n\}$, $l < n$, вершин графа G, дуги из которых входят в вершину v_i (можно сказать, что эти вершины образуют окружение вершины v_i). Каждой вершине из V_i приписывается фиксированное натуральное число $w^i_{s_j}$, $j \in \{1, \ldots, l\}$, которое отражает степень влияния агента $v_{s_j} \in V_i$ на мнение агента v_i. Полагаем, что чем больше данное число, тем больше влияние. Считаем, что степень влияния не может превосходить некоторого заданного натурального r. Помимо этого каждой вершине v_i приписывается число $\theta_i \in [0, 1]$, которое далее называем порогом конформности соответствующего агента. Вершина v_i имеет в момент времени t вес $x_i(t) = 1$, если представляемый ею агент принимает решение «действовать», и $x_i(t) = 0$, если агент принимает решение "бездействовать".

Динамика изменения веса произвольного агента v_i, $i \in \{1, \ldots, n\}$, задается следующим образом:

$$x_i(t+1) = \begin{cases} 1, \sum\limits_{v_j \in V_i} x_j(t) > \lfloor \theta_i \cdot \sum\limits_{v_j \in V_i} w^i_j \rfloor \\ 0, \sum\limits_{v_j \in V_i} x_j(t) \leqslant \lfloor \theta_i \cdot \sum\limits_{v_j \in V_i} w^i_j \rfloor \end{cases}$$

В рамках описанной модели, агитаторами являются вершины, вес которых всегда равен 1, а лоялистами - вершины, вес которых всегда равен 0. В вычислительных экспериментах мы рассмотрели следующие две задачи.

Задача 1. Для начального состояния, соответствующего бездействию всех простых агентов, найти такую расстановку не более, чем P, $P < n$, агитаторов, что число действующих агентов через k шагов составит не менее Q, $Q > P$.

Задача 2. По расстановке агитаторов, найденной в результате решения задачи 1, для начального состояния, в котором все простые агенты действуют, найти такую расстановку не более, чем U, $U < n - P$, лоялистов, что число действующих агентов через k шагов составит не более W, $P \leqslant W < n - U$.

В вычислительных экспериментах мы использовали SAT-решатель Minisat 2.2, который запускался на одном ядре процессора AMD Opteron 6276 в рамках вычислительного кластера „Академик В.М. Матросов“ ИДСТУ СО РАН. Во всех экспериментах рассматривались графы на 200 вершинах. Использовались следующие значения параметров: $k = 10$, $r = 3$, $P = 40$, $Q = 160$, $U = 60$, $W = 40$. Для каждого значения „вероятность дуги“ было сгенерировано по 10 тестов. В таблицах 1 и 2 представлены результаты экспериментов по решению приведенных задач для графов, построенных в соответствии с моделью Эрдеша-Реньи (G_{np}) и для графов, построенных в соответствии с моделью Уоттса-Строгатца (WS), соответственно.

Таблица 1. Результаты вычислительных экспериментов по поиску решений задач 1 и 2 для G_{np}-графов.

Вероятность дуги	Средний размер КНФ, Кб.	Среднее время реш-я задачи 1, с.	Среднее время реш-я задачи 2, с.
0,2	221771	156,15	107,61
0,3	396095	253,75	408,32
0,4	542102	375,88	122,12

Таблица 2. Результаты вычислительных экспериментов по поиску решений задач 1 и 2 для WS-графов.

Вероятность дуги	Средний размер КНФ, Кб.	Среднее время реш-я задачи 1, с.	Среднее время реш-я задачи 2, с.
0,2	86141	38,54	24,52
0,3	86573	37,04	22,52
0,4	89776	52,55	37,04

Работа выполнена при частичной поддержке проектов РФФИ: № 14-07-31172 мол_а и № 14-07-00403 а.

Литература

[1] *Бреер В. В.* Теоретико-игровые модели конформного поведения // Автоматика и телемеханика. — 2012. — № 10. — С. 111–126.

[2] *Семёнов А. А., Кочемазов С. Е.* О дискретно-автоматных моделях конформного поведения // Управление большими системами. — 2013. — № 46. — С. 266–292.

[3] *Григоренко Е. Д., Евдокимов А. А., Лихошвай В. А., Лобарева И. А.* Неподвижные точки и циклы автоматных отображений, моделирующих функционирование генных сетей // Вестник Томского гос. ун-та. Приложение. — 2005. — № 14. — С. 206–212.

[4] *Евдокимов А. А.* Дискретные модели генных сетей: анализ и сложность функционирования // Вычислительные технологии. — 2008. — Т. 13, № 3. — С. 31–37.

[5] *Евдокимов А. А., Кочемазов С. Е., Семёнов А. А.* Применение символьных вычислений к исследованию дискретных моделей некоторых классов генных сетей // Вычислительные технологии. — 2011. — Т. 16, № 1. — С. 30–47.

[6] *Kauffman S. A.* Metabolic stability and epigenesis in randomly constructed genetic nets // J. Theor. Biol. — 1969. — V. 22, № 3. — P. 437–467.

[7] *Erdos P., Renyi A.* On random graphs // Publ. Math. — 1959. — V. 6. — P. 290–297.

[8] *Watts D. J., Strogatz S. H.* Collective dynamics of small-world networks // Nature. — 1998. — V. 393. — P. 440–442.

[9] *Biere A., Heule V., van Maaren H., Walsh T.* Handbook of Satisfiability. — Amsterdam: IOS PRESS, 2009. — 980 p.

Об одной нижней оценке сложности вычисления элементов конечных абелевых групп

В. В. Кочергин

vvkoch@yandex.ru

МГУ им. М. В Ломоносова, Москва

Пусть G — конечная абелева группа (групповую операцию будем называть умножением). Подмножество $B = \{a_1, \ldots, a_q\}$ элементов группы будем называть *базисом* в группе G, если G раскладывается в прямое произведение циклических подгрупп, порожденных элементами множества B:

$$G = \langle a_1 \rangle_{u_1} \times \ldots \times \langle a_q \rangle_{u_q},$$

где u_i — порядок элемента a_i, $i = 1, \ldots, q$.

Для каждого элемента g группы G определим его *сложность реализации над базисом* B, обозначаемую через $L(g; B)$, как минимальное число операций умножения, достаточное для вычисления элемента g с использованием элементов множества B (при этом все уже вычисленные элементы могут быть использованы многократно).

Сложность $L(G; B)$ *конечной абелевой группы* G *над базисом* B определим так:

$$L(G; B) = \max_{g \in G} L(g; B).$$

Положим

$$LM(G) = \max_{B:\ B \text{ — базис } G} L(G, B).$$

Так как конечная абелева группа G полностью определяется вектором $\mathbf{v} = (v_1, \ldots, v_q)$ порядков примарных циклических подгрупп группы G, то вместо обозначения $LM(G)$ можно использовать обозначение $M(\mathbf{v})$.

Исследуется одна из задач, поставленных О. Б. Лупановым [1] в области сложности вычислений в конечных абелевых группах — найти числовую функцию $f(\mathbf{v})$, определенную на векторах $\mathbf{v}$, характеризующих порядки примарных циклических групп, с помощью которой выражалась бы величина $M(\mathbf{v})$ (хотя бы асимптотически или с точностью до порядка при условии, что порядок всей группы стремится к бесконечности).

Пусть $G = \langle a_1 \rangle_{u_1} \times \ldots \times \langle a_q \rangle_{u_q}$, т. е. $B = \{a_1, \ldots, a_q\}$ — базис в группе G. Из мощностных соображений (см., например, [2]) вытекает следующий факт.

Утверждение 1. *Для произвольного положительного ε найдется такое положительное $m(\varepsilon)$, что для сложности любой конечной абелевой группы G над базисом B при выполнении условия $|G| > m(\varepsilon)$ справедлива оценка*

$$L(G; M_G) \geqslant \frac{\log |G|}{\log\log |G|} \left(1 + (1-\varepsilon)\frac{\log\log\log |G|}{\log\log |G|}\right)$$

(*здесь и далее все логарифмы берутся по основанию* 2).

Из результатов работ [3, 4, 5] следует такая верхняя оценка.

Утверждение 2. *Пусть $G = \langle a_1 \rangle_{u_1} \times \ldots \times \langle a_q \rangle_{u_q}$, $B = \{a_1, \ldots, a_q\}$. Тогда при $|G| \to \infty$*

$$L(G, B) \leqslant \frac{\log |G|}{\log\log |G|}(1 + o(1)) + \log(\max_i u_i)(1 + o(1)) + O(q).$$

В работе [1] сформулировано следующее

Утверждение 3. *Пусть $G = \langle a_1 \rangle_{u_1} \times \ldots \times \langle a_q \rangle_{u_q}$, $B = \{a_1, \ldots, a_q\}$. Тогда при $|G| \to \infty$ справедливы соотношения*

$$\frac{\log |G|}{\log\log |G|} \lesssim L(G, B) \lesssim \frac{\log |G|}{\log\log |G|} + \log(\max_i u_i) + q.$$

С помощью этого утверждения и простой нижней оценки $L(G, B) \geqslant \log(\max\limits_i u_i - 1) + q - 1$ при условии $|G| \to \infty$ легко устанавливается порядок роста функционала $L(G, B)$.

Утверждение 4. *Пусть $G = \langle a_1 \rangle_{u_1} \times \ldots \times \langle a_q \rangle_{u_q}$, $B = \{a_1, \ldots, a_q\}$. Тогда при $|G| \to \infty$ справедливы соотношения*

$$\max\left(\frac{\log |G|}{\log\log |G|},\ \log(\max_i u_i) + q.\right) \lesssim L(G, B) \lesssim \frac{\log |G|}{\log\log |G|} + \log(\max_i u_i) + q.$$

Переходя к оценкам величины $M(\mathbf{v})$, для вектора $\mathbf{v} = (v_1, \ldots, v_q)$ обозначим через $\|\mathbf{v}\|$ величину $v_1 v_2 \ldots v_q$.

Утверждение 5 [1]. *При* $\|\mathbf{v}\| \to \infty$ *выполняются соотношения*

$$\frac{\log \|\mathbf{v}\|}{\log\log \|\mathbf{v}\|} \lesssim M(\mathbf{v}) \lesssim \log \|\mathbf{v}\|.$$

При этом как для нижней так и для верхней оценки несложно построить последовательности векторов $\mathbf{v}$, для которых соответствующая оценка асимптотически достигается. Однако оценки из утверждения 5 не устанавливают порядок роста величины $M(\mathbf{v})$.

Пусть теперь группа G представлена как прямое произедение своих примарных циклических компонент:

$$G = \langle a_1\rangle_{v_1} \times \ldots \times \langle a_q\rangle_{v_q}.$$

Обозначим через $q(\mathbf{v})$ длину вектора $\mathbf{v} = (v_1, v_2, \ldots, v_q)$. Далее для каждого простого делителя p_i величины $\|\mathbf{v}\|$ обозначим через $P_i(\mathbf{v})$ максимальный из порядков примарных циклических подгрупп, являющихся степенями числа p_i. Теперь положим $P(\mathbf{v}) = \prod P_i(\mathbf{v})$, где произведение берется по всем простым делителям числа $\|\mathbf{v}\|$. Легко заметить, что величина $P(\mathbf{v})$ численно равна максимальному значению порядка среди всех элементов группы G.

Очевидно, что справедливы оценки $M(\mathbf{v}) \geqslant q(\mathbf{v})-1$ и $M(\mathbf{v}) \geqslant \log(P(\mathbf{v})-1)$, но неравенство $M(\mathbf{v}) \geqslant q(\mathbf{v}) - 1 + \log(P(\mathbf{v}) - 1)$ в общем случае уже является неверным. Однако справедлива

Теорема 1. *При* $\|\mathbf{v}\| \to \infty$ *справедливо асимптотическое неравенство*

$$M(\mathbf{v}) \gtrsim q(\mathbf{v}) + \log P(\mathbf{v}).$$

Простым следствием теоремы 1 и верхней оценки из утверждения 3 является следующая

Теорема 2. *При* $\|\mathbf{v}\| \to \infty$ *выполняются соотношения*

$$\max\left(\frac{\log \|\mathbf{v}\|}{\log\log \|\mathbf{v}\|},\ q(\mathbf{v}) + \log P(\mathbf{v})\right) \lesssim M(\mathbf{v}) \lesssim \frac{\log \|\mathbf{v}\|}{\log\log \|\mathbf{v}\|} + q(\mathbf{v}) + \log P(\mathbf{v}).$$

Эта теорема уставливает порядок роста величины $M(\mathbf{v})$, причем верхняя оценка может превышать нижнюю асимптотически не более чем в два раза. Кроме того, в случае, когда одна из величин $\frac{\log \|\mathbf{v}\|}{\log\log \|\mathbf{v}\|}$ и $q(\mathbf{v}) + \log P(\mathbf{v})$ растет существенно быстрее другой, теорема 2 устанавливает асимптотику роста функционала сложности $M(\mathbf{v})$.

Работа выполнена при финансовой поддержке РФФИ, проект № 14–01–00598.

Литература

[1] *Кочергин В. В.* Некоторые задачи сложности вычисления элементов конечных абелевых групп // Материалы XI Международного семинара «Дискретная математика и ее приложения» , посвященного 80-летию со дня рождения академика О. Б. Лупанова (Москва, МГУ, 18–23 июня 2012 г.) — М.: Издательство механико-математического факультета МГУ, 2012. — С. 135–138.

[2] *Лупанов О. Б.* О синтезе некоторых классов управляющих систем // Проблемы кибернетики, вып. 10. — М.: Физматгиз, 1963. — С 63–97.

[3] *Кочергин В. В.* О сложности вычислений в конечных абелевых группах // ДАН СССР. — 1991. — Т. 317, № 2. — С. 291–294.

[4] *Кочергин В. В.* О сложности вычислений в конечных абелевых группах // Математические вопросы кибернетики. Вып. 4. — М.: Наука, 1992. — С. 178–217.

[5] *Гашков С. Б., Кочергин В. В.* Об аддитивных цепочках векторов, вентильных схемах и сложности вычисления степеней // Методы дискретного анализа в теории графов и сложности. — Новосибирск, 1992. — Вып. 52. — С. 22–40.

Кибернетический подход к изучению вероятностной модели адаптивного управления конфликтными потоками

Е. В. Кудрявцев, М. А. Федоткин

evgkudryavcev@gmail.com, fma5@rambler.ru

Нижегородский государственный университет им. Н.И. Лобачевского

Введение

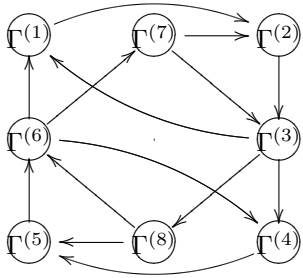
В данный работе рассматривается процесс управления конфликтными неординарными пуассоновскими потоками в классе адаптивных нециклических алгоритмов. На обслуживание поступают 2 статистически независимых потока Π_1 и Π_2. В каждый вызывающий момент по потоку Π_j, где $j = 1, 2$, поступает с интенсивностью λ_j пачка из k требований с вероятностями $Q_j(k)$

$$Q_j(1) = (1 + \alpha_j + \frac{\alpha_j\beta_j}{1-\gamma_j})^{-1}, \quad Q_j(2) = \alpha_j(1 + \alpha_j + \frac{\alpha_j\beta_j}{1-\gamma_j})^{-1}$$

$$Q_j(k) = \alpha_j\beta_j\gamma_j^{k-3}(1 + \alpha_j + \frac{\alpha_j\beta_j}{1-\gamma_j})^{-1}, \quad k \geqslant 3,$$

где параметры α_j, β_j и γ_j определены в работе [1].

В рассматриваемой системе с ожиданием и без потерь заявок входные потоки являются конфликтными, т. е. обслуживание требований различных потоков происходит в непересекающиеся промежутки времени. Выбран адаптивный алгоритм смены состояний $\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(8)}$ обслуживающего устройства. Смена состояний обслуживающего устройства задается графом вида



Состояние $\Gamma^{(3j-2)}$ (начало обслуживания заявок j-го потока) соответствует первому этапу периода обслуживания j-го потока. Длительность обслуживания одной заявки, поступившей из накопителя O_j потока Π_j, считается равной постоянной величине $\mu_{j,1}^{-1}$. Длительность пребывания в $\Gamma^{(3j-2)}$ равна T_{3j-2}. Состояние $\Gamma^{(3j-1)}$ (продолжение обслуживания заявок j-го потока) соответствует второму этапу периода обслуживания j-го потока. Длительность обслуживания в этом состоянии одной заявки считается равной постоянной величине $\mu_{j,2}^{-1} < \mu_{j,1}^{-1}$. Длительность пребывания в состоянии $\Gamma^{(3j-1)}$ — случайная величина, принимающая значения kT_{3j-1} , $k = \overline{1, n_j}$, где n_j — максимальное число продлений. Состояние $\Gamma^{(3j)}$ (дообслуживание заявок j-го потока) соответствует режиму переналадки для j-го потока. Длительность пребывания в этом состоянии равна T_{3j}. Длительность обслуживания одной заявки в состоянии $\Gamma^{(3j)}$ равна величине $\mu_{j,2}^{-1}$. Состояние $\Gamma^{(6+j)}$ (обслуживание для j-го потока) соответствует первому этапу периода обслуживания j-го потока, в случае, когда возможен мгновенный переход в состояние $\Gamma^{(3j)}$. Длительность пребывания в $\Gamma^{(6+j)}$ является случайной величиной. Максимальное время пребывания в этом состоянии составляет T_{3j-2}. Такой алгоритм управления конфликтными потоками был использован в системе «Спрут» регулирования транспортом на перекрестках.

Математическая модель управляющей системы

Кибернетический подход Ляпунова-Яблонского подразумевает представление управляющей системы в виде схемы с определенными блоками, выделение информации, координат и функций. Схема управляющей системы отражает ее структурное строение и позволяет выделить связи между ее блоками. Схема системы включает следующие структурные блоки:

1) первый тип **входных полюсов** — конфликтные входные неординарные пуассоновские потоки Π_1 и Π_2;

2) второй тип **входных полюсов** — потоки насыщения Π_1^* и Π_2^* (выходные потоки системы при ее максимальной загрузке и эффективном функционировании);

3) **внешняя память** — неограниченные накопители очередей O_1 и O_2 по входным потокам Π_1 и Π_2;

4) **блок по переработке внешней памяти** — экстремальная стратегия обслуживания, при которой из накопителя O_j на обслуживание выбирается максимально возможное число заявок;

5) **внутренняя память** — обслуживающее устройство с 8 состояниями $\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(8)}$;

6) **блок по переработке информации внутренней памяти** — адаптивный алгоритм смены состояний обслуживающего устройства.

Информация системы есть математическое описание или кодирование всех ее блоков. Например, информация о входном полюсе второго типа задается потоками насыщения. Координатами являются номер входного потока и потока насыщения, номер очереди, номер заявки в очереди на обслуживание,

номер состояния обслуживающего устройства. Функцией системы является управление входными потоками и обслуживание заявок.

Согласно кибернетическому подходу необходимо выбрать дискретное время функционирования системы. Будем рассматривать систему в моменты $\tau_i, i = 0, 1, \ldots$ смены состояний обслуживающего устройства. Для математического описания блоков системы введем следующие случайные величины: $\eta_{j,i}$ — число заявок j-го потока, которые поступили в систему на промежутке $[\tau_i; \tau_{i+1})$; η'_i — случайный вектор, принимающий значения $y_0 = (0,0)$, $y_1 = (1,0)$ и $y_2 = (0,1)$ в зависимости от того, по какому из потоков заявка поступила в систему первой на i-ом такте $[\tau_i, \tau_{i+1})$; $\overline{\xi}_{j,i}$ — число заявок j-го потока, которые реально покинут систему на промежутке $[\tau_i, \tau_{i+1})$; $\xi_{j,i}$ — максимально возможное число заявок j-го потока, которое система может обслужить на промежутке $[\tau_i, \tau_{i+1})$; $\varkappa_{j,i}$ — число заявок j-го потока, которые находятся в накопителе O_j в момент τ_i. Экстремальная стратегия обслуживания (обслуживается максимально возможное число требований) описывается следующими равенствами $\overline{\xi}_{j,i} = \min\{\varkappa_{j,i} + \eta_{j,i}; \xi_{j,i}\}$, при $\Gamma_i \in \Gamma \backslash \{\Gamma^{(3)}, \Gamma^{(6)}\}$, $i \geqslant 0$; и $\overline{\xi}_{j,i} = \min\{\varkappa_{j,i}; \xi_{j,i}\}$, если $\Gamma_i \in \{\Gamma^{(3)}, \Gamma^{(6)}\}$. Применение кибернетического подхода позволило установить, что динамика изменения состояния $(\Gamma_i, \varkappa_i)$ системы обслуживания определяется рекуррентным соотношением $(\Gamma_{i+1}, \varkappa_{i+1}) = (u(\Gamma_i, \varkappa_i, \eta'_i), v(\Gamma_i, \varkappa_i, \eta_i, \xi_i))$ для трехмерной случайной последовательности $\{(\Gamma_i, \varkappa_i); i = 0, 1, \ldots\}$, где $\varkappa_i = (\varkappa_{1,i}, \varkappa_{2,i})$. Функции $u(.)$ и $v(.)$ задаются соответственно формулами (1) и (2)

$$\Gamma_{i+1} = \begin{cases} \Gamma^{(3j-2)}, & \left\{\left[\Gamma_i = \Gamma^{(3s)}\right] \& \left[(\varkappa_{j,i} > 0) \vee (\varkappa_{s,i} \geqslant K_s) \vee (\eta'_i = y_j)\right]\right\} \vee \\ & \vee \left\{\left[\Gamma_i = \Gamma^{(3j)}\right] \& [\varkappa_{s,i} = 0] \& [\varkappa_{j,i} \leqslant K_j] \& [\eta'_i = y_j]\right\}; \\ \Gamma^{(3j-1)}, & \left\{\Gamma_i = \Gamma^{(3j-2)}\right\} \vee \left\{\left[\Gamma_i = \Gamma^{(6+j)}\right] \& [\eta'_i = y_j]\right\}; \\ \Gamma^{(3j)}, & \left\{\Gamma_i = \Gamma^{(3j-1)}\right\} \vee \left\{\left[\Gamma_i = \Gamma^{(6+j)}\right] \& [\eta'_i \neq y_j]\right\}; \\ \Gamma^{(6+j)}, & \left[\Gamma_i = \Gamma^{(3s)}\right] \& [\varkappa_{j,i} = 0] \& [\varkappa_{s,i} < K_s] \& [\eta'_i = y_0], \end{cases} \tag{1}$$

$$\varkappa_{i+1} = \begin{cases} \max\{0; \varkappa_i + \eta_i - \xi_i\}, \text{ если } \Gamma_i \in \Gamma \backslash \{\Gamma^{(3)}, \Gamma^{(6)}\}; \\ \eta_i + \max\{0; \varkappa_i - \xi_i\}, \text{ если } \Gamma_i \in \{\Gamma^{(3)}, \Gamma^{(6)}\}, \end{cases} \tag{2}$$

где $j, s = 1, 2; j \neq s$, и K_1, K_2 — целочисленные положительные константы.

Таким образом функция $u(.)$ формализует блок по переработке внутренней памяти, а функция $v(.)$ — блок по переработке внешней памяти. Как видно из полученных формул блоки имеют между собой сложные зависимости. Для построенной векторной последовательности $\{(\Gamma_i, \varkappa_i); i = 0, 1, \ldots\}$, являющейся вероятностной моделью рассматриваемой системы, была доказана следующая теорема.

Теорема 1. *Случайная векторная последовательность* $\{(\Gamma_i, \varkappa_i); i = 0, 1, \ldots\}$ *с заданным начальным распределением вектора* $\{(\Gamma_0, \varkappa_0)\}$ *является марковской.*

Работа выполнена в ННГУ по госбюджетной теме №01201456585 «Математическое моделирование и анализ стохастических эволюционных систем и процессов принятия решений».

Литература

[1] *Федоткин М. А., Кудрявцев Е. В.* Управляющие системы и механизм образования транспортных пачек на магистралях с интенсивным движением // Проблемы теоретическо кибернетики. Материалы XVI Международной конференции. — Нижний Новгород: Изд-во Нижегородского госуниверситета, 2011. — С. 503–507.

Сохранение направления движения коллективом автоматов без компаса в решётчатой среде

А. Н. Курганский, С. В. Сапунов

topologia@mail.ru, sapunov_sv@iamm.ac.donetsk.ua

ИПММ НАН Украины, Донецк

Теория автоматов в своем естественном развитии пришла к проблематике связанной с автоматным анализом дискретных и непрерывных структур или сред (изображений, графов, формальных языков и т.д) [1]. Исследование в этом направлении получили широкий спектр приложений, например, в задачах навигации мобильных роботов [2, 3]. Взаимодействие автоматов (или их коллективов) со средой зачастую представляется как процесс перемещения автоматов по среде. При этом полагается, что автоматы различают направления в среде, т.е. как бы обладают компасом [1]. В настоящей работе рассматриваются коллективы автоматов без компаса, взаимодействующие между собой в среде, представляющей собой целочисленную 2-мерную решетку. Взаимодействуя со средой, автомат получает на вход информацию о наличии или отсутствии других автоматов в окрестности своей вершины, а выходом автомата является перемещение в одну из соседних вершин. Автомат не различает направление и взаимное расположение соседних вершин, но различает занята вершина или нет. В работе приводятся как достаточные, так и необходимые условия в виде ограничений на свойства автоматов и структуру коллектива, при которых коллектив как цельный, связанный взаимодействием объект, сохраняет постоянное направление перемещения в среде.

Основные определения и результаты

Через $\mathbb{Z}$ и $\mathbb{N}$ обозначаем множества целых и натуральных чисел (с нулем) и $\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$. Произвольное множество $E \subseteq Z^n$ назовем n-мерной средой E. Элементы множества E назовем вершинами среды. Имя вершины есть ее координата. Две вершины $r = (x_1, \ldots, x_n)$ и $r' = (y_1, \ldots, y_n)$ назовем

соседними, если для единственного $i \in \{1, \ldots, n\}$ выполняется $|x_i - y_i| = 1$, а для всех остальных i $x_i = y_i$.

Однонаправленное перемещения коллектива автоматов.

Взаимодействующий со средой автомат $A = (S, X, Y, \delta_A, \lambda_A)$ состоит из конечных множеств состояний S, входных $X \subseteq N \times N$ и выходных $Y = \{0, 1, \cdot\}$ символов, а также функций переходов $\delta_A : S \times X \to S$ и выходов $\lambda_A : S \times X \to Y$. Взаимодействие автомата и среды происходит следующим образом. В каждый момент времени автомат находится в некоторой вершине. Входным сигналом автомата, находящегося в вершине r, является пара чисел $(p, q) \subseteq \mathbb{N} \times \mathbb{N}$, где p – число свободных, а q – число занятых соседних с r вершин. Вершина, в которой нет ни одного автомата, называется свободной, в противном случае — занятой. Выход автомата 0, 1 или $\cdot$ означает, что автомат переходит соответственно либо в любую свободную, либо в любую занятую вершину, либо остается в настоящей вершине r.

Под коллективом автоматов $\mathbf{A}$ понимаем конечное множество автоматов $\mathbf{A} = \{A_i | i \in I\}$ взаимодействующих со средой E, $I = \{1, 2, ..., m\}$. Пусть S_i множество состояний автомата A_i, $i \in I$. Обозначим $\mathbf{S} = S_1 \times S_2 \times \ldots \times S_m$, $\mathbf{X} = X^m$, $\mathbf{Y} = Y^m$, $\mathbf{E} = E^m$. Назовем $\mathbf{S}$ множеством состояний коллектива, т.е. если автомат A_i находится в состоянии s_i, $i \in I$, то коллектив как целое находится в состоянии $(s_1, s_2, \ldots, s_m)$. Если при этом автомат A_i находится в вершине r_i среды, то говорим, что коллектив находится в фрагменте $\mathbf{r} = (r_1, r_2, \ldots, r_m)$ среды. Пару $(\mathbf{s}, \mathbf{r})$ состояния коллектива и фрагмента среды, в котором находится коллектив, назовем конфигурацией коллектива и среды. Конфигурация $(\mathbf{s}, \mathbf{r})$ порождает некоторый входной сигнал $x_i \in X$ для каждого автомата A_i, как было определено выше. Обозначим упорядоченную совокупность $(x_1, \ldots, x_m)$ сигналов, порождаемых $(\mathbf{s}, \mathbf{r})$, через $\lambda_{\mathbf{AE}}(\mathbf{s}, \mathbf{r})$. Таким образом, мы определили функцию $\lambda_{\mathbf{AE}} : \mathbf{S} \times \mathbf{E} \to \mathbf{X}$.

По аналогии с конечными автоматами определим функцию выходов $\lambda_{\mathbf{A}}$ и переходов $\delta_{\mathbf{A}}$ коллектива $\mathbf{A}$:

$$\lambda_{\mathbf{A}} : \mathbf{S} \times \mathbf{X} \to \mathbf{Y}, \quad \delta_{\mathbf{A}} : \mathbf{S} \times \mathbf{X} \to \mathbf{S},$$

$$\lambda_{\mathbf{A}}(s_1, \ldots, s_m, x_1, \ldots, x_m) = (\lambda_{A_1}(s_1, x_1), \ldots, \lambda_{A_m}(s_m, x_m)),$$
$$\delta_{\mathbf{A}}(s_1, \ldots, s_m, x_1, \ldots, x_m) = (\delta_{A_1}(s_1, x_1), \ldots, \delta_{A_m}(s_m, x_m)).$$

Таким образом, мы рассматриваем коллектив автоматов $\{A_i | i \in I\}$ как один цельный объект в виде автомата $\mathbf{A} = (\mathbf{S}, \mathbf{X}, \mathbf{Y}, \lambda_{\mathbf{A}}, \delta_{\mathbf{A}})$.

Под взаимодействием коллектива $\mathbf{A}$ и среды E понимаем (вообще говоря, недетерминированную) систему переходов $\mathbf{AE}$ (англ. state transition system) состоящую из множества состояний и функции переходов, то есть

$$\mathbf{AE} = (\mathbf{S} \times \mathbf{E}, \delta_{\mathbf{AE}}),$$

где бинарное отношение переходов $\delta_{\mathbf{AE}} \subseteq \mathbf{S} \times \mathbf{E} \times \mathbf{S} \times \mathbf{E}$ такое, что если $\mathbf{s} = (s_1, \ldots, s_m)$, $\mathbf{s}' = (s'_1, \ldots, s'_m)$, $\mathbf{r} = (r_1, \ldots, r_m)$, $\mathbf{r}' = (r'_1, \ldots, r'_m)$, то $(\mathbf{s}, \mathbf{r}, \mathbf{s}', \mathbf{r}') \in \delta_{\mathbf{AE}}$ тогда и только тогда, когда автомат A_i, находящийся в конфигурации $(\mathbf{s}, \mathbf{r})$ переходит под воздействием соответствующего входного сигнала в состояние s'_i и может перейти в вершину r'_i среды, $i \in I$.

Координатой коллектива в среде назовем среднее арифметическое координат его автоматов. Ограничимся далее следующим неформальным определением: перемещение коллектива назовем однонаправленным, если некоторые компоненты координаты коллектива монотонно изменяются со временем, а остальные компоненты колеблются около фиксированных значений.

Теорема 1. *Для однонаправленного перемещения в среде* $\mathbb{Z} \times \mathbb{Z}_m$ *(вдоль границы) необходимо и достаточно коллектива в случае* $m = 1$ *из* 2 *автоматов, в случае* $m > 1$ *из* 4 *автоматов. Существует коллектив из* 6 *автоматов, сохраняющий направление перемещения в среде* $\mathbb{Z} \times \mathbb{Z}$.

Однонаправленное перемещения автомата с камнями.

В подразделе предлагается другой подход к проблеме направленного перемещения в среде. Вместо коллектива перемещающихся автоматов рассматривается один автомат снабженный конечным числом камней, играющих роль ограниченной внешней памяти [1].

Положим, что все вершины среды окрашены в один и тот же цвет, например, 0. Автомат, находясь в вершине среды, наблюдает цвета соседних вершин, не различая вершины одинакового цвета. Автомат может переместиться в вершину выбранного цвета (если несколько верши имеют один и тотже цвет – выбор случаен), установить или подобрать в текущей вершине камень одного из k цветов (цвет камня, установленного в вершине, щаменяет ее цвет). Обозначим через M множество все используемых цветов вершин и камней. Таким образом автомат представляет собой девятку $A = (V, S, T, P, \Omega, s_0, \delta, \lambda, \gamma)$, где $V \subseteq \{\alpha\beta | \alpha, \beta \in M\}$ – поле зрения (множество входных сигналов), S – множество состояний, $T = \{\alpha\beta | \alpha, \beta \in M\} \cup \{\text{stop}\}$ – множество перемещений (множество выходных сигналов), $P = \{1, 2, ..., k\}$ – множество камней, $\Omega = \{\text{pick}\,1, ..., \text{pick}\,k, \text{drop}\,1, ..., \text{drop}\,k, \text{none}\}$ – множество действий с камнями ($\text{pick}\,i$ – уставновить камень i, $\text{drop}\,i$ – подобрать камень i, none – ничего не делать), $s_0 \in S$ – начальное состояние, $\delta : S \times V \to S$ – функция переходов, $\lambda : S \times V \to T$ – функция перемещений, $\gamma : S \times V \to \Omega$ – функция действий с камнями.

Положим, что априори задана начальная схема размещения камней в вершинах среды. Автомат перемещается в среде перенося камни из вершины в вершину так, что в начале каждой итерации алгоритма перемещения схема размещения камней одна и та же и совпадает с начальной. Под однонаправленным перемещением здесь понимается следующее: за время работы алгоритма автомат последовательно посещает попарно различные вершины среды одна из координат которых фиксирована. Число используемых при перемещении автомата камней следующим образом зависит от размеров среды.

Теорема 2. *Для направленного перемещения в среде* $\mathbb{Z} \times \mathbb{Z}_m$ *автомату* A *в случае* $m = 1$ *необходимо и достаточно* 3 *камня, в случае* $2 \leqslant m < 5$ *необходимо и достаточно* 4 *камня, в случае* $m \geqslant 5$ *достаточно* 7 *камней.*

Заключение. Таким образом, предложены два подхода к решению задачи однонаправленного перемещения в среде автоматов без компаса. Подходы различаются структурой и поведением как самих автоматов, так и их коллективов.

Литература

[1] *Килибарда Г., Кудрявцев В. Б., Ушчумлич Ш. М.* Коллективы автоматов в лабиринтах // Дискретная математика. — 2003. — Т. 15, № 3. — С. 3–39.

[2] *Dudek G.,Jenkin M.* Computational Principles of Mobile Robotics. — Cambridge University Press, 2010. — 406 p.

[3] *Грунский И. С., Сапунов С. Б.* Диагностика местоположения мобильного робота на основе топологической информации о среде // Искусственный интеллект. — 2011. — Т. 2. — С. 15–25.

[4] *Курганский А. Н.* О мере изменения состояния коллектива взаимодействующих элементарных автоматов в дискретной среде // Кибернетика и системный анализ. — 2012. — Т. 48, № 3. — С. 349–357.

Оценки числа неизоморфных комплексов заданного вида

С. А. Лавренченко, А. Ю. Щиканов

lawrencenko@hotmail.com, au2u@ya.ru

Российский государственный университет туризма и сервиса, Черкизово

Пусть V — заданное конечное множество, содержащее $|V| = n$ элементов, которые называются *вершинами*. *Абстрактным симплициальным комплексом* (или просто *комплексом*) с множеством вершин V называется подмножество $\mathcal{K}$ множества $\mathcal{P}(V)$ всех подмножеств множества V такое, что наряду с каждым элементом из $\mathcal{K}$, называемым *гранью* комплекса, в $\mathcal{K}$ содержатся и все непустые подмножества этого элемента. Другими словами, каждая грань грани комплекса сама является гранью комплекса.

Мы рассматриваем комплексы, являющиеся *остовными* подкомплексами (т.е. подкомплесами с множеством вершин V) заданного вида или, как мы говорим, заданного *класса* в полном абстрактном симплициальном комплексе $\mathcal{P}(V)$. Два таких подкомплекса $\mathcal{K}_1$ и $\mathcal{K}_2$ называются *изоморфными*, если между ними существует *изоморфизм*, т.е. подстановка φ множества V такая, что φ переводит грани $\mathcal{K}_1$ на грани $\mathcal{K}_2$, а φ^{-1} переводит грани $\mathcal{K}_2$ на грани $\mathcal{K}_1$. Класс подкомплексов задается его характеристическими *инвариантными* свойствами, т.е. свойствами, сохраняющимися при изоморфизмах. В дальнейшем мы будем называть эти подкомплексы просто комплексами.

Примеры классов комплексов: (а) класс деревьев с данным множеством вершин, (б) класс общих *простых* (т.е. без петель и кратных ребер) графов с данным множеством вершин, (в) класс триангуляций поверхности (или поверхностей) с данным полным графом K_n.

Все подстановки множества V образуют полную симметрическую группу S_n. Пусть Ξ_n обозначает заданный класс вершинно-помеченных комплексов с заданным множеством вершин V, в котором два комплекса считаются различными, если в одном из них найдется грань $\{v_1, \ldots, v_k\}$ такая, что вершины с этими метками в другом комплексе грань не образуют. Поскольку класс Ξ_n задан инвариантными свойствами его членов, группа S_n свободно действует

на Ξ_n, т.е. *каждая* подстановка $\varphi \in S_n$ переводит любой комплекс $\mathcal{K} \in \Xi_n$ на комплекс $\varphi(\mathcal{K}) \in \Xi_n$. Те из подстановок S_n, которые переводят фиксированный комплекс $\mathcal{K}$ на себя, образуют его так называемую *группу автоморфизмов*, обозначаемую $\mathrm{Aut}(\mathcal{K})$. Далее, обозначим через $\Omega_n = \{\mathcal{K}_1, \ldots, \mathcal{K}_r\} \subseteq \Xi_n$ множество орбит при указанном действии группы. Здесь $\mathcal{K}_1, \ldots, \mathcal{K}_r$ — представители этих орбит, т.е. все попарно неизоморфные комплексы заданного типа. В описанной ситуации алгебраическая формула разложения на орбиты записывается следующим образом:

$$|\Xi_n| = \sum_{i=1}^{|\Omega_n|} \frac{|S_n|}{|\mathrm{Aut}(\mathcal{K}_i)|} = \sum_{i=1}^{|\Omega_n|} \frac{n!}{|\mathrm{Aut}(\mathcal{K}_i)|}, \tag{1}$$

откуда получаются оценки на число $|\Omega_n|$ попарно неизоморфных комплексов:

$$\frac{1}{n!}|\Xi_n| \leqslant |\Omega_n| \leqslant |\Xi_n|. \tag{2}$$

Таким образом, $|\Omega_n|$ не может отличаться от $|\Xi_n|$ более чем в $n!$ раз. Зачастую (но не всегда) удается выразить $|\Xi_n|$ в виде формулы, в то время как для $|\Omega_n|$ формула неизвестна и $|\Omega_n|$ вычисляется через производящую функцию.

В силу (1) множество $\Omega_n = \{\mathcal{K}_1, \ldots, \mathcal{K}_r\}$ превращается в вероятностное пространство путем назначения вероятностей следующим образом:

$$p(\mathcal{K}_i) = \frac{n!}{|\Xi_n|} \frac{1}{|\mathrm{Aut}(\mathcal{K}_i)|}. \tag{3}$$

Тогда математическое ожидание порядка группы автоморфизмов комплекса $\mathcal{K}$ выражается формулой:

$$E(|\mathrm{Aut}(\mathcal{K})|) = n! \frac{|\Omega_n|}{|\Xi_n|}. \tag{4}$$

Формула для $|\Xi_n|$ известна для класса деревьев (теорема Кэли):

$$|\Xi_n| = n^{n-2} \tag{5}$$

и очевидна для класса общих (простых) графов:

$$|\Xi_n| = 2^{\binom{n}{2}} = 2^{n(n-1)/2}. \tag{6}$$

Из (4) и (5) получаются математическое ожидание порядка группы автоморфизмов дерева (с n вершинами) и оценки на число неизоморфных деревьев:

$$E(|\mathrm{Aut}(\mathcal{K})|) = \frac{n!}{n^{n-2}}|\Omega_n|, \qquad \frac{n^{n-2}}{n!} \leqslant |\Omega_n| \leqslant n^{n-2},$$

а из (4) и (6) получаются ожидаемый порядок группы автоморфизмов общего графа (с n вершинами) и оценки на число неизоморфных графов:

$$E(|\mathrm{Aut}(\mathcal{K})|) = \frac{n!}{2^{n(n-1)/2}}|\Omega_n|, \qquad \frac{2^{n(n-1)/2}}{n!} \leqslant |\Omega_n| \leqslant 2^{n(n-1)/2},$$

Комплекс $\mathcal{K}$ называется *асимметричным*, если $|\mathrm{Aut}(\mathcal{K})| = 1$.

Пример: минимальное асимметричное дерево имеет 7 вершин. Оно получается соединением в одну вершину концевых вершин трех путей с длинами 1, 2 и 3 (соответственно). Это дерево имеет неожиданно большую вероятность, близкую к 0.3, т.к. по формулам (3) и (5) вероятность асимметричного дерева с n вершинами равна $n!/n^{n-2}$.

Пусть Υ_n обозначает множество всех попарно неизоморфных асимметричных комплексов в заданном классе комплексов Ξ_n. Из (1) следует, что $|\Upsilon_n| \leqslant |\Xi_n|/n!$, откуда с учетом (2) получаем следующую цепочку неравенств:

$$|\Upsilon_n| \leqslant \frac{1}{n!}|\Xi_n| \leqslant |\Omega_n| \leqslant |\Xi_n|. \tag{7}$$

В частности, для класса деревьев и класса общих графов неравенства (7) с учетом (5) и (6) принимают вид (соответственно):

$$|\Upsilon_n| \leqslant \frac{n^{n-2}}{n!} \leqslant |\Omega_n|, \qquad |\Upsilon_n| \leqslant \frac{2^{n(n-1)/2}}{n!} \leqslant |\Omega_n|.$$

Итак, мы приходим к любопытному выводу, что нецелочисленные (при $n \geqslant 3$) последовательности $\{n^{n-2}/n!\}$ и $\{2^{n(n-1)/2}/n!\}$ (полное отсутствие целочисленных членов в первой обосновывается при помощи постулата Бертрана) «зажаты» между двумя соответствующими целочисленными последовательностями $\{|\Upsilon_n|\}$ и $\{|\Omega_n|\}$, для которых формулы общего члена неизвестны (хотя их производящие функции известны).

В оставшейся части тезиса рассмотрим триангуляции замкнутых поверхностей с полным графом K_n, которые образуют класс двумерных комплексов. Известно [1], что такие триангуляции существуют на ориентируемой поверхности тогда и только тогда, когда $n \equiv 0, 3, 4$ or $7 \mod 12$, и на неориентируемой поверхности тогда и только тогда, когда $n \equiv 0$ или $1 \mod 3$, $n \geqslant 6$ и $n \neq 7$.

Из того факта, что в любой триангуляции замкнутой поверхности с графом K_n ограничивающий цикл звезды каждой вершины v есть гамильтонов цикл у графа $K_n - v$, можно вывести следующее неравенство:

$$|\Xi_n| \leqslant \frac{(n-2)!}{2}\,[(n-4)!]^{n-1}. \tag{8}$$

Далее, поскольку каждый автоморфизм триангуляции однозначно определен указанием образа одного ее флага, а число всех флагов равно $2n(n-1)$, то с учетом (4) получаем:

$$|\Omega_n| \leqslant \frac{2n(n-1)}{n!}|\Xi_n| = \frac{2}{(n-2)!}|\Xi_n|. \tag{9}$$

Комбинируя (9) и (8), получаем нашу главную оценку:

$$|\Omega_n| \leqslant [(n-4)!]^{n-1}. \tag{10}$$

Наконец, объединяя верхнюю оценку (10) с нижней оценкой, полученной в [2], приходим к следующему двойному неравенству:

$$2^{(n-1)(n-7)/54} \leqslant |\Omega_n| \leqslant [(n-4)!]^{n-1},$$

где для нижней оценки из [2] требуется следующее ограничение: $(n+2)/3$ должно быть натуральным числом $\geqslant 19$ и таким, что $(n+2)/3 \equiv 3$ or $7 \bmod 12$.

Литература

[1] *Рингель Г.* Теорема о раскраске карт. — М.: Мир, 1977. — 256 с.

[2] *Bonnington C. P., Grannell M. J., Griggs T. S., Širáň J.* Exponential families of non-isomorphic triangulations of complete graphs // J. Comb. Theory Ser. B. — 2000. — V. 78, № 2. — P. 169–184.

О сложности бесконечной надструктуры классов монотонных 4-значных функций

В. Б. Ларионов

VitalyBLarionov@yandex.ru

ООО «Атес Медика Софт», Москва

Введение

Хорошо известно [9], что решетка замкнутых относительно операции суперпозиции классов функций k-значной логики для любого $k \geqslant 3$ содержит континуальное число классов. В силу невозможности ее исчерпывающего описания представляется интересным изучение различных подмножеств этой решетки. Указанной задаче и посвящена данная работа.

Обозначим множество всех замкнутых классов k-значных функций через P_k, везде далее будем считать $k \geqslant 3$. Одним из семейств предполных в P_k классов является некоторое подмножество классов монотонных функций ([10], [6]). Ранее автором было показано, что, в зависимости от свойств порождающего частично упорядоченного множества (ЧУМ), положение в решетке класса монотонных функций может существенно меняться.

Принципиальная возможность существования классов монотонных функций с бесконечной *надструктурой* (то есть множеством содержащих их классов) была доказана в статье [4]. В [2] были получены критерии наличия бесконечной надструктуры классов монотонных функций, сохраняющих ЧУМ с ограниченным числом минимальных и максимальных элементов. В [5] исследовался вопрос сложности бесконечной надструктуры. Было показано, что для достаточно широкого множества порождающих ЧУМ надструктура класса монотонных функций содержит бесконечное число классов, не являющихся предикатно-описуемыми. Это свидетельствует о сложности указанной надструктуры. В данной работе показано, что данный результат справедлив и для минимальной логики P_4, содержащей класс монотонных функций с бесконечной надструктурой ([3]).

Основные понятия

Пусть на $E_k = \{0, 1, \ldots, k-1\}$ задано некоторое отношение частичного порядка r. Возьмем два произвольных набора $\tilde{a} = (a_1, \ldots, a_n)$ и $\tilde{b} = (b_1, \ldots, b_n)$ из E_k^n. Будем говорить, что $\tilde{a}$ не превосходит $\tilde{b}$ относительно частичного порядка r и записывать $\tilde{a} \leqslant_r \tilde{b}$, если для любого $1 \leqslant i \leqslant n$ справедливо неравенство $a_i \leqslant_r b_i$.

Функция $f(x_1, \ldots, x_n)$ называется *монотонной относительно частичного порядка* r, если для любых двух наборов $\tilde{a}, \tilde{b} \in E_k^n$ таких, что $\tilde{a} \leqslant_r \tilde{b}$, выполнено $f(\tilde{a}) \leqslant_r f(\tilde{b})$. Множество всех функций из P_k, монотонных относительно r, называется классом монотонных функций M_r.

Для наглядности везде далее будем задавать частичный порядок r частично упорядоченным множеством (ЧУМ) H из элементов E_k и соответствующий класс обозначать M_H.

В работе активно используется соответствие Галуа между замкнутыми классами функций и предикатов. Поскольку формат статьи не позволяет привести все необходимые факты, будем считать известными понятия предиката, сохранения предиката функцией, а также основные результаты соответствия Галуа (см. например [1]).

Будем обозначать через $\mathrm{Pol}(p)$ множество всех функций, сохраняющих предикат p. Класс M_H является замкнутым классом функций, сохраняющих предикат $R(x, y) = \mathrm{TRUE} \iff x \leqslant_r y$ [8]. Везде далее в выражении „монотонный класс задается предикатом R“ подразумевается именно описанный предикат $R(x, y)$.

Обозначим через H_4 ЧУМ из четырех элементов $0, 1, 2, 3$ такой, что $0 \geqslant 1, 2$, $3 \geqslant 1, 2$, пары $0, 3$ и $1, 2$ несравнимы. Надструктура класса M_{H_4} бесконечна, надструктура остальных классов монотонных функций в P_4 (кроме тех, что изоморфны M_{H_4}) конечна ([2, 3]).

Замкнутый класс A называется *предикатно-описуемым*, если существует предикат p такой, что справедливо представление $A = \mathrm{Pol}\, p$. Поясним, почему наличие в надструктуре классов, не являющихся предикатно-описуемыми (особенно, бесконечного их количества) свидетельствует о ее сложности. В работе [7] показано, что для произвольного не являющегося предикатно-описуемым класса A либо существует бесконечная цепочка классов, содержащих A, с пределом, равным A, либо бесконечное число минимальных надклассов A.

Основной результат

Теорема 1. *В P_4 существует бесконечное число различных замкнутых классов, содержащих M_{H_4} и не являющихся предикатно-описуемыми.*

Идея доказательства теоремы аналогична работе [5]. Проблема описания надструктуры сводится к проблеме выразимости предикатов над $\{R(x, y)\}$, где R – предикат, порождающий класс монотонных функций M_{H_4}. Само искомое множество классов, содержащих M_{H_4}, строится аналогично работе [5], с той лишь разницей, что доказательство невыразимости более громоздкое (за

счет отсутствия в ЧУМ H_4 единственного минимального или максимального элемента).

Итак, мы получили, что даже в минимальной логике бесконечная надструктура класса монотонных функций очень сложна. Открытым остается вопрос о ее мощности.

Работа выполнена при поддержке РФФИ, проект № 13-01-00684-а.

Литература

[1] *Боднарчук В. Г., Калужнин В. А., Котов В. Н., Ромов Б. А.* Теория Галуа для алгебр Поста // Кибернетика. — 1969. — № 3. — С. 1–10. — № 5. — С. 1–9.

[2] *Ларионов В. Б.* Замкнутые классы k-значной логики, содержащие классы монотонных или самодвойственных функций. — Диссертация на соискание степени к. ф.-м. н., 2014. — 157 с.

[3] *Ларионов В. Б.* О монотонных замкнутых классах функций многозначной логики с бесконечной надструктурой // Материалы VII молодежной научной школы по дискретной математике и ее приложениям, 18–23 мая 2009 г.. — Москва: ИПМ им. М. В. Келдыша РАН, 2009. — С. 7–12.

[4] *Ларионов В. Б.* О положении некоторых классов монотонных k-значных функций в решетке замкнутых классов // Дискретная математика. — 2009. — Т. 21, № 5. — С. 111–116.

[5] *Ларионов В. Б.* О сложности надструктуры классов монотонных k-значных функций специального вида // Известия иркутского государственного университета. — 2012. — Т. 5, № 1. — С. 70–80.

[6] *Мартынюк В. В.* Исследование некоторых классов функций в многозначных логиках // Проблемы кибернетики. — 1960. — Вып. 3. — С. 49–61.

[7] *Яблонский С. В.* О строении верхней окрестности для предикатно-описуемых классов в P_k // Доклады АН СССР. — 1974. — Т. 218, № 2. — С. 304–307.

[8] *Яблонский С. В., Гаврилов Г. П., Набебин А. А.* Предполные классы в многозначных логиках. — М.: Изд. дом МЭИ, 1997. — 144 с.

[9] *Янов Ю. И., Мучник А. А.* О существовании k-значных замкнутых классов, не имеющих конечного базиса // ДАН СССР. — 1959. — Т. 127, № 1. — С. 44–46.

[10] *Rosenberg I. G.* La structure des fonctions de plusiers variables sur un ensemble fini // Comptes Rendus Acad. Sci. Paris. — 1965. — V. 260 — P. 3817–3819.

Цикловые индексы автомата в задаче выразимости

А. А. Летуновский

alekseyletunovskiy@lsi.com

LSI Corporation, Москва

Вводится понятие цикловых индексов автомата как пары положительных чисел (b, q) эффективно вычисляемых по автомату. Показано, что множество периодов константных автоматов, выразимых суперпозициями данного автомата, булевых функций и задержки - суть делители членов геометрической

прогрессии $b, bq, bq^2....$ Отсюда следует алгоритм проверки выразимости константных автоматов, а также теорема о проверке выразимости множества всех автоматов с ограниченным числом состояний.

Обозначим через P_2 множество всех булевых функций. Класс всех автоматных функций (a-функций)обозначим через $\mathbf{P}$. В этом классе обычным образом введем операции суперпозиции.

Обозначим через $[R]$ - множество a-функций, получающихся из R с помощью операций суперпозиции.

Обозначим автоматную функцию задержки через G_0.

Обозначим через $< R >= [R \cup \{P_2, G_0\}]$.

Назовем автоматную функцию, не зависящую от входа, *константной* автоматной функцией. Множество всех константных автоматных функций обозначим $\mathbf{K}$. Без ограничения общности константную автоматную функцию можно отождествить со сверхсловом, которое является его выходом.

Пусть сверхслово β можно представить в виде $\beta = \gamma\alpha^\infty$. Выберем из всех таких представлений такое, что γ и α имеют наименьшую длину. Для выбранного представления назовем γ - наименьшим *предпериодом сверхслова* β,а α наименьшим *периодом сверхслова* β, а слова вида $\underbrace{\alpha\alpha...\alpha}_{n}$ будем также называть периодом сверхслова β. Обозначим через $|\alpha|$ длину слова α.

Для множества константных автоматных функций $K' \subseteq K$ обозначим через $\Theta(K')$- множество длин минимальных периодов сверхслов $\{\beta_{K_i} : K_i \in K'\}$. Для случая одного слова $\beta = \gamma\alpha^\infty$ будем считать, что $\Theta(\beta) = |\alpha|$.

Нашей задачей будет описание множества $\Theta(< R > \cap K)$ для произвольного множества R.

Теорема 1. *Пусть R - конечное множество автоматных функций, тогда*

$$\Theta(< R > \cap K) = \bigcup_{i=1}^{\infty} \{t | bq^i\},$$

- множество натуральных чисел, являющихся делителями чисел вида bq^i, где b,q - натуральные числа, вычисляемые эффективно по R.

Определим цикловыми индексами автомата пару чисел b, q, введенных в теореме 1.

Теорема 2. *Пусть R - конечное множество автоматных функций и β - константная автоматная функция, тогда существует алгоритм, позволяющий проверить свойство*
$\beta \in < R >$

Следствие 1 *Пусть R - конечное множество автоматных функций, тогда существует алгоритм, позволяющий проверить свойство* $| \Theta(< R > \cap K) | < \infty$

Теорема 3. *(Необходимое условие выразимости) Пусть R_1, R_2- конечные множества автоматов и $R_2 \in [R_1]$. b_1, q_1, b_2, q_2 - цикловые индексы R_1 и R_2 соответственно. Тогда*

Работа выполнена на кафедре МаТИС механико-математического факультета МГУ им. Ломоносова под руководством профессора Д.Н. Бабина.

Список литературы

1.Кудрявцев В.Б., Алешин С.В., Подколзин А.С., Введение в теорию автоматов, Наука, М., 1985.
2.Мальцев А.И. Итеративные алгебры Поста. Изд-во ИМСО АН СССР, Новосибирск, 1976
3. H.H. Loomis Jr, Completeness of sets of delayed logic devices, IEEE Trans. Electronic Computers, vol. EC-14, p.157-172, 1965.
4.Бабин Д.Н. О полноте двухместных автоматных функций относительно суперпозиции, Дискретная матетматика, том 1, выпуск 4, 1989, стр. 423-431
5.Летуновский А.А., О выразимости константных автоматов суперпозициями, Интеллектуальные системы, Том 13, выпуск 1-4, 2009 г.,397-406
6. Алешин С.В., Об одном следствии теоремы Крона-Роудза, Дискретная математика, том 11, вып.4, 1999 год, стр. 101-109
7.Бабин Д.Н. О суперпозициях ограниченно детерминированных функций ограниченного веса, Логико-алгебраические конструкции, Тверь, 1992 г. стр. 21-27

Вычислительные возможности односторонних машин Тюринга с сублогарифмическими ограничениями на память

А. Н. Лещёв

alex.leshev@gmail.com

Казанский Федеральный Университет, Казань

Введение

Сложностные характеристики машин Тюринга — время $T(n)$ работы машины и необходимая память $S(n)$ для работы машины, а также соответствующие им классы сложности $SPACE(S(n))$ и $TIME(T(n))$ являются известными инструментами для изучения возможностей разных моделей машины Тюринга. Эти классы сложности и их модификации определены в книге [1].

Рассматривается соотношение классов сложности $SPACE^{g(n)}(s_k(n))$ при $T(n) = n$, где $g(n)$ задает ограничение на количество недетерминированных шагов машины, а $s_k(n) = \log^{(k)} n \equiv \underbrace{\log \ldots \log}_{k} n$. Машины Тюринга с ограничением на время работы $T(n) = n$ называются односторонними машинами Тюринга.

Введение ограничений на память

Детерминированный вариант односторонней машины Тюринга вместе с соответствующим классом сложности R впервые был представлен в [2]. Недетерминированная модель односторонней машины и соответствующий класс

сложности Q^+ были впервые представлены в [3]. Недетерминированная модель односторонней машины с ограничением на недетерминизм была впервые представлены в [4], где через $Q^{g(n)}$ обозначался класс языков распознаваемых односторонними машинами, совершающими не более чем $g(n)$ недетерминированных шагов.

В [5] было доказано, что в случае с односторонними машинами Тюринга сублогарифмическое ограничение на память приводит к вырождению соответсвующего класса сложности $SPACE(f(n))$ до класса регулярных языков Reg. Поэтому, с целью скомпенсировать потерю мощности при ограниченной памяти, вводится новая модификация модели односторонней машины.

Определение 1. Будем говорить, что недетерминированная односторонняя машина Тюринга M использует **слабую подсказку (unary advice) длины** s, если к моменту начала работы машины M на входном слове w на всех ее рабочих лентах размечена и заполнена **одним специальным символом** область размера не более, чем s ячеек.

Определение 2. Обозначим через $SPACE_u^{g(n)}(f(n))$ класс языков распознаваемых односторонними машинами, совершающими не более чем $g(n)$ недетерминированных шагов и использующими не более, чем $f(n)$ ячеек на каждой из своих рабочих лент, со слабой подсказкой размера не более, чем $f(n)$.

Обозначим через $DC_M(n)$ количество всевозможных наборов конфигураций, в которых может находиться односторонняя машина с подсказкой M с $g(n)$ ограничением на недетерминизм и $f(n)$ ограничением на память при обработке слов длины n.

Лемма 1. *Пусть даны функции $f(n)$, $g(n)$ такие, что $f(n) = o(n)$, $g(n) \leqslant f(n)$. Пусть односторонняя машина M имеет $g(n)$ ограничение на недетерминизм и $f(n)$ ограничение на память со слабой подсказкой длины $f(n)$. Тогда справедливо:*

$$DC_M(n) \leqslant 2^{cf(n)2^{g(n)}},$$

Определение и свойства специальных языков

На основе функций $g(n)$ и $f(n)$ определим следующий язык:

Определение 3. $SP_{f(n)}^{g(n)}$ — это множество строк w, удовлетворяющих следующим свойствам:

- $|w| = n$ для некотого $n \geqslant 1$;
- $w = x_1 2 x_2 2 \ldots 2 x_s 3 u 4 y^r$, где $s \leqslant g(n) + 1$;
- $x_k \in \{0,1\}^*$ и $|x_k| = f(n)$ для всех k;
- $u \in \{0,1\}^*$;
- y^r — обратная запись слова y;
- $y \in \{0,1\}^*$, $|y| = f(n)$ и $y = x_i$ хотя бы для одного i: $1 \leqslant i \leqslant s$.

Для таких языков справедливо утверждение:

Лемма 2. *Пусть даны функции $f(n)$, $g(n)$ такие, что $f(n) = o(n)$, $g(n) = o(n)$, $g(n) \leqslant f(n)$. Тогда язык $SP_{f(n)}^{g(n)}$ содержится в классе $SPACE_u^{g(n)}(f(n))$.*

Зафиксируем некоторое значение n и рассмотрим все слова w длины n специального языка $L_1 = SP^{g(n)}_{f(n)}$ и представим их в виде: $w = A_w 3 B_w$, где $B_w = u4y^r$. Рассмотрим такие A_w, в которых подстроки x_j не повторяются.

Определение 4. Обозначим через $LC_{L_1}(n)$ количество различных подстрок A_w слов w длины n из языка $L_1 = SP^{g(n)}_{f(n)}$.

Лемма 3. *Пусть даны функции $f(n)$, $g(n)$ такие, что $f(n) = o(n)$, $g(n) \leqslant f(n)$. Тогда для языка $L_1 = SP^{g(n)}_{f(n)}$ справедливо:*

$$LC_{L_1}(n) \geqslant \left(\frac{2^{f(n)}}{g(n)}\right)^{g(n)}$$

Иерархии по памяти

Связь между сложностью специального языка $LC_{\tilde{L}}(n)$ и возможностями односторонней машины, выраженными характеристикой $DC_M(n)$, раскрывается следующим утверждением:

Лемма 4. *Пусть даны функции $f_1(n)$, $f_2(n)$, $g_1(n)$, $g_2(n)$. Рассмотрим специальный язык $L_1 = SP^{g_1(n)}_{f_1(n)}$ и любую одностороннюю машину Тюринга M с $g_2(n)$ ограничением на недетерминизм и $f_2(n)$ ограничением на память. Тогда, если выполняется условие*

$$DC_M(n) = o\left(LC_{L_1}(n)\right),$$

то справедливо:

$$SP^{g_1(n)}_{f_1(n)} \notin SPACE^{g_2(n)}_u(f_2(n))$$

Использование этой связи лежит в основе доказательств невырожденности иерархии классов $SPACE^{g(n)}_u(f(n))$ относительно ограничения на количество недетерминированных операций.

Лемма 5. *Пусть даны функции $g(n), f(n), h(n)$ такие, что $f(n) = o(n)$, $g(n) \leqslant 2^{f(n)}$ и $h(n) = o(\log g(n))$. Тогда для языка $SP^{g(n)}_{f(n)}$ справедливо:*

$$SP^{g(n)}_{f(n)} \notin SPACE^{h(n)}_u(f(n))$$

Лемма 6. *Пусть даны функции $g(n), f(n)$ такие, что $f(n) = o(n)$ и $g(n) \leqslant 2^{f(n)}$. Тогда для языка $SP^{g(n)}_{f(n)}$ справедливо:*

$$SP^{g(n)}_{f(n)} \in SPACE^{\log g(n)}_u(f(n))$$

Теорема 7. *Для любой функции $f(n) = o(n)$ и для любых функций $g(n)$ и $h(n)$ таких, что $h(n) = o(g(n))$ и $g(n) \leqslant f(n)$ справедливо:*

$$SPACE^{h(n)}_u(f(n)) \subsetneq SPACE^{g(n)}_u(f(n))$$

Следствие. Утверждение Теоремы 7 доказывает существование невырожденной иерархии классов сложности $SPACE_u^{g(n)}(f(n))$ относительно ограничения на количество недетерминированных операций при сублогарифмических ограничениях на память.

$$\ldots \subsetneq SPACE_u^{\log^{(k+1)} n}(f(n)) \subsetneq SPACE_u^{\log^{(k)} n}(f(n)) \subsetneq SPACE_u^{\log^{(k-1)} n}(f(n)) \subsetneq \ldots$$

Литература

[1] Sanjeev Arora, Boaz Barak *Computational complexity: a modern approach* (Princeton University 2006).

[2] Rosenberg A.L. *Real-time Definable Languages*, J. Assoc. Comp. Mach. (14), 645-662 (1967).

[3] Book R.V., Greibach S. A. *Quasi-Realtime Languages*, Math. Systems Theory (4), 97-111 (1970).

[4] Patrick C. Fischer, Chandra M. R. Kintala *Real-Time Computations with Restricted Nondeterminism [1979]*, Math. Systems Theory (12), 219-231 (1979).

[5] Hartmanis J., Lewis P. L. II, Stearns R. E. *Hierarchies of memory-limited computations*, Proceedings of the 6th Annual IEEE Symposium on Switching Circuit Theory and Logic Design, 179–190 (1965).

О сложности реализации мультиплексорных и квазимультиплексорных функций в некоторых классах схем

С. А. Ложкин, Н. В. Власов

lozhkin@cs.msu.ru, nikita.v.vlasov@gmail.ru

Московский государственный университет имени М. В. Ломоносова, Москва

Определим мультиплексорную функцию алгебры логики (ФАЛ) как ФАЛ, которая зависит от двух групп булевых переменных (БП) и на каждом наборе значений БП первой группы — группы «адресных» БП, — равна одной из БП второй группы — группы «информационных» БП. При этом число адресных БП (соответственно число существенных информационных БП) называется *порядком* (соответственно *рангом*) указанной ФАЛ.

Заметим, что стандартная мультиплексорная ФАЛ μ_n порядка n (см., например, [1]), которая также называется универсальной ФАЛ (см., например, [2]), представляет собой мультиплексорную ФАЛ порядка n и ранга 2^n.

Сложность и глубина ФАЛ μ_n при её реализации схемами в некоторых базисах исследовались в [2, 3, 4, 5, 6, 7] и ряде других работ. При этом в работах [5, 6, 7] были получены более точные по сравнению с известными ранее оценки как сложности, так и глубины ФАЛ μ_n, а некоторые из данных оценок являются оценками высокой степени точности (см., например, [5, 6]).

Ту ФАЛ, которая получается из мультиплексорной ФАЛ f порядка n подстановкой констант вместо части её информационных БП, будем называть *связанной* с f *квазимультиплексорной ФАЛ порядка* n *и ранга* r, где r — число

оставшихся информационных БП ФАЛ f. При этом квазимультиплексорные ФАЛ, получающиеся в результате подстановки нулей, считаются *нулевыми*, а квазимультиплексорные ФАЛ, связанные с ФАЛ μ_n, — *стандартными*. Будем называть *информационной областью* квазимультиплексорной ФАЛ множество тех наборов значений её адресных БП, на которых она равна одной из своих информационных БП.

В работе изучаются вопросы оптимальной по сложности (глубине) реализации мультиплексорных и квазимультиплексорных ФАЛ формулами и схемами из функциональных элементов (СФЭ) в стандартном базисе $Б_0 = \{x \,\&\, y, x \vee y, \overline{x}\}$ и унимодальном базисе U_2, состоящем из всех ФАЛ вида $x_1^{\sigma_1} \cdot x_2^{\sigma_2}$ и $x_1^{\sigma_1} \vee x_2^{\sigma_2}$, где σ_1, σ_2 — произвольные константы, и, как обычно, $x^0 = \overline{x}$, $x^1 = x$. При этом сложность $L_Б^{\Phi}(f)$ (соответственно $L_Б^{\mathrm{C}}(f)$) реализации ФАЛ f формулами (соответственно СФЭ) в базисе $Б$, а также её глубина $D_Б(f)$ в этом базисе определяются обычным образом.

Данная работа продолжает работы [5, 6, 7], обобщая полученные в них результаты для ФАЛ μ_n и, в частности, оценки высокой и близкой к ней степени точности на случай мультиплексорных ФАЛ общего вида и квазимультиплексорных ФАЛ.

Теорема 1. *Для последовательности квазимультиплексорных ФАЛ $\widehat{\mu}_n$ порядка n, где $n = 2, 3, \ldots$ и ранга r, $r = r(n)$, где $3 \leqslant r(n) \leqslant 2^n$, справедливы соотношения:*

$$L_{U_2}^{\Phi}(\widehat{\mu}_n) \geqslant 2r(n) + \frac{r(n)}{n+1}, \quad L_{Б_0}^{\Phi}(\widehat{\mu}_n) \geqslant 2 \cdot r(n) + \frac{4}{3} \cdot \frac{r(n)}{n} - o\left(\frac{r(n)}{n}\right),$$

$$L_{Б_0}^{C}(\widehat{\mu}_n) = L_{U_2}^{C}(\widehat{\mu}_n) \geqslant 2 \cdot r(n) + \frac{\sqrt{2}}{2}\sqrt{r(n)} - O(n),$$

$$D_{Б_0}(\widehat{\mu}_n) \geqslant D_{U_2}(\widehat{\mu}_n) \geqslant \lceil \log r(n) \rceil + 1,$$

причём второе неравенство выполняется при условии $\frac{2^n}{n} = o\,(r(n))$.

Теорема 2. *Пусть для $n = 1, 2, \ldots$ и натуральных последовательностей $r = r(n) \leqslant 2^n$, $d = d(n) \leqslant n$ выполнены условия*

$$r(n) \geqslant 2^{d(n)}, \; n - d(n) = o(n).$$

Тогда для любой последовательности нулевых стандартных квазимультиплексорных ФАЛ $\widehat{\mu}_n$, $n = 1, 2, \ldots$, порядка n и ранга $r(n)$ таких, что информационную область $\widehat{\mu}_n$ можно представить в виде объединения непересекающихся граней размерности не меньше чем $d(n)$, выполняются соотношения

$$L_{U_2}^{\Phi}(\widehat{\mu}_n) \leqslant 2r(n) + \frac{r(n)}{n} + o\left(\frac{r(n)}{n}\right),$$

$$L_{Б_0}^{\Phi}(\widehat{\mu}_n) \leqslant 2r(n) + \frac{2r(n)}{n} + o\left(\frac{r(n)}{n}\right),$$

$$L_{U_2}^{C}(\widehat{\mu}_n) \leqslant L_{Б_0}^{C}(\widehat{\mu}_n) \leqslant 2r(n) + C(n) \cdot \sqrt{r(n)} + O\left(n \sqrt[4]{r(n)}\right),$$

$$D_{U_2}(\widehat{\mu}_n) \leqslant D_{Б_0}(\widehat{\mu}_n) \leqslant \lceil \log r(n) \rceil + O(1).$$

где $C(n) = 2$, если n чётно, и $C(n) = \frac{3\sqrt{2}}{2}$, если n нечётно.

Заметим, что полученные в теоремах 1 и 2 нижние и верхние оценки величины $L^{\Phi}_{U_2}(\widehat{\mu}_n)$, которая при $n = 1, 2, \ldots$ асимптотически равна $2r(n)$, устанавливает её поведение с относительной погрешностью вида $o\left(\frac{1}{n}\right)$.

Работа выполнена при поддержке РФФИ, проект № 12-01-00964а.

Литература

[1] *Ложкин С. А.* Лекции по основам кибернетики. — М.: Издательский отдел ф-та ВМиК МГУ, 2004. — 251 с.

[2] *Коровин В. В.* О сложности реализации универсальной функции схемами из функциональных элементов // Дискретная математика. — 1995. — Т. 7, вып. 2. — С. 95–102.

[3] *Paul W. J.* A $2,5n$ lower bound on the combinational complexity of Boolean functions // SIAM, Philadelphia: SIAM Journal on Computing. — 1977. — V. 6. — P. 427–443.

[4] *Klein P., Paterson M. S* Asymptotically optimal circuit for a storage access function // IEEE Trans. on Computers, IEEE Computer Society. — 1980. — V. 29, № 8. — P. 737–738.

[5] *Ложкин С. А., Власов Н. В.* О сложности мультиплексорной функции в классе π-схем // Ученые записки Казанского университета. Сер. Физ.-матем. науки. — 2009. — Т. 151, кн. 5. — С. 98–106.

[6] *Ложкин С. А., Власов Н. В.* О глубине мультиплексорной функции // Вестник Московского университета. Сер. 15, Вычислительная математика и кибернетика. — 2011. — № 2. — P. 40–46.

[7] *Власов Н. В.* О сложности мультиплексорной функции в классе формул // Вестник Нижегородского государственного университета им. Н. И. Лобачевского. — 2012. — № 5. — С. 38–41.

О динамической активности схем из функциональных элементов и построении асимптотически оптимальных по сложности схем с оптимальной по порядку динамической активностью

С. А. Ложкин, М. С. Шуплецов

lozhkin@cs.msu.ru, miklesh@shupletsov.ru

Московский государственный университет им. М. В. Ломоносова, Москва

Введение. Оценка энергопотребления интегральных схем является одной из важных задач проектирования СБИС. В современных интегральных схемах, построенных на основе КМОП технологий, выделяют как статическое энергопотребление, связанное с рассеянием тепла и поддержанием заданного потенциала в узлах схемы, подключенных к источнику питания, так и динамическое энергопотребление, возникающее при изменении потенциалов в узлах схемы.

Одной из основных моделей комбинационных интегральных схем, предназначенных для решения различных задач анализа и синтеза, в том числе для анализа энергопотребления, является модель схем из функциональных элементов (СФЭ). Первые подходы к анализу статического энергопотребления для СФЭ были предложены в работе [1]. Основные теоретические результаты в этом направлении были получены О. М. Касим-Заде в работах [2, 3]. В указанных работах исследовался функционал сложности СФЭ, характеризующий статическое энергопотребление, — так называемая мощность СФЭ. При этом был установлен порядок роста соответствующей функции Шеннона в произвольном конечном полном базисе. Оказалось, в частности, что существуют базисы с как с линейным, так и с экспоненциальным поведением указанной функции Шеннона. Кроме того, была показана возможность построения для "типичной" функции алгебры логики (ФАЛ) такой реализующей ее СФЭ, сложность которой асимптотически оптимальна, а мощность оптимальна по порядку роста.

В свою очередь, различные подходы к оценке динамического энергопотребления схем на основе модели СФЭ были предложены в работе [4]. В работе [5] была изложена вероятностная модель для оценки среднего энергопотребления СФЭ. Исследования в области анализа динамического энергопотребления схем в основном были направлены на разработку эффективных алгоритмов расчета и оценки динамического энергопотребления конкретных схем. Обзор указанных результатов можно найти в работе [6].

В данной работе введен функционал динамической активности СФЭ, который моделирует их динамическое энергопотребление. Установлен линейный порядок роста функции Шеннона для динамической активности СФЭ в произвольном полном конечном базисе. Доказано, что для произвольной ФАЛ от n переменных можно построить такую реализующую ее СФЭ в стандартном базисе, сложность которой асимптотически не больше, чем $\frac{2^n}{n}$, а ее динамическая активность и мощность асимптотически не превосходят $5n$ и $3n$ соответственно. Заметим, что последняя оценка улучшает оценки мощности из работы [2].

Основные определения. Пусть Σ — произвольная СФЭ в базисе $\mathfrak{B}$, имеющая n входов, которым сопоставлены булевские переменные (БП) набора $(x_1, \ldots, x_n) = x$ и k функциональных элементов(ФЭ), причем на выходе ФЭ с номером i в Σ реализуется ФАЛ $\varphi_i(x)$, $i = 1, \ldots, k$. Для набора $\widetilde{\alpha} = (\alpha_1, \ldots, \alpha_n) \in B^n$, где B^n — единичный n-мерный куб, величина

$$E(\Sigma, \widetilde{\alpha}) = \sum_{i=1}^{k} \varphi_i(\widetilde{\alpha})$$

называется *статической активностью схемы* Σ *на наборе* $\widetilde{\alpha}$. Указанная величина характеризует число ФЭ СФЭ Σ на выходах которых сформировалось значение 1 при подаче на входы схемы набора $\widetilde{\alpha}$. *Статической активностью* $E(\Sigma)$ СФЭ Σ называется максимум величины $E(\Sigma, \widetilde{\alpha})$, взятый по всем наборам $\widetilde{\alpha} \in B^n$. Для произвольной ФАЛ f статической активностью $E_{\mathfrak{B}}(f)$ этой ФАЛ будем называть минимальную статическую активность СФЭ в базисе $\mathfrak{B}$, реализующих ФАЛ f.

Пусть $P_2(n)$ — множество всех ФАЛ от n переменных $x_1, x_2, \ldots, x_n$. Функцией Шеннона $E_{\mathfrak{B}}(n)$ для статической активности СФЭ в базисе $\mathfrak{B}$ будем, как обычно, называть максимальное значение статической активности $E_{\mathfrak{B}}(f)$ среди всех ФАЛ $f, f \in P_2(n)$.

Введем теперь понятие динамической активности рассматриваемой СФЭ Σ. Для произвольных наборов $\widetilde{\alpha}$ и $\widetilde{\beta}$ из B^n — наборов значений переменных x, приписанных входам СФЭ Σ, — определим величину

$$S(\Sigma, \widetilde{\alpha}, \widetilde{\beta}) = \sum_{i=1}^{k} (\varphi_i(\alpha) \oplus \varphi_i(\beta)),$$

которую назовем *динамической (переключательной) активностью СФЭ Σ на паре наборов* $(\widetilde{\alpha}, \widetilde{\beta})$. Заметим, что введенная величина характеризует число ФЭ СФЭ Σ, на выходах которых происходит изменение значения при смене набора значений на входах СФЭ с набора $\widetilde{\alpha}$ на набор $\widetilde{\beta}$ или обратно. При этом *динамической активностью* $S(\Sigma)$ СФЭ Σ назовем максимальное значение величины $S(\Sigma, \widetilde{\alpha}, \widetilde{\beta})$ взятое по всем парам наборов $(\widetilde{\alpha}, \widetilde{\beta})$ из $B^n \times B^n$. Для произвольной ФАЛ f ее динамическую активность $S_{\mathfrak{B}}(f)$ определим как минимальную динамическую активность СФЭ в базисе $\mathfrak{B}$, реализующих ФАЛ f, а затем обычным образом введем функцию Шеннона

$$S_{\mathfrak{B}}(n) = \max_{f \in P_2(n)} S_{\mathfrak{B}}(f).$$

Основные результаты. Как уже отмечалось [1, 3], функция Шеннона $E_{\mathfrak{B}}(n)$ для различных конечных полных базисов может при $n = 1, 2, \ldots$ иметь различные порядки роста. В отличие от нее функция Шеннона $S_{\mathfrak{B}}(n)$ в любом конечном полном базисе $\mathfrak{B}$ имеет линейный относительно n, $n = 1, 2, \ldots$ порядок роста.

Теорема 1. *Если $\mathfrak{B}$ — конечный полный базис, то существуют положительные константы c_1 и c_2, зависящие только от базиса $\mathfrak{B}$, что*

$$c_1 \cdot n \leqslant S_{\mathfrak{B}}(n) \leqslant c_2 \cdot n$$

при любом n, $n = 1, 2, \ldots$.

Напомним также, что в [2] была установлена возможность построения для произвольной ФАЛ f, $f \in P_2(n)$, такой реализующей ее СФЭ Σ_f в стандартном базисе $\mathfrak{B}_0 = \{\&, \vee, \neg\}$, для которой ее сложность (число ФЭ) $L(\Sigma_f)$ и статическая активность $E(\Sigma_f)$ при $n = 1, 2, \ldots$ асимптотически не больше, чем $\frac{2^n}{n}$ и $4n$ соответственно. В данной работе указанный результат уточняется и дополняется следующим утверждением.

Теорема 2. *Существует неотрицательная и стремящееся к нулю последовательность действительных чисел $\varepsilon(1), \varepsilon(2), \ldots$ такая, что для любого n, $n = 1, 2, \ldots$, любая ФАЛ f, $f \in P_2(n)$, может быть реализована некоторой СФЭ Σ_f над базисом $\mathfrak{B}_0$, удовлетворяющей неравенствам*

$$L(\Sigma_f) \leqslant (1 + \varepsilon(n))\frac{2^n}{n},\ E(\Sigma_f) \leqslant (1 + \varepsilon(n))3n,\ S(\Sigma_f) \leqslant (1 + \varepsilon(n))5n.$$

Из теоремы 2, в частности, вытекает неравенство $c_2 \leqslant 5$ для константы c_2 из теоремы 66 в случае базиса $\mathfrak{B}_0$.

Работа выполнена при поддержке РФФИ, проект № 12-01-00694а.

Литература

[1] *Вайнцвайг М. Н.* О мощности схем из функциональных элементов // Докл. АН СССР. — 1961. — Т. 139, № 2. — С. 320–323.

[2] *Касим-Заде О. М.* Об одновременной минимизации сложности и мощности схем из функциональных элементов // Проблемы кибернетики. — М.: Наука, 1978. — Вып. 33. — С. 215–220.

[3] *Касим-Заде О. М.* Об одной мере сложности схем из функциональных элементов // Проблемы кибернетики. — М.: Наука, 1981. — Вып. 38. — С. 117–179.

[4] *Devadas S., Keutzer K., White J.* Estimation of power dissipation in CMOS combinational circuits // Proc. Custom Integrated Circuits Conf. — 1990. — P. 19.7.1–19.7.6.

[5] *Ghosh A, Devadas S., Keutzer K., White J.* Estimation of average switching activity in combinational and sequential circuits // Proc. 29th Design Automation Conf. — 1992. — P. 253–259.

[6] *Najm F.* A survey of power estimation techniques in VLSI circuits (Invited paper) // IEEE Trans VLSI Syst. — 1994. — V. 2, № 4. — P. 446–455.

О билинейных отображениях малого ранга

В. В. Лысиков

lysikov-vv@yandex.ru

ВМК МГУ, Москва

Основные определения

Определение 1. Пусть F — поле, U,V,W — линейные пространства над F. Отображение $\varphi\colon U \times V \to W$ называется *билинейным*, если

$$\begin{aligned} \varphi(a_1x_1 + a_2x_2, y) &= a_1\varphi(x_1, y) + a_2\varphi(x_2, y), \\ \varphi(x, b_1y_1 + b_2y_2) &= b_1\varphi(x, y_1) + b_2\varphi(x, y_2) \end{aligned}$$

для любых $a_1, a_2, b_1, b_2 \in F$, $x, x_1, x_2 \in U$, $y, y_1, y_2 \in V$.

Вычислительная сложность билинейных отображений может быть связана с их алгебраическими свойствами (см. напр. [1]). Алгоритмы вычисления билинейных отображений соответствуют разложениям определенного вида:

Определение 2. Пусть F — поле, U,V,W — линейные пространства над F и $\varphi\colon U \times V \to W$ — билинейное отображение. *Билинейным алгоритмом* сложности r для φ называется набор троек $(f_1, g_1, w_1; \ldots; f_r, g_r, w_r)$, где $f_s \in U^*$, $g_s \in V^*$, $w_s \in W$, для которого выполняется соотношение

$$\varphi(x, y) = \sum_{s=1}^{r} f_s(x) g_s(y) w_s$$

при произвольных $x \in U$, $y \in V$.

Рангом отображения φ называется минимально возможная сложность билинейного алгоритма для φ.

Структура билинейных алгоритмов

Пусть U, V, W — линейные пространства над полем F, размерности $\dim U = n$, $\dim V = m$, $\dim W = l$, а $\varphi\colon U \times V \to W$ — билинейное отображение.

Определение 3. Будем говорить, что в билинейном алгоритме

$$(f_1, g_1, w_1; \ldots; f_r, g_r, w_r)$$

есть базисы с перекрытием p, если существуют наборы индексов I и J такие, что $\{f_i | i \in I\}$ и $\{g_j | j \in J\}$ — базисы пространств U^* и V^* соответственно, а $|I \cap J| = p$.

Минимальное возможное перекрытие базисов в алгоритме ранга r равно $n + m - r$, если $r < n + m$, и 0 иначе. Рассмотрим случаи, в которых этот минимум не достигается.

Теорема 1. *Пусть $R(\varphi) = r$. Если в оптимальном билинейном алгоритме для φ есть базисы с перекрытием p ($p > 0$), но нет базисов с перекрытием $p-1$, то существуют разбиения $U = U_1 \oplus U_2 \oplus U_3$, $V = V_1 \oplus V_2 \oplus V_3$ такие, что*

$$\begin{gathered}
\dim U_1 = \dim V_1 = R(\varphi|_{U_1 \times V_1}) = p, \\
\varphi(U_1, V_3) = \varphi(U_3, V_1) = 0, \\
\varphi(U_2, V_2) \subset \varphi(U_1 + U_3, V_2) + \varphi(U_2, V_1 + V_3), \\
R(\varphi|_{U_3 \times V_3}) - \dim U_3 - \dim V_3 = r - m - n + p
\end{gathered}$$

Доказательство. Пусть в оптимальном алгоритме $(f_1, g_1, w_1; \ldots; f_r, g_r, w_r)$ есть базисы с перекрытием p. Пусть I и J — множества индексов, фигурирующие в определении пары базисов с перекрытием, $R = \{1, \ldots, r\} \setminus (I \cup J)$ — множество индексов, не участвующих в этой паре базисов.

Пусть $f = \sum_{i \in I} \alpha_i f_i$, $g = \sum_{j \in J} \beta_j g_j$ — разложения некоторых функционалов $f \in U^*$, $g \in V^*$ по рассматриваемым базисам. Введем обозначения $\operatorname{supp}_1 f = \{i | \alpha_i \neq 0\}$, $\operatorname{supp}_2 g = \{j | \beta_j \neq 0\}$.

Построим последовательности наборов индексов I_k, J_k следующим образом:

$$\begin{gathered}
I_0 = J_0 = R, \\
I_{k+1} = I_k \cup \bigcup\nolimits_{j \in J_k} \operatorname{supp}_1 f_j, \\
J_{k+1} = J_k \cup \bigcup\nolimits_{i \in I_k} \operatorname{supp}_1 g_i
\end{gathered}$$

Последовательности I_k и J_k монотонно возрастающие. Обозначим их объединения $\hat{I}$ и $\hat{J}$ и рассмотрим два случая:

1 случай. Найдется индекс $t \in I \cap J$, входящий в $\hat{I}$ (тогда он обязательно входит и в $\hat{J}$). В этом случае есть цепочка индексов $t = t_0, t_1, t_2, \ldots t_d \in$

R такая, что для натуральных k выполняется $t_{2k} \in \operatorname{supp}_1 f_{t_{2k+1}}$, $t_{2k+1} \in \operatorname{supp}_1 g_{t_{2k+2}}$. Это значит, что в базисе $\{f_i | i \in I\}$ функционал f_{t_0} можно заменить на f_{t_1}, получив новую пару базисов с перекрытием p. Аналогично, функционал g_{t_1} в базисе $\{g_j | j \in J\}$ можно заменить на g_{t_2}. Последовательно проводя такие замены, мы в конце концов заменим какой-то элемент какого-то из базисов на функционал с индексом $t_d \in R$. При этом перекрытие базисов уменьшится на 1. Таким образом, в этом случае условие теоремы не выполняется.

2 случай. $\hat{I}$ и $I \cap J$, а также $\hat{J}$ и $I \cap J$, не имеют общих элементов. Множество I разделяется на три непересекающиеся части: $I \cap J$, $I \cap \hat{I}$ и $I' = I \setminus (\hat{I} \cup J)$. Аналогично, J делится на $I \cap J$, $J \cap \hat{J}$ и $J' = J \setminus (\hat{J} \cup I)$.

Рассмотрим базисы $\{x_i\}$ и $\{y_j\}$ пространств U и V соответственно, двойственные базисам $\{f_i | i \in I\}$ и $\{g_j | g \in J\}$. Для подпространств

$$U_1 = \operatorname{lin}\{x_i | i \in I \cap J\},\ U_2 = \operatorname{lin}\{x_i | i \in I'\},\ U_3 = \operatorname{lin}\{x_i | i \in I \cap \hat{I}\},$$

$$V_1 = \operatorname{lin}\{x_i | i \in I \cap J\},\ V_2 = \operatorname{lin}\{x_i | i \in J'\},\ V_3 = \operatorname{lin}\{x_i | i \in J \cap \hat{J}\}$$

выполняются соотношения из условия теоремы. При этом более оптимального алгоритма для $\varphi|_{U_3 \times V_3}$ не существует, так как заменив тройки (f_s, g_s, w_s) с $s \in \hat{I} \cup \hat{J}$, составляющие алгоритм ранга $\dim U_3 + \dim V_3 + r - m - n + p$, на более эффективный алгоритм, мы получим более эффективный алгоритм для исходного отображения φ. ∎

Применения

Доказанная теорема полезна для исследования билинейных отображений малого ранга. В частности

Теорема 2. *Над полем действительных чисел ранг умножения комплексных матриц размера* 2×2 *на комплексный вектор равен* 12, *а ранг умножения кватерниона на пару кватернионов равен* 16.

Нижние оценки доказываются перебором возможных вариантов значения перекрытия базисов в алгоритмах меньшего ранга. Полученные результаты улучшают нижние оценки для указанных отображений, следующие из результатов [2].

Также удалось доказать, что перекрытие базисов в оптимальных алгоритмах для умножения квадратных матриц минимально.

Теорема 3. *В оптимальном алгоритме умножения квадратных матриц размера* $n \times n$ *при* $n \geqslant 3$ *есть пара базисов с перекрытием* 0.

Небольшой модификацией доказательства леммы 5 из [3] из этого факта можно получить нижнюю оценку

Теорема 4. *Пусть* U, V *— подпространства* $F^{n \times n}$. *Тогда*

$$R(\langle n, n, n \rangle) \geqslant \dim U + \dim V + \min_{\substack{U \oplus X = F^{n \times n}, \\ V \oplus Y = F^{n \times n}}} \dim \operatorname{lin}(\{\varphi(x, y) | x \in X, y \in Y\}).$$

Работа выполнена при поддержке РФФИ, проект № 12–01–91331–ННИО_а.

Литература

[1] *Bürgisser P. et al.* Algebraic Complexity Theory. — Berlin: Springer-Verlag, 1997. — 618 p.

[2] *Hartmann W.* On the multiplicative complexity of modules over associative algebras // SIAM J. Comput. — 1985. — V. 14, № 2. — P. 383–395.

[3] *Bläser M.* The complexity of bivariate power series arithmetic // Theor. Comp. Sci. — 2003. — V. 295, № 1. — P. 65–83.

(6,3)-бирегулярные графы, нераскрашиваемые интервально 6 цветами

А. М. Магомедов

magomedtagir1@yandex.ru

Дагестанский государственный университет, Махачкала

В сообщении используется терминология из [1]. *Интервальной раскраской* графа G t цветами будем называть сюръективное отображение множества ребер графа G в множество $\{1, 2, \ldots, t\}$ такое, что в каждой вершине G все представленные в ней цвета попарно различны и образуют целочисленный интервал.

В [2] доказана NP-полнота задачи об интервальной раскрашиваемости 6 цветами двудольного графа $G = (X, Y, E)$, в котором степени всех вершин X равны 6, а степени всех вершин Y равны 3. Такие графы будем называть $(6,3)$-*бирегулярными* или, кратко, $(6,3)$-*графами*; $(6,3)$-граф $G = (X, Y, E)$, где $|X| = n$, будем называть $(6,3)_n$-*графом*. $(6,3)$-граф G будем называть *раскрашиваемым* (*нераскрашиваемым*), если G допускает (не допускает) интервальной раскраски 6 цветами.

Из результата [2], очевидно, следует существование нераскрашиваемых $(6,3)_n$-графов. В [3] построен пример нераскрашиваемого $(6,3)_{11}$-графа. В данном сообщении уточнено наименьшее значение n, для которого существует нераскрашиваемый $(6,3)_n$-граф. Более точно, доказана следующая теорема.

Теорема 1. *Для любого $n \geqslant 6$ существует нераскрашиваемый $(6,3)_n$-граф. При $n < 5$ любой $(6,3)_n$-граф раскрашиваем.*

Работа выполнена при финансовой поддержке гос.задания № 2014/33 Минобрнауки РФ.

Литература

[1] *Свами М., Тхуласираман К.* Графы, сети и алгоритмы. — М.: Мир, 1984. — 455 с.

[2] *Casselgren C. J.* On Some Graph Coloring Problems // Doctoral Thesis (Department of Mathematics and Mathematical Statistics Umea University). — 2011. № 48.

[3] *Магомедов А. М.* О реберной раскраске двудольного графа // Материалы XI Международного семинара «Дискретная математика и ее приложения», посвященного 80-летию со дня рождения академика О. Б. Лупанова / Под редакцией О. М. Касим-Заде. — М.: Изд-во механико-математического факультета МГУ, 2012. — С. 298—299.

О размерности пространства стационарных функций в трехзначной логике

А. А. Мазуров

anat-mazurov@mail.ru

МГУ имени М. В. Ломоносова, Москва

Введение

Пусть k – натуральное число, $k \geqslant 2$. Множество всех натуральных чисел от 0 до $k-1$ обозначается через E_k: $E_k = \{0, \ldots, k-1\}$. *Функцией* k-значной логики называется отображение $\varphi : E_k^n \to E_k$, где $E_k = \{0, \ldots, k-1\}$. Множество всех функций k-значной логики от n переменных обозначается $P_k(n)$. Множество всех функций k-значной логики (от любого количества переменных) обозначается P_k. *Вектором значений* функции f, зависящей от переменных $x_1, \ldots, x_n$, называется последовательность значений функции на всех наборах от $(0, \ldots, 0)$ до $(k-1, \ldots, k-1)$. *Полиномом* в k-значной логике называется формула вида

$$f(x_1, \ldots, x_n) = \sum_{\tilde{\alpha} \in E_k^n} c_\alpha x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n} \pmod{k}$$

где $x^\alpha = \begin{cases} 1, \alpha = 0 \\ \underbrace{x \cdot x \cdot \ldots \cdot x}_{\alpha}, \alpha \neq 0 \end{cases}$, $c_\alpha \in E_k$, а $\tilde{\alpha} = \{\alpha_1, \ldots, \alpha_n\}$. Числа c_α называются *коэффициентами полинома*.

Преобразование Мёбиуса – это отображение, переводящее вектор значений функции в вектор коэффициентов соответствующего ей полинома.

$Q_l(n) = \{f \in E_3(n) \mid \mu(f) \equiv l \cdot f\}$, $l \in \{1, 2\}$ – это класс функций, являющихся неподвижными точками преобразования Мебиуса. Мы будем называть такие функции *стационарными с коэффициентом* l.

Кронекеровым произведением $A \times B$ матриц A и B называется матрица C размера $mp \times nq$ следующего блочного строения:

$$C = \begin{bmatrix} a_{11}B & a_{12}B & \ldots & a_{1n}B \\ a_{21}B & a_{22}B & \ldots & a_{2n}B \\ \ldots & \ldots & \ldots & \ldots \\ a_{m1}B & a_{m2}B & \ldots & a_{mn}B \end{bmatrix}.$$

Кронекеровой k-й степенью квадратной матрицы A будем называть выражение

$$A^{\times k} = \begin{cases} A, & k = 1, \\ A \times A^{[k-1]}, & k \geqslant 2. \end{cases}$$

Подсчет стационарных функций

Переформулируя результаты, описанные в [4], получаем следующую теорему.

Теорема 1. *Размерность подпространства стационарных функций трехзначной логики равна*

$$\frac{3^n + (-1)^n + 4 \cdot 3^{\frac{n}{2}} \arccos(\frac{1}{\sqrt{3}}) + 2}{8}.$$

Докажем эту теорему при помощи других рассуждений.

Доказательство. Вектора значений стационарных функций – это собственные вектора пространства векторов значений всех функций, соответствующие собственному значению 1 матрицы преобразования. Поскольку матрица преобразования для функций n переменных является кронекеровой n-й степенью матрицы T_1 ([1]), жорданова форма которой диагональна, то множество ее собственных значений можно найти как кронекерову степень вектора, состоящего из собственных значений матрицы T_1 [2], где

$$T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 2 & 2 & 2 \end{bmatrix}$$

Этот вектор имеет вид $(1, x, 2x+1)$, где x и $2x+1$ являются элементами поля $\mathbb{F}_3|_{(x^2+2x+2)}$.

Таким образом множество собственных значений (без учета кратности) матрицы $T_1^{\times n}$ равно $(1, x, 2x+1)^{\times n}$. Выписав таблицу умножения поля $\mathbb{F}_3|_{(x^2+2x+2)}$, можно получить рекуррентную систему уравнений для нахождения числа единиц в этом множестве. Приведем рассуждения для первого уравнения, остальные полностью аналогины.

Обозначим $s_y(n)$ – количество собственных значений y матрицы T_1^n. Чтобы получить собственное значение 1 в матрице T_1^n, нужно либо собственное значение 1 матрицы T_1 умножить на (любое) собственное значение 1 матрицы T_{n-1}, либо x на $x+2$, либо $2x+1$ на $2x$. Поэтому количество с. з. 1 матрицы T_1^n равно

$$s_1(n) = s_1(n-1) + s_{x+2}(n-1) + s_{2x+1}(n-1).$$

Точно таким же образом можно составить остальные уравнения системы. В итоге получаем систему, которую проще записать в матричном виде: $\tilde{s}(n) = L\tilde{s}(n-1)$, или, что то же самое, $\tilde{s}(n) = L^n\tilde{s}(0)$, где $\tilde{s}(n) = (s_1(n), s_{x+2}(n), s_{2x}(n), s_2(n), s_{2x+2}(n), s_{x+1}(n), s_x(n), s_{2x+1}(n))^T$, а L – матрица

из нулей и единиц:

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

Нетрудно видеть, что $\tilde{s}(0) = (1,0,0,0,0,0,0,0)^T$.

Таким образом, для нахождения количества стационарных функций нужно получить общий вид для L^n. Для этого воспользуемся жордановой формой этой матрицы: $L = SJS^{-1}$, где

$$S = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & -1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 1 & 0 & 1 & 0 & 1 \\ -1 & 0 & 1 & 1 & 1 & \sqrt{2}\,i & 1 & -\sqrt{2}\,i \\ 0 & -1 & -1 & 1 & -\sqrt{2}\,i & 1 & \sqrt{2}\,i & 1 \\ 0 & 1 & -1 & 1 & 0 & -1 & 0 & -1 \\ 0 & -1 & -1 & 1 & \sqrt{2}\,i & -1 & -\sqrt{2}\,i & -1 \\ -1 & 0 & 1 & 1 & -1 & -\sqrt{2}\,i & -1 & \sqrt{2}\,i \end{bmatrix},$$

а J – диагональная матрица

$$J = diag(1, 1, -1, 3, 1 - i\sqrt{2}, 1 - i\sqrt{2}, 1 + i\sqrt{2}, 1 + i\sqrt{2}).$$

Вид этих матриц получен на компьютере при помощи пакета с открытым исходным кодом Maxima [3].

Получаем, что $L^n = SJ^nS^{-1}$. С учетом того, что $\tilde{s}(0) = (1,0,0,0,0,0,0,0)^T$, нас интересует только первый столбец, в котором $s_1(n)$ является первым элементом. Теорема доказана. ∎

Попутно была также получена и размерность класса $Q_2(n)$ – это четвертый элемент столбца, соответствующий $s_2(n)$. Выражение для него отличается только знаком при комплексных слагаемых.

Заключение

В данной статье описан более простой способ вычисления размерности классов стационарных функций, по сравнению с приведенным в [4]. При описанном подходе теряется структура стационарных классов и взаимосвязи между ними, указанные в [5], однако данный способ проще и универсальнее для подсчета количества стационарных функций.

Работа выполнена при поддержке РФФИ, проект № 13-01-00684-а.

Литература

[1] *Мазуров А. А.* О стационарных классах функций трехзначной логики // Вестник Московского университета. Серия 15 Вычислительная математика и кибернетика — 2012. — № 2. — С. 37–43.

[2] *Воеводин В. В., Кузнецов Ю. А.* Матрицы и вычисления. — М.: Наука, 1984.

[3] http://maxima.sourceforge.net/ru/

[4] *Мазуров А. А.* О числе стационарных точек преобразования Мёбиуса в трехзначной логике // Материалы молодежного научного форума „Ломоносов-2013“. — М.: МАКС Пресс. 2013. — ISBN: 978-5-317-04429-9. [Электронный ресурс]

[5] *Мазуров А. А.* Структура стационарных классов функций трехзначной логики // Вестник Московского университета. Серия 15: Вычислительная математика и кибернетика. — 2013. — № 2. — Р. 33–38.

Обучение линейной комбинации метрик на конечной выборке

А. И. Майсурадзе, М. А. Суворов

maysuradze@cs.msu.su, severe013@gmail.com

МГУ имени М.В. Ломоносова, Москва

В задачах интеллектуального анализа данных всё чаще встречаются ситуации, когда на одних и тех же объектах распознавания заданы различные способы измерения их сходства. Такие ситуации характерны для задач компьютерного зрения, биологии, в социальных системах. Одним из способов формализовать понятие сходства является метрика. Тогда можно сказать, что объекты распознавания являются элементами мультиметрического пространства. По аналогии с задачами распознавания, в которых объект описывается несколькими признаками, при решении прикладных задач в случае наличия нескольких метрик разумно надеяться, что даже если отдельные метрики не очень качественно разделяют классы, то их совместное использование позволит повысить качество разделения.

Для решения фундаментальных задач интеллектуального анализа данных, в первую очередь задач обучения с учителем, уже классическими стали некоторые метрические информационные модели, особенно методы ближайших соседей и парзеновского окна, методы потенциальных функций. Важно отметить, что эти классические методы формулируются, получают теоретическое обоснование, исследуются для ситуаций с одной метрикой. Следовательно, приходится решать задачу агрегирования первичной метрической информации, т.е. по набору расстояний на одних и тех же объектах требуется построить новую метрику на всей генеральной совокупности. В интеллектуальном анализе данных авторы обычно допускают равенство расстояния нулю на несовпадающих объектах. Таким образом, формально правильно было бы везде говорить о полуметриках.

При конкретизации указанной задачи, как и в случае признаковых описаний, можно идти разными путями. Метрические аналоги методов разделения сигналов (blind signal separation), например, метрический метод главных компонент, метрическая неотрицательная матричная факторизация, рассматривались ранее в работах [1, 2, 3].

В данной работе рассматривается случай, когда исходные попарные расстояния вычислены на конечной выборке, для объектов этой конечной выборки заданы истинные метки классов, требуется построить новую метрику *на всей генеральной совокупности*, которая наилучшим образом соответствует распределению прецедентов по классам. Эту задачу можно отнести к группе задач обучения метрик (metric learning). От распространенных в литературе задач обучения метрик она отличается тем, что строится новая метрика по исходным метрикам, а не метрика по исходным признакам. Предлагается подход, в котором указанная задача сводится к задаче линейного программирования. Рассматриваются теоретические свойства задачи, приводятся эмпирические данные.

В подавляющем большинстве работ последнего времени, посвященных обучению метрик, одна метрика строится по признаковому описанию конечного набора прецедентов. При этом в качестве модели метрики выбирается метрика Махаланобиса, рассматриваемая как обобщение евклидовой метрики. Наибольшую известность в этом направлении получили методы, базирующиеся на идеях Large-Margin nearest neighbor [4] и Information-theoretic metric learning [5]. Положительной стороной указанных подходов является то, что они напрямую пытаются повысить ту или иную величину, характеризующую "компактность" классов. Например, требуют, чтобы ближайший сосед был из своего же класса. Оборотной стороной этой медали является то, что задачи сводятся к неудобным задачам оптимизации, которые трудно исследовать и решать. Соответственно, часто авторы довольствуются некоторым количеством итераций градиентного спуска, даже не проверяя сходимость.

В рассматриваемом в данной работе случае размеры задач ещё больше. Расстояния считаются для пар объектов, многообразие способов сопоставить объекты между собой приводит к большим объемам сравнительной информации. Поэтому определенный интерес представляют те способы формализовать рассматриваемую задачу агрегирования метрик, которые сводят её к удобной с точки зрения оптимизации модели. Одну такую формализацию мы сейчас и рассмотрим.

Когда пытаются формализовать идею о том, что внутриклассовые расстояния должны быть маленькими, а межклассовые — большими, то обычно переходят к одной скалярной величине, характеризующей отношение внутриклассовых и межклассовых расстояний, и решают задачу безусловной оптимизации. Или, по аналогии с SVM, можно и внутриклассовые, и межклассовые расстояния ввести в ограничения и требовать их наибольшего разделения. Мы предлагаем рассмотреть внутриклассовые и межклассовые расстояния асимметрично: внутриклассовые расстояния идут в целевую функцию, межклассовые расстояния идут в ограничения.

В работах [6, 7] было показано, что удобно теоретически исследовать и вычислительно эффективно реализовывать методы, в которых новая метрика является линейной комбинацией исходных.

Пусть $x_1, \ldots, x_M$ — объекты выборки. Пусть на множестве объектов заданы исходные метрики $\rho_1, \ldots, \rho_N$. Пусть новая метрика r является линейной комбинацией исходных: $r(x,y) = w_1\rho_1(x,y) + \cdots + w_N\rho_N(x,y)$. Требуется ми-

нимизировать среднее внутриклассовое расстояние при условии того, что все межклассовые расстояния не менее 1 и веса w неотрицательны. Это задача линейного программирования, соответственно, существует широкий спектр программного обеспечения, позволяющий решить её. Будет найден глобальный оптимум.

Неотрицательность весов гарантирует выполнение аксиом полуметрики. Более того, она позволяет смотреть на данную задачу как на задачу отбора метрик.

Хочется отметить, что в предложенной постановке не требуется вводить переменные натяжения, привычные по SVM. Назовём пару объектов из разных классов непротиворечивой, если хоть в одной из исходных метрик расстояние между ними ненулевое.

Теорема 1. *В указанной задаче линейного программирования всегда существует допустимый план, если только в выборке нет противоречивых объектов.*

Рассмотрим задачу из области компьютерного зрения, в которой на множестве объектов вычисляется 8 исходных метрик. Решается задача классификации. Новая метрика настраивается на обучении. Для сравнения исходных и новых метрик используется простой метод 1NN (сравнение с эталоном), поскольку наша цель — оценить качество метрик, а не построить лучший классификатор. В таблице представлены проценты ошибок для каждой из исходных метрик и для новой метрики. Качество классификации оценивается на контроле, всё обучение является эталонами.

ρ_1	84.2	ρ_2	89.7	ρ_3	86.2	ρ_4	89.5	ρ_5	86.6
ρ_6	84.8	ρ_7	88.7	ρ_8	85.0			r	23.0

Если линейную комбинацию настраивать именно под 1NN, то можно получить ошибку 19.4%. Разумеется, время оптимизации вырастает на 2 порядка, нет гарантии, что был найден глобальный оптимум.

Работа выполнена при поддержке РФФИ, проект № 13-01-00751-а.

Литература

[1] *Суворов М. А., Майсурадзе А. И.* Методы агрегирования метрических описаний // ММРО: 16-я Всероссийская конференция, г. Казань, 6-12 сентября 2013г.: Тезисы докладов. — М.: Торус Пресс, 2013. — С. 11.

[2] *Майсурадзе А. И.* Метрический метод главных компонент для генеральной совокупности // Интеллектуализация обработки информации: 9-я международная конференция. Черногория, г.Будва, 2012 г.: Сборник докладов. — М.: Торус Пресс, 2012. — С. 168–170.

[3] *Суворов М. А.* Методы агрегации метрических описаний на основе оптимальной матричной факторизации // Ломоносов-2012: XIX Международная научная конференция студентов, аспирантов и молодых ученых; секция Вычислительная математика и кибернетика: Москва, МГУ имени М.В. Ломоносова, 9-13 апреля 2012 г.: Сб. тезисов. — М.: Издательский отдел факультета ВМиК МГУ; МАКС Пресс, 2012. — С. 109–110.

[4] *Weinberger K. Q., Saul L. K.* Distane metric learning for large margin nearest neighbor classification // Journal of Mahine Learning Research. — 2009. — V. 10. — P. 207–244.

[5] *Davis J. V., Kulis B, Jain P., Sra S., Dhillon I. S.* Information-theoretic metric learning // Proceedings of the 24th international conference on Machine learning. — ACM, 2007. — P. 209–216.

[6] *Maysuradze A. I.* On optimal decompositions of finite metric configurations in pattern recognition problems // J. Comput. Math. Math. Phys. — 2004. — V. 44, No. 9 — P. 1615–1624.

[7] *Maysuradze A. I.* Homogeneous and rank bases in spaces of metric configurations // J. Comput. Math. Math. Phys. — 2004. — V. 46, No. 2 — P. 330–344.

О сложности задач о раскраске для наследственных классов с запретами небольшого размера

Д. С. Малышев

dsmalyshev@rambler.ru

ННГУ, НИУ ВШЭ, Н.Новгород

Классом называется любое множество непомеченных обыкновенных графов. Класс графов называется *наследственным*, если он замкнут относительно удаления вершин. Любой наследственный класс (и только наследственный класс) графов $\mathcal{X}$ может быть задан множеством своих *запрещенных порожденных подграфов* $\mathcal{S}$ (т.е. графов, которые нельзя получить удалением вершин в графах из $\mathcal{X}$), при этом принята запись $\mathcal{X} = \mathcal{F}ree(\mathcal{S})$.

Раскраской вершин (соответственно, *ребер*) *графа* $G = (V, E)$ *в* k *цветов* называется отображение $c : V \longrightarrow \{1, 2, \ldots, k\}$ (соответственно, $c : E \longrightarrow \{1, 2, \ldots, k\}$), где значения функции c называются *цветами*. Раскраска вершин (соответственно, ребер) называется *правильной*, если соседние вершины (соответственно, ребра) покрашены в разные цвета. *Хроматическим числом графа* G называется наименьшее количество цветов, необходимое для правильного раскрашивания его вершин. Оно обозначается через $\chi(G)$.

Задача о вершинной k*-раскраске* состоит для заданного графа G в том, чтобы определить, имеет ли G правильную раскраску вершин в k цветов. По аналогии определяется задача о реберной k-раскраске. *Задача о хроматическом числе* (иногда называемая *задачей о раскраске*) состоит для заданных графа G и натурального числа k в том, чтобы определить выполнение неравенства $\chi(G) \leqslant k$. Реберным аналогом задачи о хроматическом числе является задача о хроматическом индексе. Все перечисленные задачи являются NP-полными в классе всех графов и остаются таковыми даже при значительных сужениях этого множества. Например, задача о реберной 3-раскраске NP-полна в классе графов со степенями всех вершин не более чем 3 [1], а задача о вершинной 3-раскраске NP-полна в классе планарных графов [2].

По-видимому, получение полной сложностной дихотомии в семействе наследственных классов для задач о вершинной и реберной k-раскраске ($k > 2$), задач о хроматическом числе и индексе невозможно в принципе [3, 4, 5, 6]. Но, для некоторых из этих задач известны полные классификации в семействе наследственных классов с запрещенными фрагментами небольшого размера.

Как обычно, P_n, C_n, O_n означает простой путь, простой цикл и пустой граф с n вершинами соответственно. Граф $K_{p,q}$ — полный двудольный граф с p вершинами в одной доле и q вершинами в другой. Формула $G_1 + G_2$ обозначает объединение двух графов G_1 и G_2 с непересекающимися множествами вершин.

Теорема 1. *[7] В семействе классов $\{\mathcal{F}ree(\{H\}) : |V(H)| \leqslant 4\}$ задача о хроматическом числе является полиномиально разрешимой, если H — порожденный подграф графа C_4 или графа $P_3 + P_1$. Она является NP-полной для всех остальных классов данного семейства.*

Линейным лесом называется граф, каждая компонента связности которого является простым путем.

Теорема 2. *[8] В семействе классов $\{\mathcal{F}ree(\{H\}) : |V(H)| \leqslant 6\}$ задача о вершинной 3-раскраске является полиномиально разрешимой, если H — линейный лес. Она является NP-полной для всех остальных классов данного семейства.*

Теорема 3. *[9] В семействе классов $\{\mathcal{F}ree(\{H\}) : |V(H)| \leqslant 5\}$ задача о вершинной 4-раскраске является полиномиально разрешимой, если H — линейный лес. Она является NP-полной для всех остальных классов данного семейства.*

До недавнего времени для задачи о реберной k-раскраске и для задачи о хроматическом индексе автору не было известно утверждений, подобных приведенным выше теоремам. По-видимому, первым фактом такого рода является полная сложностная дихотомия для задачи о реберной 3-раскраске в семействе $\{\mathcal{F}ree(\mathcal{S}) :$ каждый граф из $\mathcal{S}$ имеет не более 6-ти вершин$\}$.

Через $\mathcal{T}$ будем обозначать множество графов, каждая компонента связности которых является деревом с не более чем тремя листьями, а $\mathcal{D}$ множество реберных графов к графам из $\mathcal{T}$. Графы *hantel* и *barbell* — графы на множестве вершин $\{1, 2, 3, 4, 5, 6\}$ с множествами ребер $\{(1, 2), (1, 3), (4, 5), (4, 6), (1, 4)\}$ и $\{(1, 2), (1, 3), (2, 3), (4, 5), (4, 6), (5, 6), (1, 4)\}$ соответственно. Графы *bull* и *butterfly* — графы на множестве вершин $\{1, 2, 3, 4, 5\}$ и имеют множества ребер $\{(1, 2), (1, 3), (2, 3), (1, 4), (2, 5)\}$, $\{(1, 2), (1, 3), (2, 3), (1, 4), (1, 5), (4, 5)\}$ соответственно.

Теорема 4. *[10] Пусть $\mathcal{S}$ — произвольное множество, содержащее только графы с не более чем 6-ю вершинами. Задача о реберной 3-раскраске полиномиально разрешима в классе $\mathcal{X} = \mathcal{F}ree(\mathcal{S})$, если $\mathcal{T} \nsubseteq \mathcal{X}, \mathcal{D} \cup \{barbell\} \nsubseteq \mathcal{X}$ или $\mathcal{F}ree(\{hantel\}) \nsubseteq \mathcal{X}, \mathcal{D} \cap \mathcal{F}ree(\{bull\}) \nsubseteq \mathcal{X}$. Во всех остальных случаях она NP-полна в классе $\mathcal{F}ree(\mathcal{S})$.*

Вернемся к задачам о вершинной k-раскраске и о хроматическом числе. Применительно к задаче о хроматическом числе построение полной

классификации наталкивается на серьезные трудности уже для пар запрещенных графов маленького размера. Именно, для всех классов семейства $\{\mathcal{F}ree(\mathcal{S}) : \text{ каждый граф из } \mathcal{S} \text{ имеет не более 4-х вершин}\}$ известен вычислительный статус задачи о хроматическом числе, за исключением $\mathcal{S} = \{C_4, O_4\}, \mathcal{S} = \{K_{1,3}, O_4\}, \mathcal{S} = \{K_{1,3}, P_2 + P_1 + P_1, O_4\}$ [12]. Там же показано, что задача о хроматическом числе в классе $\mathcal{F}ree(\{K_{1,3}, O_4\})$ полиномиально эквивалентна той же задаче в классе $\mathcal{F}ree(\{K_{1,3}, P_2 + P_1 + P_1, O_4\})$. По всей видимости, оба этих случая являются NP-полными, а $\mathcal{F}ree(\{C_4, O_4\})$ является полиномиальным. В [11] были получены некоторые результаты, касающиеся сложности задачи о хроматическом числе для наследственных классов с двумя пятивершинными связными запретами.

Недавно автору удалось получить полную характеризацию сложности задачи о вершинной 3-раскраске для наследственных классов с двумя запретами, содержащими 5 вершин.

Теорема 5. *[13] Пусть G_1 и G_2 — произвольные графы с не более чем 5-ю вершинами. Задача о вершинной 3-раскраске является полиномиально разрешимой в классе $\mathcal{F}ree(\{G_1, G_2\})$, если среди G_1, G_2 один из графов является лесом, а другой принадлежит $\mathcal{D} \cup \{butterfly\}$ и при этом $\{G_1, G_2\} \neq \{K_{1,4}, bull\}$. Во всех остальных случаях данная задача будет NP-полной в классе $\mathcal{F}ree(\{G_1, G_2\})$.*

Работа выполнена при поддержке РФФИ, проект № 14-01-00515-а; при поддержке гранта Президента РФ МК-1148.2013.1; при поддержке лаборатории алгоритмов и технологий анализа сетевых структур НИУ ВШЭ, грант правительства РФ, дог. 11.G34.31.0057. Исследование осуществлено в рамках Программы "Научный фонд НИУ ВШЭ" в 2013-2014 гг., проект № 12-01-0035.

Литература

[1] *Holyer I.* The NP-completeness of edge-coloring // SIAM Journal on Computing. — 1981. — V. 10, № 4. — P. 718–720.

[2] *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982. — 318 с.

[3] *Малышев Д.С.* Континуальные множества граничных классов графов для задач о раскраске // Дискретный Анализ и Исследование Операций. — 2009. — Т. 16, № 5. — С. 41–51.

[4] *Korpelainen N., Lozin V.V., Malyshev D.S., Tiskin A.* Boundary properties of graphs for algorithmic graph problems // Theoretical Computer Science. — 2011. — V. 412. — P. 3545–3554.

[5] *Малышев Д.С.* Исследование граничных классов графов для задач о раскраске // Дискретный Анализ и Исследование Операций. — 2012. — Т. 19, № 6. — С. 37–48.

[6] *Малышев Д.С.* О пересечении и симметрической разности семейств граничных классов для задач о раскраске и о хроматическом числе // Дискретная Математика. — 2012. — Т. 24, № 2. — С. 75–78.

[7] *Kral' D., Kratochvil J., Tuza Z., Woeginger G.* Complexity of coloring graphs without forbidden induced subgraphs // Lecture Notes in Computer Science. — 2001. — V. 2204. — P. 254–262.

[8] *Broersma H., Golovach P., Paulusma D., Song J.* Updating the complexity status of coloring graphs without a fixed induced linear forest // Theoretical Computer Science. — 2012. — V. 420. — P. 28–35.

[9] *Golovach P., Paulusma D., Song J.* 4-coloring H-free graphs when H is small // Discrete Applied Mathematics. — 2013. — V. 161, № 1-2. — P. 140–150.

[10] *Malyshev D.S.* A complete complexity dichotomy for the edge 3-colorability problem in the absence of induced fragments with at most 6 vertices // Сибирские Электронные Математические Известия. — 2013 (направлено в журнал).

[11] *Malyshev D.S.* The coloring problem for classes with two small obstructions // Optimization Letters. — 2014. doi: 10.1007/s11590-014-0733-y (accepted).

[12] *Lozin V.V., Malyshev D.S.* Vertex coloring of graphs with few obstructions // Discrete Applied Mathematics. — 2013 (submitted).

[13] *Malyshev D.S.* The complexity of the 3-colorability problem in the absence of a pair of small forbidden induced subgraphs // Discrete Mathematics. — 2013 (submitted).

Алгоритмические задачи, связанные с полнотой в функциональной системе L(Z)

А. И. Мамонтов, Д. Г. Мещанинов

MamontovAI@yandex.ru, MeshchaninovDG@mpei.ru

Московский энергетический институт, Москва

В настоящей работе исследуется функциональная система $L(\mathbb{Z})$ линейных полиномов над кольцом $\mathbb{Z}$ с операциями суперпозиции [1]. Важнейшими в любой универсальной алгебре являются вопросы полноты и выразимости. Функциональная система $L(\mathbb{Z})$ (замкнутый класс в счетнозначной логике) конечнопорождаема, множество $\{1, x - y\}$ образует её базис, поэтому критерий полноты в этой алгебре можно сформулировать в терминах предполных классов (максимальных собственных подалгебр). Установлено, что в функциональной системе $L(\mathbb{Z})$ предполные классы образуют счетное множество, все они описаны в [1], что позволило вывести критерий полноты в этой алгебре. Несмотря на бесконечность множества всех предполных классов, проблема полноты в $L(\mathbb{Z})$ оказалась алгоритмически разрешимой.

Основными объектами нашего рассмотрения являются функции $f : \mathbb{Z}^n \to \mathbb{Z}$, задаваемые полиномами

$$f(x_1, \ldots, x_n) = a_0 + a_1 x_1 + \ldots + a_n x_n, \quad a_0, a_1, \ldots, a_n \in \mathbb{Z}. \tag{1}$$

Каждый такой полином полностью определяется вектором коэффициентов $(a_0, a_1, \ldots, a_n)$ и представляет ровно одну функцию Мы отождествляем функцию f, реализующий ее полином вида (1) и все конгруэнтные и равные f функции (т. е. получаемые из f переименованием переменных, а также введением и изъятием фиктивных переменных). Множество всех таких функций мы обозначаем как $L(\mathbb{Z})$. Такое же обозначение применяем и для функциональной системы (алгебры) $L(\mathbb{Z})$ с операциями суперпозиции. Буквами p,

возможно, с индексами, обозначаем простые числа, $НОД(c_1, \ldots, c_k)$ — наибольший общий делитель целых $c_1, \ldots, c_k$.

В [1] найдены все предполные в $L(\mathbb{Z})$ классы. Перечислим их.

1. Класс L^+ функций, у которых все коэффициенты при переменных неотрицательны.

2. Класс D, содержащий все константы и унарные функции, а также все n-местные ($n \geqslant 2$) функции, имеющие $НОД(a_1, \ldots, a_n) > 1$.

3. Классы $C(p_1 \cdots p_r)$, где $p_1, \ldots, p_r$ — различные простые числа, $r \geqslant 1$, состоят из функций, у которых все коэффициенты $a_1, \ldots, a_n$ кратны некоторому p_i, $1 \leqslant i \leqslant r$, или все коэффициенты $a_1, \ldots, a_n$ кроме, возможно, одного кратны $p_1 \cdots p_r$. Каждый класс $C(p_1 \cdots p_r)$ содержит все константы и унарные функции.

4. Классы $S(p)$ функций, у которых $a_1 + \cdots + a_n \equiv 1 \pmod{p}$.

5. Классы $U(b, p)$, $b = 0, 1, \ldots, p-1$, состоят из функций, сохраняющих множество $\{c \in \mathbb{Z} \mid c \equiv b \pmod{p}\}$.

Пусть далее $F = \{f_1, \ldots, f_m\}$ — конечная система функций из $L(\mathbb{Z})$. Требуется определить, полна ли система F в $L(\mathbb{Z})$ (выдать ответ „Да“или „Нет“). Каждую из функций системы F считаем зависящей от одного множества переменных $\{x_1, \ldots, x_n\}$ и представляем в виде

$$f_i(\tilde{x}) = a_{i0} + a_{i1}x_1 + \ldots + a_{in}x_n,$$

$i = 1, \ldots, m$, при этом некоторые из коэффициентов a_{ij}, $j = 0, \ldots, n$, могут быть нулевыми.

Нами получен алгоритм (будем называть его A) распознавания полноты в $L(\mathbb{Z})$ и

Теорема 1. *Если все коэффициенты функций системы F ограничены по абсолютной величине константой t и максимальное количество переменных функций есть n, то размером задачи является $N = mnt$ и алгоритм имеет при реализации машиной с произвольным доступом к памяти временную сложность $O(N^2 \log^2 N)$ (двоичных операций) и емкостную сложность $O(N \log N)$ (битов).*

Алгоритм можно применить и для распознавания относительной полноты.

Проблема *полноты относительно* заданного *класса* K (не обязательно замкнутого) состоит в выяснении полноты системы, содержащей класс K. Рассмотрим эту проблему в $L(\mathbb{Z})$ для классов $K = K_0, K_1, K_E, K_M, K_S, K_c$, где
K_0 — класс всех функций с коэффициентом $a_0 = 0$ (класс всех нечетных функций $f(x_1, \ldots, x_n) = -f(-x_1, \ldots, -x_n)$);
K_E — классы всех функций с коэффициентом a_0, кратным E, $E \geqslant 2$;
K_1 — класс всех функций, зависящих не более чем от одной переменной;
K_M — класс всех функций, сохраняющих модуль, т. е. эквивалентность $x \sim y \Leftrightarrow |x| = |y|$ на $\mathbb{Z}$;
K_S — класс всех сюръекций (он не замкнут);
K_C — класс функций, сохраняющих фиксированную константу c из $\mathbb{Z}$.

Теорема 2. *Если $K_0 \subset F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов $U(0, p)$.*

Если $K_E \subset F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов $U(0,p)$. для которых $p|E$.

Если $K_1 \subset F$ или $K_M \subset F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов D и $C(p_1 \cdots p_r)$.

Если $K_S \subset F$, то система F полна в $L(\mathbb{Z})$.

Если $K_C \subset F$, то система F полна в $L(\mathbb{Z})$ тогда и только тогда, когда она не содержится ни в одном из классов $U(b,p)$ таких. что $c \equiv b \pmod{p}$.

Теорема 3. *Пусть конечная система F_1 функций из $L(\mathbb{Z})$ имеет размер N, K — один из классов K_0 или K_E, K' — один из классов K_1, K_M, K_C. Тогда распознавание полноты в $L(\mathbb{Z})$ системы $K \cup F_1$ имеет временную сложность $O(N)$ и емкостную сложность $O(1)$, а для временной и емкостной сложности распознавания полноты системы $K' \cup F_1$ справедливы оценки $O(N^2 \log^2 N)$ и $O(N \log N)$.*

Рассмотрим еще одну группу проблем, решаемых с применением некоторых фрагментов алгоритма A. Эти проблемы связаны с распознаванием свойств одной фиксированной функции из $L(\mathbb{Z})$, а именно принадлежности ее некоторым из предполных в $L(\mathbb{Z})$ классов. Поставим следующие вопросы.

1. Существует ли класс $C(p_1 \cdots p_r)$, содержащий функцию f?
2. Существует ли класс $S(p)$, содержащий функцию f?
3. Существует ли класс $U(b,p)$, содержащий функцию f?

Эти свойства не обязательно проверять, пользуясь определениями классов. Для ответа на вопросы 1 и 2 можно применить фрагменты алгоритма A, при этом остаются в силе оценки сложности, полученные в теореме 1, с учетом условия $m = 1$. Ответ же на вопрос 3 можно получить проще, если использовать следующее утверждение.

Теорема 4. *Функция вида (1) не содержится ни в одном из классов $U(b,p)$ тогда и только тогда, когда $a_0 = \pm 1$ и $a_1 + \cdots + a_n = 1$.*

Следствие 1. *Если t есть максимальное значение абсолютной величины коэффициентов функции f вида (1), то ответ на вопрос 3 можно получить с временной сложностью $O(n \log t)$ и емкостной сложностью $O(1)$.*

Работа выполнена при поддержке РФФИ, проект № 13-01-00684.

Литература

[1] *Мамонтов А. И., Мещанинов Д. Г.* Проблема полноты в функциональной системе линейных полиномов с целыми коэффициентами // Дискретная математика. — 2010. — Т. 22, № 4. — С. 64–82.

О распределении нетерминалов в деревьях вывода стохастической КС-грамматики вида «цепочки»

И. М. Мартынов

murbidodrus@gmail.com

ННГУ им. Н. И. Лобачевского, Нижний Новгород

В работе исследуются вероятностные свойства деревьев вывода стохастической КС-грамматики специального вида. Рассматривается случай, когда матрица первых моментов A грамматики разложима и имеет перронов корень равный 1. Целью работы ялвляется исследование распределения нетерминальных символов в деревьях вывода высоты t, при $t \to \infty$.

Стохастической КС–грамматикой называется система $G = \langle V_N, V_N, R, s\rangle$, где V_T и V_N — конечные алфавиты терминальных и нетерминальных символов соответственно, $s \in V_N$ — аксиома, $R = \cup_{i=1}^{k} R_i$, где k — мощность алфавита V_N и R_i — множество правил с одинкаовой левой частью вида

$$r_{ij} : A_i \xrightarrow{p_{ij}} \beta_{ij},\ j = 1, 2, \ldots, n,$$

где $A_i \in V_N$, $\beta_{ij} \in (V_T \cup V_N)^*$ и p_{ij} — вероятность применения правила r_{ij}, причём $0 < p_{ij} \leqslant 1$ и $\sum_{j=1}^{n_i} p_{ij} = 1$.

Применение правила грамматики к слову состоит в замене вхождения нетерминала из левой части правила на слово, стоящее в его правой части.

Каждому слову α КС-языка соответствует последовательность правил грамматики (вывод), с помощью которой α выводится из аксиомы s. Выводу слова соответствует дерево вывода, вероятность которого определяется как произведение вероятностей правил, образующих вывод.

По стохастической КС-грамматике строится матрица A первых моментов. Для неё элемент a_j^i определяется как $\sum_{l=1}^{n_i} p_{il} s_{il}^j$, где величина s_{il}^j равна числу нетерминальных символов A_j в правой части правила r_{il}. Перронов корень матрицы A обозначим через r.

Введём отношение на множестве нетерминалов. Будем обозначать $A_i \to A_j$, если в грамматике существует правило вида $A_i \xrightarrow{p_{il}} \alpha_1 A_j \alpha_2$, где $\alpha_1, \alpha_2 \in (V_T \cup V_N)^*$. Рефлексивное транзитивное замыкание отношения $\to$ обозначим $\to_*$.

Классом нетерминалов назовём максимальное по включению подмножество $K \in V_N$ такое, что $A_i \to_* A_j$ для любых $A_i, A_j \in K$. Отношение $\to_*$ на множестве нетерминалов порождает отношение на множестве их классов. Будем обозначать $K_1 \prec K_2$, если существуют $A_1 \in K_1$ и $A_2 \in K_2$, такие, что $A_1 \to A_2$. Рефлексивное транзитивное замыкание отношения $\prec$ обозначим через $\prec_*$.

Пусть $\mathcal{K} = \{K_1, K_2, \ldots, K_m\}$ — множество классов нетерминалов грамматики, $m \geqslant 2$. Будем говорить, что грамматика имеет вид «цепочки», если $K_i \prec K_j$ тогда и только тогда, когда $i + 1 = j$.

Матрица первых моментов грамматики вида «цепочки» имеет вид:

$$A = \begin{pmatrix} A_{11} & A_{12} & 0 & \cdots & 0 & 0 \\ 0 & A_{22} & A_{23} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & A_{n-1,n-1} & A_{n-1,n} \\ 0 & 0 & 0 & \cdots & 0 & A_{n,n} \end{pmatrix}$$

Один класс нетерминалов представлен в матрице множеством подряд идущих строк и соответствующим множеством столбцов с теми же номерами. Для класса K_i квадратная подматрица, образованная соответствующими строками и столбцами, обозначается через A_{ii}. Подматрица A_{ij} $(i \neq j)$ является нулевой, если $K_i \not\prec K_j$. Блоки, расположенные ниже главной диагонали, нулевые в силу упорядоченности классов.

Для каждого класса K_i матрица A_{ii} неразложима. Без ограничения общности будем считать, что она строго положительна и непериодична. Обозначим через r_i перронов корень матрицы A_{ii}. Для неразложимой матрицы перронов корень является вещественным и простым [1]. Очевидно, $r = \max\limits_i \{r_i\}$.

Пусть $J = \{i_1, i_2, \ldots, i_l\}$ — множество всех номеров i_j классов, для которых $r_{ij} = 1$. Классы K_l такие, что $l \in J$, будем называть критическими. Также обозначим через q_l число критических классов среди подцепочки $K_l, K_{l+1}, \ldots, K_m$. Тогда верна следующая теорема.

Теорема 1. *Математические ожидания $M_i(t)$ числа нетерминалов A_i в деревьях вывода высоты t, порождённых стохастической КС-грамматикой вида «цепочки», при $t \to \infty$ удовлетворяют условию:*

$$M_i(t) \sim d_i \cdot t^{\left(\frac{1}{2}\right)^{q_l^* - 1}},$$

где $q_l^* = q_l - 1$ при $l \in J$, и $q_l^* = q_l$ при $l \notin J$, $A_i \in K_l$, и d_i — некоторые константы.

Обозначим через $q_i(t)$ число нетерминалов A_i в случайном дереве высоты t, порождённом грамматикой. Рассмотрим произвольную пару нетерминалов A_i и A_j таких, что математические ожидания $M_i(t)$ и $M_j(t)$ имеют один порядок по t, при $t \to \infty$. Верна следующая теорема.

Теорема 2. *Для любых двух нетерминалов $A_i \in K_l$, $A_j \in K_s$ таких, что $q_l^* = q_s^*$, при $t \to \infty$ выполняется условие:*

$$D\left(\frac{q_i(t)}{q_j(t)} - \frac{d_i}{d_j}\right) \to 0,$$

где $q_i(t)$, $q_j(t)$ — количество нетерминалов A_i, A_j соответственно в деревьях высоты t, d_i и d_j — некоторые константы.

Другими словами, при $t \to \infty$ соотношение частот между двумя нетерминалами, имеющими одинаковую асимптотику математических ожиданий, становится всё ближе к фиксированному значению.

Литература

[1] *Гантмахер Ф. Р.* Теория матриц. 5-е изд. — М.: ФИЗМАТЛИТ, 2010. — 560 с.

Сложность реализации некоторых классов бент-функций в модели упорядоченных один раз читающих ветвящихся программ

А. А. Марченко

anton.marchenko@kpfu.ru

Казанский Федеральный Университет, Казань

Введение

Нелинейность булевой функции – это расстояние Хэмминга до всех аффинных булевых функций. Булевские бент-функции – максимально нелинейные булевы фукнции, имеющие многочисленные теоретические и практические приложения. Достаточно полный обзор результатов по бент-функциям представлен в работе [1].

Ветвящиеся программы являются известной моделью представления булевых функций. Сокращенные упорядоченные один раз читающие ветвящиеся программы (ROBDD) [2] являются компактным каноническим представлением булевых функций(при фиксированном порядке) и позволяют добиться эффективной реализации операций над ними. Количество внутренних вершин в OBDD функции f при порядке считывания переменных π обозначается $\pi OBDD(f)$. $OBDD(f) = \min\limits_{\pi}(\pi OBDD(f))$, где минимум берется по всем порядкам считывания переменных. Подробную информацию о ветвящихся программах можно найти в работе [3].

Сложность представления бент-фукнций в OBDD

В данной работе рассматриваются вопросы сложности представления в модели сокращенной детерминированной OBDD двух важных конструкций бент-функций: итеративной конструкции и конструкции Мэйорана-МакФарланда. Первая позволяет строить бент-функции на основе известных бент-функций от меньшего числа переменных, а вторая позволяет строить бент-функции напрямую.

Итеративная конструкция бент-функций [4] представляет собой конкатенацию[1] двух бент-функций, не имеющих общих переменных. Получающаяся в результате применения конструкции функция $f(\mathbf{u}, \mathbf{v}) = g(\mathbf{u}) \oplus h(\mathbf{v})$, $|\mathbf{u}| + |\mathbf{v}| = n$ является бент-функцией от n переменных.

Поскольку у функций g и h нет общих переменных, $f = \overline{g}h \vee g\overline{h}$. Ветвящаяся программа такой конкатенации будет иметь вид рис.1, но может оказаться не сокращенной.

[1] Сумму по модулю 2 соответствующих полиномов Жегалкина

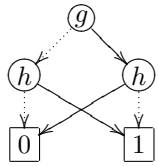


Рис. 1. OBDD конкатенации функций не имеющих общих переменных.

Сложность OBDD функции $f(\mathbf{u},\mathbf{v}) = g(\mathbf{u}) \oplus h(\mathbf{v})$ не превысит

$$\min\{OBDD(g) + OBDD(h,\overline{h}), OBDD(h) + OBDD(g,\overline{g})\} \qquad (1)$$

где $OBDD(\varphi,\overline{\varphi})$-срез OBDD, реализующий функцию φ и её отрицание $\overline{\varphi}$, имеющий две начальных и две конечных вершины. Причем, $OBDD(\varphi,\overline{\varphi}) \leqslant 2 \cdot OBDD(\varphi)$. А $OBDD(\varphi,\overline{\varphi}) = 2 \cdot OBDD(\varphi)$ в том случае, когда полином Жегалкина функции φ – конкатенация монома и константы.

Пусть класс функций, получающихся применением итеративной конструкции к парам бент-функций из некоторых классов $\mathcal{C}_1, \mathcal{C}_2$, обозначается $\mathbf{IT}(\mathcal{C}_1, \mathcal{C}_2)$.

Квадратичные бент-функции

Известной квадратичной бент-функцией является функция скалярного произведения векторов: $IP(\mathbf{x},\mathbf{y}) = \langle \mathbf{x},\mathbf{y}\rangle = \bigoplus_{i=1}^{n} x_i y_i,\ |\mathbf{x}| = |\mathbf{y}| = n$.

Все квадратичные бент-функции аффинно эквивалентны IP и образуют класс $\mathcal{B}_2$ [5].

В работе [6] получена следующая оценка для функции IP: $OBDD(IP) = 2n - 2$. Эту оценку можно получить, рассмотрев IP как результат последовательного применения итеративной конструкции.

Стоит отметить, что линейная сложность OBDD функции скалярного произведения достигается при "хорошем" порядке считывания переменных. При "плохом" порядке сложность может достигать $\mathcal{O}(2^{n/2})$.

Известно, что OBDD отрицания функции f получается из OBDD функции f переменой мест терминальных вершин 0 и 1. Также известно [3, 6], что для произвольной булевой функции выполняется:

Утверждение 1. *$OBDD(f(\mathbf{x})) = OBDD(f(\mathbf{a} * \mathbf{x}))$, где $\mathbf{a}$–константный двоичный вектор, а "$*$" – покомпонентное умножение векторов.*

Так как, после инверсии некоторых переменных, инвертируются выходящие ребра из вершин OBDD, помеченных этими переменными.

Несложно заметить, что OBDD квадратичных бент-функций обладают следующим свойством:

Утверждение 2. *$OBDD(f(\mathbf{x})) = OBDD(f(\mathbf{x}) \oplus \langle \mathbf{a},\mathbf{x}\rangle \oplus b)$, где $f \in \mathcal{B}_2$, $\mathbf{a}$–константный двоичный вектор, b–константа.*

Поскольку добавление аффинной функции к квадратичной сводится к случаю утверждения 1 и добавления константы, оно не влияет на размер OBDD.

Конструкция Мэйорана-МакФарланда [7] представляет собой конкатенацию $\langle \mathbf{u}, h(\mathbf{v})\rangle \oplus g(\mathbf{v})$, где h-перестановка вектора $\mathbf{v}$, g–произвольная функция от $n/2$ переменных. Результатом применения конструкции являются бент-функции от n переменных.

Пусть класс бент-функций от n переменных, получающихся в результате применения конструкции Мэйорана-МакФарланда, обозначается $\mathbf{MM_n}$.

Для конструкции получены следующие оценки сложности реализации в OBDD:

Теорема 1. *Если* $f \in \mathbf{MM_n}$, *то* $2n-2 \leqslant OBDD(f) \leqslant C \cdot 2^{(n/2)}$, *где* C-*некоторая константа.*

Доказательство.

1)Поскольку $OBDD(IP) = 2n-2$, а конкатенация $IP(\mathbf{u},\mathbf{v})$ с любой булевой функцией $g(\mathbf{v})$ не может уменьшить число подфункций, для всех бент-функций $f \in \mathbf{MM_n}$ $OBDD(f) \geqslant 2n-2$.

2) Известно [3], что для произвольной булевой операции $\otimes$ и функций g, h от n переменных выполняется $OBDD(g(\mathbf{x}) \otimes h(\mathbf{x})) \leqslant OBDD(g) \cdot OBDD(h)$. Так как $OBDD(\varphi) \leqslant C \cdot 2^{(n-\log n)}$, где $\varphi : \mathbb{Z}_2^n \mapsto \mathbb{Z}_2$, а C-константа и $OBDD(IP(\mathbf{u},\mathbf{v})) = 2n-2$, то для $f \in \mathbf{MM_n}$ выполняется $OBDD(f(\mathbf{x},\mathbf{y})) \leqslant (2n-2) \cdot C_1 \cdot 2^{(n/2)-\log(n/2)} \leqslant C_2 \cdot 2^{(n/2)}$, где $|\mathbf{x}| + |\mathbf{y}| = n$, а C_1, C_2-некоторые константы, причем $C_2 < 12$. ∎

Теорема 2. *Если* $g \in \mathbf{MM_r}$, $h \in \mathbf{MM_k}$, $r+k=n$, $f = g \oplus h$, *то* $OBDD(f) \geqslant 2n-2$.

Нижняя оценка достигается при конкатенации функций с минимально возможными размерами OBDD из класса Мэйорана-МакФарланда, то есть при конкатенации функций скалярного произведения, возможно, с добавленными к ним аффинными функциями.

Верхняя оценка $\mathbf{IT}(\mathbf{MM_k}, \mathbf{MM_r})$ будет соответствовать выражению 1.

Литература

[1] *Токарева Н. Н.* Нелинейные булевы функции: бент-функции и их обобщения. — М.: Издательство LAP LAMBERT Academic Publishing (Saarbrucken, Germany), 2011. ISBN: 978-3-8433-0904-2. — 180 с.

[2] *Bryant R. E.* Graph-based algorithms for boolean function manipulation // Computers, IEEE Transactions on. — 1986. — V. 100, №. 8. — p. 677-691.

[3] *Ingo Wegener.* BDDs–design, analysis, complexity, and applications // Discrete Applied Mathematics. — 2004. — V. 138, № 1-2. — p. 229-251.

[4] *Rothaus O.* On bent functions // J. Combin. Theory. Ser. A. — 1976. — V. 20, № 3. — p. 300–305.

[5] *Chase P. J., Dillon J. F., Lerche K. D.* Bent functions and difference sets // NSA R41 Technical Paper. September, — 1970.

[6] *Schafer N. B.* Characteristics of Binary Decision Diagrams of Boolean Bent Functions // master thesis. — 2009. — Naval Postgraduate School, Monterey California, US.

[7] *McFarland R. L.* A family of difference sets in non-cyclic groups // J. Combin. Theory. Ser. A. — 1973. — V. 15, № 1. — p. 1–10.

Об автоматическом построении Ватерлоо-подобных конечных автоматов

Б. Ф. Мельников

bormel@rambler.ru

СамГУ, Тольятти

В статье показано, как на основе известного автомата Ватерлоо (для которого т.н. сокращённый универсальный автомат неэквивалентен исходному), а также т.н. автомата $L^{\#}$ построить подобную конструкцию автоматически. Более того, подобные построения возможны для любого варианта неэквивалентности сокращённого универсального автомата исходному. Это даёт возможность описания переборного алгоритма получения таких конструкций. Кроме того, удаётся описать переборный алгоритм проверки необходимого условия такой конструкции для произвольного недетерминированного автомата.

Ватерлоо-подобные автоматы

В задачах минимизации недетерминированных конечных автоматов может возникать ситуация, когда покрывающее множество блоков (см. [1]) определяет автомат, неэквивалентный исходному. Впервые подобный пример был получен в 1970 г. Камедой и Вайнером, и, согласно опубликованной ими статье [2], получил название Ватерлоо. В нашей терминологии подробное описание этого факта приведено в [3]; подобную конструкцию мы называем Waterloo-like badness (*walibad*), а любое покрывающее подмножество множества блоков, включающее *не все* блоки, называем прото-walibad. Наличие walibad'ов сильно усложняет описание практических алгоритмов вершинной минимизации автоматов – поэтому возникают задачи поиска и описания таких конструкций, причём, по возможности, *до* применения самих алгоритмов минимизации.

В данной статье впервые показано, как *построить конструкцию walibad автоматически* – на основе:

— некоторого конкретного заранее известного примера walibad'а, и, следовательно, соответствующей ему таблице бинарного отношения #, см. [1];
— а также автомата $L^{\#}$, определённого на основе такой таблицы в [4].

Это даёт возможность получения на основе только одной таблицы отношения #, обладающей свойством прото-walibad:

— во-первых, переборного алгоритма получения конструкции walibad;
— во-вторых, переборного алгоритма проверки необходимого условия такой конструкции для *произвольного* недетерминированного автомата;
— и, в-третьих, классификации всех возможных таблиц отношения # (обладающих этим свойством) по условию существования конструкции walibad, соответствующей этой таблице.

Краткое описание примера

Описание примера базируется на основе материала, приведённого в [3, 4].

	X	Y	Z	U	V	W	P	Q	R	S
A		#						#		
B		#			#					
C				#	#					
D						#	#			
E	#					#	#			
F	#		#				#		#	
G			#						#	#
H			#							#

Мы повторили таблицу отношения #, соответствующего автомату Ватерлоо. В следующей таблице приведём только необходимые для дальнейшего построения буквы автомата $L^{\#}$.

	a	a	a	a	b	a	a	a	a	b	a	a	a	b	a
	E Y	F Y	A Z	B Z	B U	G U	F V	G V	C W	H W	B P	C P	E Q	D R	A S
$\to A$	***E***	F											***E***		
B	E	***F***					***F***	G							
C					***B***	***G***	F	***G***							
D									***C***	***H***	B	***C***			
$\leftarrow E$									***C***	***H***	B	***C***			
$\leftarrow F$			A	***B***							***B***	C		***D***	
G			***A***	B										***D***	***A***
H			***A***	B											***A***

Далее производим *один из возможных* вариантов объединения некоторых столбцов (причём с «поглощением» «простых» состояний «жирными»).

	a	b	c	d	e	f	g	h	i	j	k
$\to A$	***E***	F								***E***	
B	E	***F***					***F***	G			
C					***B***	***G***	F	***G***			
D				***C***					***H***		
$\leftarrow E$				***C***					***H***		
$\leftarrow F$			A	***B***							***D***
G			***A***	***A***							***D***
H			***A***	***A***							

(Мы переобозначили 11 букв получившегося алфавита символами a, b, …, k.) Итак, теперь мы считаем последний автомат *заданным*.

COM(L)	a	b	c	d	e	f	g	h	i	j	k
$\to$ (1) $A \times YQ$	6 7 8 14	8 9 10 13 14								6 7 8 14	
$\to$ (2) $AB \times Y$	6 7 8 14	8 9 10 13 14									
(3) $B \times YV$	6 7 8 14	8 9 10 13 14					8 9 10 13 14	10 13			
(4) $BC \times V$							8 9 10 13 14	10 13			
(5) $C \times UV$					2 3 4	10 11 12 13	8 9 10 13 14	10 11 12 13			
(6) $DE \times WP$				4 5					12 13		
$\leftarrow$ (7) $E \times XWP$				4 5					12 13		
$\leftarrow$ (8) $EF \times XP$				4							
$\leftarrow$ (9) $F \times XZPR$			1 2	2 3 4							
(10) $FG \times ZR$			1 2	2							6 14
(11) $G \times ZRS$			1 2	1 2							6
(12) $GH \times ZS$			1 2	1 2							
(13) $FGH \times Z$			1 2	2							
(14) $DEF \times P$				4							

Описание процесса его детерминизации мы опускаем; при этом его получаемая таблица отношения # (при обозначении получающихся состояний канонического автомата для инверсного языка L^R совершенно так же, как было сделано в [3]) совпадает с исходной, т.е. приведённой выше.

После дальнейших преобразований (также аналогичных [3]) мы получаем следующий универсальный автомат, приведённый в предыдущей таблице. На основе универсального автомата строим следующий сокращённый – выбирая блоки универсального автомата {1, 3, 5, 6, 8, 10, 12} также аналогично [3]:

	a	b	c	d	e	f	g	h	i	j	k
$\rightarrow$ (1) $A \times YQ$	6 8	8 10								6 8	
(3) $B \times YV$	6 8	8 10					8 10	10			
(5) $C \times UV$					3	10 12	8 10	10 12			
(6) $DE \times WP$				5					12		
$\leftarrow$ (8) $EF \times XP$											
(10) $FG \times ZR$			1								6
(12) $GH \times ZS$			1	1							

Последний автомат *не*эквивалентен исходному: в нём отсутствует цикл, соответствующий циклу базисного автомата

$$B\#Y \xrightarrow{a} F\#P \xrightarrow{d} B\#V \xrightarrow{g} F\#Z \xrightarrow{c} B\#Y\,.$$

Отметим, что дуги этого «отсутствующего» цикла соответствуют дугам «отсутствующего» цикла в сокращённом автомате из [3] – *для получения этого результата и выполнялись все построения настоящей статьи.*

Краткая формулировка результатов

Строгие формулировки основываются на определениях соответствующих дуг, приведённых в [3, 4]; в связи с ограниченным объёмом публикации приведём лишь краткое описание результатов.

Теорема 1. *Для некоторой заданной таблицы бинарного отношения #, обладающего свойством прото-walibad, существует переборный алгоритм, описываемый на основе объединения дуг автомата $L^{\#}$, определяющий, существует ли walibad с заданной таблицей.*

Литература

[1] *Melnikov B.* Once more about the state-minimization of the nondeterministic finite automata // The Korean Journal of Computational and Applied Mathematics. — 2000. — V. 7, No. 3. — P. 655–662.

[2] *Kameda T., Weiner P.* On the state minimization of nondeterministic finite automata // IEEE Transactions on Computers. — 1970. — V. C-19. — P. 617–627.

[3] *Melnikov B., Tsyganov A.* The state minimization problem for nondeterministic finite automata: the parallel implementation of the truncated branch and bound method // Proceedings – International Symposium on Parallel Architectures, Algorithms and Programming, PAAP. — 2012. — P. 194–201.

[4] *Melnikov B., Melnikova A.* Some more on the basis finite automaton // Acta Informatica, Univ. Sapientiae. — 2013. — V. 5, No. 2. — P. 227–244.

Projective simulation agent in real-world tasks

A. Melnikov, A. Makmal, H.- J. Briegel

alexey.melnikov@uibk.ac.at

Institute for Quantum Optics and Quantum Information, Austrian Academy of Sciences, Innsbruck, Austria, Institute for Theoretical Physics, University of Innsbruck, Innsbruck, Austria

We study the model of projective simulation (PS), a novel approach to artificial intelligence (AI) based on the stochastic processing of episodic memory, which was first introduced in [1]. The model is applied to reinforcement learning problems, but at the same time, the notion of PS is more general and can be seen as a building block for complete agent architectures. Episodic memory is a central component of the PS agent and is composed of so called "clips", which are the units of episodic memory. Inputs are represented by percept-clips, and actions are represented by action-clips. Once a percept-clip is excited, the excitation hops between clips probabilistically until the excitation reaches an action-clip. Then, at the end of a random walk through the clip network, the final action clip couples out, and the PS agent makes a real action. At the beginning no action is preferred over the other, but as experience is built up the clip-network is dynamically changed according to rewards from the environment, such that the probability to take rewarded actions is increased.

The PS performance on a number of discrete toy-problems has been studied in a previous work [2] where it was compared with those of Q-learning [3] and learning classifier systems [4], two popular representatives of AI models. For these problems the PS agent was shown to perform very well, thereby suggesting the PS as a competitive AI model.

In this work we study the PS model in more complicated, real-world, scenarios. To that end we chose two canonical, well studied tasks, namely the "grid world" [5] and "mountain car" [6]. The grid world task is to find a path to a fixed goal in a maze of size 6 by 9, where in each trial the agent starts from a fixed position. After reaching the goal, the agent is rewarded. The performance in this game is measured by the number of steps the agent makes on average before it reaches the goal. We compared the performances of the PS model with those of Dyna-PI [5] after the same number of trials and found that the PS agent finds the goal with number of steps comparable to the reference model.

In the mountain car task we test how PS handles the situation of continuous variable inputs. The agent needs to go up the hill by pushing a car back and forth in order to gain sufficient potential energy. At each step, the agent perceives a position coordinate and velocity, and chooses among three actions, which correspond to forward thrust, no thrust, and reverse thrust. After arriving at the top of the hill the PS agent is rewarded once. We have shown that in this game the PS agent manages to go up the hill twice as fast as with the SARSA algorithm [7], after the

same number of trials. We thus conclude that the PS model performs well also in real world scenarios with large and even continuous input space.

Литература

[1] *Briegel H. J., De las Cuevas G.* Projective simulation for artificial intelligence // Sci. Rep. 2, 400 (2012)

[2] *Mautner J., Makmal A., Manzano D., Tiersch M., Briegel H.-J.* Projective simulation for classical learning agents: a comprehensive investigation // New Generation Computing, accepted (2014)

[3] *Russel S.J., Norvig P.* Artificial intelligence - A modern approach. Second edition // Prentice Hall, New Jersey (2003)

[4] *Wilson S.W.* Classifier Fitness Based on Accuracy // Evol. Comput. 3(2) (1995)

[5] *Sutton R.S.* Integrated Architectures for Learning, Planning, and Reacting Based on Approximating Dynamic Programming // Machine learning (1990)

[6] *Moore A.* Efficient Memory-Based Learning for Robot Control // PhD thesis, University of Cambridge (1990)

[7] *Singh S.P., Sutton R.S.* Reinforcement learning with replacing eligibility traces // Machine learning 22 (1996)

О функциях из P_3, порожденных $(1,2)$-самодвойственными двухслойными симметрическими функциями

А. В. Михайлович

avmikhailvoich@gmail.com

Национальный исследовательский университет «Высшая школа экономики», Москва

Известно [1], что все замкнутые классы булевых функций имеют конечный базис. В [2] приведены примеры множеств функций k-значной логики, $k \geqslant 3$, которые являются порождающими системами класса без базиса и класса со счетным базисом. Функции из этих множеств являются симметрическими, принимают значения из множества $\{0,1\}$ и принимают нулевое значение на наборах, содержащих хотя бы одну нулевую компоненту. В [3, 4, 5] рассмотрены некоторые семейства замкнутых классов, порождающие системы которых обладают аналогичными свойствами; для них приведены критерии базируемости и конечной порожденности. В данной работе рассматривается семейство замкнутых классов, порожденных $(1,2)$-самодвойственными двухслойными симметрическими функциями принимающими значения из множества $\{0,1,2\}$. Все необходимые определения можно найти в [3, 4, 5].

Функции f и g называются *конгруэнтными* (обозначение $f \cong g$), если одна из них получается из другой переименованием переменных без отождествления. Пусть $f(x_1,\ldots,x_n) \in P_3$, $n \in \mathbb{N}$. Будем обозначать через N_f^1 множество всех наборов из E_3^n, на которых функция f принимает значение 1, а через N_f^2 — множество всех наборов из E_3^n, на которых функция f принимает

значение 2. Множество $\mathcal{L}$ всех наборов из E_3^n, которые получаются друг из друга перестановкой компонент, называется *слоем*. Слой $\mathcal{L} \subseteq \{1,2\}^n$, содержащий e единиц и d двоек, $e+d=n$, будем обозначать через $\mathcal{L}(e,d)$. Функцию $f(x_1,\ldots,x_n)$ из P_3 будем называть *симметрической*, если для любого слоя $\mathcal{L} \subseteq E_3^n$ и любых двух наборов $\widetilde{\alpha}$, $\widetilde{\beta} \in \mathcal{L}$ выполняется равенство $f(\widetilde{\alpha}) = f(\widetilde{\beta})$. Функцию $f(x_1,\ldots,x_n)$, $n \in \mathbb{N}$, из P_3 будем называть $(1,2)$-*самодвойственной двухслойной симметрической функцией*, если для некоторой пары чисел e,d, таких, что $e+d=n$ и $e \neq d$, $N_f^1 = \mathcal{L}(e,d)$, $N_f^2 = \mathcal{L}(d,e)$. Множество всех таких функций, для которых $e \geqslant 1$, $d \geqslant 1$, обозначим через $QNS_{1,2}^1$. Пусть $f(x_1,\ldots,x_n)$, $g(x_1,\ldots,x_n) \in QNS_{1,2}^1$. Будем говорить, что функция g является $(1,2)$-*инверсией* функции f (обозначение $f = \mathrm{not}_{1,2}(g)$), если $N_f^1 = N_g^2$, $N_f^2 = N_g^1$. Пусть $F \subseteq QNS_{1,2}^1$. Будем говорить, что класс F является $(1,2)$-*самоинвенрсным* если для любой фукнции f из F существует функция g из F, такая, что $g = \mathrm{not}_{1,2}(f)$.

Пусть $f,g \in QNS_{1,2}^1$. Будем говорить, что функция f не превосходит функцию g относительно частичного порядка $\preceq$ (обозначение $f \preceq g$), если $f \in [\{g\}]$. Будем говорить, что функция f не превосходит функцию g относительно частичного порядка $\trianglelefteq$ (обозначение $f \trianglelefteq g$), если $f \in [\{g\} \cup \{\mathrm{not}_{1,2}(f)\}]$. Будем говорить, что функция f строго меньше g относительно частичного порядка $\triangleleft$ (обозначение $f \triangleleft g$), если $f \trianglelefteq g$ и $f \not\cong g$. Множество $H \subset QNS_{1,2}^1$ называется *цепью относительно порядка* $\preceq$ (соответственно, относительно порядка $\trianglelefteq$), если любые два элемента множества H сравнимы относительно частичного порядка $\preceq$ (соотв. $\trianglelefteq$). Цепь $H \subset QNS_{1,2}^1$ называется *максимальной цепью* множества G, если для любой цепи $H_1 \subset QNS_{1,2}^1$ такой, что $H \subseteq H_1$, $H \neq H_1$, цепь H_1 не является подмножеством множества G. Функция $f \in H$ называется *верхней гранью* цепи H, если для любой функции $g \in H$ выполняется неравенство $g \preceq f$. Цепь называется *ограниченной*, если она имеет верхнюю грань. Пусть $G \subseteq QNS_{1,2}^1$. Обозначим через $U(G)$ множество всех верхних граней множества G.

Лемма 1. *Пусть $f(x_1,\ldots,x_n) \in QNS_{1,2}^1$, Φ — формула над $QNS_{1,2}^1$, реализующая функцию f, а Φ_1 — некоторая подформула формулы Φ, имеющая вид $g(\mathcal{B}_1,\ldots,\mathcal{B}_m)$, где $\mathcal{B}_1,\ldots,\mathcal{B}_m$ — формулы над $QNS_{1,2}^1$, $g \in QNS_{1,2}^1$. Тогда справедливы следующие утверждения.*

1. *Среди формул $\mathcal{B}_1,\ldots,\mathcal{B}_m$ символ каждой переменной из множества $\{x_1,\ldots,x_n\}$ встречается одинаковое число раз (в том числе ни разу).*
2. *Формула Φ_1 реализует либо функцию f, либо функцию $\mathrm{not}_{1,2}(f)$.*

Основным результатом является следующая теорема.

Теорема 2. *Пусть G — некоторое множество попарно неконгруэнтных функций из $QNS_{1,2}^1$, $F = [G]$. Тогда имеют место следующие утверждения.*

1. *Класс F имеет конечный базис тогда и только тогда, когда множество G конечно.*
2. *Класс F имеет счетный базис тогда и только тогда, когда каждая функция f из максимального $(1,2)$-самоинверсного подмножества множества $U(G)$ содержится в ограниченной максимальной цепи этого подмножества отно-*

сительно частичного порядка $\trianglelefteq$ и для каждой функции g из G выполняется по крайней мере одно из условий:

- *функция g содержится в ограниченной максимальной цепи множества G относительно $\preceq$;*
- *функция $\mathrm{not}_{1,2}(g)$ содержится в ограниченной максимальной цепи множества G и при этом существует функция h из G, такая, что $g \lhd h$ и функция h содержтся в ограниченной максимальной цепи множества G относительно частичного порядка $\preceq$.*

3. *Класс F не имеет базиса тогда и только тогда, когда либо существует функция f из максимального $(1,2)$-самоинверсного подмножества множества $U(G)$, которая не содержится ни в какой ограниченной максимальной цепи этого подмножества относительно порядка $\trianglelefteq$, либо существует функция g из G, не содержащаяся ни в какой ограниченной максимальной цепи множества G, для которой выполняется по крайней мере одно из условий:*
   - *функция $\mathrm{not}_{1,2}(g)$ не содержится ни в какой ограниченной максимальной цепи множества G относительно частичного порядка $\preceq$;*
   - *множество G не содержит функцию h, такую, что $g \lhd h$;*
   - *для любой функции $h \in G$, такой, что $g \lhd h$, функция h не содержится ни в какой ограниченной максимальной цепи множества G относительно частичного порядка $\preceq$.*

Достаточность утверждения 1 теоремы очевидна. Доказательство необходимости утверждения 1 теоремы опирается на первую часть леммы 1.

Доказательство достаточности утверждения 2 теоремы проводится аналогично доказательству достаточности утверждения 2 теоремы из [3]. Доказательство необходимости утверждения 2 теоремы опирается на лемму 1.

Доказательство утверждения 3 теоремы следует из утверждений 1 и 2.

Данное научное исследование выполнено при поддержке Программы «Научный фонд НИУ ВШЭ» (проект №14-01-0144) в 2014/2015гг.

Литература

[1] *Post E. L.* The two-valued iterative systems of mathematical logic. — Annals of Math. Studies. — Princeton Univ. Press, 1941. — 122 p.

[2] *Янов Ю. И., Мучник А. А.* О существовании k-значных замкнутых классов, не имеющих конечного базиса. // ДАН СССР. — 1959. — 127, № 1. — С. 44–46.

[3] *Михайлович А. В.* О замкнутых классах трехзначной логики, порожденных симметрическими функциями. // Вестн. Моск. ун-та. Матем. Механ. — 2008. — № 4. — С. 54–57.

[4] *Михайлович А. В.* О некоторых классах, порожденных однослойными симметрическими функциями многозначной логики // Материалы VII молодежной научной школы по дискретной математике и её приложениям (Москва, 18 – 23 мая 2009 г.). — Москва, 2009. — С. 21–26.

[5] *Михайлович А. В.* О базируемости замкнутых классов функций трехзначной логики, порожденнх симметрическими функциями с ограниченным числом слоев // Материалы IX молодежной научной школы по дискретной математике и её приложениям (Москва, 16 – 21 сентября 2013 г.). — Москва, 2013. — С. 80–85.

Кёниговы графы относительно 4-пути

Д. Б. Мокеев

MokeevDB@gmail.com

НИУ ВШЭ, Нижний Новгород

Характеризуется класс графов, у которых для каждого порождённого подграфа максимальное число непересекающихся порождённых путей с четырьмя вершинами равно минимальному числу вершин, покрывающих все такие пути.

Введение

Пусть $\mathcal{X}$ – множество графов. Множество попарно непересекающихся порожденных подграфов графа G, принадлежащих $\mathcal{X}$, называется *$\mathcal{X}$-упаковкой* графа G. Подмножество множества вершин графа G, покрывающее все порожденные подграфы из $\mathcal{X}$ называется его *$\mathcal{X}$-покрытием. Кёниговым графом относительно $\mathcal{X}$* называется граф, каждый порожденный подграф которого обладает свойством: наибольшая мощность $\mathcal{X}$-упаковки равна наименьшей мощности $\mathcal{X}$-покрытия. Класс всех кёниговых графов относительно множества $\mathcal{X}$ обозначаем через $\mathcal{K}(\mathcal{X})$. Если $\mathcal{X}$ состоит из единственного графа H, то будем говорить об H-упаковках и т.п.

Задаче об упаковке графа посвящено немало работ, особенно её алгоритмическим аспектам (см., например, [1, 2]). Известно, что задача поиска мощности максимальной H-упаковки NP-полна для любого графа H, имеющего компоненту связности с тремя или более вершинами. Будучи сформулированы как задачи ЦЛП, задачи о $\mathcal{X}$-упаковке и $\mathcal{X}$-покрытии образуют пару двойственных задач. Кёниговы графы, таким образом, суть графы, у которых для любого порождённого подграфа отсутствует разрыв двойственности, что способствует эффективному решению этих задач для таких графов.

Класс $\mathcal{K}(\mathcal{X})$ при любом $\mathcal{X}$ является наследственным и, следовательно, может быть описан множеством запрещенных графов (минимальных по отношению «быть порожденным подграфом» графов, не принадлежащих $\mathcal{X}$). Для P_2 такую характеризацию даёт теорема Кёнига вместе с известным критерием двудольности. Кроме этой классической теоремы автору известны следующие результаты такого рода для обыкновенных графов: в [3] эта задача решена для класса $\mathcal{K}(P_3)$; в [4] – для класса $\mathcal{K}(\mathcal{C})$, где $\mathcal{C}$ – множество всех простых циклов.

Цель настоящей работы охарактеризовать класс графов $\mathcal{K}(P_4)$. Применяется два подхода к описанию этого класса. Один из них – конструктивный: показано, как можно построить графы данного класса с помощью операций подразбиения рёбер и замены кографами вершин и висячих путей. При другом подходе ищется стандартное описание наследственного класса запрещёнными подграфами. Найденное множество запрещённых подграфов состоит из пяти бесконечных семейств и 64 отдельных графов.

Далее под кёниговым графом подразумеваем кёнигов граф относительно P_4. Рассматривая цикл C_n, предполагаем, что его вершины пронумерованы

вдоль цикла числами $0, 1, \ldots, n-1$. Каждый класс вычетов номеров вершин по модулю 4 называем 4-классом.

Свойства класса

Назовём класс графов *самодополнительным*, если вместе с каждым своим графом он содержит также дополнение этого графа.

Лемма 1. *Класс кёниговых графов является самодополнительным.*

Заметим, что свойство самодополнительности выполняется и для множества минимальных запрещённых подграфов. Если граф F является минимальным запрещённым для класса $\mathcal{K}(P_4)$, то $\bar{F}$ – тоже минимальный запрещённый граф для этого класса.

Будем называть связный граф G *приведённым*, если его дополнение связно и через каждую его вершину проходит хотя бы один порождённый 4-путь.

Лемма 2. *Чтобы получить ответ на вопрос, является ли граф кёниговым, достаточно получить этот ответ для каждого из его максимальных по включению приведённых подграфов.*

Расширенные подразбиения двудольных графов

Операция замены кографом вершины x состоит в следующем: эта вершина удаляется из графа, к нему добавляются несколько новых вершин. Каждая из них соединяется ребром с каждой вершиной, смежной x в исходном графе. Новые вершины соединены между собой так, что образуют кограф.

Назовём путь графа *висячим*, если степень одной из его вершин 1, а остальных – 2. Смежной вершиной висячего пути назовём вершину графа, смежную одной из вершин пути, но ему не принадлежащую.

Операция замены кографом висячего пути из 3 вершин, смежного вершине y состоит в том, что вершины этого пути удаляются из графа, затем к графу добавляется несколько новых вершин. Новые вершины соединены между собой так, что образуют кограф. Также новые вершины соединены с вершиной y так, чтобы максимальный путь, содержащий y и добавленные вершины имел длину 3.

Пусть H – двудольный граф. Каждое ребро этого графа, принадлежащее какому-нибудь циклу, подразобьём одной вершиной. Заменим произвольными кографами некоторые вершины, добавленные при подразбиении и вершины степени 1 или 2, не принадлежащие ни одному циклу. Заменим кографами также некоторые старые вершины степени 2, принадлежащие циклам, но при этом если в цикле есть вершина u, смежная с тремя и более вершинами степени больше 1, в этом цикле вершины 4-класса, содержащего u, не могут быть подвергнуты замене, а так же если в цикле есть вершина v степени 3 и более, не могут быть подвергнуты замене одновременно вершина 4-класса, содержащего v и вершина 4-класса, содержащая $v+2$. Последним шагом заменим некоторые висячие пути длины 3 произвольными кографами. Полученный таким образом граф будем называть *расширенным подразбиением* исходного двудольного графа.

Теорема 3. *Любой существенный граф, полученный расширенным подразбиением произвольного двудольного графа, отличного от простого цикла, является кёниговым.*

Запрещённые графы

Обозначим $\mathcal{A}$ множество циклов и их дополнений с числом вершин большим 5 и не кратным 4.

Обозначим $\mathcal{B}$ множество графов и дополнений графов, полученных из цикла длины кратной 4 добавлением двух вершин, не смежных между собой, каждая из которых соединяется ребром с одной вершиной цикла, причём расстояние между добавленными вершинами нечётно.

Обозначим $\mathcal{C}$ множество графов и дополнений графов, полученных из цикла длины кратной 4 добавлением висячего пути длины 2, смежного с вершиной с номером 0 и заменой кографом из двух вершин (K_2 или O_2) вершины цикла с номером $4k, k \in \mathbb{N}$, а так же полученных из цикла длины кратной 4 добавлением вершины, смежной с вершиной с номером 0 и заменой кографами из двух вершин (K_2 или O_2) вершин цикла с номерами $4k, k \in \mathbb{N}$ и $4l+2, l \in \mathbb{N} \cup \{0\}$.

Обозначим $\mathcal{D}$ множество графов и дополнений графов, полученных из цикла длины $k_1+k_2+k_3+k_4$ заменой кографами из двух вершин (K_2 или O_2) вершин с номерами $0, k_1, k_1+k_2, k_1+k_2+k_3$, причём $k_1 \equiv k_2 \equiv k_3 \equiv k_4 \equiv 1(mod 4), k_i \geqslant 5, i = 2,3,4$ или $k_1 \equiv 1(mod 4), k_1 \geqslant 5, k_2 \equiv k_4 \equiv 2(mod 4), k_3 \equiv 3(mod 4)$.

Обозначим $\mathcal{E}$ множество графов и дополнений графов, полученных из цикла длины кратной 4 добавлением вершины, смежной с вершинами, имеющими номера 0 и $4k+2, k \in \mathbb{N}$, и заменой кографами из двух вершин (K_2 или O_2) вершины цикла с номерами $4p, p \in \mathbb{N}, p \leqslant k$ и $4q+2, q \in \mathbb{N}, q > k$.

Теорема 4. *Графы множества $\mathcal{A} \cup \mathcal{B} \cup \mathcal{C} \cup \mathcal{D} \cup \mathcal{E}$ являются минимальными запрещёнными графами для класса $\mathcal{K}(P_4)$. Существует ровно 4 графа из 6 вершин и 60 графов из 7 вершин, не входящих в данное множество, которые так же являются минимальными запрещёнными графами для класса $\mathcal{K}(P_4)$.*

Работа выполнена при финансовой поддержке лаборатории ЛАТАС, НИУ ВШЭ, грант правительства ag. 11.G34.31.0057; гранта президента России МК-1148.2013.1; РФФИ, грант №14-01-00515-а.

Литература

[1] Hell P. Graph packing // Electronic Notes in Discrete Mathematics, 2000, V. 5, P. 170-173.

[2] Yuster R. Combinatorial and computational aspects of graph packing and graph decomposition // Computer Science Review, 2007, V. 1, P. 12-26.

[3] Алексеев В.Е., Мокеев Д.Б. Кёниговы графы относительно 3-пути // Дискретный анализ и исследование операций.– 2012.– 19(4).– С. 3-14.

[4] Ding G., Xu Z., Zang W. Packing cycles in graphs, II // J. Comb. Theory. B,. 2003, V. 87, P. 244-253.

Сведение проблемы эквивалентности в перегородчатой модели программ к проблемам для порождающей модели

А. Э. Молчанов

gurux13@gmail.com

МГУ, Москва

В докладе исследуются алгебраические модели программ, исследовавшиеся в работах [1] и [2] как абстракция последовательных программ. Существуют классы моделей программ, позволяющие переносить выявленные свойства на реальные программы.

Модель программ строится над заданным базисом операторных символов и предикатов. *Алгебраическая модель программ* – это множество схем программ со введенным отношением эквивалентности. Под *схемой программы* здесь понимается абстрактный вычислитель, заданный графом переходов, в котором вершинам приписаны операторные символы, а переходам – все возможные значения предикатов. Две вершины графа выделены особо и названы входом и выходом.

Вычисление схемы программы задается на *функции разметки* – оценке предикатов на всех словах из алфавита операторных символов. При вычислении схемы переход выбирается исходя из текущего *операторного слова* или *цепочки* – последовательности уже пройденных операторных символов. Для выбора перехода вычисляется значение функции разметки на текущей цепочке, и осуществляется переход в вершину, связанную с текущей ребром, помеченным этими значениями предикатов.

Эквивалентность схем – это эквивалентность их результатов вычислений на любой функции разметки из заданного множества L допустимых функций разметки. Так как результатом вычисления схемы является операторное слово, полученное в момент достижения выхода или неопределенность, если выход не был достигнут, то эквивалентность вводится на множестве $Y^* \cup \{\omega\}$, где Y – множество операторных символов, а ω – обозначение неопределенности. На отношение эквивалентности накладывается ограничение: $\omega \sim h \Rightarrow h = \omega$. То есть, если одна из схем не остановилась, то другая ей эквивалентная тоже должна зациклиться. Эквивалентность схем обозначим ν.

При задании параметров ν, L (базис считается фиксированным) алгебраическая модель программ полностью определена.

Так введенные схемы программ используют все традиционные композиции операторов последовательных программ, кроме процедур. Для описания рекурсивных программ применяются модели программ с процедурами. В этом случае базис дополняется множествами символов вызовов и возвратов, а схема содержит описание процедур (в качестве подграфов). Один подграф выделен как главный, остальные называются процедурными. В процедурных подграфах вместо входа выделена инициальная вершина, вместо выхода – финальная. В главном подграфе остаются вход и выход схемы. Кроме того, вершинам могут быть приписаны вместо операторных символов символы

вызова и возврата. В таком случае вершины называются вызовами и возвратами, соответственно. Естественно, что вызовы и возвраты образуют пары, и дуги из вызовов ведут в инициальные вершины, а в возвраты ведут дуги из финальных вершин.

Соответствующим образом меняется определение функции разметки – теперь она определяет значение предикатов на цепочках символов из операторных, вызовов и возвратов. Выполнение тоже изменяется: используется магазин, в который загружаются номера проходимых вызовов и извлекаются номера для правильного выхода из финальных вершин – это нужно для того, чтобы после вызова процедуры вернуться в парный возврат.

Отношение эквивалентности ν задается теперь на словах из $(Y \cup C \cup R)^* \cup \{\omega\}$, где C, R – множества символов вызовов и возвратов, соответственно. Эквивалентность схем вводится так же, как и для схем без процедур.

Схемы и модели программ без процедур будем называть *простыми*.

Для моделей программ, простых и с процедурами, фундаментальной является *проблема эквивалентности* – определить по паре схем, являются ли они эквивалентными в модели или нет. В общем случае (без ограничения на параметры модели) эта проблема неразрешима. Однако, показано, что при некоторых ограничениях эта проблема разрешима (см., например, [3], [4]) и, при еще более строгих ограничениях, может сводиться к проблеме эквивалентности конечных автоматов ([5]).

Проблема эквивалентности достаточно хорошо изучена для простых моделей программ, поэтому интересна задача переноса результатов с простых моделей программ на модели программ с процедурами. Один из способов это сделать – введение так называемых *перегородчатых моделей программ*, строящихся по заданной простой модели программ, называемой *индуцирующей*.

Эквивалентность ν перегородчатой модели определяется по эквивалентности τ порождающей модели, а множество допустимых функций разметки L – по множеству допустимых функций разметки порождающей модели l. Определение параметров таково, что схемы модели на всех участках, на которых они могут считаться простыми схемами, ведут себя как схемы из индуцирующей модели.

Формальное определение перегородчатой модели программ дано в [6].

Традиционно исследовался случай однопараметрических моделей, в которых множество функций разметки l строится по отношению эквивалентности. В данном докладе, однако, рассматриваются двухпараметрические модели, в которых τ и l могут выбираться независимо. Этот случай является обобщением однопараметрических моделей программ.

Перед тем как решать проблему эквивалентности, нужно обеспечить *процедурную свободу* схем. То есть, удалить из схемы все невызываемые процедурные подграфы и бесполезные вызовы и возвраты. Для этого доказана следующая теорема:

Теорема 1. *Если в индуцирующей простой модели программ разрешима проблема эквивалентности, существует алгоритм, который для схемы из перегородчатой модели программ строит эквивалентную ей процедурно свободную схему.*

Доказательство заключается в сведении поставленной задачи к задаче освобождения контекстно-свободной грамматики от бесполезных символов. Последняя решена, в частности, в [7]. Каждый нетерминал получаемой КС грамматики соответствует какому-либо вызову, возврату или процедуре исходной схемы, и тогда бесполезные символы отражают нефункционирующие элементы схемы. Терминалы же соответствуют участкам схемы, на которых она функционирует как простая.

После приведения схем к процедурно свободному виду возможно доказательство следующей теоремы:

Теорема 2. *В перегородчатой модели программ эквивалентность разрешима, если в индуцирующей простой модели разрешимы проблемы эквивалентности и непустоты пересечения.*

Проблема *непустоты пересечения*, упомянутая в теореме – вопрос о существовании какой-либо допустимой функции разметки, на которой обе схемы останавливаются.

Доказательство теоремы состоит в построении такого алгоритма. При работе алгоритма исследуются пары вызовов схем, и на них вводится отношение, показывающее, что эти вызовы должны быть связаны с эквивалентными подграфами. Все такие пары подграфов двух схем проверяются на эквивалентность. При обнаружении неэквивалентной пары подграфов схемы являются неэквивалентными. Если все пары вызовов, находящихся в указанном отношении, связаны с эквивалентными подграфами, то схемы эквивалентны.

Оба указанные алгоритма, рассматриваемые как алгоритмы с оракулами, являются полиномиальными.

В заключение отметим, что при дополнительном ограничении на схемы можно построить алгоритм, которому не требуется разрешимость проблемы непустоты пересечения в индуцирующей модели.

Литература

[1] *Ляпунов А. А.* О логических схемах программ // Проблемы кибернетики. — М: Физматгиз, 1958. — Вып. 1, с. 46–74.

[2] *Янов Ю. И.* О логических схемах алгоритмов // Проблемы кибернетики. — М: Физматгиз, 1958. — Вып. 1, с. 75–127.

[3] *Подловченко Р. И.* Об одной методике распознавания эквивалентности в алгебраических моделях программ // Программирование. — 2011. — № 6. — С. 33–43.

[4] *Подловченко Р. И.* Специальные перегородчато-автоматные модели рекурсивных программ // Программирование. — 1994. — № 3. — С. 3–26.

[5] *Подловченко Р. И.* От схем Янова к теории моделей программ // Математические вопросы кибернетики. — 1998. — Вып. 7, — с. 281–302.

[6] *Подловченко Р. И., Молчанов А. Э.* О теории алгебраических моделей программ с процедурами // Моделирование и анализ информационных систем. — 2012. — Т. 19, № 5. — С. 100–114.

[7] *Hopcroft J. E., Motwani R, Ullman J. D.* Introduction to Automata Theory, Languages, and Computation. — Melbourne: Addison-Wesley Publishing Company, 2006. — 750 pp.

О тестах относительно монотонных симметрических слипаний переменных в булевых функциях

Е. В. Морозов

morozov_msu@mail.ru

Факультет Вычислительной Математики и Кибернетики Московского Государственного Университета им. М. В. Ломоносова, Москва

Будем говорить, что в булевой функции $f(x_1, ..., x_n)$ произошло Ф-*слипание* переменных $x_{i_1}, ..., x_{i_k}$, если вместо исходной функции реализуется булева функция, полученная из нее подстановкой вместо каждой из переменных $x_{i_1}, ..., x_{i_k}$ функции $\varphi(x_{i_1}, ..., x_{i_k})$ от $x_{i_1}, ..., x_{i_k}$, где функция $\varphi \in \Phi$, φ будем также называть *функцией слипания*. Пусть q, $p \in \mathbb{N}$, $q \leqslant p \leqslant n$, $i_1, i_2, ..., i_{j_1}, i_{j_1+1}, ..., i_{j_2}, ..., i_{j_{q-1}}, i_{j_{q-1}+1}, ..., i_{j_q}$ — попарно различные натуральные числа из отрезка $[1, n]$. Будем говорить, что в булевой функции $f(x_1, ..., x_n)$ произошло множественное Ф-слипание переменных $x_{i_1}, ..., x_{i_{j_1}}, ..., x_{i_{j_{q-1}+1}}, ..., x_{i_{j_q}}$, если вместо исходной функции реализуется булева функция, полученная из нее подстановкой вместо каждой из переменных $x_{i_1}, x_{i_2}, ..., x_{i_{j_1}}$ функции $\varphi_1(x_{i_1}, x_{i_2}, ..., x_{i_{j_1}}) \in \Phi$, вместо каждой из переменных $x_{i_{j_1+1}}, ..., x_{i_{j_2}}$ функции $\varphi_2(x_{i_{j_1+1}}, ..., x_{i_{j_2}}) \in \Phi$ и так далее, вместо каждой из переменных $x_{i_{j_{q-1}+1}}, ..., x_{i_{j_q}}$ функции $\varphi_q(x_{i_{j_{q-1}+1}}, ..., x_{i_{j_q}}) \in \Phi$, функции $\varphi_1, ..., \varphi_q$ также будем называть функциями слипания. Через $\Psi = \Psi_{n,f,\varphi}$ обозначим множество функций, в которое входит $f(x_1, ..., x_n)$ и всевозможные булевы функции, получающиеся из $f(x_1, ..., x_n)$ в результате множественных Ф-слипаний переменных при любых допустимых значениях чисел $p, q, i_1, i_2, ..., i_{j_1}, i_{j_1+1}, ..., i_{j_2}, ..., i_{j_{q-1}}, i_{j_{q-1}+1}, ..., i_{j_q}$. Множество наборов T назовем *проверяющим (диагностическим) тестом для функции* $f(x_1, ..., x_n)$, если для функции f и любой $g \in \Psi$, отличной от f, (соответственно, для любой пары неравных функций из $\Psi \cup f$) в T найдется набор, на котором эти функции принимают разные значения. Традиционным образом введем *функцию Шеннона длины проверяющего (диагностического) теста относительно множественных* Ф-*слипаний переменных* $L^{detect}(n)$ (соответственно $L^{diagn}(n)$), как максимум по всем булевым функциям длины минимального проверяющего (диагностического) теста относительно множественных Ф-слипаний. Множество наборов, у которых ровно p единиц, называется p-м слоем булева куба. Булева функция $\varphi(x_{i_1}, ..., x_{i_k})$ называется *симметрической*, если на всяких двух наборах, принадлежащих одному слою булева куба, значения функции совпадают. Все неопределенные понятия можно найти в [1]. В данной работе рассматривается случай, когда Ф — всевозможные монотонные симметрические функции, а слипания, соответственно, называются монотонными симметрическими. Получены нетривиальные верхняя и нижняя оценка функции Шеннона длины проверяющего теста и точное значение функции Шеннона длины диагностического теста относительно монотонных симметрических слипаний.

Теорема 1. *При $n \longrightarrow \infty$ имеет место неравенство:*

$$2n \leqslant L^{detect}(n) \leqslant \frac{n^2}{2} + O(n \log n).$$

Доказательство.

Пусть имеется некоторая булева функция $f(x_1, ..., x_n)$.

Сначала заметим, что если в каком-либо константном слипании участвует существенная переменная x_i, то неисправность обнаруживается на одном из двух наборов, соседних по x_i, на которых функция принимает разные значения. Если в константных слипаниях участвуют только фиктивные переменные, это не влияет на реализуемую функцию. Обозначим через T_{exist} множество наборов, содержащее для каждой существенной переменной соответствующую пару наборов и далее будем считать, что константных слипаний нет.

Проверяющей парой для переменных x_i, x_j, $i \neq j$, назовем пару наборов $\tilde{\alpha} = (\alpha_1, ..., \alpha_n), \tilde{\beta} = (\beta_1, ..., \beta_n)$ таких, что $f(\tilde{\alpha}) \neq f(\tilde{\beta})$, $a_i \neq a_j$ и при $k \in \{i, j\}$ $\alpha_k \neq \beta_k$, а при всех остальных k имеет место равенство $\alpha_k = \beta_k$.

Введем на множестве булевых переменных x_1,...,x_n бинарное отношение Q_f. Всегда верно, что $x_i Q_f x_i$. Если $i \neq j$, то $x_i Q_f x_j$ тогда и только тогда, когда для переменных x_i, x_j функции $f(\tilde{x}^n)$ не существует проверяющей пары. Можно показать, что Q_f — отношение эквивалентности, следовательно, множество переменных x_1,...,x_n разбивается на классы эквивалентности. Каждый класс эквивалентности назовем *множеством симметричности*.

Без ограничения общности будем считать, что переменные $x_1, ..., x_{m_1}$ образуют первое множество симметричности, $x_{m_1+1}, ..., x_{m_2}$ — второе множество симметричности, ..., $x_{m_{r-1}+1}, ..., x_{m_r}$ — r-е множество симметричности, а, начиная с x_{m_r+1}, каждая переменная образует отдельное множество симметричности.

Можно показать, что для множества T_{detect}, содержащего проверяющие пары для всех пар переменных из разных множество симметричности, имеет место неравенство $|T_{detect}| \leqslant C_n^2 - \sum\limits_{i=0}^{r} C_{m_{i+1}-m_i}^2 + O(n \log n)$, где $m_0 = 0$. Множество T_{detect} обнаруживает всевозможные неисправности, при которых в некотором слипании участвуют переменные из разных множеств симметричности.

Построим множество T_1, которое обнаруживает всевозможные слипания среди переменных первого множества симметричности $x_1, ..., x_{m_1}$.

Рассмотрим случаи.

Случай 0. Переменные $x_1, ..., x_{m_1}$ фиктивны. Их слипания не меняют исходную функцию, переходим к следующему множеству симметричности.

Случай 1. Существуют такие числа $a_{m_1+1}, ..., a_n$, что функция $h(x_1, ..., x_{m_1}) = f(x_1, ..., x_{m_1}, a_{m_1+1}, ..., a_n)$ не является ни монотонной, ни антимонотонной.

Подслучай 1а. Функция $h(x_1, ..., x_{m_1})$ на k-м слое куба равна a, на всех остальных слоях — $\bar{a}$. Заметим, что k не может равняться 0 или n.

Проверяющей тройкой для переменных x_i, x_j назовем наборы $\tilde{\alpha}_1, \tilde{\alpha}_2, \tilde{\alpha}_3$ такие, что: 1) $h(\tilde{\alpha}_1) = h(\tilde{\alpha}_3) \neq h(\tilde{\alpha}_2)$; 2) в позициях i, j набора $\tilde{\alpha}_1$ стоят нули;

3) набор $\tilde{\alpha}_2$ является соседним с $\tilde{\alpha}_2$ по одной из компонент i или j, а набор $\tilde{\alpha}_3$ соседний с $\tilde{\alpha}_2$ по другой компоненте.

Можно показать, что существует множество наборов, содержащее все проверяющие тройки для функции $h(x_1, ..., x_{m_1})$, мощности не более, чем $\frac{m_1^2}{2} + O(m_1)$. Приписав к каждому набору числа $a_{m_1+1}, ..., a_n$, получим множество T_1 той же мощности, содержащее проверяющие тройки для переменных $x_1, ..., x_{m_1}$ функции $f(x_1, ..., x_n)$. Несложно показать, что множество T_1 проверяет всевозможные слипания среди переменных первого множества симметричности.

Подслучай 1b. Существуют числа $k_0, k_1, k_0 < k_1$ такие, что функция $h(y_1, ..., y_{m_1})$ принимает значение a на слоях $0, 1, ..., k_0 - 1$, значение $\bar{a}$ на слое k_0, значение b на слое k_1 и значение $\bar{b}$ на слоях $k_1 + 1, ..., n$.

Проверяющей четверкой для переменных x_i, x_j назовем наборы $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\alpha}_3$, $\tilde{\alpha}_4$ такие, что: 1) $h(\tilde{\alpha}_1) \neq h(\tilde{\alpha}_2)$, $h(\tilde{\alpha}_3) \neq f(\tilde{\alpha}_4)$; 2) у набора $\tilde{\alpha}_1$ стоят нули в разрядах i, j, а у $\tilde{\alpha}_3$ стоит нуль в одном из разрядов i, j (обозначим этот разряд через t) и единица в другом разряде; 3) наборы $\tilde{\alpha}_1$ и $\tilde{\alpha}_2$ являются соседними по компоненте t, наборы $\tilde{\alpha}_3$ и $\tilde{\alpha}_4$ также являются соседними по компоненте t; 4) $\tilde{\alpha}_1 \prec \tilde{\alpha}_2 \prec \tilde{\alpha}_4$, $\tilde{\alpha}_1 \prec \tilde{\alpha}_3 \prec \tilde{\alpha}_4$.

Аналогично предыдущему подслучаю множество T_1 содержит проверяющие четверки для всех переменных из первого множества симметричности, имеет мощности не более, чем $\frac{m_1^2}{2} + O(m_1)$ и проверяет всевозможные слипания переменных из первого множества симметричности.

Случай 2. При любых $b_{m_1+1}, ..., b_n$ функция $f(x_1, ..., x_{m_1}, b_{m_1+1}, ..., b_n)$ будет монотонной или антимонотонной симметрической функцией.

Можно показать, что в этом случае строится множество T_1 мощности не более, чем $\frac{m_1^2}{2} + O(m_1 \log m_1)$, которое проверяет всевозможные слипания переменных первого множества симметричности.

Остальные множества симметричности рассматриваются аналогично.

Объединив множества T_{exist},T_{detect},T_1,...,T_r, получаем проверяющий тест требуемой мощности.

Нижняя оценка достигается на функции $x_1...x_n \vee \bar{x}_1...\bar{x}_n$. Эта же функция была использована в [2] для оценки длины теста на конъюнктивно-дизъюнктивные слипания.

∎

Теорема 2.

$$L^{diagn}(n) = 2^n.$$

Доказательство. Верхняя оценка тривиальна, нижняя достигается на функции $x_1...x_n$. ∎

Работа выполнена при поддержке РФФИ, проект № 12-01-00964-а.

Литература

[1] *Редькин Н. П.* Надежность и диагностика схем.. — М.: Изд-во МГУ, 1992. — 191 с.

[2] *Икрамов А. А.* О сложности тестирования логических устройств на некоторые типы неисправностей // Интеллектуальные системы. — 2013. — Т. 13, № 1-4. — С. 311–313.

Оценка количества состояний последовательности, порожденной вероятностным автоматом

Р. Г. Мубаракзянов

Rustam.Mubarakzyanov@kpfu.ru

Казанский Федеральный университет, Казань

Автономные конечные вероятностные автоматы порождают случайные последовательности. Чтобы оценить количество состояний такой последовательности, требуется решать различные задачи. В данной работе приводятся оценки, связанные с исследованием пересечения конуса и подпространства.

Введение

Автономный конечный вероятностный автомат (КВА) с n состояниями определяется множеством линейных операторов в n-мерном пространстве R_n. При минимизации КВА это множество может меняться, но для исходного и получаемого множество будет существовать инвариантный полиэдральный конус. Исходный конус обозначим через R_n^+.

Определение 1. [1, 2] Выпуклый остроконечный конус K_r в R_n будем называть r-конусом, если он образован r векторами, т.е. для некоторых $\mu_1, \dots \mu_r$ $K_r = \{\sum\limits_{i=1}^{r} \alpha_i \mu_i | \alpha_i \geqslant 0, i = \overline{1,r}\}$ и при этом никакое множество из $(r-1)$ вектора не образует K_r. По определению конус отроконечен, если из $\{-\mu, \mu\} \subset K_r$ следует, что $\mu = 0$.

В данной работе приводятся оценки числа $\mathrm{КО}(r,n)$ – максимально возможного количества образующих конуса, являющегося пересечением некоторого r-конуса K_r, то есть остроконечного конуса, лежашего в пересечении r полупространств, и подпространства размерности n, а также числа $\mathrm{КОГ}(r,n)$ – максимально возможного количества образующих конуса, являющегося пересечением некоторого r-гранного конуса K_r^*(то есть остроконечного конуса, являющимся пересечением r полупространств) и подпространства размерности n. Исходный конус R_n^+ имеет как n образующих, так и n граней. При переходе к предельно минимальному автомату соответствующее линейное преобразование переводит R_n^+, в конус, имеющий по-прежнему n граней.

Пусть $\mathrm{КВМ}(r,n)$ – максимальное количество вершин выпуклого r-гранника, то есть ограниченного r $(n-1)$-мерными гранями, в n-мерном пространстве. В [3] приведена функция, равная максимальному числу k-мерных граней n-мерного многогранника с r вершинами (обозначим ее $\mathrm{КГМ}(r,n,k)$).

Оценка, связанная с количеством образующих

Теорема 1. *$KBM(r, n-1) \leqslant KO(r,n)$, $r \geqslant n$.*

Доказательство. Выпуклый r-гранник в $(n-1)$-мерном пространстве взаимно-однозначно соотвествует решению системы r линейных неравенств

S. Неотрицательный ортант R_r^+ образован r векторами. Линейная комбинация этих векторов с использованием коэффициентов системы неравенств S позволяет получить некоторое n-мерное пространство. Любой вектор μ этого пространства однозначно соответствует некоторому вектору $(x_1, \ldots, x_n)$. Причем μ принадлежит R_r^+ тогда и только тогда, когда $(x_1, \ldots, x_n)$ соответствует решению системы S. Следовательно, $\mathrm{KBM}(r, n-1) \leqslant \mathrm{KO}(r, n)$. ∎

Теорема опровергает результат работы [4], утверждающий, что всегда $\mathrm{KO}(r, n) \leqslant r$. Например, 4-мерное пространство с базисом:

$$(-1, 0, 0, 1, 0, 0), (0, -1, 0, 0, 1, 0), (0, 0, -1, 0, 0, 1), (1, 1, 1, 0, 0, 0),$$

пересекаясь с 6-конусом R_6^+, дает 8-конус с образующими

$$(1, 1, 1, 0, 0, 0), (1, 1, 0, 0, 0, 1), (1, 0, 1, 0, 1, 0), (1, 0, 0, 0, 1, 1),$$

$$(0, 1, 0, 1, 0, 1), (0, 1, 1, 1, 0, 0), (0, 0, 1, 1, 1, 0), (0, 0, 0, 1, 1, 1).$$

Оценка, связанная с количеством граней

Ошибка работы [4], основана на неверном утверждении, что r-конус всегда ограничен r гиперплоскостями. Следующие 5 гиперплоскостей:

$$x \geqslant 0, y \geqslant 0, z \geqslant 0, t \geqslant 0, x \leqslant y + z + t$$

образуют 6-конус с образующими

$$(0, 1, 0, 0), (0, 0, 1, 0), (0, 0, 0, 1), (1, 0, 1, 0), (1, 1, 0, 0), (1, 0, 0, 1).$$

Теорема 2. *$КОГ(r, n) = КВМ(r, n-1) =$*

$$= КГМ(r, n-1, n-2) = \begin{cases} 2\frac{r}{n-1}\binom{r-(n+1)/2}{(n-3)/2}, & \text{если } n \text{ нечетно,} \\ 2\binom{r-n/2}{n/2-1}, & \text{если } n \text{ четно.} \end{cases}$$

Доказательство. Основная идея доказательства состоит в том, что в пространстве R_n может быть проведена $(n-1)$-мерная гиперплоскость, пересекающая все образующие r-гранного конуса K_r^* и, следовательно, дающая в пересечении с K_r^* выпуклый многогранник, вершины которого однозначно соответствуют образующим некоторого полиэдрального конуса.

То, что $\mathrm{KOГ}(r, n) \geqslant \mathrm{KBM}(r, n-1)$, было показано в Теореме 1. Неотрицательный ортант R_r^+ является r-гранным конусом.

Покажем, что $\mathrm{KOГ}(r, n) \leqslant \mathrm{KBM}(r, n-1)$. Рассмотрим в R_h произвольный остроконечный r -гранный конус K_r^*.

Лемма 3. *Для остроконечного полиэдрального конуса K, заданного в пространстве размерности h, существует подпространство размерности $(h-1)$, пересекающееся с K только в начале координат.*

Доказательство. Полиэдральный остроконечный конус задается при помощи конечного числа неравенств вида $cx \geqslant 0$, где c есть h-вектор, определяющий грань конуса. Существует множество векторов $\{x_1, \ldots, x_{h-1}\}$ такое, что $c_i x_i > 0, 1 \leqslant i \leqslant (h-1)$; $c_i x_j = 0$ при $i \neq j$. Для образующей x_0 существует грань, ее не содержащая. Пусть эта грань соответствует c_0. Тогда $c_0 x_0 = a > 0$. Пусть $c_0 x_i = a_i \geqslant 0, 1 \leqslant i \leqslant (h-1)$. Рассмотрим любое число $a_0 > a_i, 1 \leqslant i \leqslant (h-1)$. Для вектора $y_0 = x_0 \frac{a_0}{a}$ верно $c_0 y_0 = a_0 > a_i, 1 \leqslant i \leqslant (h-1)$, $c_i y_0 = 0, 1 \leqslant i \leqslant (h-1)$. Рассмотрим пространство размерности $(h-1)$: $Q = \mathrm{Lin}\{x_i - y_0 | 1 \leqslant i \leqslant (h-1)\}$. ∎

Рассмотрим пространство Q с базисными векторами $\{x_1, \ldots, x_{h-1}\}$, пересекающееся с K_r^* только в начале координат. Пусть некоторое пространство S размерности n пересекается с конусом K_r^*. Очевидно, вектор $x \in K_r^* \cap S - \{0\}$ линейно независим с $\{x_1, \ldots, x_{h-1}\}$. Гиперплоскость $Q' = \{x + \sum_{1 \leqslant i \leqslant (h-1)} \alpha_i x_i | \alpha_i \in R_1\}$ пересекает все образующие конуса. Действительно, для любой образующей y конуса K_r^* существует разложение $y = \beta x + \sum_{1 \leqslant i \leqslant (h-1)} \alpha_i x_i, \beta \neq 0$. Если $\beta > 0$, то Q' пересекает y. Пусть $\beta < 0$, тогда вектор $y - \beta x$ принадлежит $Q \cap K_r^*$. Поэтому $y = \beta x$ и из отрицательности β и остроконечности K_r^* следует $y = x = 0$.

Пересечение $Q \cap K_r^* = M'$ есть выпуклый r-гранник размерности $(h-1)$. Грани этого многогранника соответствуют граням исходного конуса (являются пересечениями этих граней с Q'), а вершины – образующим конуса.

В S можно выбрать базис $\{x, s_1, s_2, \ldots, s_{n-1}\}$ так, что $s_i \in Q, 1 \leqslant i \leqslant (n-1)$. Очевидно, что $Q' \cap K_r^* \cap S = M' \cap S = M$ есть выпуклый многогранник не более, чем с r гранями. Множество вершин M взаимно однозначно соответствует множеству образующих конуса $K_r^* \cap S = K_{r,S}^*$. Поэтому $\mathrm{KO}(r, n) \leqslant \mathrm{KBM}(r, n-1)$.

Последнее равенство теоремы $\mathrm{KBM}(r, n-1) = K(r, n-1, n-2)$ означает, что максимальное количество вершин выпуклого r-гранника в $(n-1)$-мерном пространстве равно максимальному числу граней многогранника с r вершинами в том же пространстве (его значение, как уже упомяналось, получено Кли [3]). Это следует из двойственности многогранника: любому n-граннику в с r вершинами можно поставить в соответствие r-гранник в с n вершинами в том же пространстве [5]. ∎

Литература

[1] *Глазман И. М.* Конечномерный линейный анализ. — М.: Наука, 1969. — 476 с.

[2] *Рокафеллар Р.* Выпуклый анализ. — М.: Мир, 1973. — 470 с.

[3] *Grunbaum B., Klee V., Perles M. A., Shephard G. A.* Convex polytopes. — Sydney,London,N.Y.: A Division of John Wiley & Sons. 1967.— 456 p.

[4] *Bancilhon F.* A geometric model for stochastic automata // ISEE. — 1974. — V. 23, № 12. — P. 1290–1299.

[5] *Edelsbrunner H.* Algorithms in combinatorial Geometry. — Berlin, Heidelberg: Springer, 1987. — 424 p.

О свойствах пересечений предполных классов, сохраняющих разбиения, в пятизначной логике

А. С. Нагорный

anagorny@list.ru

МГУ им. М. В. Ломоносова, ф-т ВМК, Москва

Пусть $E_k = \{0, 1, \ldots, k-1\}$, P_k — множество всех конечноместных функций на E_k. Известно [1], что все классы функций из P_k, сохраняющих нетривиальные разбиения множества E_k, являются попарно различными, замкнутыми (относительно операции суперпозиции) и предполными в P_k при любом $k \geqslant 3$. Обозначим через U^5 семейство всех таких классов для $k = 5$.

В статье изучается вопрос о минимальном числе классов из U^5, пересечение которых может быть тривиальным (т.е. состоять только из константных и селекторных функций), а также вопрос о том, в каких случаях пересечение двух классов из U^5 вложено в какой-нибудь другой класс из U^5. На оба вопроса в работе получен исчерпывающий ответ.

Основные обозначения

Обозначим классы функций пятизначной логики, сохраняющие разбиения $\{\{0\},\{1,2,3,4\}\}$, $\{\{0,1\},\{2,3,4\}\}$, $\{\{0\},\{1\},\{2,3,4\}\}$, $\{\{0\},\{1,2\},\{3,4\}\}$ и $\{\{0\},\{1\},\{2\},\{3,4\}\}$ множества E_5, через $U_{0\{1234\}}$, $U_{\{01\}\{234\}}$, $U_{01\{234\}}$, $U_{0\{12\}\{34\}}$ и $U_{012\{34\}}$, соответственно. Аналогично обозначаются все остальные классы семейства U^5 (легко видеть, что все они получаются из перечисленных классов с помощью некоторой подстановки индексов на E_5, такие вложения будем называть *двойственными*). Семейство U^5 содержит ровно 50 классов.

Тривиальные пересечения

Селекторной функцией назовем функцию из P_k, равную одной из своих переменных. Класс функций из P_k назовем *тривиальным*, если он содержит только все константные и все селекторные функции из P_k. Легко доказать, что класс, равный пересечению всех классов из семейства U^5, является тривиальным. Однако тривиальным может быть и пересечение меньшего числа таких классов.

Теорема 1. *Ни одно пересечение двух классов из семейства U^5 не является тривиальным.*

Теорема 2. *Все тривиальные пересечения трех классов из семейства U^5 суть пересечения двух видов $U_{\{ab\}\{cde\}} \cap U_{c\{da\}\{eb\}} \cap U_{d\{ea\}\{bc\}}$ (60 троек) и $U_{a\{bc\}\{de\}} \cap U_{b\{cd\}\{ea\}} \cap U_{d\{eb\}\{ac\}}$ (20 троек).*

Тривиальные пересечения *четырех и более* классов из семейства U^5 можно найти в [2].

Вложения пересечений

Теперь рассмотрим вопрос о том, в каких случаях пересечение двух классов из U^5 вложено в какой-нибудь другой класс из U^5. Исчерпывающий ответ на этот вопрос дают следующие пять теорем.

Теорема 3. *Имеется ровно* 45 *вложений пересечений пар классов из* U^5 *в класс* $U_{0\{1234\}}$, *они перечислены в нижеследующей таблице:*

№	Вложение в класс $U_{0\{1234\}}$	Двойственных вложений
1	$U_{01\{234\}} \cap U_{02\{134\}} \subseteq U_{0\{1234\}}$	6
2	$U_{01\{234\}} \cap U_{0\{12\}\{34\}} \subseteq U_{0\{1234\}}$	12
3	$U_{01\{234\}} \cap U_{023\{14\}} \subseteq U_{0\{1234\}}$	12
4	$U_{0\{12\}\{34\}} \cap U_{0\{13\}\{24\}} \subseteq U_{0\{1234\}}$	3
5	$U_{0\{12\}\{34\}} \cap U_{013\{24\}} \subseteq U_{0\{1234\}}$	12

Теорема 4. *Имеется ровно* 13 *вложений пересечений пар классов из* U^5 *в класс* $U_{\{01\}\{234\}}$, *они перечислены в нижеследующей таблице:*

№	Вложение в класс $U_{\{01\}\{234\}}$	Двойственных вложений
1	$U_{01\{234\}} \cap U_{2\{34\}\{01\}} \subseteq U_{\{01\}\{234\}}$	3
2	$U_{01\{234\}} \cap U_{234\{01\}} \subseteq U_{\{01\}\{234\}}$	1
3	$U_{2\{34\}\{01\}} \cap U_{3\{42\}\{01\}} \subseteq U_{\{01\}\{234\}}$	3
4	$U_{2\{34\}\{01\}} \cap U_{013\{24\}} \subseteq U_{\{01\}\{234\}}$	6

Теорема 5. *Имеется ровно* 6 *вложений пересечений пар классов из* U^5 *в класс* $U_{01\{234\}}$, *они перечислены в нижеследующей таблице:*

№	Вложение в класс $U_{01\{234\}}$	Двойственных вложений
1	$U_{0\{1234\}} \cap U_{1\{2340\}} \subseteq U_{01\{234\}}$	1
2	$U_{0\{1234\}} \cap U_{\{01\}\{234\}} \subseteq U_{01\{234\}}$	2
3	$U_{012\{34\}} \cap U_{013\{24\}} \subseteq U_{01\{234\}}$	3

Теорема 6. *Имеется ровно* 4 *вложения пересечений пар классов из* U^5 *в класс* $U_{0\{12\}\{34\}}$, *они перечислены в нижеследующей таблице:*

№	Вложение в класс $U_{0\{12\}\{34\}}$	Двойственных вложений
1	$U_{0\{1234\}} \cap U_{\{12\}\{034\}} \subseteq U_{0\{12\}\{34\}}$	2
2	$U_{\{12\}\{034\}} \cap U_{\{34\}\{012\}} \subseteq U_{0\{12\}\{34\}}$	1
3	$U_{012\{34\}} \cap U_{034\{12\}} \subseteq U_{0\{12\}\{34\}}$	1

Теорема 7. *Имеется ровно* 42 *вложения пересечений пар классов из* U^5 *в класс* $U_{012\{34\}}$, *они перечислены в нижеследующей таблице:*

№	Вложение в класс $U_{012\{34\}}$	Двойственных вложений
1	$U_{0\{1234\}} \cap U_{12\{034\}} \subseteq U_{012\{34\}}$	3
2	$U_{0\{1234\}} \cap U_{1\{20\}\{34\}} \subseteq U_{012\{34\}}$	6
3	$U_{\{01\}\{234\}} \cap U_{\{02\}\{134\}} \subseteq U_{012\{34\}}$	3
4	$U_{\{01\}\{234\}} \cap U_{02\{134\}} \subseteq U_{012\{34\}}$	6
5	$U_{\{01\}\{234\}} \cap U_{0\{12\}\{34\}} \subseteq U_{012\{34\}}$	6
6	$U_{\{34\}\{012\}} \cap U_{01\{234\}} \subseteq U_{012\{34\}}$	3
7	$U_{01\{234\}} \cap U_{02\{134\}} \subseteq U_{012\{34\}}$	3
8	$U_{01\{234\}} \cap U_{0\{12\}\{34\}} \subseteq U_{012\{34\}}$	6
9	$U_{01\{234\}} \cap U_{2\{34\}\{01\}} \subseteq U_{012\{34\}}$	3
10	$U_{0\{12\}\{34\}} \cap U_{1\{20\}\{34\}} \subseteq U_{012\{34\}}$	3

Заметим, что тривиальные пересечения классов из семейства U^5 вложены в каждый класс из семейства U^5.

Ясно также, что имеет смысл искать только те пересечения и вложения, которые являются *неприводимыми*, т.е., из которых ни один класс нельзя удалить (все перечисленные здесь пересечения и вложения неприводимыми являются в силу нетривиальности и предполноты всех классов из U^5).

Неприводимые вложения нетривиальных пересечений *трех и более* классов семейства U^5 в некоторые классы из U^5 более многочисленны и поэтому в текст этой статьи не вошли. Они размещются автором в [2].

В заключение автор хотел бы выразить благодарность А. А. Вороненко за постановку задачи и С. С. Марченкову за постоянное внимание к работе.

Литература

[1] *Яблонский С. В.* Функциональные построения в k-значной логике // Тр. МИАН СССР им. В. А. Стеклова. — 1958. — Т. 51. — С. 5–142.

[2] *Nagorny A. S.* Intersections and embedding of intersections of U^5 classes to some U^5 classes // `https://googledrive.com/host/0B7d_hlk8RpGET1ZKNUtxLUJuZHM/`, 2014.

О сложности задачи решения линейных уравнений над конечными подстановками

Т. А. Новикова, В. А. Захаров

`taniaelf@mail.ru, zakh@cs.msu.su`

Казахстанский филиал Московского государственного университета имени М.В. Ломоносова, Астана

Различные отношения подобия программ находят применение в решении задач реорганизации (рефакторинга) программ (см. [1]) и эффективного выделения клонов [2]. Одно из возможных определений подобия программ можно сформулировать так. Предположим, что на множестве программ введено некоторое отношение эквивалентности и выделен некоторый класс простых программ П. Тогда пара программ π_1 и π_2 считается подобной, если существуют две такие пары программ π_1^{in}, π_2^{in} и π_1^{out}, π_2^{out} из класса П, для которых эквивалентны последовательные композиции $\pi_1^{in}; \pi_1; \pi_1^{out}$ и $\pi_2^{in}; \pi_2; \pi_2^{out}$. В качестве отношения эквивалентности разумно выбрать отношение функциональной эквивалентности программ или любую его аппроксимацию; в этом случае из эквивалентности программ будет следовать равенство вычисляемых этими программами функций. Пары программ π_1^{in}, π_2^{in} и π_1^{out}, π_2^{out} в формулировке задачи проверки подобия могут мыслиться как интерфейсы, преобразующие формат представления входных и выходных данных.

Обычные императивные программы строятся из операторов присваивания вида `x:=t`, где `x` — переменная, а `t` — терм. Семантика оператора присваивания определятся подстановкой $\{x/t\}$ терма t вместо переменной x. Простейшая последовательная программа π это последовательная композиция операторов присваивания `x1:=t1; x2:=t2;...xn:=tn;` ее семантика определяется

композицией подстановок $\theta_\pi = \{x1/t1\}\{x2/t2\}\cdots\{xn/tn\}$. Рассматривая простейшие программы как конечные подстановки, мы можем сформулировать задачу проверки подобия программ как задачу проверки разрешимости уравнений вида $X_1\theta_{\pi_1}Y_1 = X_2\theta_{\pi_2}Y_2$ над множеством конечных подстановок с операцией композиции, где θ_{π_1} и θ_{π_2} — заданные подстановки, соответствующие анализируемым программам π_1 и π_2, а X_1, X_2, Y_1, Y_2 — неизвестные подстановки, соответствующие интерфейсам преобразования входных и выходных данных. Поскольку каждая неизвестная подстановка имеет ровно одно вхождение в уравнение, такого вида уравнения называются линейными. Таким образом, для решения задачи проверки подобия программ представляют интерес алгоритмы решения линейных уравнений над конечными подстановками.

Приведем строгие определения используемых понятий из теории подстановок. Для конечных множеств переменных $\mathcal{X}, \mathcal{Y}$ и множества функциональных символов $\mathcal{F}$ обозначим записью $Term(\mathcal{F}, \mathcal{X})$ множество термов, а записью $Subst(\mathcal{X}, \mathcal{Y})$ — множество подстановок вида $\theta : \mathcal{X} \to Term(\mathcal{F}, \mathcal{Y})$. Операция композиции подстановок определяется следующим образом. Пусть $\theta \in Subst(\mathcal{X}, \mathcal{Y})$ и $\eta \in Subst(\mathcal{Y}, \mathcal{Z})$. Тогда применение $t\eta$ подстановки η к терму $t, t \in Term(\mathcal{F}, \mathcal{Y})$, состоит в одновременной замене для всех переменных из $\mathcal{Y}$ каждого вхождения переменной y на терм $\eta(y)$. Композицией $\theta\eta$ указанных подстановок называется такая подстановка $\mu, \mu \in Subst(\mathcal{X}, \mathcal{Z})$, которая для каждой переменной $x, x \in \mathcal{X}$, удовлетворяет равенству $\mu(x) = (\theta(x))\eta$. Размер подстановки θ — это суммарный размер всех термов из ее области значений.

К задачам решения линейных уравнений сводится, в частности, задача унификации термов: для двух заданных термов t_1, t_2 найти такие подстановки θ_1, θ_2, чтобы выполнялось равенство $t_1\theta_1 = t_2\theta_2$. Задача унификации термов сводится к решению уравнения $\{x/t_1\}Y_1 = \{x/t_2\}Y_2$, и ее решение может быть получено за почти линейное время (см. [3]). Насколько известно авторам настоящей статьи, сложность решения других видов линейных уравнений над подстановками ранее не исследовалась.

Нами была рассмотрена задача решения линейных уравнений над подстановками вида $X\theta_1 Y = \theta_2$ и установлена следующая

Теорема 1. *Задача о разрешимости уравнения $X\theta_1 Y = \theta_2$ над подстановками является NP-полной.*

Принадлежность рассматриваемой задачи классу сложности NP очевидна: достаточно найти такое разложение $\theta_2 = \theta_2'\theta_2''$ подстановки θ_2 для которого задача унификации $\theta_1 Y = \theta_2'' Y'$ имеет решение $Y' = \varepsilon$, где ε — пустая (тождественная) подстановка. Подстановка θ_2'' конструируется из подтермов тех термов, которые входят в область значений подстановки θ_2; число таких подтермов ограничено размером подстановки θ_2.

NP-трудность задачи проверки разрешимости уравнений вида $X\theta_1 Y = \theta_2$ следует из того, что к этой задаче сводится NP-полный вариант ограниченной проблемы домино (bounded tiling problem). Этот вариант проблемы мозаики состоит в следующем. Пусть задан конечный алфавит пометок (красок) $A = \{a_1, a_2, \ldots, a_k\}$. Домино называется квадрат размера 1×1, каждая сторона которого помечена одной из букв алфавита A. Предположим, что задано некоторое конечное множество типов домино $Tiles = \{T_1, T_2, \ldots, T_L\}$ и

прямоугольный участок плоскости размера $n \times m$, разбитый на единичные квадраты. Правильным заполнением участка называется такое размещение домино заданных типов в квадратах заданного участка, при котором смежные стороны любой пары домино имеют одинаковую окраску. В статье [4] было установлено, что описанный вариант проблемы домино является NP-полной задачей. Чтобы промоделировать прямоугольный вариант проблемы домино можно выбрать подстановки θ_1 и θ_2 таким образом, чтобы термы подстановки θ_1 соответствовали расположениям всех домино заданных типов во всех квадратах участка, а подстановка θ_2 выражала требование правильного заполнения прямоугольной области монохроматическими домино, все стороны которых окрашены в цвет a_k. Тогда решение уравнения $X\theta_1 Y = \theta_2$ состоит в размещении подходящих типов домино из множества $Tiles$ в квадратах участка (подстановка X) и перекрашивании сторон домино путем увеличения на одну и ту же величину номера цвета у каждой смежной пары сторон домино (подстановка Y). Такое синхронное монохроматическое перекрашивание смежных сторон домино возможно в том и только том случае, если было выбрано правильное заполнение участка.

Теорема 1 завершает исследование сложности проблемы разрешимости уравнений вида $X_1^{\sigma_1}\theta_1 X_2^{\sigma_2} = X_3^{\sigma_3}\theta_2 X_4^{\sigma_4}$ в полугруппе конечных подстановок первого порядка, где $\sigma_i \in \{0,1\}$, и при этом $X^1 = X$ и $X^0 = \varepsilon$, а ε — тождественная подстановка (нейтральный элемент полугруппы). Действительно, уравнения вида $X_1\theta_1 X_2 = X_3\theta_2 X_4$, $X_2 = X_3\theta_2 X_4$ и $X_1\theta_1 = X_3\theta_2 X_4$ имеют очевидные тривиальные решения вида $(X_1 = \theta_2, X_2 = X_3 = \varepsilon, X_4 = \theta_1)$, $(X_2 = \theta_2, X_3 = \theta_1, X_4 = \varepsilon)$ и $(X_1 = \theta_2, X_3 = \theta_1, X_4 = \varepsilon)$ соответственно. Уравнения вида $\theta_1 X_2 = \theta_2 X_4$ и $\theta_1 X_2 = \theta_2$ соответствуют проблеме унификации и, как показано в работах [3], разрешимы за почти линейное время. Уравнения вида $X_1\theta_1 = X_3\theta_2$ и $X_1\theta_1 = \theta_2$ были исследованы в статье [5] в связи с изучением проблемы эквивалентности в одном классе последовательных программ. Эти уравнения разрешимы за полиномиальное время. И, как установлено в теореме 1, лишь для уравнений вида $X_1\theta_1 X_2 = \theta_2$ задача их разрешимости является NP-полной.

Теорема 1 позволяет оценить сложность одного варианта задачи проверки подобия программ в модели стандартных схем программ с отношением логико-термальной эквивалентности. Как известно (см. [6, 7, 8]), логико-термальная эквивалентность является полиномиально разрешимой аппроксимацией отношения функциональной эквивалентности в классе стандартных схем программ. Поэтому это отношение удобно использовать для решения задач верификации и оптимизации программ. Стандартная схема программ π_2 называется специализацией схемы программ π_1, если существуют такие подстановки θ_1, θ_2, соответствующие линейным программам (последовательностям операторов присваивания), для которых программа π_2 логико-термально эквивалентна последовательной композиции программ $\theta_1; \pi_1; \theta_2$.

Теорема 2. *Задача проверки логико-термальной специализируемости стандартных схем программ является NP-полной.*

Работа выполнена при поддержке гранта РФФИ 12-01-00707.

Литература

[1] *Фаулер М.* Рефакторинг. Улучшение существующего кода. — М.: Символ-Плюс, 2008. — 432 с.

[2] *Roy C. K., Cordy J. R.* A survey on software clone detection research // Technical report TR 2007-541, School of Computing, Queen's University. — 2007. — v. 115.

[3] *Paterson M. S., Wegman M. N.* Linear unification // The Journal of Computer and System Science. — 1978. — v. 16. — N 2. — p. 158-167.

[4] *Lewis H.* Complexity of solvable cases of the decision problem for predicate calculus // Proceedings of the 19-th Conference Foundation of Computer Science. — 1978. — p. 38-47.

[5] *Zakharov V. A.* On the decidability of the equivalence problem for orthogonal sequential programs. // Grammars. — 2000. — v. 2 — N 3. — p. 271-281.

[6] *Иткин В. Э.* Логико-термальная эквивалентность схем программ // Кибернетика. — 1972. — N 1. — с. 5-27.

[7] *Котов В. Е., Сабельфельд В. К.*. Теория схем программ. — М.:Наука, 1991. — 348 с.

[8] *Новикова Т. А., Захаров В. А.* Полиномиальный по времени алгоритм проверки логико-термальной эквивалентности программ. // Труды Института системного программирования РАН. — 2012. — т. 22 — с. 435-455.

Регулярность частотных языков

Д. В. Пархоменко

dcdenis@rambler.ru

МГУ им. М.В.Ломоносова, Москва

$P-$язык это множество слов, встречающихся на выходе детерминированного автомата не менее p раз. Известно, что для каждого натурального p он является регулярным, однако, оставался вопрос, проверяемо ли свойство произвольного регулярного языка быть $p-$языком? Как будет показано в докладе, это свойство проверяемо. Более того, можно установить все такие p для данного регулярного языка, для которых он является $p-$языком.

Рассмотрим конечные алфавиты $A, B : A = \{a_1, \ldots, a_{|A|}\}$, $B = \{b_1, \ldots, b_{|B|}\}$ и конечный детерминированный инициальный автомат (см. [1]):

$$V = (A, Q, B, \varphi, \psi, q_0).$$

Этот автомат порождает ограниченно детерминированную словарную функцию

$$f_V : A^* \to B^*.$$

Точное определение и свойства автоматной функции описаны в [1]. Для простоты изложения в докладе автор ограничился рассмотрением случая $|A| = |B| > 1$ (в случае, $|A| \neq |B|$ получаются аналогичные результаты).

Определение 1. Пусть задан конечный детерминированный инициальный автомат V. *Гистограммной автоматной функцией* автомата V назовем функцию $\varkappa_V : B^* \to N \cup 0$, определенную формулой:

$$\varkappa_V(\beta) = |\{\alpha \in A^* | f_V(\alpha) = \beta\}|.$$

Функция $\varkappa_V$ каждому слову выходного алфавита сопоставляет мощность множества его прообразов при отображении f_V. Фактически, если использовать множество с кратными элементами $f_V(A^*)$, можно каждому слову из B^* поставить в соответствие его кратность во множестве $f_V(A^*)$. Другими словами, $\varkappa_V$ −это функция кратности выходных слов автомата V.

Определение 2. Произвольный непустой язык L алфавита B назовем *продолжаемым*, если для каждого слова $\beta \in B^*$ найдется такая буква $b \in B$, что $\beta b \in B^*$.

Определение 3. Для произвольных конечного детерминированного автомата V и натурального p обозначим

$$L_p(V) = \{\beta \in B^* | \varkappa_V(\beta) \geqslant p\}.$$

Таким образом, $L_p(V)$ - множество выходных слов, которые на выходе автомата V возникают не менее p раз. Регулярные языки с частотными свойствами ранее рассматривались в [2].

Рассмотрим вопрос о структуре множества выходных слов автомата. Полагая фиксированными алфавиты A,B, введём класс языков типа $\mathscr{L}_p$.

Определение 4. Для натурального p положим $\mathscr{L}_p = \{L_p(V) | V \in K(A, B)\}$, где K(A,B)- класс всех автоматов вида $(A, Q, B, \varphi, \psi, q_0)$.

В [3] установлено, что для любых конечного автомата V и натурального $p > 1$ язык $L_p(V)$ является регулярным и продолжаемым. Однако не каждое регулярное продолжаемое множество является языком типа $\mathscr{L}_p$.

При фиксированном натуральном p введем понятие p-раскраски автомата. Традиционно, рассмотрим события, представимые в автомате без выхода с помощью набора финальных состояний.

Определение 5. Пусть задан автомат

$$W = (B, Q, \varphi, q_0, Q_F) \tag{1}$$

без выхода, с выделенным множеством финальных состояний Q_F. Введем функцию

$$k : Q \to N \cup \{0\},$$

сопоставляющую каждому состоянию из Q неотрицательное целое число. Будем говорить, что функция k задает на автомате V *правильную p-раскраску состояний*, если для любой вершины $q \in Q$:

1. $k(q) < p$ для любого $q \notin Q_F$,
   $k(q) \geqslant p$ для любого $q \in Q_F$ и
   $k(q_0) = 1$,
2. $k(q) \cdot |B| = \sum k(\varphi(q, b))$, если $\varphi(q, b) \notin Q_F$ для всех $b \in B$,
   $k(q) \cdot |B| \geqslant s \cdot p + \sum\limits_{b \in B, \varphi(q,b) \notin Q_F} k(\varphi(q, b))$, где $s = |\{b \in B | \varphi(q, b) \in Q_F\}|$.

Автомат V в таком случае назовём *правильно p-раскрашенным.* Ясно, что введенное понятие p-раскраски обобщается и на автоматы с бесконечным числом состояний.

Заметим, что не всякий автомат типа (1) может быть правильно p-раскрашен.

Теорема 1. *Пусть $p > 1$ — натуральное, а автомат $V' = (B, Q', \varphi', q'_0, Q'_F)$ представляет некоторый язык L. Тогда $L \in \mathscr{L}_p$ в том и только том случае, когда найдётся правильно p-раскрашенный автомат $V = (B, Q, \varphi, q_0, Q_F)$, также представляющий L. Причем число состояний нового автомата сравнимо с числом состояний исходного, а именно: $|Q| \leqslant \frac{p(p-1)}{2} \cdot |Q'| + p \cdot |Q'_F|$.*

Теорема 2. *Существует алгоритм, который для любого языка $L \in Reg$ определяет все такие p, что $L \in \mathscr{L}_p$.*

Работа выполнена на кафедре МАТИС под руководством профессора д.ф.м.н. Бабина Д.Н.

Литература

[1] *Кудрявцев В. Б., Алешин С. В., Подколзин А. С.* Введение в теорию автоматов. — М.: Наука, 1985.

[2] *Бабин Д. Н., Холоденко А. Б.* Об автоматной аппроксимации естественных языков // Интеллектуальные системы. — 2007. — Т. 12 — С. 125–136.

[3] *Пархоменко Д. В* Спектральная автоматная функция и связанные с нею автоматные языки // Интеллектуальные системы в производстве. — 2012. — Т. 9, № 1. — С. 165-175.

О биграммных языках с закольцовыванием

А. А. Петюшко

petsan@newmail.ru

МГУ им М. В. Ломоносова, Москва

Биграммные языки

Введение. Еще в начале 20 века выдающимся русским ученым А. А. Марковым был создан аппарат цепей, впоследствии названных цепями Маркова, и опробован [1] на вычислении переходных вероятностей между соседними буквами в поэме А. С. Пушкина „Евгений Онегин“. В дальнейшем этот аппарат получил широкое применение для распознавания и статистического моделирования естественных языков [2]. Тем не менее, в детерминированном случае, за редким исключением прикладных задач (например, для подсчета ДНК-последовательностей [3]), биграммы для исследования формальных языков практически не применялись. В данной статье автор изучает языки, состоящие из слов с фиксированными частотами пар соседних букв.

Определение биграммных языков. Пусть $A, |A| < \infty$ — конечный алфавит, A^* — множество всех конечных слов (включая пустое) в данном алфавите.

Определение 1. Биграммой в алфавите A называется двухбуквенное слово $ab \in A^*, a, b \in A$ (порядок вхождения букв в биграмму имеет значение, т.е. биграмма ab не равна биграмме ba при $a \neq b$).

Определение 2. Обозначим через $\theta_\beta(\eta)$, где $\beta \in A^*, \eta \in A^*$, причем β — непустое слово, отображение $A^* \to N \cup \{0\}$, сопоставляющее слову η число подслов β в слове η, т.е. количество различных разложений слова η в виде $\eta = \alpha'\beta\alpha''$ (α' и α'' могут быть пустыми). При длине слова η, меньшей длины слова β, значение $\theta_\beta(\eta)$ положим равным 0. Само же значение $\theta_\beta(\eta)$ при данных β и η назовем кратностью β в слове η.

С учетом введенных определений, по каждому слову $\alpha \in A^*$ можно построить квадратную матрицу биграмм $\Theta(\alpha) = (\theta_{a_i a_j}(\alpha))_{i,j=1}^{|A|}$ размера $|A| \times |A|$ при условии, что все буквы алфавита $A = \{a_1, a_2, ..., a_{|A|}\}$ пронумерованы и нумерация зафиксирована.

Обозначим через Ξ множество квадратных матриц размера $|A| \times |A|$, каждый элемент которых является целым неотрицательным числом. Таким образом, для каждого $\alpha \in A^*$ имеем $\Theta(\alpha) \in \Xi$. Также, здесь и далее через $\Theta(\alpha)$ будем обозначать матрицу биграмм, построенную по конкретному слову α, а через Θ — просто некоторую матрицу из Ξ, при этом будем считать, что на месте (i, j) матрицы Θ будет стоять значение $\theta_{a_i a_j}$ (т. е. для произвольной матрицы из Ξ мы опустили зависимость от α как для самой матрицы биграмм, так и для отдельных ее элементов).

Определение 3. Назовем биграммным языком $L(\Theta)$, порожденным матрицей $\Theta \in \Xi$, множество всех слов, имеющих одну и ту же матрицу биграмм Θ, т.е. $L(\Theta) = \{\beta \in A^* | \Theta(\beta) = \Theta\}$.

Определение 4. Назовем частотным языком, заданным матрицей биграмм $\Theta \in \Xi$, язык $F_\Theta = \bigcup_{k=1}^{\infty} L(k\Theta)$, т.е. язык, состоящий из всех таких слов β, что набор кратностей этих слов $\Theta(\beta)$ кратен набору Θ, а именно, $F_\Theta = \{\beta \in A^* | \Theta(\beta) = k\Theta, k \in N\}$, где умножение k на Θ понимается как умножение скаляра на матрицу.

Подробно языки $L(\Theta)$ и F_Θ были изучены в [4]. В данной же работе изучаются свойства языков, в которых задающая их матрица биграмм учитывает так называемую „закольцованность“.

Биграммные языки с закольцовыванием

Определение 5. Элементарной матрицей кратностей биграмм $\Theta_{ij} \in \Xi$ будем называть матрицу из пространства матриц биграмм Ξ, имеющую единственный ненулевой элемент на месте (i, j): $\theta_{a_i a_j} = 1, \theta_{a_k a_l} = 0, (k, l) \neq (i, j), 1 \leqslant i, j, k, l \leqslant n$.

Определение 6. Назовем $\Omega(\alpha)$ матрицей кратностей биграмм с закольцовыванием для непустого слова $\alpha \in A^*$ следующую матрицу:
1) при однобуквенном слове $\alpha = a_t, 1 \leqslant t \leqslant n$, $\Omega(\alpha) = \Theta_{tt}$,
2) при длине слова α не меньше 2, то есть $\alpha = a_i\alpha_1 a_j, 1 \leqslant i, j \leqslant n$, где $\alpha_1 \in A^*$ (в том числе α_1 может быть пустым), $\Omega(\alpha) = \Theta(\alpha) + \Theta_{ji}$.

Здесь и далее будем считать, что на месте (i, j) матрицы $\Omega(\alpha)$ будет стоять значение $\omega_{a_i a_j}(\alpha)$.

Содержательно, матрица биграмм с закольцовыванием — это та же обычная матрица биграмм за исключением единичной добавки в одной ячейке,

которая отвечает за биграмму, связывающую последнюю букву слова с первой (отсюда и название — „с закольцовыванием“, поскольку мы как бы считаем биграммы не на линейном слове, а на слове, начало и конец которого объединены в кольцо).

Определение 7. Назовем биграммным языком с закольцовыванием $K(\Omega)$ множество всех слов, имеющих одну и ту же матрицу Ω кратностей биграмм с закольцовыванием, т.е. $K(\Omega) = \{\beta \in A^* | \Omega(\beta) = \Omega\}$.

Рассмотрим основные свойства биграммных языков с закольцовыванием, которые во многом повторяют аналогичные утверждения для биграммных языков.

Построим по матрице $\Omega(\alpha)$ (или по произвольной матрице $\Omega \in \Xi$) ориентированный граф $G_{\Omega(\alpha)}$ (соответственно, G_Ω) на плоскости. Вершинами у этого графа будут все буквы из алфавита A, при этом ребра будут соответствовать биграммам с учетом их кратностей, т.е. кратность $\omega_{ab}(\alpha)$ будет порождать $\omega_{ab}(\alpha)$ ориентированных ребер $a \to b$. Аналогично, кратность $\omega_{cc}(\alpha)$ будет порождать $\omega_{cc}(\alpha)$ петель $c \to c$.

Также нам понадобятся понятия эйлеровых циклов и необходимых и достаточных условий существования оных в ориентированных графах из [5].

Теорема 1. *Биграммный язык с закольцовыванием $K(\Omega)$ состоит не более чем из конечного числа слов одинаковой длины $l_\Omega = \sum_{a_i,a_j \in A} \omega_{a_i a_j}$. При этом язык $K(\Omega)$ непуст, если построенный по Ω ориентированный граф G_Ω — эйлеров.*

Теорема 2. *Пусть задана матрица $\Omega \in \Xi$ с эйлеровым графом G_Ω. Тогда существует взаимно-однозначное соответствие между словами языков $K(\Omega)$ и $L(\Theta)$, где $\Omega = \Theta$.*

Следствие 1. *Пусть для матрицы $\Omega \in \Xi$ ориентированный граф G_Ω — эйлеров. Тогда количество слов в языках $K(\Omega)$ и $L(\Theta)$, где $\Omega = \Theta$, одинаково: $|K(\Omega)| = |L(\Theta)|$.*

Определение 8. Назовем частотным биграммным языком с закольцовыванием, заданным матрицей биграмм $\Omega \in \Xi$ с закольцовыванием, язык $E_\Omega = \bigcup_{k=1}^{\infty} K(k\Omega)$, т.е. язык, состоящий из всех таких слов β, т.ч. матрица биграмм $\Omega(\beta)$ с закольцовыванием этих слов кратна набору Ω, а именно $E_\Omega = \{\beta \in A^* | \Omega(\beta) = k\Omega, k \in N\}$, где умножение k на Ω понимается как умножение скаляра на матрицу.

Теорема 3. *Если задана матрица биграмм $\Omega \in \Xi$ с закольцовыванием, то:*
1) *если ориентированный граф G_Ω является эйлеровым, то в частотном языке E_Θ с закольцовыванием счетное множество слов;*
2) *если ориентированный граф G_Ω не является эйлеровым, то в частотном языке E_Θ с закольцовыванием нет ни одного слова.*

Определение 9. Назовем две ненулевые матрицы Ω_1 и Ω_2 из Ξ неколлинеарными, если не существует ненулевых действительных коэффициентов $c_1, c_2 \in R, (c_1, c_2) \neq (0, 0)$, для которых $c_1\Omega_1 + c_2\Omega_2 = 0$.

Теорема 4. *Пусть задана матрица биграмм Ω с закольцовыванием такая, что ориентированный граф G_Ω — эйлеров. Тогда:*

1) *если существует такое разложение* Ω *в сумму двух ненулевых неколлинеарных матриц* $\Omega = \Omega_1 + \Omega_2$ *такое, что оба ориентированных графа* G_{Ω_1} *и* G_{Ω_2} *— эйлеровы, то частотный язык с закольцовыванием* E_Ω *нерегулярен;*
2) *в противном случае язык* E_Ω *регулярен. При этом для* $\forall k \in N$ *существуют ровно* l *слов* $\beta_{k,i}, i = 1..l$, *т.ч.* $\Omega(\beta_{k,i}) = k\Omega$, *а* l *— число ненулевых элементов в матрице* Ω.

Литература

[1] *Марков А. А.* Пример статистического исследования над текстом „Евгения Онегина“, иллюстрирующий связь испытаний в цепь // Известия Императорской Академии наук. — 1913. — Сер. 6, Т. 7, № 3. — С. 153–162.

[2] *Essen U., Steinbiss V.* Cooccurrence smoothing for stochastic language modeling // IEEE International Conference on Acoustics, Speech, and Signal Processing. — 1992. — Vol. 1. — P. 161–164.

[3] *Hutchinson J. P., Wilf H. S.* On eulerian circuits and words with prescribed adjacency patterns // Journal of Combinatorial Theory. — 1975. — Ser. A, Vol. 18. — P. 80–87.

[4] *Петюшко А. А.* О биграммных языках // Дискретная математика. — 2013. — Т. 25, № 3. — С. 64–77.

[5] *Оре О.* Теория графов. — М.: Наука, 1980. — 336 с.

Решение проблемы эквивалентности и проблемы эквивалентных преобразований в одной двухпараметрической алгебраической модели программ

Р. И. Подловченко

podlovchenko.rimma@gmail.com

МГУ, Москва

Рассматриваемые нами алгебраические модели программ предназначены для изучения семантических свойств последовательных программ, использующих все стандартные композиции операторов, кроме аппарата процедур [1].

Сами программы, будучи записаны в формализованном виде, представляют собой конечный ориентированный граф со входом и выходом, все остальные вершины которого нагружены операторами и логическими условиями.

При моделировании программы объектом, называемым его схемой, сохраняется её структура, а операторы и логические условия заменяются соответственно операторными символами и логическими переменными, принимающими значения 0 и 1. Подходящая интерпретация тех и других возвращает схему в программу.

Алгебраическая модель программ состоит из множества схем программ, построенных над базисом операторных символов и логических переменных. Семантическая трактовка схемы связана с выполнением её на функциях разметки; такой функцией каждой операторной цепочке приписываются значения на ней всех логических переменных. Процедурой выполнения схемы на

функции разметки схеме сопоставляется отображение множества всех функций разметки в множество операторных цепочек, воспринимаемых как результат выполнения схемы в случае, когда оно завершаемо.

Отдельная алгебраическая модель характеризуется своим отношением эквивалентности схем. Оно определяется двумя параметрами - подмножеством функций разметки, допустимых при выполнении схемы, и введённой эквивалентностью в множестве всех операторных цепочек; результаты выполнения двух схем на допустимой функции разметки обязаны быть эквивалентными цепочками. Так возникает двухпараметрическая алгебраическая модель программ.

В проблематике теории этих моделей фундаментальными являются две проблемы – проблема эквивалентности схем в модели, состоящая в поиске разрешающего эквивалентность алгоритма, и проблема эквивалентных преобразований (э.п.), заключающаяся в построении системы э.п., полной в данной модели; средствами полной системы любая схема алгоритмически трансформируема в любую ей эквивалентную. Вторая проблема рассматривается при разрешимости первой.

Возвратимся к вопросу о том, как решения этих проблем используются для программ. В этих целях рассматриваются только аппроксимирующие алгебраические модели. Для такой модели существует класс программ с выполнением условия: из эквивалентности схем в модели всегда следует эквивалентность программ, структура которых совпадает со структурой этих схем. Таким образом, алгоритм, разрешающий эквивалентность схем в аппроксимирующей модели, частично разрешает эквивалентность программ из аппроксимируемого их класса, а все э.п. схем одновременно являются и э.п. программ. Поэтому в первую очередь в теории алгебраических моделей построен приемлемый достаточный признак того, что модель является аппроксимирующей [1].

В основном рассматривались однопараметрические модели - в них допустимость функции разметки полностью определяется отношением эквивалентности операторных цепочек. Для них разработаны две методики – методика распознавания эквивалентности в модели [2] и методика построения системы э.п., полной в модели [3]. Задачей данной статьи является применение этих методик к двухпараметрической модели программ.

В качестве таковой нами рассматривается модель M со следующими параметрами. Эквивалентными объявляются две операторные цепочки, обладающие свойством: одна может быть получена из другой перестановками соседних операторных символов. Допустимой для модели считается всякая функция разметки μ, которая сохраняет своё значение на эквивалентных операторных цепочках и удовлетворяет требованию: какими бы ни были логическая переменная p, операторная цепочка h и операторный символ y. значение переменной p в $\mu(h)$ не превышает её значения в $\mu(hy)$.

Применением разработанной в [2] методики установлена разрешимость в M проблемы эквивалентности (см. [4]). Отметим выполненные при этом этапы исследований.

1. В схеме из M рассматривались ориентированные пути с началом в её входе; они названы маршрутами. Маршруты в двух схемах, прокладываемые

общей для них допустимой функцией разметки, названы сочетаемыми. Отношение эквивалентности схем из M переведено на язык требований к сочетаемым маршрутам.

2. Для всякого натурального числа N введено алгоритмически распознаваемое отношение N-эквивалентности схем из M, выполняемое при их эквивалентности.
3. Исследованием структуры эквивалентных схем построен алгоритм, который для любых схем G_1, G_2 из модели M находит натуральное число $N(G_1, G_2)$ такое , что из $N(G_1, G_2)$-эквивалентности этих схем следует их эквивалентность в M.

Для модели M по методике, описанной в [3], построена полная в ней система э.п. (см. [4]). Согласно этой методике, создано требуемое формальное исчисление. Его объектами являются фрагменты схем из M, обобщающие понятие схемы. Единственным правилом вывода в этом исчислении является правило подстановки во фрагмент вместо одного его подфрагмента другого фрагмента. Фрагменты названы эквивалентными, если применением подстановки любая схема преобразуется в эквивалентную ей схему, и условно эквивалентными, когда исходная схема удовлетворяет некоторому условию. Аксиомы исчисления выявляются путём трансформации схемы к каноническому виду. Последние вводятся в конечном числе в каждом классе эквивалентных схем в M, на основании чего появляется дополнительная условная аксиома.

В заключение отметим, что наличие нескольких канонических форм в классе эквивалентных схем принципиально отличает модель M от исследуемых ранее однопараметрических алгебраических моделей программ.

Литература

[1] *Подловченко Р. И.* От схем Янова к теории моделей программ // Математические вопросы кибернетики. — М: Физматниз, 1998. — Вып. 7. — С. 281–302.

[2] *Подловченко Р. И.* Об одной методике распознавания эквивалентности в алгебраических моделях программ // Программирование. — 2011. — № 6. — С. 33–43.

[3] *Подловченко Р. И.* Методология построения системы эквивалентных преобразований, полной в модели вычислений, и её применение для алгебраических моделей программ // Труды семинара “Семантика, спецификация и верификация программ: теория и приложения”. — Казань: Отечество, 2010. — С. 82–87.

[4] *Подловченко Р. И.* Исследование двухпараметрической алгебраической модели программ по методикам, разработанным для однопараметрических моделей // Программирование (в печати).

О мощности некоторых семейств β-замкнутых классов функций многозначной логики

Д. К. Подолько

podolko_dk@mail.ru

Московский госуниверситет им. М. В. Ломоносова, г. Москва

Известно [1, 2], что семейство замкнутых классов функций k-значной логики является континуальным при $k \geqslant 3$. Поэтому возникают значительные сложности при изучении данного семейства классов, и для их исследования часто рассматриваются различные усиления оператора замыкания, которые позволяют получать множества замкнутых классов с более обозримой структурой (см., например, [3, 4, 5]).

Настоящая работа относится к аналогичному направлению исследований. В ней изучаются функции k-значной логики, где $k = 2^m$, $m \geqslant 2$. Для этого определяется специальный оператор β-замыкания на основе кодирования данных функций в двоичной системе счисления и строится отображение семейства всех β-замкнутых классов в семейство замкнутых классов булевых функций. В работе для каждого замкнутого класса $\mathcal{B}$ булевых функций исследуется мощность семейства различных β-замкнутых классов, которые отображаются в класс $\mathcal{B}$ и содержат только функции, принимающие не более четырех значений.

Сформулируем основные определения (подробнее см. [6]). Пусть $k = 2^m$, где $m \geqslant 2$. Тогда каждое число α из множества $\{0, 1, \ldots, k-1\}$ можно записать в двоичной системе счисления. Это означает, что ему взаимно-однозначно сопоставляется двоичный вектор $(\alpha_1, \ldots, \alpha_m)$ из $\{0,1\}^m$, который будем обозначать через $\langle \alpha_1, \ldots, \alpha_m \rangle$. Переменной x, принимающей значения из множества $\{0, 1, \ldots, k-1\}$, можно поставить в соответствие вектор-переменную $\langle x_1, \ldots, x_m \rangle$, где $x_1, \ldots, x_m$ являются переменными, принимающими значения из множества $\{0, 1\}$, таким образом, что каждому значению α переменной x ставится в соответствие значение $\langle \alpha_1, \ldots, \alpha_m \rangle$ вектор-переменной $\langle x_1, \ldots, x_m \rangle$. Данную вектор-переменную будем обозначать также через $\widehat{x}$.

Обозначим через P_k множество всех функций k-значной логики. При рассматриваемых нами значениях k произвольной n-местной функции $F(x^1, \ldots, x^n)$ из P_k можно взаимно-однозначно сопоставить вектор-функцию $\langle f_1, \ldots, f_m \rangle$, где функции $f_1, \ldots, f_m$ являются булевыми и каждая из них зависит от всех булевых переменных $x_1^j, \ldots, x_m^j$, $j = 1, \ldots, n$ (здесь $\widehat{x}^j = \langle x_1^j, \ldots, x_m^j \rangle$). Данную вектор-функцию будем также обозначать через $\widehat{F}(\langle x_1^1, \ldots, x_m^1 \rangle, \ldots, \langle x_1^n, \ldots, x_m^n \rangle)$ или $\widehat{F}$.

Описанные представления будем называть *двоичным представлением числа α, переменной x* и *функции F* соответственно.

Пусть $F \in P_k$ и $\widehat{F} = \langle f_1, \ldots, f_m \rangle$. Каждую из булевых функций $f_1, \ldots, f_m$ будем называть *компонентой функции F*. Множество всех компонент функции F обозначим через $b(F)$.

Пусть $\mathcal{A} \subseteq P_k$. Класс булевых функций, совпадающий с замыканием множества $\bigcup\limits_{F \in \mathcal{A}} b(F)$ относительно операций суперпозиции и введения несущественной переменной, будем называть *булевым замыканием множества* $\mathcal{A}$ и обозначать через $B(\mathcal{A})$.

Будем говорить, что функция H из P_k получена из функций множества $\mathcal{A}$ при помощи *операции двоичной суперпозиции*, если найдутся функция F из множества $\mathcal{A}$ и функции $g_1, \ldots, g_{mn}$ из множества $B(\mathcal{A})$ (где n — число переменных функции F), такие, что выполняется следующее равенство:

$$\widehat{H} = \widehat{F}\big(\langle g_1, \ldots, g_m\rangle, \ldots, \langle g_{m(n-1)+1}, \ldots, g_{mn}\rangle\big).$$

Множество всех функций, которые могут быть получены из функций системы $\mathcal{A}$ при помощи операций двоичной суперпозиции и введения несущественной переменной, будем называть *β-замыканием множества* $\mathcal{A}$ и обозначать через $[\mathcal{A}]_\beta$. В работе [6] установлено, что введенный таким образом оператор удовлетворяет всем необходимым свойствам оператора замыкания. Множество $\mathcal{A}$ назовем *β-замкнутым*, если выполняется равенство $[\mathcal{A}]_\beta = \mathcal{A}$.

Для каждого r, $1 \leqslant r \leqslant k$, обозначим через $P_{k|r}$ множество всех функций k-значной логики, принимающих не более r значений. Для данных значений числа r обозначим также через $\mathcal{C}(k, r)$ множество всех замкнутых классов $\mathcal{B}$ булевых функций, для которых семейство различных β-замкнутых классов $\mathcal{A}$ функций из $P_{k|r}$, удовлетворяющих условию $B(\mathcal{A}) = \mathcal{B}$, является конечным, а через $\mathcal{Q}(k, r)$ — множество всех классов булевых функций, для которых аналогичное семейство β-замкнутых классов являются континуальными.

В работе [6] получены следующие результаты.

Теорема 1. *Пусть $k = 2^m$, где $m \geqslant 2$. Тогда множество $\mathcal{C}(k, 2)$ совпадает с множеством всех замкнутых классов булевых функций.*

Теорема 2. *Пусть $k = 2^m$, где $m \geqslant 2$. Тогда семейство различных β-замкнутых классов функций из $P_{k|2}$ является счетным.*

При доказательстве теоремы 2 в работе [6] для каждого замкнутого класса $\mathcal{B}$ булевых функций приведен пример β-замкнутого класса функций из $P_{k|2}$ с булевым замыканием, равным $\mathcal{B}$, и показано, что все такие классы различны.

Настоящая работа является продолжением статьи [6], и в ней установлены следующие утверждения для семейств β-замкнутых классов функций из $P_{k|r}$ при $r = 3, 4$.

Теорема 3. *Пусть $k = 2^m$, где $m \geqslant 2$. Тогда имеет место равенство:*

$$\mathcal{Q}(k, 3) = \{O^\mu, I^\mu \mid \mu \geqslant 3\} \bigcup \{O^\infty, I^\infty\}.$$

Теорема 4. *Множество $\mathcal{Q}(4, 4)$ содержит все замкнутые классы из множества $\{O^\mu, O_0^\mu, MO^\mu, MO_0^\mu \mid \mu \geqslant 3\} \bigcup \{O^\infty, O_0^\infty, MO^\infty, MO_0^\infty\}$, двойственные им классы, и только такие классы.*

Теорема 5. *Пусть $k = 2^m$, где $m \geqslant 3$. Тогда имеет место равенство:*

$$\mathcal{Q}(k, 4) = \{O^2, I^2\} \bigcup \mathcal{Q}(4, 4).$$

Теорема 6. *Пусть $3 \leqslant r \leqslant 4$, $k = 2^m$, где $m \geqslant 2$, и $\mathcal{B}$ — замкнутый класс булевых функций, такой, что $\mathcal{B} \notin \mathcal{Q}(k, r)$. Тогда класс $\mathcal{B}$ содержится в множестве $\mathcal{C}(k, r)$.*

Таким образом для каждой пары чисел k и r, где $k = 2^m$, $m \geqslant 2$, $r \leqslant 4$, и каждого замкнутого класса $\mathcal{B}$ булевых функций определено, конечным или континуальным (счетным быть не может) является семейство различных β-замкнутых классов $\mathcal{A}$ функций из $P_{k|r}$, удовлетворяющих условию $B(\mathcal{A}) = \mathcal{B}$. В частности, приведена полная классификация по мощности (в рассмотренном выше смысле) семейств всех β-замкнутых классов функций четырехзначной логики с одинаковым булевым замыканием.

Работа выполнена при финансовой поддержке РФФИ (проект № 14-01-00598) и программы фундаментальных исследований ОМН РАН «Алгебраические и комбинаторные методы математической кибернетики и информационные системы нового поколения» (проект «Задачи оптимального синтеза управляющих систем»).

Литература

[1] *Яблонский С. В.* Введение в дискретную математику. — М.: Высшая школа, 2001. — 384 с.

[2] *Янов Ю. И., Мучник А. А.* О существовании k-значных замкнутых классов, не имеющих конечного базиса // Докл. АН СССР. — 1959. — Т. 127, № 1. — С. 44–46.

[3] *Нгуен Ван Хоа.* Описание замкнутых классов k-значной логики, сохраняемых всеми автоморфизмами // Докл. АН Беларуси. — 1994. — Т. 38, № 3. — С. 16–19.

[4] *Марченков С. С.* S-классификация функций многозначной логики // Дискретная математика. — 1997. — Т. 9, № 3. — С. 125–152.

[5] *Тарасова О. С.* Классы функций k-значной логики, замкнутые относительно операций суперпозиции и перестановок // Математические вопросы кибернетики. Вып. 13. — М.: Физматлит, 2004. — С. 59–112.

[6] *Подолько Д. К.* О классах функций, замкнутых относительно специальной операции суперпозиции // Вестник Московского университета. Серия 1. Математика. Механика. — 2013. — № 6. — С. 54–57.

Быстрый алгоритм проверки эквивалентности программ с коммутативными и подавляемыми операторами

В. В. Подымов

valdus@yandex.ru

МГУ им. М.В. Ломоносова, Москва

Рассматривается модель пропозициональных последовательных программ, синтаксис которых близок к синтаксису “реальных” программ в императивных языках программирования и семантика которых определяется на основе структур динамической логики (шкал, моделей) [1]. Рассматриваются шкалы (состояния данных вместе с интерпретацией операторов), основанные на упо-

рядоченных конечнопорожденных моноидах, задаваемых тождествами коммутативности ($ab = ba$) и подавления ($ab = b$). Содержательно, тождество $ab = ba$ означает, что результат выполнения операторов a и b не зависит от порядка их выполнения, тождество $ab = b$ — что влияние оператора a на текущее состояние данных всегда "подавляется" влиянием оператора b. Такие тождества можно легко найти при помощи анализа переменных, используемых и определяемых операторами, а упорядоченность полученного моноида можно легко проверить, используя результат статьи [2]. Основной результат данной работы — полиномиальный по времени алгоритм проверки эквивалентности программ в рассматриваемой модели и, как следствие, обоснование полиномиальной разрешимости проблемы эквивалентности. В [3] предложен быстрый разрешающий алгоритм, предполагающий существенно более строгие ограничения на стурктуру рассматриваемых моноидов. Этот алгоритм не может быть применим напрямую к рассматриваемому случаю, однако основная его идея — построение графа, описывающего всевозможные вычисления программ, реализуемые в общих моделях — лежит и в основе результата данной статьи.

Считаем заданными конечные множества операторов $\mathfrak{A}$ и логических условий Δ. Пропозициональная последовательная программа (далее — программа) — это система $\pi = (L, l_{en}, l_{ex}, T, B)$, где L — конечное множество точек программы, включающее вход l_{en} и выход l_{ex}, $T : (L \setminus \{l_{ex}\}) \times \Delta \to L$ — функция переходов и $B : (L \setminus \{l_{en}, l_{ex}\}) \to \mathfrak{A}$ — функция привязки. Вместо $T(l_1, \delta) = l_2$ будем писать $l_1 \xrightarrow{\delta} l_2$ и $l_1 \to l_2$. Для простоты выкладок считаем, что $B(l_{ex}) = B(l_{ex}) = \lambda$, где λ — пустое слово. Трассой программы назовем любую ее последовательность точек вида $tr = l_1 \to l_2 \to \dots$. Непродолжаемую трассу назовем вычислением. Записью $B(tr)$ обозначим слово $B(l_1)B(l_2)\dots$. Детерминированная динамическая шкала (далее — шкала) — это система $\mathcal{F} = (S, s_0, R)$, где S — множество состояний данных, $s_0 \in S$ — начальное состояние и $R : S \times \mathfrak{A} \to S$ — функция переходов. Обобщим функцию переходов: $R^*(s, \lambda) = s$, $R^*(s, ah) = R^*(R(s, a), h)$. Записью $[h]$ обозначим состояние $R^*(s_0, h)$. Состояние s_2 достижимо из s_1 (обозначим это записью $s_2 \leqslant s_1$), если найдутся слова h_1, h_2, такие что $s_1 = [h_1]$ и $s_2 = [h_1 h_2]$. Шкалу $\mathcal{F}$ назовем упорядоченной, если $\leqslant$ — отношение нестрогого частичного порядка. Детерминированная динамическая модель (далее — модель) — это система $\mathcal{M} = (\mathcal{F}, \xi)$, где $\mathcal{F} = (S, s_0, R)$ — шкала, и $\xi : S \to \Delta$ — оценка логических условий. Вычисление cp программы π назовем $\mathcal{M}$-вычислением, если для любого его префикса $tr \xrightarrow{\delta} l$ верно $\xi([B(tr)]) = \delta$. Результатом конечного вычисления cp назовем состояние данных $[B(cp)]$. Бесконечные вычисления безрезультатны. Программы π_1, π_2 назовем $\mathcal{F}$-эквивалентными, где $\mathcal{F}$ — шкала, если для любой модели $\mathcal{M} = (\mathcal{F}, \xi)$ результаты их $\mathcal{M}$-вычислений совпадают.

Тождество вида $(ab = ba)$, где $a, b \in \mathfrak{A}$, назовем тождеством коммутативности, вида $(ab = b)$ — тождеством подавления. Рассмотрим конечнопорожденный моноид $\mathfrak{M}(C, A)$ с множеством образующих $\mathfrak{A}$, задаваемый набором тождеств коммутативности C и подавления A. Пусть $S_{\mathfrak{M}}$ — множество элементов этого моноида и $\circ : S_{\mathfrak{M}} \times S_{\mathfrak{M}} \to S_{\mathfrak{M}}$ — задаваемая им операция. Элемент моноида $\mathfrak{M}(C, A)$, порожденный словом h, обозначим записью $\langle h \rangle$. Слово h

назовем минимальным представителем (элемента $\langle h\rangle$), если для любого слова g, такого что $\langle g\rangle$, верно $|g| \geqslant |h|$. Записью $\|s\|$ обозначим длину минимальных представителей элемента s. Будем говорить, что шкала $\mathcal{F} = (S, s_0, R)$ основывается на моноиде $\mathfrak{M}(C, A)$, если $S = S_{\mathfrak{M}}$, $s_0 = \langle\lambda\rangle$ и $R(s, a) = s \circ \langle a\rangle$. Такую шкалу назовем (C, A)-шкалой. Отметим без доказательства, что если (C, A)-шкала упорядочена, то моноид $\mathfrak{M}(C, A)$ обладает свойством левого сокращения: если $s \circ s_1 = s \circ s_2$, то $s_1 = s_2$. Далее считаем заданными упорядоченную (C, A)-шкалу $\mathcal{F} = (S, s_0, R)$ и программы $\pi_i = (L_i, l^i_{en}, l^i_{ex}, T_i, B_i)$, $i \in \{1, 2\}$.

Опишем граф совместных вычислений Γ. Вершины графа Γ суть четверки $w = (l_1, l_2, s_1, s_2)$, где $l_1 \in L_1$, $l_2 \in L_2$ и $s_1, s_2 \in S$. Вход графа Γ — вершина $(l^1_{en}, l^2_{en}, s_0, s_0)$. Вершину w назовем терминальной, если: либо l_1 — выход, l_2 не выход и $s_2 \neq s_0$; либо l_1 не выход, l_2 — выход и $s_1 \neq s_0$; либо l_1, l_2 — выходы. Терминальные вершины, отличные от $(l^1_{ex}, l^2_{ex}, s_0, s_0)$, назовем опровергающими. Метки дуг, исходящих из нетерминальной вершины w, образуют множество: $\Delta \times \{\varepsilon\}$, если l_2 — выход или $s_2 \neq s_0$; $\{\varepsilon\} \times \Delta$, если либо l_1 — выход, либо $s_1 \neq s_0$ и $s_2 = s_0$; $\{(\delta, \delta) \mid \delta \in \Delta\}$, если l_1 и l_2 не выходы и $s_1 = s_2 = s_0$. Различные дуги, исходящие из одной вершины, помечены различными метками. Из терминальных вершин не исходит ни одной дуги.

Если $w \xrightarrow{d_1, d_2} w'$, то вершина $w' = (l'_1, l'_2, s'_1, s'_2)$ определяется следующим образом (здесь $i \in \{1, 2\}$). Если $d_i = \varepsilon$, то $l'_i = l_i$ и $s''_i = s_i$. Если $d_i \neq \varepsilon$, то $l_i \xrightarrow{d_i} l'_i$ и $s''_i = s_i \circ [l'_i]$. Если $s''_1 = s''_2$, то $s'_1 = s'_2 = s_0$; если $s''_i < s''_{3-i}$, то $s'_i = s_0$ и $s'_{3-i} = s'''$, где $s''_{3-i} = s''_i \circ s'''$; иначе $s'_1 = s''_1$ и $s'_2 = s''_2$. Корректность описания обеспечивается свойством левого сокращения моноида $\mathfrak{M}(C, A)$.

Корневым маршрутом графа Γ назовем маршрут, начинающийся в его корне. Корневой маршрут назовем опровергающим, если либо он оканчивается в опровергающей вершине, либо он бесконечен и для некоторого i, $i \in \{1, 2\}$, и некоторого натурального N i-е компоненты меток его дуг, начиная с N-й, равны ε, и i-е компоненты его вершин, начиная с N-й, завершаемы.

Теорема 1. *Программы π_1, π_2 $\mathcal{F}$-эквивалентны тогда и только тогда, когда граф Γ не содержит опровергающих маршрутов.*

Эффектом подавления, индуцированным состоянием данных s, назовем функцию $æ_s : S \to S$, такую что $æ_s(s') = s''$, где $s' \circ s = s'' \circ s$ и значение $\|s''\|$ минимально. Корректность определения основана на том факте, что уравнение $X \circ s_1 = s_2$ имеет не более одного решения с минимальным значением $\|X\|$. Эффекты подавления можно частично упорядочить: $æ_1 \leqslant æ_2$, если существуют состояния данных s_1, s_2, такие что $s_1 \leqslant s_2$, $æ_1 = æ_{s_1}$ и $æ_2 = æ_{s_2}$. Множество эффектов подавления конечно — это обосновывается леммой 2.

Автоматом назовем систему $A = (Q, q_0, T_A, S_A, B_A)$, где Q — конечное множество состояний, $q_0 \in Q$ — начальное состояние, $T_A : Q \times \mathfrak{A} \to Q$ — функция переходов, S_A — множество меток и $B_A : Q \to S_A$ — разметка состояний. Записью $A(h)$, где $h \in \mathfrak{A}^*$, обозначим состояние, в которое автомат A приводится словом h. Автоматом, распознающим эффекты подавления, или æ-автоматом, назовем автомат A, обладающий следующими свойствами: если

$[h_1] = [h_2]$, то $A(h_1) = A(h_2)$; S_A есть множество всех эффектов подавления; $B_A(A(h)) = æ_{[h]}$.

Лемма 2. *Существует æ-автомат.*

Считаем заданными æ-автомат $A = (Q, q_0, T_A, S_A, B_A)$ и число $n = \max(|\pi_1|, |\pi_2|)$. Согласно определению æ-автомата будем вместо $A(h)$ писать $A([h])$. Разобьем вершины графа Γ на $O(n^2)$ групп по следующему признаку: в одну группу попадают все вершины (l_1, l_2, s_1, s_2) с совпадающими значениями l_1, l_2, s_2 и совпадающими состояниями $A(s_1)$. Рассмотрим произвольный обход графа Γ (например, в глубину), начинающийся в корне. Определим на его основе k-обход: если в процессе обхода в группу была добавлена k-я вершина, то остальные вершины этой группы игнорируются. В процессе k-обхода будет просмотрено $O(kn^2)$ вершин графа Γ.

Лемма 3. *Существует число $k = O(n^3)$, такое что ответ об $\mathcal{F}$-эквивалентности программ π_1, π_2 можно определить в результате k-обхода графа* Γ.

Теорема 4. *Пусть $\mathcal{F}$ — упорядоченная (C, A)-шкала. Тогда проблема $\mathcal{F}$-эквивалентности программ разрешима за полиномиальное время.*

Теорему обосновывают последняя лемма и полиномиальная разрешимость нахождения минимального представителя заданного элемента моноида $\mathfrak{M}(C, A)$ и проверки равенства его элементов. Алгоритм проверки эквивалентности состоит в k-обходе графа Γ и вынесении решения в зависимости от того, найдены ли опровергающие маршруты в обойденном фрагменте и накоплено ли k вершин в одной из определенных перед обходом групп.

Литература

[1] *Harel D.* Dynamic logic. — The MIT Press, 2000. — 459 p.

[2] *Подымов В. В., Захаров В. А.* Об одной полугрупповой модели программ, определяемой при помощи двухленточных автоматов // Научные ведомости БелГУ. Серия История. Политология. Экономика. Информатика. — 2010. — № 7, вып. 14/1. — С. 94–101.

[3] *Захаров В. А., Щербина;В. Л.* Эффективные алгоритмы проверки эквивалентности программ в моделях, связанных с обработкой прерываний // Вестн. Моск. ун-та. Сер. 15. Вычисл. матем. и киберн. — 2008. — № 2. — С. 33–41.

О проверяющих и диагностических тестах для функциональных элементов

К. А. Попков

kirill-formulist@mail.ru

МГУ им. М. В. Ломоносова, Москва

Введение

Рассматриваются задачи проверки исправности и распознавания состояний функциональных элементов с использованием экспериментов, заключаю-

щихся в составлении произвольных схем из заданных функциональных элементов с последующим "прозваниванием" этих схем, т. е. нахождением булевых функций, реализуемых составляемыми схемами. Суть общепринятой математической модели схемы из функциональных элементов и тех элементов, из которых строятся эти схемы, с исчерпывающей полнотой и ясностью представлена в [1]; именно такая математическая модель является объектом исследования и рассматривается ниже.

Представим, что имеются N функциональных элементов $E_1, \ldots, E_N$ ($N \geqslant 1$). Каждый элемент, рассматриваемый как простейшая схема из функциональных элементов, имеет $n \geqslant 1$ входов $v_1, \ldots, v_n$ и один выход и в исправном состоянии реализует на выходе заданную булеву функцию $f(x_1, \ldots, x_n)$, где $x_1, \ldots, x_n$ — переменные, подаваемые на его входы соответственно $v_1, \ldots, v_n$ (считаем, что функция $f(x_1, \ldots, x_n)$ существенно зависит от всех своих переменных и, как следствие, отлична от константы). В неисправном состоянии каждый элемент реализует одну из констант 0 или 1. Неисправность элемента E_i, при которой он реализует константу 0 (или 1), будем называть неисправностью E_i типа 0 (соответственно, 1). Предполагается, что среди данных N функциональных элементов не более k элементов могут быть неисправны, где k – заданное натуральное число, $k \leqslant N$. Можно составлять любые схемы с одним выходом из данных функциональных элементов и наблюдать выдаваемые схемами значения на любых наборах значений переменных.

Задача заключается в том, чтобы протестировать функциональные элементы, то есть для каждого из них определить, исправен данный элемент или неисправен (задача проверки), и, в дополнение к этому, определить тип неисправности каждого неисправного элемента (задача диагностики), используя при тестировании по возможности меньшее число схем.

Основные определения и предварительные замечания

Диагностическим тестом назовём такой набор схем $S_1, \ldots, S_l$, составленных из заданных функциональных элементов, что по набору функций, реализуемых этими схемами, можно однозначно определить состояние каждого из N элементов. Число l назовём *длиной* этого теста. (Здесь используется терминология, общепринятая для диагностики управляющих систем: см., например, [2].)

Проверяющим тестом назовём такой набор схем $S_1, \ldots, S_l$, составленных из заданных функциональных элементов, что по набору функций, реализуемых этими схемами, можно однозначно определить исправность или неисправность каждого из N элементов. Число l назовём *длиной* этого теста.

Отметим, что проверяющий тест, в отличие от диагностического, не обязан определять тип неисправности (0 или 1) каждого неисправного элемента.

Введём функции $L_c(f, N, k)$ и $L_d(f, N, k)$, равные длинам самого короткого, соответственно, проверяющего и диагностического тестов для N функциональных элементов, среди которых не более чем k неисправных (в исправном состоянии каждый элемент реализует функцию f). Основной задачей в дальнейшем будет нахождение оценок величин $L_c(f, N, k)$ и $L_d(f, N, k)$ при различных f, N и k.

Отметим, что для любых f, N и k выполняется соотношение $L_d(f,N,k) \geqslant L_c(f,N,k)$, поскольку любой диагностический тест, очевидно, является проверяющим.

В качестве тривиального диагностического (и проверяющего) теста длины N, очевидно, всегда можно взять множество из N схем, каждая из которых представляет собой один из заданных функциональных элементов. Отсюда $L_c(f,N,k) \leqslant N$ и $L_d(f,N,k) \leqslant N$ для любых f, N и k.

Формулировки основных теорем

Теорема 1. *Для любых f, N и k выполняются неравенства $L_c(f,N,k) \geqslant k$, $L_d(f,N,k) \geqslant k$.*

Теорема 2. *Пусть булева функция $f(x_1,\ldots,x_n)$ не совпадает ни с одной из функций $x_1\&\ldots\&x_n$, $x_1 \vee \ldots \vee x_n$, $\overline{x_1}$, и выполнено условие $8k + \frac{5}{2} \leqslant \sqrt{N}$. Тогда:*

1) $L_c(f,N,k) \leqslant 2k+1$,

2) если функция $f(x_1,\ldots,x_n)$ нелинейна, то $L_d(f,N,k) \leqslant 2k+1$.

Теорема 3. *Пусть $f \in \{x_1\&\ldots\&x_n,\ x_1 \vee \ldots \vee x_n, \overline{x_1}\}$, $n \geqslant 1$. Тогда для любых N и k выполняются неравенства $L_c(f,N,k) \geqslant c(f)k(\log_2 N - \log_2 k)$, $L_d(f,N,k) \geqslant c(f)k(\log_2 N - \log_2 k)$, где*

$$c(f) = \begin{cases} 1 \text{ для } f \in \{x_1\&\ldots\&x_n,\ x_1 \vee \ldots \vee x_n\}, \\ \log_3 2 \text{ для } f = \overline{x_1}. \end{cases}$$

Замечание. В случае $n = 1$ и $f \in \{x_1\&\ldots\&x_n,\ x_1 \vee \ldots \vee x_n\}$ получаем, что $f(x_1) = x_1$, т. е. элементы $E_1,\ldots,E_N$ реализуют в исправном состоянии тождественную функцию. Хотя такие элементы, как правило, не рассматриваются, теоретически такой случай возможен.

Заключение

Отметим некоторые очевидные следствия из теорем 1-3.

1. Теоремы 1 и 2 позволяют утверждать, что при условиях $8k + \frac{5}{2} \leqslant \sqrt{N}$ и $f \notin \{x_1\&\ldots\&x_n,\ x_1 \vee \ldots \vee x_n, \overline{x_1}\}$ значения функций $L_c(f,N,k)$ и (при дополнительном условии, что f нелинейна) $L_d(f,N,k)$ содержатся в отрезке $[k; 2k+1]$.

2. Пусть k фиксировано, $N > (8k+\frac{5}{2})^2$ и возрастает. Теоремы 2 и 3 позволяют установить, что при $f \notin \{x_1\&\ldots\&x_n,\ x_1 \vee \ldots \vee x_n, \overline{x_1}\}$ функции $L_c(f,N,k)$ и — при дополнительном условии, что f нелинейна — $L_d(f,N,k)$ ограничены сверху величиной $2k+1$; в то же время, при $f \in \{x_1\&\ldots\&x_n,\ x_1 \vee \ldots \vee x_n, \overline{x_1}\}$ данные функции ограничены снизу величиной $c(f)k(\log_2 N - \log_2 k)$, растущей с ростом N (при фиксированном k). Таким образом, видна принципиальная разница в поведении функций $L_c(f,N,k)$ и $L_d(f,N,k)$ в двух случаях:

1) когда f — либо конъюнкция, либо дизъюнкция, либо инверсия,

2) когда f — любая другая неконстантная булева функция (за исключением, быть может, поведения величины $L_d(f,N,k)$ в случае, когда f — линейная функция от двух или более переменных).

Автор выражает глубокую благодарность своему научному руководителю профессору Н. П. Редькину за постановку задачи и внимание к работе.

Литература

[1] *Лупанов О. Б.* Асимптотические оценки сложности управляющих систем. — М.: Изд-во МГУ, 1984. — 137 с.

[2] *Редькин Н. П.* Надёжность и диагностика схем. — М.: Изд-во МГУ, 1992. — 192 с.

О синтезе контактных схем, допускающих проверяющие тесты линейной длины

Д. С. Романов

romanov@cs.msu.ru

Факультет ВМК МГУ имени М. В. Ломоносова, Москва

Пусть $F(\tilde{x}^n) = (f_1(\tilde{x}^n), \ldots, f_t(\tilde{x}^n))$ — произвольная система булевых функций, отличных от констант и зависящих от переменных $x_1, x_2, \ldots, x_n$ ($F \in P_2^t(n)$), а S — $(1,t)$-контактная схема (т.е. контактная схема с одним входным полюсом и t выходными полюсами), реализующая систему F (все не введенные в работе определения можно найти в книгах [1], [2]). Пусть на схему S действует источник неисправностей U, способный вызывать размыкания и замыкания контактов. Схема S называется *тестопригодной (относительно обнаружения неисправностей для источника неисправностей U)* тогда и только тогда, когда при любой неисправности схемы S, вызванной действием на нее источника U, полученная вследствие этой неисправности схема S' реализует систему функций $F'(\tilde{x}^n) = (f'_1(\tilde{x}^n), \ldots, f'_t(\tilde{x}^n))$, не равную F ($F \neq F'$, т.е. $(f_1(\tilde{x}^n), \ldots, f_t(\tilde{x}^n)) \neq (f'_1(\tilde{x}^n), \ldots, f'_t(\tilde{x}^n))$). Обозначим через W множество всех попарно неравных систем функций, каждая из которых может быть реализована в результате действия на схему S источника неисправностей U (в частности, $F \in W$). Множество T наборов значений переменных $x_1, x_2, \ldots, x_n$ называется *проверяющим тестом* для схемы S относительно источника неисправностей U тогда и только тогда, когда для любой системы функций $F' \in W$ такой, что F' не равна F, найдется набор $\tilde{\alpha}$ из T, для которого выполнено неравенство $F'(\tilde{\alpha}) \neq F(\tilde{\alpha})$. Количество различных наборов в тесте T называется его *длиной* и обозначается через $l(T)$ или через $|T|$. Тест минимальной длины называется *минимальным*. Тест называется полным, если источник неисправностей может повреждать произвольное количество контактов в схеме, и единичным, если в схеме может быть поврежден не более чем один контакт. Обозначим через $D_U(S)$ длину минимального проверяющего теста относительно источника неисправностей U в схеме S, через $D_U(F(\tilde{x}^n))$ — минимум величины $D_U(S)$ по всем тестопригодным (относительно обнаружения неисправностей для источника неисправностей U) реализующим $F(\tilde{x}^n)$ контактным схемам S. Через $D_U^t(n)$ обозначим *функцию Шеннона длины проверяющего теста относительно источника неисправностей U*, т. е. функцию $D_U(n) = \max\limits_{F(\tilde{x}^n) \in P_2^t(n)} D_U(F(\tilde{x}^n))$.

Первые оценки функций Шеннона появились уже в пионерской работе С. В. Яблонского и И. А. Чегис [3]. Фактически, было доказано, что функция Шеннона длины единичного диагностического (а, значит, и проверяющего) теста для двухполюсных контактных схем есть $O(\frac{2^n}{n})$. Х. А. Мадатяном [4] установлено, что точное значение функции Шеннона длины полного диагностического теста для контактных схем равно 2^n. Н. П. Редькиным доказано, что полный проверяющий тест для контактных схем не обязан содержать все наборы и что соответствующая функция Шеннона не превосходит $\frac{15}{16} \cdot 2^n$ [5]. Им же в [6] было установлено, что функция Шеннона длины полного диагностического теста размыкания есть $O(2^{n/2})$, а также была получена нетривиальная верхняя оценка $O(2^{2n \log_2 n/(1+2 \log_2 n)})$ функции Шеннона длины полного диагностического теста замыкания. Как видно, приведенные оценки функций Шеннона являются быстрорастущими с ростом n. Отметим, что введение дополнительных переменных и дополнительных выходных полюсов (контрольных точек), а также использование надфункций (т.е. функций, для которых данная функция является подфункцией) — это классические способы понижения трудоемкости тестирования. В настоящей работе предлагаются методы построения легкотестируемых контактных схем, допускающих проверяющие тесты некоторых типов, имеющие линейную по числу переменных длину.

Именно, верны следующие теоремы.

Теорема 1. *Пусть $f(\tilde{x}^n)$ — произвольная булева функция, отличная от константы. Тогда систему функций $(f,\ \bar{f})$ можно реализовать тестопригодной контактной схемой, допускающей единичный проверяющий тест длины $2n+2$.*

Теорема 2. *Пусть $f(\tilde{x}^n)$ — произвольная булева функция, отличная от константы. Тогда функцию $\hat{f}(\tilde{x}^{n+1}) = f(\tilde{x}^n) \oplus x_{n+1}$ можно реализовать тестопригодной контактной схемой, допускающей единичный проверяющий тест длины $4n + 4$.*

Теорема 3. *Пусть $f(\tilde{x}^n)$ — произвольная булева функция, отличная от константы. Тогда систему функций $(f,\ \bar{f})$ можно реализовать тестопригодной контактной схемой, допускающей*

а) полный проверяющий тест размыкания длины $2n + 2$,

б) полный проверяющий тест замыкания длины $2n + 2$.

Теорема 4. *Пусть $f(\tilde{x}^n)$ — произвольная булева функция, отличная от константы. Тогда функцию $\hat{f}(\tilde{x}^{n+1}) = f(\tilde{x}^n) \oplus x_{n+1}$ можно реализовать тестопригодной контактной схемой, допускающей*

а) полный проверяющий тест размыкания длины $4n + 4$,

б) полный проверяющий тест замыкания длины $4n + 4$.

Идеи доказательств базируются на разложении произвольной отличной от константы булевой функции $f(\tilde{x}^n)$ в полином Жегалкина и на построении на этой основе схемы, обобщающей схему Кардо счетчика четности следующим образом. Для каждого отличного от константы слагаемого K полинома Жегалкина строится (как в схеме Кардо для переменной) свой собственный блок с двумя входными («левыми») и двумя выходными («правыми») полюсами (блоки соединяются как в схеме Кардо: выходные полюсы предыдущего блока отождествляются с входными блоками следующего). Конъюнкция, пред-

ставляющая собой рассматриваемое слагаемое K, реализуется цепочкой контактов, и две копии такой цепочки соединяют левый верхний полюс блока с правым нижним, а левый нижний — с правым верхним. Отрицание слагаемого K реализуется как параллельное соединение (пучок) контактов, которым приписаны отрицания входящих в K переменных, и две копии такого пучка соединяют левый верхний полюс блока с правым верхним, а левый нижний — с правым нижним. При этом первый блок — это (1,2)-контактное дерево для фиктивной переменной x_0, последний блок для теорем 1, 3 — это стандартный блок схемы Кардо для переменной x_0; в теоремах 2, 4 к последнему блоку присоединяется (2,1)-контактное дерево для переменной x_{n+1}.

Проверяющие тесты строятся так: наборы значений переменных $x_1, x_2, \ldots, x_n$ — это наборы $(1, 1, \ldots, 1)$, $(0, 1, \ldots, 1)$, $(1, 0, 1, \ldots, 1)$, $\ldots$, $(1, \ldots, 1, 0)$, а переменные x_0 и x_{n+1} (x_{n+1} — только для теорем 2, 4) при этом принимают всевозможные значения. Легко показать, что указанные множества наборов образуют для теорем 1, 2 единичные проверяющие тесты. А поскольку единичный проверяющий тест замыкания (размыкания) является и полным проверяющим тестом замыкания (соответственно. размыкания), то справедливы и теоремы 3, 4.

Автор выражает глубокую благодарность профессору Сергею Андреевичу Ложкину за обсуждение работы и ценные замечания.

Работа поддержана РФФИ (проекты № 12-01-00964-а и № 13-01-00958-а).

Литература

[1] *Редькин Н. П.* Надежность и диагностика схем. — М: Изд-во МГУ, 1992. — 192 с.

[2] *Ложкин С. А.* Лекции по основам кибернетики. — М: МАКС Пресс, 2004. — 256 с.

[3] *Чегис И. А., Яблонский С. В.* Логические способы контроля работы электрических схем // Труды МИАН СССР. — 1958. — Т. LI. — С. 270–360.

[4] *Мадатян Х. А.* Полный тест для бесповторных контактных схем // Проблемы кибернетики. — Вып. 23. М.: Наука, 1970. — С. 103–118.

[5] *Редькин Н. П.* О полных проверяющих тестах для контактных схем // Методы дискретного анализа в исследовании экстремальных структур. — Вып. 39. Новосибирск: Изд-во ИМ СО АН СССР, 1983. — С. 80–87.

[6] *Редькин Н. П.* О проверяющих тестах замыкания и размыкания // Методы дискретного анализа в оптимизации управляющих систем. — Вып. 40. Новосибирск: Изд-во ИМ СО АН СССР, 1983. — С. 87–99.

Теорема о статической полноте СУБД DIM

В. С. Рублев

roublev@mail.ru

Ярославский госуниверситет, Ярославль

Вводятся формализация OD-модели, при помощи которой может быть заданы данные произвольной базы данных, формализация схемы классов DIM и формализация статического описания OD-модели схемой

классов DIM. Обосновывается алгоритм сведения произвольной OD-модели к схеме классов DIM.

Объектная СУБД DIM

Недостатки имеющихся моделей СУБД позволили поставить задачу о создании новой технологии СУБД, которая использует достоинства имеющихся технологий темпоральной, реляционной, объектно-ориентированной и объектно-реляционной [1]– [4]. Новый объектный подход к созданию СУБД [5] предполагает не только изменение данных объектов, но и возможность изменения типов объектов, т. е. схемы базы данных. В этом подходе мы выделили 6 базовых отношений объектов: *наследования, включения, внутреннего наследования, внутреннего включения, истории и взаимодействия* и назвали эту СУБД *динамической информационной моделью* (DIM).

Эволюция DIM привела к необходимости расширения отношения включения: помимо видов простого и функционального включения, позволяющих описать отношение только для двух объектов, введен вид мультивключения, позволяющий описывать это отношение для большего числа объектов. Это заставило переработать теорему о полноте DIM [5], которая показывает как произвольную *дискретную детерминированную систему* адекватно описать в БД DIM. Формализация дискретной детерминированной модели привела к построению *объектно-динамической модели* (OD-модели), а для адекватного описания ее данных в DIM введены формализация схемы классов DIM и формализация статического описания OD-модели схемой классов DIM.

Объектно-динамическая модель

OD-модель описывается как девятка

$$(O, A, V, L_p, L_o, L_f, V_{L_f}, F, T),$$

где

O – конечное множество объектов,

A – конечное множество свойств объектов с типами этих свойств (элемент этого множества пара (a, V^a) – свойство, тип свойства),

$V = \bigcup_o \{V_o\}$ – множество кортежей значений свойств объектов,

$L_p = \bigcup_{j \in L_p} \{l_j^p = \{o, o1\}\}$ – множество простых связей объектов,

L_o – множество объектов-связей $(O \cap L_o = \varnothing)$,

$L_f = \bigcup_{j \in L_f} \{(l_j^f, o_l^f \in L_o)\}$ – множество функциональных связей объектов,

$V_{L_f} = \bigcup_{j \in L_f} \{V_{o_l^j}\}$ – множество кортежей значений атрибутов объектов-связей o_l^j функциональных связей L_f,

F – конечное множество алгоритмических процедур изменения значений свойств объектов и изменения объектов,

T – дискретная шкала времени[1].

[1] Подробное рассмотрение динамических свойств OD-модели опущено, так как для данной работы не требуется

Класс DIM и схема классов

Подмножество объектов $O_c \subseteq O$, имеющих одинаковые свойства-атрибуты и одинаковое *поведение* (законы взаимодействия с другими объектами определяются одинаковым подмножеством множества F алгоритмических процедур этой модели), мы называем *классом с* объектов, а совокупность всех классов – множеством C классов.

Тип объекта DIM определяется множеством параметров классов и множеством взаимодействий, для каждого из которых одну из *ролей* взаимодействия выполняет класс объекта или его родительские классы. Множество свойств типа определяется:

1) *параметрами класса* объекта;
2) *параметрами родительских классов* для класса объекта, если таковые имеются;
3) *связями* со всеми классами, с которыми имеется отношение *включения* для класса объекта и его родительских классов.

Под *схемой классов DIM* мы будем понимать конечное множество C классов DIM с введенными отношениями наследования P и включения I классов и их объектов.

$$S = (C^s, O^s, A^s, V^s = \bigcup_o \{V_o^s\}, P, p, I_c^p, I_o^p, I_c^k, I_o^k, O_l^s, V_l^s),$$

где

C^s – множество классов схемы S,

O^s – множество основных объектов классов DIM схемы S, не являющихся объектами-связями из O_l^s,

A^s – множество свойств классов C^s (элемент множества – пара (a, V^a) свойство - тип свойства) схемы S,

$V^s = \bigcup_{o \in O^s} V_o^s$ – множество кортежей значений свойств основных объектов схемы S,

P – отношение наследования классов схемы S,

p – отношение наследования объектов схемы S,

I_c^p – отношение простого включения классов схемы S,

I_o^p – отношение простого включения объектов схемы S,

I_c^k – отношение качественного включения классов схемы S,

I_o^k – отношение качественного включения объектов схемы S,

O_l^s – множество объектов-связей включения схемы S $(O_l^s \cap O^s = \varnothing)$,

V_l^s – множество значений атрибутов кортежей объектов-связей включения схемы S.

Статическая полнота DIM

Будем говорить, что некоторая *схема S классов DIM статически описывает некоторую OD-модель*

$OD = (O^d, A^d, V^d = \bigcup_{o \in O^d} V_o^d, L_p = \bigcup_o L_o^p, L_o, L_f = \bigcup_o L_o^f, V^d_{L_o^f}, F, T))$

в некоторый момент времени $t \in T$ с ее конечными множествами (на этот момент) объектов O^d, свойств-атрибутов A^d, значений свойств-атрибутов V_o^d,

свойств-связей L_o^p, L_o^f для каждого объекта $o \in O^d$ и кортежа значений $V^d_{o_l^j}$ каждого объекта-связи из L_o, *если существует отображение* G *OD-модели в схему* S *классов DIM, при котором*

1. $\forall\ o \in O^d:\quad G(o) \in O^s,\quad A^s_{G(o)} = A^d_o,$
2. $\forall\ a \in A^d,\ o \in O^d:\ v_{G(a),G(o)}(t) = v^a_o(t),$
3. $\forall\ l_j = \{o, o_k\} \in L_p:\ (G(o), G(o_k)) \in I^p_{G(o)} \vee (G(o_k), G(o)) \in I^p_{G(o_k)},$
4. $\forall\ (l^f_j, o^j_l) \in L_f,\ \exists\{o, o1\} \subseteq l^f_j \in L_f,\ o^j_l \in L_o:$

$$((G(o), G(o1), G(o^j_l)) \in I^k_{G(o)}\ \vee\ (G(o1), G(o), G(o^j_l)) \in I^k_{G(o1)})$$
$$\wedge\ (p(G(o^j_l)) = G(l^f_j) \setminus \{G(o), G(o1)\},\ A^s_{c(G(o^j_l))} = G(A^d_{o^j_l}),$$
$$\forall\ a_{o^j_l} \in A^d_{o^j_l},\ v_{G(a),G(o^j_l)} \in V^s_{G(o^j_l)}).$$

Схема классов DIM находится в нормальной форме, если она отвечает ограничениям *определенности*[2] и *однозначности*[3] .

Теорема 1. *Произвольная OD-модель OD для произвольного момента* $t \in T$ *в ней может быть статически описана с помощью некоторой схемы* S *классов DIM, находящейся в нормальной форме.*

Литература

[1] *Codd E. F.* A relational model for large shared data banks // Comm.ACM. — 1970.

[2] *Аткинсон М. и др.* Манифест систем объектно-ориентированных баз данных // СУБД. — 1995.

[3] *Грин Г. Р.* Архитектуры ООСУБД. Анализ реализаций (перевод С.Д.Кузнецова) — 2007.

[4] Б.Б. Костенко, С.Д. Кузнецов История и актуальные проблемы темпоральных баз данных. сайт *www.citforum.ru* (точный адрес: *http* : *//www.citforum.ru/database/articles/temporal*), 2007.

[5] *Писаренко Д. С., Рублев В. С.* Объектная СУБД Динамическая информационная модель и ее основные концепции. // Моделирование и анализ информационных систем, т.16, № 1. –Ярославль: ЯрГУ, 2009. – С. 62 – 91.

[2] ациклический граф наследования любого класса не должен содержать класса, который в своем ациклическом графе включения содержит исходный класс

[3] любой атрибут объектов системы должен быть параметром только одного класса системы

Исследование задачи регулярной реализуемости для контекстно-свободных языков

А. А. Рубцов

rubtsov99@gmail.com

Московский Физико-Технический Институт (государственный университет), Москва

Введение

Задачами регулярной реализуемости называются задачи, состоящие в проверке непустоты пересечения регулярного языка, поданного на вход задаче, с фиксированным языком-фильтром, который является параметром задачи. Задачи регулярной реализуемости разделяют в зависимости от способа описания регулярного языка. В случае когда вход задан описанием детерминированного конечного автомата, задача называется задачей (детерминированной) регулярной реализуемости $\mathrm{RR}(F)$, в случае когда вход задан описанием недетерминированного конечного автомата, задача называется задачей недетерминированной регулярной реализуемости $\mathrm{RR}^n(F)$.

Основная цель изучения задач регулярной реализуемости состоит в определении их алгоритмической сложности. В зависимости от фильтра F задачи $\mathrm{RR}(F)$ и $\mathrm{RR}^n(F)$ оказываются полны в таких классах сложности как **NL**, **P**, **NP**, **PSPACE** [1, 2]. В [3] представлен ряд классов сложности, в которых полна детерминированная версия задачи. Чёткое разделение сложности детерминированной и недетерминированной версии задач в зависимости от фильтра остаётся открытым вопросом, немногое из известного – в случае когда фильтр F является некоторым регулярным языком, например 0^*, задача $\mathrm{RR}(F)$ полна в классе **L** [4], в то время как задача $\mathrm{RR}^n(F)$ полна в классе **NL** для любого регулярного языка.

Вопрос об изучении сложности задач регулярной реализуемости в случае контекстно-свободных фильтров является естественным. Сосредоточимся на задаче $\mathrm{RR}^n(F)$ – результаты данной работы переносятся и на детерминированную версию задачи либо напрямую, либо путём замены используемой сводимости на фильтрах на более ограниченную. Поскольку сложность задачи $\mathrm{RR}^n(F)$ ограничена снизу классом **NL** и, как мы покажем далее, любая задача $\mathrm{RR}^n(F)$ с контекстно-свободным фильтром лежит в классе **P**, то в качестве сводимости мы будем использовать сводимость на логарифмической памяти $\leqslant_{\log}$.

Вспомогательные сведения

Введём вспомогательные понятия из области КС-языков, а также зафиксируем основные используемые нами КС-языки. Будем говорить, что язык $L' \subseteq B^*$ рационально доминирует язык $L \subseteq A^*$, если существует такое рациональное отношение $R \subseteq A^* \times B^*$, что $L = \{u \in A^* \mid \exists v \in L'\ (v, u) \in R\}$. Будем обозначать отношение «рационально доминирует» как $\leqslant_{\mathrm{rat}}$. Будем называть

рациональным конусом класс языков, замкнутых относительно рационального доминирования и будем обозначать $\mathcal{T}(L)$ наименьший рациональный конус, содержащий язык L. Будем говорить, что язык L является генератором рационального конуса $\mathcal{T}$, если $\mathcal{T}(L) = \mathcal{T}$. Основной КС-язык, который мы будем использовать, это язык Дика с n типами скобок, который задан грамматикой

$$D_n^* = \langle S \to TS + \varepsilon\,;\, T \to a_1 S \bar{a}_1 + \ldots + a_n S \bar{a}_n \rangle.$$

Утверждение 1. *Если* $F_1 \leqslant_{\text{rat}} F_2$, *то* $\text{RR}(F_1) \leqslant_{\log} \text{RR}(F_2)$.

Таким образом сложность недетерминированной задачи регулярной реализуемости для фильтра F является верхней оценкой сложности задачи недетерминированной регулярной реализуемости с фильтром из рационального конуса $\mathcal{T}(F)$.

Мы используем следующую формулировку теоремы Хомского-Щютценберже.

Теорема 1 (Хомский, Щютценберже). $\mathcal{T}(D_2^*) = \mathsf{CFL}$.

Задача регулярной реализуемости $\text{RR}(D_2^*)$ сводится к **P**-полной задаче о проверке непустоты языка, порождаемого КС-грамматикой, отсюда следует

Теорема 2. *Задача недетерминированной регулярной реализуемости для КС-фильтров лежит в классе* **P**. *Если КС-язык* L *является генератором конуса* CFL, *то задача* $\text{RR}^n(F)$ *полна в классе* **P**.

Одной из сложностных характеристик языков является рациональный индекс. *Рациональным индексом* $\rho_L(n)$ языка L называется максимальная длина самого короткого слова из пересечения языка L с регулярным языком, распознаваемым автоматом $\mathcal{A}$ с n состояниями, причём язык $L(\mathcal{A}) \cap L$ не пуст. Формально

$$\rho_L(n) = \max_{\mathcal{A}:|Q_{\mathcal{A}}|=n,\ L(\mathcal{A})\cap L\neq\varnothing} \min\{|u| \mid u \in L(\mathcal{A}) \cap L\}.$$

Классификация по рациональному индексу согласуется с рациональным доминированием, что следует из следующей теоремы.

Теорема 3 (Boasson, Courcelle, Nivat, 1981, [5]). *Если* $L' \leqslant_{\text{rat}} L$ *тогда существует такая константа* c, *что*

$$\rho_{L'}(n) \leqslant cn(\rho_L(cn) + 1).$$

Отсюда и из следующей теоремы следует ограничение на рациональный индекс для КС-языков.

Теорема 4 (Pierre, 1992, [6]). *Рациональный индекс генератора рационального конуса* CFL *лежит в* $\exp(\Theta(n^2/\log n))$.

Основные результаты

Утверждение 2. *Пусть* L_c *– КС-язык, распознаваемый автоматом со счётчиком, тогда задача регулярной реализуемости* $\text{RR}^n(F)$ *лежит в классе* **NL**.

Мы благодарим A. Yakaryilmaz за указание на верность этого результата.

Известные примеры КС-языков, для которых задача регулярной реализуемости лежит в **NL**, таких как D_1^* или язык палиндромов, имеют малый (полиномиальный) рациональный индекс, в то время как известные примеры КС-языков, для которых задача **P**-полна, имеют большой (субэкспоненциальный) рациональный индекс. Следующая теорема даёт верхнюю оценку на сложность задачи в случае когда КС-язык имеет полиномиальный рациональный индекс.

Теорема 5. *КС-языки с полиномиально-ограниченным рациональным индексом лежат в классе* $\mathbf{NSPACE}(\log^2 n)$.

Работа выполнена при поддержке РФФИ, проект №14–01–00641.

Литература

[1] Anderson T., Loftus J., Rampersad N., Santean N., Shallit J. Detecting palindromes, patterns and borders in regular languages. Information and Computation. Vol. 207, 2009. P. 1096–1118.

[2] Vyalyi M.N. On regular realizability problems. Problems of Information Transmission. Vol. 47, issue 4, 2011. P. 342–352.

[3] Vyalyi M.N. Universality of regular realizability problems. CSR 2013, LNCS 7913, 2013. P. 271–282.

[4] *Рубцов А.А.* О регулярных языках-подсказках в модели обобщенных недетерминированных автоматов. Математические модели и задачи управления: сборник научных трудов. — М.: МФТИ,2011. С. 61-67.

[5] L. Boasson, B. Courcelle, M. Nivat. The rational index, a complexity measure for languages. SIAM J. Comput. Vol. 10(2), 1981. P. 284–296.

[6] L. Pierre. Rational indexes of generators of the cone of context-free languages. TCS. Vol. 95, 1992. P. 279–305.

[7] J. Berstel. Transductions and context-free languages. Teubner Verlag, 1979.

[8] J. Berstel, L. Boasson, Context-Free Languages, in: J. van Leewen (ed.), Handbook of Theoretical Computer Science Vol. B, Elsevier, 1990, 59–102.

[9] L. Boasson, Non-générateurs algébriques et substitution, RAIRO Informatique théorique 19 (1985), 125–136.

[10] Raymond Greenlaw, H. James Hoover, and Larry Ruzzo. Limits to Parallel Computation: P-completeness Theory. Oxford Univ. Press, 1995.

[11] Lewis P.M., Stearns R.E., Hartmanis J. Memory bounds for recognition of context-free and context-sensitive languages. Switching Circuit Theory and Logical Design, 1965. SWCT 1965. P. 191-202.

[12] L. Pierre, J.M. Farinone. Rational index of Context-free languages with rational index in $\Theta(n^\gamma)$ for algebraic numbers γ. Informatique théorique et applications. Tome 24 (3), 1990. P. 275–322.

[13] A. Yakaryilmaz. One-counter verifiers for decidable languages. arXiv:1207.3880, 2012.

О множествах, свободных от нулей, в группе Z_n

А. А. Сапоженко

sapozhenko@mail.ru

Факультет ВМК МГУ имени М. В. Ломоносова, Москва

Множество M элементов группы Z_n называется *свободным от нулей* (МСН), если оно не содержит троек, удовлетворяющих условию $a+b+c=0 \pmod n$. Задача о числе МСН относится к классу комбинаторных задач о числе множеств с запрещенными подмножествами. Классической задачей такого типа является известная проблема Камерона-Эрдёша [1] о числе $s(n)$ подмножеств натуральных чисел из отрезка $[1,n]$, *свободных от сумм* (МСС). Проблема заключается в подсчете числа множеств (называемых МСС), не содержащих троек (a,b,c), связанных соотношением $a+b=c$. Камерон и Эрдёш предположили, что $s(n)=O(2^{n/2})$. Эта гипотеза была доказана независимо в работах [2] и [3]. В данной статье решается задача о числе $ZF(Z_n)$ множеств, свободных от нулей, в группе Z_n вычетов по модулю n. Пусть $ZF(Z_n)$ — число МСН в группе Z_n. Основным утверждением работы является следующая теорема.

Теорема 1. *Для любого $\alpha \in \{-1,0,1\}$ существует константа $c_a>0$ такая, что для всякого достаточно большого $n=3k+\alpha \pmod 3$ выполнено асимптотическое равенство*

$$ZF(Z_n) \sim c_\alpha \varphi(n) \cdot 2^{n/3},$$

где $\varphi(n)$ — функция Эйлера.

Положим $B_n=[-t, n/3+t]$, $C_n=[-t+n/3, 2n/3+t]$, $D_n=[-t+2n/3, n+t]$, где $[a,b]$ — множество целых чисел x, таких, что $a \leqslant x \leqslant b$. Пусть d — натуральное число, и A — множество натуральных чисел. Положим $dA=\{bd : b \in A\}$. Множество dA называется *d-расширением* множества A. Пусть $ZF(A)$ — число МСН, содержащихся во множестве A натуральных чисел. Основным при оценке $ZF(A)$ следующее утверждение.

Лемма 2. *Для всякого достаточно большого n и $t=\log n$ существует натуральное d, такое что*

$$ZF(Z_n) \sim ZF(dB_n)+ZF(dC_n)+ZF(dD_n). \qquad (*)$$

Величина каждого из слагаемых правой части асимптотического равенства $(*)$ оценивается аналогично тому, как это делалось в статье [4] при оценке числа $|S(Z_n)|$ МСС в Z_n.

Работа поддержана РФФИ (проект № 13-01-00958-а).

Литература

[1] *Cameron P., Erdős P.* On the number of integers with various properties // in R. A. Mollin (ed). Number Theory: Proc. First Conf. Can. Number Th. Ass., Banff, de Gruyter. — 1990. — P. 61–79.

[2] *Сапоженко А. А.* Гипотеза Камерона-Эрдёша // ДАН. — 2003. — Т. 393, № 6. — С. 749–752.

[3] *Green B.* The Cameron–Erdős Conjecture // Bull. London Math. Soc. — 2004. — V. 36 (6). — С. 769–778.

[4] *Сапоженко А. А.* Решение проблемы Камерона-Эрдёша для групп простого порядка // Журнал вычислительной математики и математической физики. — 2009. — Т. 49, № 8. — С. 1–7.

Асимптотика логарифма числа множеств, свободных от решений линейных, в абелевой группе

В. Г. Саргсян

vahe_sargsyan@ymail.com

Московский государственный университет им. М.В. Ломоносова, факультет вычислительной математики и кибернетики, кафедра математической кибернетики, Москва

Подмножество A элементов группы G называется (k,l)-свободным от сумм, если уравнение $x_1 + \cdots + x_k = y_1 + \cdots + y_l$ не имеет рещений в множестве A. Получена асимптотика логарифма числа множеств, (k,l)-свободных от сумм, в абелевой группе.

Введение

Пусть G — множество с определенной на нем операцией сложения, а k и l — неотрицательные целые числа, удовлетворяющие условию $k + l \geqslant 3$.. Подмножество $A \subseteq G$ называется (k,l)-*свободным от сумм* ((k,l)-МСС), если уравнение

$$x_1 + \ldots + x_k = y_1 + \ldots + y_l \tag{1}$$

не имеет решений в множестве A. Семейство всех (k,l)-МСС в G обозначим через $SF_{k,l}(G)$.

В данной работе с исползованием методов [1] получена асимптотика логарифма числа (k,l)-МСС для произвольной абелевой группы. В частности, доказана

Теорема 1. *Пусть G — абелева группа порядка n. Тогда справедливо равенство*

$$\log |SF_{k,l}(G)| = \mu_{k,l}(G) + \bar{o}(n),$$

где $\mu_{k,l}(G)$ — максимальная мощность (k,l)-МСС в G.

Определения и вспомогательные утверждения

Пусть G — абелева группа порядка n. *Характером* группы G называется отображение $\gamma : G \to \mathbb{C}$ такое, что для любого $x \in G$ имеет место $|\gamma(x)| = 1$ и $\gamma(x+y) = \gamma(x)\gamma(y)$. Обозначим через Γ множество всех характеров группы G. Заметим, что Γ образует группу с операцией $\gamma_1 * \gamma_2(x) = \gamma_1(x)\gamma_2(x)$.

Пусть $f : G \to \mathbb{R}$. *Преобразованием Фурье* f называется функция $\widehat{f} : G \to \mathbb{C}$, определяемая равенством $\widehat{f}(\gamma) = \sum\limits_{x \in G} f(x)\gamma(x)$.

Для доказательства теоремы 1 используем метод гранулирования. Сущность этого метода состоит в том, что для оценки мощности множества $SF_{k,l}(G)$ строится семейство $\mathcal{F}$, так называемых «гранул» $F \in \mathcal{F}$, такое, что каждый элемент множества $SF_{k,l}(G)$ содержится в некоторой грануле F из построенного семейства $\mathcal{F}$, при этом $\log|\mathcal{F}| = \overline{o}(n)$, и в каждой грануле $F \in \mathcal{F}$ есть $\overline{o}(n^{k+l-1})$ решений уравнения (1). Будем рассматривать два типа «гранулы».

L-гранулой типа смежного класса называется объединение смежных классов группы G по некоторой подгруппе порядка не меньше L.

Пусть L — целое число и $d \in G$, причем $ord(d) \geqslant L$, где $ord(d)$ — порядок элемента d. Рассмотрим подгруппу G, порожденную элементом d, и разобьём каждый её смежный класс на $\lfloor ord(d)/L \rfloor$ прогрессий вида $\{x + id \mid 0 \leqslant i \leqslant L-1\}$ и одно «остаточное» множество мощности менее L. Для каждого $d \in G$ фиксируем одно такое разбиение. Объединение полученных прогрессий называется *L-гранулой типа прогрессии*.

Естественно, что если в множестве «мало» решений уравнения 1, то в нем существует подмножество, свободное от решений уравнения 1, с «большой» мощностью. Доказательство этого факта можно найти в работе [2].

Теорема 2. *Пусть $m \geqslant 3$, $A_1, \ldots, A_m$ — такие подмножества абелевой группы G порядка n, что существует $\overline{o}(n^{m-1})$ решений уравнения $x_1 + \ldots + x_m = 0$ при $x_i \in A_i$, $i = 1, \ldots, m$. Тогда существуют подмножества $A'_1, \ldots, A'_m$ такие, что $A'_i \subseteq A_i$, $|A_i \setminus A'_i| = \overline{o}(n)$, и решений уравнения $x_1 + \ldots + x_m = 0$ при $x_i \in A'_i$ нет.*

Гранулирование

Сущность следующей леммы состоит в том, что для каждого $A \in SF_{k,l}(G)$ строится «подходящая» гранула.

Лемма 3 (Гранулирование). *Пусть G — абелева группа порядка n, и $A \in SF_{k,l}(G)$, а $\varepsilon \in \left(0, \frac{1}{2}\right)$, L и L' — положительные числа, удовлетворяющие неравенству*

$$n > L' \left(4L/\varepsilon\right)^{4^{2(k+l)+1}(k+l)^2 \varepsilon^{-2(k+l+1)}}.$$

Тогда существует подмножество $A' \subseteq G$ такое, что

(i) *A' — либо L-гранула типа прогрессии, либо L'-гранула типа смежного класса;*

(ii) $\left|A \setminus A'\right| \leqslant \varepsilon n$;

(iii) *A' содержит не более εn^{k+l-1} решений уравнения* (1).

Заметим, что если $\mathcal{F}$ состоит из множеств, являющихся объединением L-гранулы и некоторого подмножества G мощности $\overline{o}(n)$, то имеет место равенство $\log|\mathcal{F}| = \overline{o}(n)$. Построим эти множества так, чтобы $\mathcal{F}$ удовлетворяло остальным двум требованиям.

В следующей теореме доказывается существование семейства гранул.

Теорема 4. *Пусть G — абелева группа порядка n, и n достаточно велико. Тогда существует семейство $\mathcal{F}$ подмножеств группы G, удовлетворяющее условиям*

(i) $\log|\mathcal{F}| \leqslant 2n(k+l-1)^{-1/2}(\log n)^{-(4(k+l)+6)^{-1}}$;

(ii) *для каждого $A \in SF_{k,l}(G)$ существует $F \in \mathcal{F}$ такое, что $A \subseteq F$;*

(iii) *всякое $F \in \mathcal{F}$ содержит не более $n^{k+l-1}(\log n)^{-(2(k+l)+3)^{-1}}$ решений уравнения* (1).

Доказательство теоремы 1

Пусть G — абелева группа порядка n. По теореме 4 существует семейство $\mathcal{F}$ подмножеств (гранул) группы G, удовлетворяющее условиям (i)-(iii). Пусть $F \in \mathcal{F}$. Фиксируя F и применяя теорему 2 при $A_1 = \ldots = A_k = F$ и $A_{k+1} = \ldots = A_{k+l} = -F$, получим, что существует $F' \subseteq F$ такое, что $|F \setminus F'| = \overline{o}(n)$ и $F' \in SF_{k,l}(G)$. Отсюда следует, что $|F| \leqslant \mu_{k,l}(G) + \overline{o}(n)$, где $\mu_{k,l}(G)$ — максимальная мощность множества из $SF_{k,l}(G)$. Из того, что $\log|\mathcal{F}| = \overline{o}(n)$ (пункт (i) теоремы 4) получим, что количество подмножеств всех множеств семейства $\mathcal{F}$, не превосходит $2^{\mu_{k,l}(G)+\overline{o}(n)}$.

В силу пункта (ii) теоремы 4 все множества из $SF_{k,l}(G)$ являются подмножествами некоторого множества из семейства $\mathcal{F}$. Отсюда следует, что

$$\log|SF_{k,l}(G)| = \mu_{k,l}(G) + \overline{o}(n).$$

Теорема 1 доказана.

Автор выражает признательность профессору А.А. Сапоженко за постановку задачи и внимание к этой работе.

Работа выполнена при поддержке РФФИ, проект № 13-01-00958а.

Литература

[1] *Green B., Ruzsa I.* Sum-free sets in abelian groups // Israel J. Math. — 2005. — V. 147. — P. 157–188.

[2] *Green B.* A Szemeredi-type regularity lemma in abelian groups // GAFA. — 2005. — V. 15, № 2. — P. 340–376.

О мультипликативной сложности некоторых булевых функций

С. Н. Селезнева

selezn@cs.msu.su

МГУ имени М.В. Ломоносова, Москва

В настоящей работе найдено точное значение мультипликативной сложности некоторых булевых функций и некоторых систем булевых функций. *Мультипликативной* (или *конъюнктивной*) сложностью $\mu(f)$ булевой функции $f(x_1, \ldots, x_n)$ называется минимальное число элементов конъюнкции в

схемах из функциональных элементов в базисе $\{x\&y, x \oplus y, 1\}$, которые реализуют функцию f. *Мультипликативной* (или *конъюнктивной*) сложностью $\mu(F)$ системы булевых функций F называется минимальное число элементов конъюнкции в схемах из функциональных элементов в базисе $\{x\&y, x \oplus y, 1\}$, каждая из которые реализует все функции системы F. В [1, 2, 3, 4] изучалась мультипликативная сложность некоторых булевых функций. В [1, 2] было найдено точное значение $\lfloor n/2 \rfloor$ функции Шеннона мультипликативной сложности булевых функций степени два, зависящих от n переменных. В [1] доказано, что мультипликативная сложность произвольной булевой функции всегда не меньше, чем степень этой функции минус единица. В [3] доказано, что мультипликативная сложность произвольной симметрической булевой функции, зависящей от n переменных, не превосходит $n + O(\sqrt{n})$. В [4] найдено точное значение мультипликативной сложности пороговых булевых функций с порогом два.

Пусть $B = \{0, 1\}$. Отображение $f : B^n \to B$ называется булевой функцией, зависящей от n переменных, $n = 0, 1, 2, \ldots$. Каждая булева функция $f(x_1, \ldots, x_n)$ может быть однозначно задана полиномом Жегалкина, т.е. в виде суммы по модулю два монотонных элементарных конъюнкций. *Степенью* $\deg(f)$ булевой функции f назовем максимальное число сомножителей в слагаемых ее полинома Жегалкина. Булева функция f называется *аффинной*, если $\deg(f) \leqslant 1$. Булева функция называется *мульти-аффинной*, если она может быть представлена в виде произведения аффинных функций.

Теорема 1. *Если $n \geqslant 3$, и булева функция $f(x_1, \ldots, x_n)$ может быть представлена в виде $x_1 \ldots x_n \oplus q(x_1, \ldots, x_n)$, где $\deg(q) = 2$, то $\mu(f) = n - 1$.*

Теорема 2. *При всех $n \geqslant 2$ для булевой функции $f(x_1, \ldots, x_n) = x_1 \ldots x_n \vee \bar{x}_1 \ldots \bar{x}_n$ верно, что $\mu(f) = n - 2$.*

Теорема 3. *При всех $n \geqslant 1$ для системы булевых функций $F = \{x_1 \ldots x_n,\ \bar{x}_1 \ldots \bar{x}_n\}$ верно, что $\mu(F) = n - 1$.*

Теорема 4. *Если $n \geqslant 1$, булева функция $f(x_1, \ldots, x_n)$ может быть представлена в виде суммы по модулю два двух мульти-аффинных функций, и $\deg(f) = n$, то $\mu(f) = n - 1$.*

Работа поддержана РФФИ, гранты 12–01–00786-а, 13–01–00684-а, 13–01–00958-а.

Литература

[1] *Schnorr C.P.* The multiplicative complexity of Boolean functions // Proc. 1st Internat. Joint Conf of ISSAC '88 and AAECC-6, Rome (1988). Lecture Notes in Computer Science. **357**. 1989. P. 45–58.

[2] *Mirwald R., Schnorr C.P.* The multiplicative complexity of quadratic boolean forms // Theoretical Computer Science. **102**. 1992. P. 307–328.

[3] *Boyar J., Peralta R., Pochuev D.* On the Multiplicative Complexity of Boolean Functions over the Basis $\{\wedge, \oplus, 1\}$ // Theor. Comp. Sci. **235**. 2000. P. 43–57.

[4] *Краснова Т.И.* О конъюнктивной сложности схем в базисе Жегалкина для одной последовательности булевых функций // Материалы XI Международного

семинара "Дискретная математика и ее приложения". М.: Из-во мех-мат. ф-та МГУ, 2012. С. 138–141.

Порядок функции Шеннона длины функций k-значной логики в классе полиномиальных нормальных форм по модулю k

С. Н. Селезнева, М. А. Башов

selezn@cs.msu.su, max.bashov@gmail.com

МГУ имени М.В. Ломоносова, Москва

Полиномиальные формы являются одним из способов представления булевых и конечнозначных функций. Полиномиальные представления булевых функций находят применения при синтезе схем, в программируемых логических матрицах (ПЛМ) [1, 2]. Эффективность ПЛМ зависит от числа слагаемых в полиномиальной форме, по которой она построена. В ряде работ изучается сложность представления булевых функций и функций k-значной логики полиномиальными формами различных видов [3, 4, 5, 6, 7, 8]. Полиномиальные нормальные формы (п.н.ф.) для булевых функций – это представление булевых функций суммами по модулю два элементарных конъюнкций. Это представление булевых функций аналогично дизъюнктивным нормальным формам (д.н.ф.), только вместо операции дизъюнкции применяется операция сложения по модулю два. Теоретические и экспериментальные исследования показывают, что представление булевых в виде п.н.ф. более эффективно, чем в виде д.н.ф.: и в худшем случае (при росте числа переменных функций), и для функций малого числа переменных п.н.ф. имеют меньшее число слагаемых, чем соответствующие д.н.ф [2, 7]. Полиномиальные нормальные формы по модулю k для функций k-значной логики обобщают понятие п.н.ф. для булевых функций.

Пусть $k \geqslant 2$ – натуральное число, $E_k = \{0, 1, \ldots, k-1\}$. Определим на наборах из множества E_k^n частичный порядок: если $\alpha = (a_1, \ldots, a_n) \in E_k^n$ и $\beta = (b_1, \ldots, b_n) \in E_k^n$, то $\alpha \leqslant \beta$ при $a_i \leqslant b_i$ $i = 1, \ldots, n$. *Весом* набора $\alpha = (a_1, \ldots, a_n) \in E_k^n$ назовем число $|\alpha| = \sum\limits_{i=1}^{n} a_i$ (здесь рассматривается сумма целых чисел). Для набора $\alpha \in E_k^n$ назовем его *тенью* «вниз» множество

$$\check{S}(\alpha) = \{\beta \in E_k^n : \beta \leqslant \alpha, |\beta| = |\alpha| - 1\}.$$

Если $T \subseteq E_k^n$, то $\check{S}(T) = \bigcup\limits_{\alpha \in T} \check{S}(\alpha)$. Множество T, $T \subseteq E_k^n$, назовем *затеняющим* на множестве E_k^n, если $\check{S}(T) = E_k^n \setminus \widetilde{(k-1)}$, где $\widetilde{(k-1)} = (k-1, \ldots, k-1)$ – наибольший набор из E_k^n. Множество T, $T \subseteq E_k^n$, назовем *покрытием* (*асимметричным покрытием «вниз»*) множества E_k^n, если $T \cup \check{S}(T) = E_k^n$.

Моном $\prod\limits_{a_i \neq 0} x_i^{a_i}$ назовем *соответствующим* набору $\alpha = (a_1, \ldots, a_n) \in E_k^n$ и обозначим через X_α. По определению положим, что константа 1 соответствует набору из всех нулей.

Функцией k-значной логики называется отображение $f^n : E_k^n \to E_k$, $n = 0, 1, \ldots$. Множество всех k-значных функций обозначим через P_k, множество всех k-значных функций, зависящих от переменных $x_1, \ldots, x_n$, обозначим через P_k^n.

Далее в формулах для функций k-значной логики операции сложения и умножения рассматриваются по модулю k.

Если k – простое число, то каждая функция k-значной логики $f(x_1, \ldots, x_n)$ может быть однозначно задана формулой вида

$$f(x_1, \ldots, x_n) = \sum_{\alpha \in E_k^n : c_f(\alpha) \neq 0} c_f(\alpha) X_\alpha,$$

где $c_f(\alpha) \in E_k$ – коэффициенты, $\alpha \in E_k^n$ [9]. Это представление функций k-значной логики называется ее *полиномом по модулю* k.

Определим полиномиальные нормальные формы по модулю k. Под *поляризованной переменной* x_i будем понимать выражение вида $x_i + h$, где $h \in E_k \setminus \{0\}$. Произведение вида $y_{i_1}^{m_1} \cdot \ldots \cdot y_{i_r}^{m_r}$, где y_{i_j} есть либо переменная x_{i_j}, либо поляризованная переменная x_{i_j}, $y_{i_s} \neq y_{i_t}$ при $s \neq t$, $1 \leqslant m_1, \ldots, m_r \leqslant k-1$, назовем *мономом с поляризованными переменными*. Обычный моном является частным случаем монома с поляризованными переменными.

Выражение вида $\sum\limits_{i=1}^{l} c_i \cdot X_i$, где $c_i \in E_k \setminus \{0\}$ – коэффициенты, X_i – различные мономы с поляризованными переменными, $i = 1, \ldots, l$, назовем *полиномиальной нормальной формой по модулю* k (п.н.ф.). *Длиной* п.н.ф. называется число ее слагаемых с ненулевыми коэффициентами. Мы будем полагать константу 0 п.н.ф. с длиной, равной 0.

Каждую функцию k-значной логики (при простых k) можно представить различными п.н.ф. по модулю k. Назовем длиной функции f в классе п.н.ф. величину $l^{\text{p.n.f.}}(f)$, равную минимальной длине среди всех п.н.ф., представляющих функцию f. Пусть $L_k^{\text{p.n.f.}}(n) = \max l^{\text{p.n.f.}}(f)$, где максимум берется по всем функциям $f \in P_k^n$.

К.Д. Кириченко [7] была исследована длина булевых функций в классе п.н.ф. и получена оценка $L_2^{\text{p.n.f.}}(n) \leqslant 2 \cdot \frac{2^n}{n}(\log_2 n + 1)$. Эта оценка была получена методом построения п.н.ф. на основе затеняющего множества на E_2^n. С учетом результата из [10] можно получить, что $L_2^{\text{p.n.f.}}(n) = O\left(\frac{2^n}{n}\right)$. В [8] доказана оценка длины k-значных функций в классе п.н.ф. по модулю k (при простых k): $L_k^{\text{p.n.f.}}(n) \leqslant (2 + o(1)) \cdot \frac{k^n}{n} \cdot \ln n$. В [8] при построении затеняющего множества на E_k^n был применен градиентный алгоритм.

В настоящей работе мы обобщаем результат из [10] на множество E_k^n и получаем порядок функции Шеннона $L_k^{\text{p.n.f.}}(n)$ при простых k.

Теорема 1. *Пусть k – простое число. Если T, $T \subseteq E_k^n$, – покрытие множества E_k^n, то для каждой функции $f(x_1, \ldots, x_n) \in P_k^n$ можно построить п.н.ф. по модулю k, ее представляющую, с длиной, не большей $2 \cdot |T|$.*

Теорема 2. *При каждом натуральном $k \geqslant 2$ и при каждом целом $n \geqslant 0$ существует множество мощности $\frac{c \cdot k^n}{n}$, покрывающее множество E_k^n, где $c = c(k)$ – некоторая константа, зависящая только от числа k.*

Теорема 3. *Если k – простое число, то* $L_k^{\text{p.n.f.}}(n) = O\left(\frac{k^n}{n}\right)$.

Применяя нижнюю мощностную оценку для $L_k^{\text{p.n.f.}}(n)$ из [8], получаем

Теорема 4. *Если k – простое число, то* $L_k^{\text{p.n.f.}}(n) = \Theta\left(\frac{k^n}{n}\right)$.

Работа поддержана РФФИ, грант 13–01–00958-а и частично грант 13–01–00684-а.

Литература

[1] *Угрюмов Е. П.* Цифровая схемотехника. СПб.: БХВ-Петербург, 2004.

[2] *Astola J. T., Stankovich R. S.* Fundamentals of Switching Theory and Logic Design. Dordrechht, Netherlands, Springer, 2006.

[3] *Sasao T., Besslich P.* On the complexity of mod-2 sum PLA's // IEEE Trans. on Comput. **39**. N 2. 1990. P. 262–266.

[4] *Супрун В. П.* Сложность булевых функций в классе канонических поляризованных полиномов // Дискретная математика. **5**. № 2. 1993. С. 111–115.

[5] *Перязев Н. А.* Сложность булевых функций в классе полиномиальных поляризованных форм // Алгебра и логика. **34**. № 3. 1995. С. 323–326.

[6] *Селезнева С. Н.* О сложности представления функций многозначных логик поляризованными полиномами // Дискретная математика. **14**. № 2. 2002. С. 48–53.

[7] *Кириченко К. Д.* Верхняя оценка сложности полиномиальных нормальных форм булевых функций // Дискретная математика. **17**. № 3. 2005. С. 80–88.

[8] *Селезнева С. Н., Дайняк А. Б.* О сложности обобщенных полиномов k-значных функций // Вестник Московского университета. Серия 15. Вычислительная математика и кибернетика. № 3. 2008. С. 34–39.

[9] *Яблонский С. В.* Введение в дискретную математику. М.: Высшая школа, 2001.

[10] *Cooper J. N., Ellis R. B., Kahng A. B.* Asymmetric binary covering codes // Journal of Combinatorial Theory, Series A. **100**. N 2. 2002. P. 232–249.

О дистанционной магической разметке декартового произведения графов

М. Ф. Семенюта

marina_semenyuta@mail.ru

Кировоградская летная академия НАУ, Кировоград

В теории разметок возникают такие ситуации, когда одна и та же терминология используется для разных понятий или некоторые из разметок были предложены разными авторами. Одним из таких случаев является дистанционная магическая разметка, которая была исследована под разными названиями, такими как сигма разметка и 1-вершинно магическая вершинная разметка. Ее введение мотивировано, с одной стороны, изучением магических квадратов, с другой стороны, для нужд радиотрансляции возникла необходимость в разметках, связанных с расстоянием между вершинами. В работе [1] представлен обзор существующих результатов по дистанционным магическим

графам и выделены нерешенные проблемы, а также сформулированы гипотезы. К одной из проблем относится характеристика графов G и H таких, что их декартовое произведение $G \times H$ будет дистанционным магическим графом. Ее исследованием занимались авторы работ [2-4]. Мы начали рассмотрение нерешенных случаев, описанных в [1,4].

Пусть $N(x)$ - множество смежности вершины x. Дистанционной магической разметкой графа $G = (V, E)$ порядка n называют биекцию f: $V(G) \longrightarrow \{1, 2, \ldots, n\}$, для которой существует такое положительное целое число k, что для каждой вершины x, $k = \sum_{y \in N(x)} f(y)$. Постоянная k называется магической постоянной разметки f, сумма $\sum_{y \in N(x)} f(y)$- весом вершины x и обозначается $w(x)$. Граф, допускающий дистанционную магическую разметку, называют дистанционным магическим графом.

Для r-регулярного дистанционного магического графа порядка n магическая постоянная определяется по формуле $k = r(n+1)/2$. r-Регулярный граф с нечетным r не имеет дистанционной магической разметки. $K_n \times C_m$ является $(n+1)$-регулярным графом порядка nm. Граф $K_n \times C_m$ при четном n и $K_n \times C_3$ при нечетном n не допускают дистанционную магическую разметку [4]. Рассмотрим случай нечетного n и $m \geqslant 4$. При $n = 3$ получим граф $K_3 \times C_m = C_3 \times C_m$, который, как известно, не является дистанционным магическим для любого натурального числа $m \geqslant 4$.

Пусть $V(K_n \times C_m) = \{(x_1, y_1), (x_1, y_2), \ldots, (x_1, y_m), (x_2, y_1), (x_2, y_2), \ldots, (x_2, y_m), \ldots, (x_n, y_1), (x_n, y_2), \ldots, (x_n, y_m)\}$.

Предположим, что для графа $K_n \times C_m$ существует дистанционная магическая разметка $f : V(K_n \times C_m) \longrightarrow \{1, 2, \ldots, mn\}$ и $f_{ij} = f(x_i, y_j)$ - метка вершины $(x_i, y_j) \in V(K_n \times C_m)$, где $i = 1, 2, \ldots, n, j = 1, 2, \ldots, m$. Обозначим $\sum_i = \sum_{j=1}^{m} f_{ij}$, $\sum_j^{'} = \sum_{i=1}^{n} f_{ij}$.

Теорема 1. *Если граф $K_n \times C_m$, где $n \geqslant 5$ - нечетное, $m \geqslant 4$, является дистанционным магическим, то $\sum_1 = \sum_2 = \sum_3 = \cdots = \sum_n = m(mn+1)/2$.*

Доказательство. Пусть для графа $K_n \times C_m$, где $n \geqslant 5$ - нечетное, $m \geqslant 4$, существует дистанционная магическая разметка f с магической постоянной k. Из условия магичности, например, для вершин $(x_1, y_1), (x_1, y_2), \ldots, (x_1, y_m)$, находим

$$k = w(x_{1j}) = \Sigma_j^{'} - f_{1j} + f_{1,j-1} + f_{1,j+1},$$

индексы $j = 1, 2, \ldots, m$ берутся по модулю m. Тогда

$$\Sigma_1^{'} = k + f_{11} - f_{12} - f_{1m}; \Sigma_2^{'} = k + f_{12} - f_{11} - f_{13};$$

$$\Sigma_3^{'} = k + f_{13} - f_{12} - f_{14}; ..; \Sigma_m^{'} = k + f_{1m} - f_{11} - f_{1m-1},$$

где $k = (n+1)(mn+1)/2$. Суммируя правые и левые части последних равенств, получим

$$\Sigma_1^{'} + \Sigma_2^{'} + \Sigma_3^{'} + \cdots + \Sigma_m^{'} = mk - \Sigma_1,$$

откуда

$$\Sigma_1 = m(mn+1)/2.$$

Для вершин $(x_2, y_1), (x_2, y_2), \ldots, (x_2, y_m)$, имеем

$$\Sigma_1^{'} = k + f_{21} - f_{22} - f_{2m}; \Sigma_2^{'} = k + f_{22} - f_{21} - f_{23};$$

$$\Sigma_3^{'} = k + f_{23} - f_{22} - f_{24}; ..; \Sigma_{m1}^{'} = k + f_{2m} - f_{21} - f_{2m-1},$$

следовательно,

$$\Sigma_2 = m(mn+1)/2.$$

Аналогично для вершин $(x_3, y_1), (x_3, y_2), \ldots, (x_3, y_m)$ найдем

$$\Sigma_3 = m(mn+1)/2$$

и так далее, для $(x_n, y_1), (x_n, y_2), \ldots, (x_n, y_m)$ получим

$$\Sigma_n = m(mn+1)/2.$$

Таким образом,

$$\Sigma_1 = \Sigma_2 = \Sigma_3 = \cdots = \Sigma_n = m(mn+1)/2.$$

∎

В [5] доказано, что для четного n, r-регулярный дистанционный магический граф порядка n существует тогда и только тогда, когда $2 \leqslant r \leqslant n-2$, $r \equiv 0(mod 2)$ и $n \equiv 0(mod 4)$ или $n \equiv r+2 \equiv 2(mod 4)$.

Теорема 2. *Граф $K_n \times C_{4m}$ является дистанционным магическим для любого нечетного натурального числа $n \geqslant 5$.*

Доказательство. Пусть задан граф $K_n \times C_{4m}$ с нечетным $n \geqslant 5$. Он представляет собой r-регулярный граф четного порядка $4mn$, где $r = n+1$ - четное, $2 \leqslant r \leqslant 4mn-2$, для которого $4mn \equiv 0(mod 4)$. Следовательно, $K_n \times C_{4m}$ является дистанционным магическим графом. ∎

Литература

[1] *Arumugam S., Froncek D., Kamatchi N.* Distance magic graphs — a survey // J. Indones. Math. Soc. — 2011.— Special Edition. — P. 1–9.

[2] *Rao S. B., Singh T., Parameswaran V.* Some sigma labelled graphs I, In Graphs, Combinatorics, Algorithms and Applications, eds.. — Narosa Publishing House, New Delhi, 2004. — P. 125–133.

[3] *Beena S.* On $\sum$ and $\sum^{'}$ labelled graphs // JDiscrete Math. — 2009. — V. 309, № 6. — P. 1783–1787.

[4] *Seoud M. A., AbdelMaqsoud A. E. I., Aldiban Y. I.* New classes of graphs with and without 1-vertex magic vertex labeling // Proc. Pakistan Acad. Sci. — 2009. — V. 46, № 3. — P. 159–174.

[5] *Froncek D., Kovar P., Kovarova T.* Fair incomplete tournaments // Bull. Inst. Combin. Appl. — 2006. — V. 48. — P. 31–33.

Некоторые свойства сигнатурных операций табличных алгебр

А. С. Сенченко

senchenko@pisem.net

Киевский национальный университет им. Тараса Шевченко

Процесс информатизации общества имеет объективный характер. Ядром для подавляющего большинства современных информационных систем являются базы данных. В настоящее время наиболее распространенными остаются реляционные базы данных, математическая модель которых была впервые предложена Э. Коддом в 1970 году [1]. С математической точки зрения реляционная база данных является конечным множеством конечных отношений различной размерности (арности) между заранее определёнными множествами элементарных данных. Табличные алгебры, введённые В.Н. Редько и Д.Б. Буем, построены на основе реляционных алгебр Э. Кодда и существенно их развивают. Они составляют теоретический фундамент языков запросов современных табличных баз данных, их сигнатурные операции построены на базе основных табличных манипуляций в реляционных алгебрах и SQL-подобных языках. В монографии [2] установлено значительное количество различных свойств операций табличных алгебр, большинство из которых для общего случая выполняются в виде включений. В настоящей работе приведены критерии перехода некоторых таких включений в равенства. Эти критерии представляют интерес для теории табличных алгебр по той причине, что только на основе равенств можно осуществлять эквивалентные преобразования выражений для решения актуальной задачи оптимизации запросов [3], [4].

Зафиксируем некоторое непустое множество атрибутов $A = \{A_1, \ldots, A_n\}$. Произвольное конечное подмножество множества A назовем схемой, причем схема может быть пустым множеством. Строкой s схемы R называется множество пар $s = \{(A'_1, d_1), \ldots, (A'_k, d_k)\}$, проекция которого по первой компоненте равна R, причем атрибуты $A'_1, \ldots, A'_k$ попарно различны. Таблицей схемы R называется конечное множество строк схемы R. Далее в работе рассматриваем таблицы схемы R с количеством атрибутов k.

Рассмотрим основные сигнатурные операции на множестве всех таблиц. Операции объединения, пересечения и разности таблиц вводятся как ограничение одноименных теоретико-множественных операций на множество таблиц. Насыщением $C(T)$ называется таблица $\prod\limits_{A \in R} D_{A,T}$, где R – схема таблицы T, $D_{A,T} = \{d | \exists s \in T \wedge (A, d) \in s\}$ – активный домен атрибута A относительно таблицы T, состоящий из всевозможных значений атрибута A в таблице T, а $\prod$ – оператор прямого (декартового) произведения, отвечающий индексированию $A \mapsto D_{A,T}$, $A \in R$ [5]. Проекцией по множеству атрибутов $X \subseteq R$ называется унарная параметрическая операция π_X , значением которой является таблица, состоящая из ограничений по X всех строк исходной таблицы: $\pi_X(T) = \{s \mid x \mid s \in T$. Пусть $X = \{X_1, \ldots, X_p\}$. Далее в работе через $O = \{O_1, \ldots, O_{k-p}\}$ обозначим множество атрибутов $R - X$, не участвующих в проекции. Для введения операции соединения необходимо одно вспомога-

тельное понятие. Бинарные отношения ρ и τ называются совместными (обозначается $\rho \approx \tau$), если $\rho \mid x = \tau \mid x$, где $X = pr_1\rho \bigcap pr_1\tau$ [2]. Соединением называется бинарная операция $\otimes$, значением которой является таблица, состоящая из всевозможных объединений совместных строк исходных таблиц, то есть $T_1 \otimes T_2 = \{s_1 \bigcup s_2 \mid s_1 \in T_1 \wedge s_2 \in T_2 \wedge s_1 \approx s_2\}$. Табличной алгеброй называют частичную алгебру с носителем — множеством всех таблиц произвольной схемы, приведёнными выше операциями, а также операциями селекции, деления таблиц и переименования атрибутов. В табличной алгебре выделяют две особые таблицы: таблицу $T_\varepsilon = \{\varepsilon\}$, схема которой является пустым множеством, и таблицу $T_\varnothing = \varnothing$ – пустое множество строк любой схемы.

В монографии [2] сформулирован и доказан ряд свойств насыщения, активном дополнения, проекции и соединения. В настоящей работе найдены необходимые и достаточные условия, при которых некоторые включения превращаются в равенства для таблиц, не являющихся особыми; для особых таблиц эти равенства тоже выполняются, но могут не выполняться критерии.

Теорема 1. *(Закон двойного отрицания.) Равенство $\tilde{\tilde{T}} = T$ выполняется тогда и только тогда, когда для каждого значения x активного домена каждого атрибута A_q существуют такие значения $d_1, \ldots, d_{q-1}, d_{q+1}, \ldots, d_k$ активных доменов атрибутов $A_1, \ldots, A_{q-1}, A_{q+1}, \ldots, A_k$ соответственно, что $s = \{(A_1, d_1), \ldots, (A_{q-1}, d_{q-1}), (A_q, x), (A_{q+1}, d_{q+1}), \ldots, (A_k, d_k)\} \notin T$.*

Теорема 2. *(Первый закон де Моргана.) Равенство $\tilde{T_1} \bigcap \tilde{T_2} = (T_1 \tilde{\bigcup} T_2)$ при $(T_1 \tilde{\bigcup} T_2) \neq T_\varnothing$ выполняется тогда и только тогда, когда выполняется одно из четырёх взаимоисключающих условий:*

1) $\forall A(A \in R \Rightarrow D_{A,T_1} = D_{A,T_2})$;

2) $\forall A(A \in R \Rightarrow D_{A,T_2} \subseteq D_{A,T_1})$, и для всех индексов q, значений $x \in D_{A_q,T_1} - D_{A_q,T_2}$ и всех значений $d_1, \ldots, d_{q-1}, d_{q+1}, \ldots, d_k$, принадлежащих соответственно активным доменам $D_{A_1,T_1}, \ldots, D_{A_{q-1},T_1}, D_{A_{q+1},T_1} \ldots, D_{A_k,T_1}$, строка $\{(A_1, d_1), \ldots, (A_{q-1}, d_{q-1}), (A_q, x), (A_{q+1}, d_{q+1}), \ldots, (A_k, d_k)\} \in T_1$;

3) $\forall A(A \in R \Rightarrow D_{A,T_1} \subseteq D_{A,T_2})$, и для всех индексов q, значений $x \in D_{A_q,T_2} - D_{A_q,T_1}$ и всех значений $d_1, \ldots, d_{q-1}, d_{q+1}, \ldots, d_k$, принадлежащих соответственно активным доменам $D_{A_1,T_2}, \ldots, D_{A_{q-1},T_2}, D_{A_{q+1},T_2} \ldots, D_{A_k,T_2}$, строка $\{(A_1, d_1), \ldots, (A_{q-1}, d_{q-1}), (A_q, x), (A_{q+1}, d_{q+1}), \ldots, (A_k, d_k)\} \in T_2$;

4) Существует такой атрибут A_q и значения $x \in D_{A_q,T_1} - D_{A_q,T_2}$, $y \in D_{A_q,T_2} - D_{A_q,T_1}$, причем $D_{A_q,T_1} \bigcap D_{A_q,T_2} \neq \varnothing$; кроме того, для всех $i \neq q$ выполняются равенства $D_{A_i,T_1} = D_{A_i,T_2}$; наконец, для всех $z_1 \in D_{A_q,T_1} - D_{A_q,T_2}$, всех $z_2 \in D_{A_q,T_2} - D_{A_q,T_1}$ и всех $d_1, \ldots, d_{q-1}, d_{q+1}, \ldots, d_k$, принадлежащих соответственно активным доменам $D_{A_1,T_1}, \ldots, D_{A_{q-1},T_1}, D_{A_{q+1},T_1} \ldots, D_{A_k,T_1}$, строка $\{(A_1, d_1), \ldots, (A_{q-1}, d_{q-1}), (A_q, z_1), (A_{q+1}, d_{q+1}), \ldots, (A_k, d_k)\} \in T_1$, а строка $\{(A_1, d_1), \ldots, (A_{q-1}, d_{q-1}), (A_q, z_2), (A_{q+1}, d_{q+1}), \ldots, (A_k, d_k)\} \in T_2$.

Теорема 3. *(Второй закон де Моргана.) Равенство $(T_1 \tilde{\bigcap} T_2) = \tilde{T_1} \bigcup \tilde{T_2}$ при $(T_1 \tilde{\bigcap} T_2) \neq T_\varnothing$ выполняется тогда и только тогда, когда выполняется одно из двух взаимоисключающих условий:*

1) для каждого атрибута $A \in R$ выполняются равенства $D_{A,T_1} = D_{A,T_1 \bigcap T_2}$ и $D_{A,T_2} = D_{A,T_1 \bigcap T_2}$.

2) для всех атрибутов $A_q \in R$, таких x, что $x \in D_{A_q,T_1} - D_{A_q,T_1 \bigcap T_2}$ и всех $d_1, \ldots, d_{q-1}, d_{q+1}, \ldots, d_k$, принадлежащих соответственно $D_{A_1,T_1}, \ldots, D_{A_{q-1},T_1}$, $D_{A_{q+1},T_1} \ldots, D_{A_k,T_1}$, строка $\{(A_1, d_1), \ldots, (A_{q-1}, d_{q-1}), (A_q, x), (A_{q+1}, d_{q+1}), \ldots, (A_k, d_k)\} \in T_1$. Для всех атрибутов $A_w \in R$, таких y, что $y \in D_{A_w,T_2} - D_{A_w,T_1 \bigcap T_2}$ и всех $d_1, \ldots, d_{w-1}, d_{w+1}, \ldots, d_k$, принадлежащихт соответственно $D_{A_1,T_2}, \ldots, D_{A_{w-1},T_2}, D_{A_{w+1},T_2} \ldots, D_{A_k,T_2}$, строка $\{(A_1, d_1), \ldots, (A_{w-1}, d_{w-1}), (A_w, y), (A_{w+1}, d_{w+1}), \ldots, (A_k, d_k)\} \in T_2$.

Теорема 4. *(Перестановочность проекции и активного дополнения.) При $T \neq T_\varnothing$ равенство $(\pi_X(\tilde{T})) = \pi_X(\tilde{T})$ выполняется тогда и только тогда, когда для каждой $s = \{(X_1, x_1), \ldots, (X_p, x_p)\} \in \pi_X(T)$ и любых $o_1 \in D_{O_1,T}, \ldots, o_{k-p} \in D_{O_{k-p},T}$, строка $s' = \{(X_1, x_1), .., (X_p, x_p), (O_1, o_1), .., (O_{k-p}, o_{k-p})\} \in T$.*

Теорема 5. *(Дистрибутивность проекции по пересечению.) При $\bigcap\limits_i T_i \neq T_\varnothing$ равенство $\pi_X(\bigcap\limits_i T_i) = \bigcap\limits_i \pi_X(T_i)$ выполняется тогда и только тогда, когда для каждой строки $s = \{(X_1, x_1), \ldots, (X_p, x_p)\} \in \bigcap\limits_i \pi_X(T)$ существуют такие значения $o_1 \in D_{O_1,\bigcap\limits_i T_i}, \ldots, o_{k-p} \in D_{O_{k-p},\bigcap\limits_i T_i}$, что строка $s' = \{(X_1, x_1), \ldots, (X_p, x_p), (O_1, o_1), \ldots, (O_{k-p}, o_{k-p})\} \in \bigcap\limits_i T_i$.*

Теорема 6. *(Дистрибутивность проекции по разности.) При $\pi_X(T_1 - T_2) \neq T_\varnothing$ равенство $\pi_X(T_1) - \pi_X(T_2) = \pi_X(T_1 - T_2)$ выполняется тогда и только тогда, когда для каждой строки $s = \{(X_1, x_1), \ldots, (X_p, x_p)\} \in \pi_X(T_1) \bigcap \pi_X(T_2)$ и всех значений $o_1 \in D_{O_1,T_1 \bigcup T_2}, \ldots, o_{k-p} \in D_{O_{k-p},T_1 \bigcup T_2}$, из принадлежности $s' = \{(X_1, x_1), \ldots, (X_p, x_p), (O_1, o_1), \ldots, (O_{k-p}, o_{k-p})\} \in T_1$ следует $s' \in T_2$.*

Теорема 7. *(Дистрибутивность насыщения по соединению.) Пусть R_1 и R_2 – схемы таблиц T_1 и T_2, $R' = R_1 \bigcap R_2$ и $T_1 \otimes T_2 \neq T_\varnothing$. Равенство $C(T_1 \otimes T_2) = C(T_1) \otimes C(T_2)$ выполняется тогда и только тогда, когда $\pi_{R'}(T_1) = \pi_{R'}(T_2)$.*

Теорема 8. *(Дистрибутивность активного дополнения по соединению.) Равенство $\tilde{T_1} \otimes \tilde{T_2} = (T_1 \tilde{\otimes} T_2)$ выполняется тогда и только тогда, когда выполняется хотя бы одно из двух условий:*

1) $T_1 = T_2$;

2) $\tilde{T_1} \otimes \tilde{T_2} = T_\varnothing$ и $(T_1 \tilde{\otimes} T_2) = T_\varnothing$.

Автор благодарит своего научного руководителя Дмитрия Борисовича Буя за постановку задачи и обсуждение результатов.

Литература

[1] *Codd E. F.* A Relational Model of Data for Large Shared Data Banks // Communications of the ACM. — 1970. — V. 13, № 6. — P. 377–387.

[2] *Редько В. Н., Брона Ю. Й., Буй Д. Б., Поляков С. А.* Реляційні бази даних: табличні алгебри та SQL-подібні мови. — К.: Академперіодика, 2001. — 198 с.

[3] *Кнут Дональд.* Искусство программирования, том 4, А. Комбинаторные алгоритмы, часть 1. — М.: Вильямс, 2013. — 960 с.

[4] *Мендекович Н. А., Кузнецов С. Д.* Обзор развития методов лексической оптимизации запросов // Труды ИСП РАН. — 2012. — Т. 23. — С. 195–214.

[5] *Куратовский К.* Топология, том 1. — М.: Мир, 1966. — 594 с.

О сравнительной сложности реализации матрицы и ее дополнения вентильными схемами

И. С. Сергеев

isserg@gmail.com

ФГУП «НИИ «Квант», Москва

В настоящей работе строятся примеры квадратных булевых матриц A, таких, что сложность реализации матрицы A и дополнительной матрицы $\bar{A}$ вентильными схемами различается существенно.

Напомним, что *вентильная* (m,n)*-схема* — это ориентированный ациклический граф, в котором n вершин отмечены как входы и m вершин отмечены как выходы. Вентильная схема реализует булеву $m \times n$-матрицу A тогда и только тогда, когда при любых i и j выполнено: $A[i,j] = 1$ равносильно тому, что в схеме имеется ориентированный путь из j-го входа в i-й выход. Сложностью схемы называется число ребер в ней, а глубиной — максимальная длина ориентированного пути. Подробнее см. в [1].

Сложность минимальной по числу ребер схемы, реализующей матрицу A, обозначаем через $L(A)$; сложность минимальной схемы глубины d — через $L_d(A)$.

В работе [2] методом [3] доказано существование $n \times n$-матриц A, удовлетворяющих соотношению

$$L(\bar{A})/L(A) = \Omega(n/\log^3 n).$$

Отметим, что согласно результатам об асимптотической сложности [1, 4] класса булевых матриц подобное отношение не может по порядку превосходить $n/\log n$.

Назовем k*-прямоугольником* сплошь единичную матрицу размера $k \times k$. Матрица называется k*-редкой*, если она не содержит k-прямоугольников в качестве подматриц.

В [2] доказано существование матрицы A, допускающей простую реализацию в глубине 2, $L_2(A) = O(n\log^2 n)$, дополнительная матрица $\bar{A}$ к которой является 2-редкой и имеет при этом достаточно большой вес (число единиц), $|\bar{A}| = \Omega(n^{5/4})$. Как следствие из [4], $L(\bar{A}) = L_2(\bar{A}) = |\bar{A}|$.

Ниже матрицы, обладающие аналогичными свойствами, строятся явно.

Теорема 1.

(*i*) *Для некоторой конкретно заданной булевой* $n \times n$*-матрицы* C *выполнено* $L(\bar{C})/L(C) = n \cdot 2^{-O(\sqrt{\ln n \ln\ln n})}$.

(*ii*) *Для некоторой конкретно заданной булевой* $n \times n$*-матрицы* C *выполнено:* $L(C) = O(n)$, *матрица* $\bar{C}$ *является* 2*-редкой и* $|\bar{C}| = \Omega(n^{4/3})$.

(Напомним, что 2-редкая матрица не может иметь вес выше $n^{3/2} + n$.)

Доказательство теоремы опирается на простую комбинаторную лемму.

Лемма 1. *Пусть $n \times n$-матрица A имеет вес $|A| \geqslant 2n^{3/2}$. Тогда она содержит $\Omega((|A|/n)^4)$ 2-прямоугольников.*

Доказательство. Скажем, что строка матрицы покрывает пару столбцов u, если в позициях на пересечении строки и этих столбцов стоят единицы. Если обозначить через a_i число единиц в i-й строке матрицы A, то общее число покрытий строками пар столбцов можно оценить как

$$\sigma = \sum_{i=1}^{n} \binom{a_i}{2} = \frac{1}{2}\sum_{i=1}^{n} a_i^2 - \frac{|A|}{2} \geqslant \frac{\left(\sum_{i=1}^{n} a_i\right)^2}{2n} - \frac{|A|}{2} = \frac{|A|^2}{2n} - \frac{|A|}{2} \geqslant \frac{|A|^2}{4n}.$$

Обозначим через b_u число строк, покрывающих пару столбцов u. Тогда $\sum_u b_u = \sigma$. Число 2-прямоугольников в матрице A при этом равно

$$\sum_u \binom{b_u}{2} = \frac{1}{2}\sum_u b_u^2 - \frac{\sigma}{2} \geqslant \frac{\left(\sum_u b_u\right)^2}{n(n-1)} - \frac{\sigma}{2} = \frac{\sigma^2}{n(n-1)} - \frac{\sigma}{2} \geqslant \frac{\sigma^2}{2n^2} = \Omega\left(\left(\frac{|A|}{n}\right)^4\right).$$

∎

Пусть $n = \binom{m}{2}$. По булевой $m \times m$-матрице A построим $n \times n$-матрицу B следующим образом. Занумеруем строки и столбцы матрицы B 2-элементными подмножествами множества $[m]$. Положим $B[a, b] = 1$ в том и только том случае, когда на пересечении строк a и столбцов b в матрице A расположен 2-прямоугольник.

Лемма 2. *Если матрица A является k-редкой, то матрица B является K-редкой, $K = \binom{k-1}{2} + 1$.*

Доказательство. Если матрица B содержит K-прямоугольник на пересечении строк $s_1, \ldots, s_K$ и столбцов $t_1, \ldots, t_K$, то матрица A содержит прямоугольник на пересечении строк $\cup s_i$ и столбцов $\cup t_i$. При этом $|\cup s_i|, |\cup t_i| \geqslant k$, что противоречит k-редкости матрицы A. ∎

Лемма 3. *Если матрица A является k-редкой и $|A| \geqslant 2m^{3/2}$, то*

$$L(B) = \Omega\left(\left(\frac{|A|}{kn}\right)^4\right),$$

при этом $L_3(\bar{B}) = O(n)$.

Доказательство. Согласно лемме 1, $|B| = \Omega((|A|/n)^4)$, а из леммы 2 следует, что B является K-редкой. Поэтому по теореме Нечипорука [4]

$$L(B) \geqslant \frac{|B|}{K^2} = \Omega\left(\left(\frac{|A|}{kn}\right)^4\right).$$

Покажем, что матрицу $\bar{B}$ можно реализовать схемой глубины 3 и линейной сложности. На втором и третьем уровнях схемы разместим по m вершин и занумеруем их числами из $[m]$. Вход $a = \{i, j\}$ соединим с вершинами i, j второго уровня. Аналогично поступим с выходами и вершинами третьего уровня. Вершины второго и третьего уровня соединим ребрами согласно матрице $\bar{A}$.

По построению, схема имеет $O(m^2)$ ребер. То, что схема реализует матрицу $\bar{B}$, вытекает из того, что путь, соединяющий вход a и выход b содержится в схеме в том и только том случае, когда подматрица матрицы $\bar{A}$, образованная строками b и столбцами a, не является сплошь нулевой. ∎

Для доказательства п. (i) теоремы в качестве матрицы A выберем норм-матрицу из работы [5], которая при подходящем выборе параметров является Δ-редкой и имеет вес m^2/Δ, где $\Delta = 2^{O(\sqrt{\log m \log\log m})}$. Положим $C = \bar{B}$.

Для доказательства п. (ii) выберем в качестве матрицы A 3-редкую матрицу Брауна [6] веса $\Theta(m^{5/3})$. Положим $C = \bar{B}$. Теорема доказана.

Автор благодарен С. Юкне за предложения по усовершенствованию доказательства.

Работа выполнена при поддержке РФФИ, проект № 14-01-00671а.

Литература

[1] *Лупанов О. Б.* О вентильных и контактно-вентильных схемах // ДАН СССР. — 1956. — Т. 111, № 6. — С. 1171–1174.

[2] *Jukna S., Sergeev I.* Complexity of linear boolean operators // Foundations and Trends in TCS. — 2013. — V. 9, № 1. — P. 1–123.

[3] *Katz N. H.* On the CNF-complexity of bipartite graphs containing no squares // Lithuanian Math. Journal. — 2012. — V. 52, № 4. — P. 385–389.

[4] *Нечипорук Э. И.* О топологических принципах самокорректирования // Проблемы кибернетики. Вып. 21. — М.: Наука, 1969. — С. 5–102.

[5] *Kóllar J., Rónyai L., Szabó T.* Norm-graphs and bipartite Turán numbers // Combinatorica. — 1996. — V. 16, № 3. — P. 399–406.

[6] *Brown W. G.* On graphs that do not contain a Thomsen graph // Canad. Math. Bull. — 1966. — V. 9. — P. 281–285. [Русский перевод: *Браун У. Г.* Графы, не содержащие графа Томсена // Кибернетический сборник. Вып. 18. — М.: Мир, 1981. — С. 34–38.]

О подобии блочно-диагональных матриц над кольцом целых чисел

С. В. Сидоров

sesidorov@yandex.ru

Нижегородский государственный университет им. Н.И. Лобачевского, национальный исследовательский университет, Нижний Новгород

Определение. Пусть A и B — две целочисленные матрицы порядка n. Они называются подобными над кольцом целых чисел $\mathbf{Z}$, если существует такая матрица $X \in \mathbf{Z}^{n\times n}$, что $AX = XB$ и $\det X = \pm 1$ (при этом пишут $A \sim B$). Матрица X называется трансформирующей матрицей.

Пусть матрица $A \in \mathbf{Z}^{n\times n}$ имеет характеристический многочлен $d(x) = \det(xE - A)$, который раскладывается на неприводимые над полем рациональных чисел $\mathbf{Q}$ множители. Будем считать, что кратность каждого множителя

равна единице. Итак, $d(x) = p_1(x) \cdot \ldots \cdot p_s(x)$, $\deg p_i(x) = k_i$, $i = 1, \ldots, s$, $k_1 + \ldots + k_s = n$. Известно (см., например, [1]), что над полем $\mathbf{Q}$ такая матрица подобна блочно-диагональной матрице $F = \begin{pmatrix} F_1 & 0 & \ldots & 0 & 0 \\ 0 & F_2 & \ldots & 0 & 0 \\ & & \ldots & & \\ 0 & 0 & \ldots & 0 & F_s \end{pmatrix}$, где $F_i = \begin{pmatrix} -f_{i1} & -f_{i2} & \ldots & -f_{i,k_i-1} & -f_{i,k_i} \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ & & \ldots & & \\ 0 & 0 & \ldots & 0 & 0 \\ 0 & 0 & \ldots & 1 & 0 \end{pmatrix}$ — матрица Фробениуса. Характеристический многочлен матрицы Фробениуса равен $\det(xE - F_i) = p_i(x) = x^{k_i} + f_{i1}x^{k_i-1} + \ldots + f_{i,k_i}$.

Над кольцом целых чисел такой факт, вообще говоря, не имеет места. Приведем контрпример. Рассмотрим матрицы $A = \begin{pmatrix} 0 & 2 \\ 5 & 0 \end{pmatrix}$ и $F = \begin{pmatrix} 0 & 10 \\ 1 & 0 \end{pmatrix}$. Они подобны над $\mathbf{Q}$, но не подобны над $\mathbf{Z}$.

Теорема 1. *Пусть $A = diag\{A_1, A_2, \ldots, A_s\}$ и $B = diag\{B_1, B_2, \ldots, B_s\}$ — две блочно-диагональные целочисленные матрицы, где $p_i(x) = |xE - A_i| = |xE - B_i|$ — неприводимые над $\mathbf{Q}$ и попарно различные характеристические многочлены. Тогда $A \sim B$ тогда и только тогда, когда $A_i \sim B_i$ для всех $i = 1, \ldots, s$.*

Доказательство. Пусть $X = (X_{ij})$ — трансформирующая матрица, разбитая на блоки X_{ij}, $i = 1, \ldots, s$, $j = 1, \ldots, s$. Поскольку $AX = XB$, то $A_iX_{ij} = X_{ij}B_j$. Из условия теоремы следует, что матрицы A_i и B_j не имеют общих собственных чисел при $i \neq j$, поэтому X_{ij} — нулевая матрица при $i \neq j$ (см., например, [1]). Следовательно, X — блочно-диагональная матрица и $\det X = \det X_{11} \cdot \ldots \cdot \det X_{ss}$. Из условия $\det X \in \{-1, 1\}$ вытекает, что $\det X_{ii} \in \{-1, 1\}$. Поскольку при этом $A_iX_{ii} = X_{ii}B_i$, то матрицы A_i и B_i должны быть подобны над $\mathbf{Z}$ для всех $i = 1, \ldots, s$. ∎

Теорема неверна, если матрицы, стоящие по диагонали, имеют одинаковые характеристические многочлены. Приведем пример. Пусть $A = \begin{pmatrix} 0 & d_1 \\ d_2 & 0 \end{pmatrix}$ и $F = \begin{pmatrix} 0 & D \\ 1 & 0 \end{pmatrix}$, где d_1, d_2 — взаимно простые числа и $d_1d_2 = D$. Если A и F не подобны над $\mathbf{Z}$ (например, при $d_1 = 2$, $d_2 = 5$), то блочные матрицы $\begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix} = \begin{pmatrix} 0 & d_1 & 0 & 0 \\ d_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & d_1 \\ 0 & 0 & d_2 & 0 \end{pmatrix}$ и $\begin{pmatrix} F & 0 \\ 0 & F \end{pmatrix} = \begin{pmatrix} 0 & D & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & D \\ 0 & 0 & 1 & 0 \end{pmatrix}$ подобны над $\mathbf{Z}$. Действительно, в качестве трансформирующей матрицы можно

взять следующую $X = \begin{pmatrix} 0 & d_1 & v & 0 \\ 1 & 0 & 0 & vd_2 \\ 1 & 0 & 0 & ud_1 \\ 0 & d_2 & u & 0 \end{pmatrix}$, где u и v удовлетворяют условию $ud_1 - vd_2 = 1$.

Литература

[1] *Гантмахер Ф. Р.* Теория матриц. — М.: Наука, 1988. — 552 с.

[2] *Шевченко В. Н., Сидоров С. В.* О подобии матриц второго порядка над кольцом целых чисел // Известия ВУЗ. Математика. — 2006. — № 4. — С. 57–64.

Об оптимальном кодировании в классе локально-префиксных кодов

Т. Г. Смирнова

smirnova-tg@mail.ru

ННГУ им. Н.И. Лобачевского, Нижний Новгород

В работах Ал.А. Маркова [1, 2] было введено понятие обобщенно-префиксного кодирования локальных моделей языка сообщений произвольной глубины и доказано, что использование локальных моделей может повышать эффективность алфавитного кодирования в сравнении с другими алгоритми экономного кодирования.

В настоящей работе рассмотрим задачу оптимального кодирования в классе локально-префиксных кодов, которые соответствуют алфавитному кодированию локальных моделей глубины 1 и представляют собой простейший класс кодов в классе обобщенно-префиксных.

Пусть $B = \{b_1, ..., b_m\}$ – алфавит языка сообщений $L \subseteq B^*$, где B^* – множество всех слов над алфавитом B. Алфавитное кодирование $f = f_V$ задается схемой: $b_i \to v_i$, где $v_i \in \{0,1\}^+$ $(i = 1, ..., m)$. Множество $V = \{v_1, \ldots, v_m\}$ называется кодом, определяющим f_V, а множество $d(V) = \{d_1, ..., d_m\}$, где $d_i = |v_i|$ – длина элементарного кода v_i, называется спектром длин кода V.

Будем говорить, что слова α и β находятся в отношении антипрефиксности $\overline{\pi}$, и обозначать $\alpha \,\overline{\pi}\, \beta$, если никакое из них не является префиксом другого. Префиксные коды, т. е. коды, у которых $v_i \,\overline{\pi}\, v_j$ для любых $i \neq j$, составляют один из основных классов взаимно-однозначных кодов переменной длины.

Локальная модель глубины 1 языка $L \subseteq B^*$ полностью характеризуется обыкновенным графом $G = (B, E)$, где E задает отношение антипрефиксности на V. Код $V = \{v_1, \ldots, v_m\}$ называется локально-префиксным относительно графа антипрефиксности G, если для любых вершин b_i, b_j графа G из $(b_i, b_j) \in E$ следует $v_i \,\overline{\pi}\, v_j$.

Множество всех спектров, реализуемых двоичными локально-префиксными относительно G кодами, монотонно отросительно отношения частичного порядка, определенного покомпонентной сравнимостью векторов. Минимальные элементы этого множества образуют матрицу $M(G)$ оптимального двоич-

ного локально-префиксного кодирования относительно графа G. Это множество конечно для любого графа по теореме Диксона и для любого графа может быть расшифровано.

Пусть на множестве букв алфавита $B = \{b_1, \ldots, b_m\}$ определено распределение вероятностей $P = (p_1, \ldots, p_m)$, где $0 < p_i < 1$ для всех $i = 1, \ldots, m$, $\sum_{i=1}^{m} p_i = 1$, тогда величина

$$C(V, P) = \sum_{i=1}^{m} p_i \cdot d_i$$

называется избыточностью кода V, а код V^*, минимизирующий $C(V, P)$, называется оптимальным для заданного распределения P.

Задача оптимального локально-префиксного кодирования в общей постановке относится к классу NP-трудных задач [3]. С ней связаны две проблемы: во-первых, построение матрицы оптимального локально-префиксного кодирования и, во-вторых, задача минимизации линейной формы на конечном множестве наборов с неотрицательными целыми компонентами. Представляет интерес выделение полиномиально разрешимых подклассов или поиск эффективных приближенных алгоритмов.

Если граф антипрефиксности $G = K_m$ - полный граф с m вершинами, тогда задача оптимального локально-префиксного кодирования совпадает с классической задачей оптимального кодирования, для которой известен алгоритм Хаффмана [4], который находит оптимальный код в классе префиксных кодов за время $O(m \log m)$.

В [5] выделен подкласс класса кографов, относительно которого задача построения оптимального локально-префиксного кодирования решается за полиномиальное время.

В данной работе задача оптимального локально-префиксного кодирования изучается для случая, когда граф антипрефиксности G содержит ровно две клики. Показано, что в этом случае матрица оптимального двоичного локально-префиксного кодирования может быть построена с помощью системы неравенств Мак-Миллана, записанных для каждой из клик графа H_1, H_2:

$$\sum_{b_i \in H_j} 2^{-d_i} \leqslant 1 \quad (j = 1, 2),$$

и число строк этой матрицы растет экспоненциально с ростом m. В работе предлагается полиномиальный алгоритм оптимального локально-префиксного кодирования относительно графа G с двумя кликами, который не предусматривает построения матрицы оптимального кодирования.

Литература

[1] *Марков Ал. А., Смирнова Т. Г.* Алгоритмические основания обобщенно-префиксного кодирования // ДАН СССР. — 1984. — Т. 274, № 4. — С. 790–793.

[2] *Марков Ал. А.* Локальные модели языков и системы ограничений в словах // Материалы Всесоюзного семинара по дискретной математике и ее приложениям. — М.: МГУ. — 1986. — С. 19–23.

[3] *Марков Ал. А., Смирнова Т. Г.* О словарных раскрасках и некоторых совершенных графах // Дискретная математика. — 1990. — Т. 2, № 2. — С. 16–32.

[4] *Huffman D. A.* A method for the construction of minimum redundancy codes // Proc. IRE. — 1952. — V. 40. — P. 1098–1101.

[5] *Смирнова Т. Г.* Оптимальное кодирование в классе локально-префиксных кодов // Материалы IX Международного семинара "Дискретная математика и ее приложения". — М.: МГУ. — 2007. — С. 457–459.

О реализации булевых функций обобщенными формулами

Л. Н. Сысоева

s-luba@mail.ru

мех-мат МГУ им М. В. Ломоносова, Москва

В работе рассматривается задача о реализации булевых функций обобщенными α-формулами. Вводится понятие универсального множества обобщенных α-формул для заданного множества булевых функций. Формулируется принцип двойственности для обобщенных α-формул. Показывается, что для каждого $n \geqslant 2$ для множеств $T_0(n)$ и $T_1(n)$ всех булевых функций от переменных $x_1, x_2, \ldots, x_n$, сохраняющих константу 0 и 1 соответственно, существуют универсальные множества. Необходимые определения можно найти в книгах [1, 2].

Понятие α-*формул*, т. е. таких формул, в которых любая подформула содержит не более одной нетривиальной главной подформулы, было введено в работе [3]. В работах [3, 4, 5] показано наличие конечных α-полных систем в множестве P_k всех функций k-значной логики ($k \geqslant 3$) и отсутствие их в P_2. В работах [6, 7] свойства α-формул изучены с точки зрения теории сложности. В работе [8] построены универсальные множества обобщенных α-формул для множества $T_{01}(n)$ всех булевых функций от переменных $x_1, x_2, \ldots, x_n$, сохраняющих константы 0 и 1.

Обобщенной α-формулой называется α-формула над некоторым множеством автоматных функций со специальным образом сопоставленной ей функцией алгебры логики. Рассматриваются автоматные функции, которые в каждом состоянии реализуют некоторую булеву функцию. Пусть $\Phi(x_1, x_2, \ldots, x_n)$ — α-формула над некоторым множеством автоматных функций. На формулу Φ подаются все двоичные наборы длины n в определенной последовательности $\alpha_1, \alpha_2, \ldots, \alpha_{2^n}$, $n \geqslant 1$. Таким образом задается последовательность $\Phi(\alpha_1), \Phi(\alpha_2), \ldots, \Phi(\alpha_{2^n})$ значений формулы Φ на всех двоичных наборах длины n. Формуле Φ сопоставляется функция $f(x_1, x_2, \ldots, x_n)$ алгебры логики, такая, что выполнены равенства $f(\alpha_i) = \Phi(\alpha_i)$ для всех $i = 1, \ldots, 2^n$. В момент времени $t = 0$, каждая автоматная функция, входящая в обобщенную α-формулу Φ, находится в состоянии, отвечающем некоторой функции алгебры логики. Заменим все символы автоматных функций, входящих в Φ, на

символы соответствующих булевых функций. Получим некоторую α-формулу $\Phi_0(x_1, x_2, \ldots, x_n)$. Назовем Φ_0 *начальной формулой* обобщенной α-формулы Φ.

Пусть A — некоторое множество булевых функций. Обобщенная α-формула $\Phi(x_1, x_2, \ldots, x_n)$ называется *универсальной* для A, если для любой булевой функции $f(x_1, x_2, \ldots, x_n)$ из A, существует последовательность всех двоичных наборов длины n, такая, что формула Φ при такой последовательности подаваемых наборов реализует функцию f, $n \geqslant 1$.

Пусть V — конечный автомат с двумя входами и одним выходом, такой, что $\{q_1, q_2\}$ — множество его состояний, в состоянии q_1 автомат реализует функцию $x_1 \vee x_2$, в состоянии q_2 — функцию $0(x_1, x_2)$ и автомат в момент времени t переходит из состояния q_i в состояние q_j тогда и только тогда, когда входные символы в этот момент времени совпадают и равны 0, $i \neq j$. Обозначим через F автоматную функцию, реализуемую инициальным автоматом V_{q_1}.

Обозначим через $\mathcal{A}_n$ множество всех обобщенных α-формул над $\{F\}$ от n переменных, в которую каждая переменная входит ровно один раз.

Теорема 1. *Любая обобщенная α-формула из множества $\mathcal{A}_n$ является универсальной для множества $T_0(n)$, $n \geqslant 2$.*

Следует отметить, что ни для какой булевой функции f, не принадлежащей множеству $T_0(n)$, не существует обобщенной α-формулы из множества $\mathcal{A}_n$, реализующей эту функцию, $n \geqslant 2$.

Введем понятие двойственных обобщенных α-формул. Пусть булевы функции $f_1(x, y), f_2(x, y)$ двойственны к функциям $g_1(x, y), g_2(x, y)$ соответственно. Пусть R — конечный автомат с двумя входами и одним выходом, такой, что $\{q_1, q_2\}$ — множество его состояний, в состоянии q_1 автомат реализует функцию $f_1(x, y)$, в состоянии q_2 — функцию $f_2(x, y)$. Пусть $\hat{R}$ — конечный автомат с двумя входами и одним выходом, такой, что $\{\hat{q}_1, \hat{q}_2\}$ — множество его состояний, в состоянии $\hat{q}_1$ автомат реализует функцию $g_1(x, y)$, в состоянии $\hat{q}_2$ — функцию $g_2(x, y)$. И функции переходов φ и $\hat{\varphi}$ автоматов R и $\hat{R}$ таковы, что $\varphi(q_i, (\alpha, \beta)) = q_j$ тогда и только тогда, когда $\hat{\varphi}(\hat{q}_j, (\overline{\alpha}, \overline{\beta})) = \hat{q}_i$, где $i, j \in \{1, 2\}$. Обозначим через F_1 и F_2 автоматные функции, реализуемые инициальными автоматами R_{q_1} и R_{q_2} соответственно. А через $\hat{F}_1$ и $\hat{F}_2$ — автоматные функции, реализуемые инициальными автоматами $\hat{R}_{\hat{q}_1}$ и $\hat{R}_{\hat{q}_2}$ соответственно. Тогда две обобщенные α-формулы Φ над $\{F_1, F_2\}$ и $\hat{\Phi}$ над $\{\hat{F}_1, \hat{F}_2\}$ будем называть двойственными, если $(\widetilde{\alpha}_1, \widetilde{\alpha}_2, \ldots \widetilde{\alpha}_{2^n})$ является последовательностью, в которой наборы подаются на формулу Φ, и обобщенная формула Φ имеет начальную формулу

$$f_{i_{n-1}}(x_{i_1}, f_{i_{n-2}}(x_{i_2}, \ldots f_{i_1}(x_{i_{n-1}}, x_{i_n})) \ldots),$$

а последовательностью, в которой наборы подаются на формулу $\hat{\Phi}$, является последовательность $(\widetilde{\overline{\alpha}}_1, \widetilde{\overline{\alpha}}_2, \ldots \widetilde{\overline{\alpha}}_{2^n})$, $\hat{\Phi}$ имеет начальную формулу

$$g_{i_{n-1}}(x_{i_1}, g_{i_{n-2}}(x_{i_2}, \ldots g_{i_1}(x_{i_{n-1}}, x_{i_n})) \ldots).$$

Утверждение 1. *Двойственные обобщенные формулы Φ и $\hat{\Phi}$ реализуют двойственные функции.*

Из этого утверждения следует, что для обобщенных α-формул верен принцип двойственности. По принципу двойственности из теоремы 1 можно вывести следующую теорему.

Пусть W — конечный автомат с двумя входами и одним выходом, такой, что $\{q_1, q_2\}$ — множество его состояний, в состоянии q_1 автомат реализует функцию $x_1 \& x_2$, в состоянии q_2 — функцию $1(x_1, x_2)$ и автомат в момент времени t переходит из состояния q_i в состояние q_j тогда и только тогда, когда входные символы в этот момент времени совпадают и равны 1, $i \neq j$. Обозначим через G автоматную функцию, реализуемую инициальным автоматом W_{q_1}.

Обозначим через $\mathcal{B}_n$ множество всех обобщенных α-формул над $\{G\}$ от n переменных, в которую каждая переменная входит ровно один раз.

Теорема 2. *Любая обобщенная α-формула из множества $\mathcal{B}_n$ является универсальной для множества $T_1(n)$, $n \geqslant 2$.*

Следует отметить, что ни для какой булевой функции f, не принадлежащей множеству $T_1(n)$, не существует обобщенной α-формулы из множества $\mathcal{B}_n$, реализующей эту функцию, $n \geqslant 2$.

Автор благодарен А. Б. Угольникову за постановку задачи и выражает искреннюю признательность О. С. Дудаковой за обсуждение результатов работы.

Работа выполнена при поддержке РФФИ, проект № 14-01-00598.

Литература

[1] *Яблонский С. В.* Введение в дискретную математику. — М.: Высшая школа, 2006. — 384 с.

[2] Конспект лекций О. Б. Лупанова по курсу “Введение в математическую логику” // Отв. ред. А. Б. Угольников. — М.: Изд-во ЦПИ при механико-математическом факультете МГУ имени М. В. Ломоносова, 2007. — 191 с.

[3] *Глухов М. М.* Об α-замкнутых классах и α-полных системах функций k-значной логики // Дискретная математика. — 1989. — Т. 1, № 1. — С. 16–21.

[4] *Чернышев А. Л.* Условия α-полноты систем функций многозначной логики // Дискретная математика. — 1992. — Т. 4, № 4. — С. 117–130.

[5] *Шабунин А. Л.* Примеры α-полных систем k-значной логики при $k = 3, 4$ // Дискретная математика. — 2006. — Т. 18, № 4. — С. 45–55.

[6] *Трущин Д. В.* О глубине α-пополнения систем булевых функций // Вестник Московского университета. Сер. 1. Математика. Механика.. — 2009. — № 2. — С. 72–75.

[7] *Трущин Д. В.* Об оценках глубины α-пополнений систем функций трехзначной логики // Проблемы теоретической кибернетики: Мат-лы XVI международной конф. — Н. Новгород: Изд-во ННГУ, 2011.— С. 484–487.

[8] *Сысоева Л. Н.* О некоторых свойствах обобщенных α-формул // Вестник Московского университета. Сер. 1. Математика. Механика.. — 2013. — № 4. — С. 51–55.

Несократимые разложения однородных произведений двучленов для построения m-устойчивых функций с максимально возможной нелинейностью

Ю. В. Таранников

taran@butovo.com

Механико-математический факультет, Московский государственный университет им. М. В. Ломоносова, г. Москва

Пусть $n, k \in \mathbf{Z}$, $0 \leqslant 2k \leqslant n$. Будем называть (n,k)-продуктом (или просто *продуктом*) произведение двучленов: $P = \prod\limits_{i=1}^{k}(x_{i,1} + x_{i,2})$, где $x_{i,1}, x_{i,2}$, $i = 1, \ldots, k$, — неповторяющиеся переменные из множества $x_1, \ldots, x_n$. *Разложением* (n,k)-продукта P назовем совокупность 2^k мономов длины k, получающихся после раскрытия скобок в продукте P. Считаем, что разложением $(n,0)$-продукта является моном длины 0. Будем говорить, что разложение суммы продуктов $\sum\limits_{i=1}^{s} P_i$ *несократимо*, если разложения никаких двух продуктов P_i и P_j, $i \neq j$, не содержат общих мономов. Число s продуктов в сумме назовем *длиной* суммы продуктов. Через $A_{n,k}$ обозначим максимально возможное значение длины суммы (n,k)-продуктов с несократимым разложением.

Утверждение 1. *Справедливы следующие соотношения:*

а) $A_{n,k} \leqslant \frac{\binom{n}{k}}{2^k}$;
б) $A_{n,k} \leqslant \binom{n}{2k}$;
в) $A_{n,k} \geqslant \binom{\lfloor \frac{n}{2} \rfloor}{k}$;
г) $A_{n,k} \geqslant A_{n-2,k} + A_{n-2,k-1}$ *при* $2 \leqslant 2k \leqslant n-2$;
д) $A_{n,0} = 1$;
е) $A_{n,1} = \lfloor \frac{n}{2} \rfloor$;
ж) $A_{n,2} = \binom{\frac{n}{2}}{2}$ *при четном* n;
з) $A_{n,\lfloor \frac{n}{2} \rfloor} = 1$;
и) $A_{n,\frac{n}{2}-1} = \frac{n}{2}$ *при четном* n;
к) $A_{10,3} = 15$.

Пример суммы продуктов, на которой достигается значение $A_{10,3} = 15$, приведен (в другой терминологии) в [1]:

$$\begin{aligned}
&(x_1+x_2)(x_3+x_4)(x_5+x_6)+(x_1+x_2)(x_4+x_6)(x_8+x_9)+(x_1+x_2)(x_7+x_9)(x_8+x_{10})+\\
&+(x_1+x_3)(x_2+x_5)(x_7+x_8)+(x_1+x_4)(x_5+x_7)(x_6+x_9)+(x_1+x_5)(x_2+x_3)(x_9+x_{10})+\\
&(x_1+x_6)(x_3+x_{10})(x_4+x_8)+(x_1+x_7)(x_2+x_{10})(x_5+x_6)+(x_1+x_{10})(x_2+x_7)(x_3+x_4)+\\
&+(x_2+x_8)(x_3+x_7)(x_4+x_9)+(x_2+x_9)(x_5+x_{10})(x_6+x_8)+(x_3+x_5)(x_4+x_9)(x_7+x_{10})+\\
&+(x_3+x_5)(x_6+x_8)(x_7+x_{10})+(x_3+x_8)(x_4+x_6)(x_5+x_9)+(x_4+x_7)(x_6+x_{10})(x_8+x_9).
\end{aligned} \tag{1}$$

Верхняя оценка в утверждении 1, к) следует из оценки утверждения 1, а). Пример (1) можно рассматривать как *совершенную упаковку* продуктов, поскольку каждый из $\binom{10}{3} = 120 = 15 \cdot 2^3$ мономов длины 3 от 10 переменных встречается в разложении ровно одного продукта.

Булева функция от n переменных — это отображение из $\mathbf{F}_2^n$ в $\mathbf{F}_2$.

Вес $\mathrm{wt}(f)$ функции f над $\mathbf{F}_2^n$ — это число наборов x из $\mathbf{F}_2^n$, для которых $f(x) = 1$. *Подфункцией* булевой функции f называется функция f', полученная подстановкой в f некоторых констант 0 или 1 вместо некоторых переменных.

Для двух булевых функций f_1 и f_2 на $\mathbf{F}_2^n$ расстояние между f_1 и f_2 определяется как $d(f_1, f_2) = |\{x \in \mathbf{F}_2^n | f_1(x) \neq f_2(x)\}|$. Для заданной функции f из $\mathbf{F}_2^n$ минимум расстояний $d(f, l)$, где l пробегает множество всех аффинных функций (т. е. функций, в полиноме Жегалкина которых отсутствуют слагаемые длины больше чем 1), называется *нелинейностью* функции f и обозначается через $\mathrm{nl}(f)$.

Булева функция f от n переменных называется m-устойчивой, если $\mathrm{wt}(f') = 2^{n-m-1}$ для любой ее подфункции f' от $n-m$ переменных. Нелинейность и m-устойчивость относятся к числу важнейших криптографических характеристик булевых функций.

Если f является m-устойчивой булевой функцией на $\mathbf{F}_2^n$, $m \leqslant n-2$, то выполнено

$$\mathrm{nl}(f) \leqslant 2^{n-1} - 2^{m+1}. \tag{2}$$

Область значений параметров, для которых построены функции, на которых достигается равенство в (2), неоднократно расширялась. В 2014 году в [1] построены функции, достигающие равенства в (2), для $m \geqslant 0.5789...n(1+o(1))$.

Теорема 1. *Пусть* $n, C_k \in \mathbf{N}$, $C_k \leqslant A_{n,k}$, $k = 0, 1, 2, \ldots, \lfloor \frac{n}{2} \rfloor$. *Положим* $C = \frac{1}{1+\log_2 X_{max}}$, *где* X_{max} — *старший корень многочлена* $x^n - \sum\limits_{k=0}^{\lfloor \frac{n}{2} \rfloor} C_k x^k$. *Тогда для любого* $\varepsilon > 0$ *начиная с некоторого* n_0 *для всех пар* (n, m), *таких что* $\frac{m}{n} > C + \varepsilon$, $n \geqslant n_0$, $m \leqslant n-2$, *существует* m-*устойчивая функция от* n *переменных, на которой достигается равенство в (2).*

Частный случай теоремы 1 для $n = 10$, $C_0 = 1$, $C_1 = 5$, $C_2 = 10$, $C_3 = 15$, $C_4 = 5$, $C_5 = 1$, $X_{\max} = 1.6556...$, $C = 0.5789...$ был доказан в [1].

Работа выполнена при поддержке РФФИ, проект № 13-01-00183-а.

Литература

[1] *Tarannikov Y. V.* Generalized proper matrices and constructing of m-resilient Boolean functions with maximal nonlinearity for expanded range of parameters // Siberian Electronic Mathematical Reports. — 2014. — V. 11. — P. 229–245. (http://semr.math.nsc.ru/v11/p229-245.pdf)

О некоторых необходимых условиях равномерности систем функций многозначной логики

П. Б. Тарасов

tarasov.p.b@gmail.com

МГУ имени М. В. Ломоносова , Москва

В работе рассматривается задача о реализации функций k-значной логики из замкнутых классов формулами в конечных базисах, состоящих из функций, принадлежащих этим же классам. Все необходимые определения можно найти в работах [1, 2, 3].

Обозначим через P_k множество всех функций k-значной логики, а через $P_{k,s}$ — множество всех функции k-значной логики, принимающих значения из множества $E_s = \{0, 1, \ldots, s-1\}$, $k \geqslant s \geqslant 2$. Пусть A — конечная система функций из P_k. Через $[A]$ обозначим замкнутый класс, порожденный системой A. Пусть Φ — формула над A. Обозначим через $L(\Phi)$ (сложность формулы Φ) число символов переменных, входящих в Φ, а через $D(\Phi)$ — глубину формулы Φ. Пусть $f \in [A]$. Положим $D_A(f) = \min D(\Phi)$, $L_A(f) = \min L(\Phi)$, где минимум берется по всем формулам Φ над A, реализующим f. Конечную систему функций A будем называть равномерной, если существуют такие константы c и d (зависящие только от A), что для любой функции $f \in [A]$ выполнено неравенство

$$D_A(f) \leqslant c \log_2 L_A(f) + d.$$

В работах [4, 5] доказана равномерность всех конечных полных систем булевых функций (см. также [6]). В [7] установлена равномерность всех конечных систем, порождающих класс M всех монотонных булевых функций. В работе [3] доказана равномерность всех конечных систем булевых функций (см. также [8, 9]).

Ряд публикаций посвящен задаче о соотношении глубины и сложности формул над конечными системами функций многозначной логики. Известно (см. [3]), что не все конечные системы функций многозначной логики равномерны. В работах [10, 11, 12, 13] изучается вопрос о равномерности конечных систем, порождающих предполные классы в P_k, $k \geqslant 3$. Более подробный обзор результатов по данной тематике дано в работе автора [13].

Функцию $f(x_1, \ldots, x_n) \in P_k$ будем называть мажоритарной функцией, если для любых $\alpha, \beta \in E_k$ выполнены равенства

$$f(\beta, \alpha, \ldots, \alpha) = f(\alpha, \beta, \alpha, \ldots, \alpha) = \ldots = f(\alpha, \ldots, \alpha, \beta) = \alpha.$$

Пусть $f(x_1, \ldots, x_n)$ — функция из $P_{k,s}$, а $g(x_1, \ldots, x_n)$ — функция из P_s, такие, что для любого $\widetilde{\alpha} \in E_s^n$ выполнено равенство $f(\widetilde{\alpha}) = g(\widetilde{\alpha})$. Функцию g будем называть проекцией функции f на множество P_s и обозначать через $\mathrm{pr_s} f$. Для произвольной системы $A \subseteq P_{k,s}$ положим $\mathrm{pr_s} A = \cup \mathrm{pr_s}\{f\}$, где объединение берется по всем функциям f из A. В работе [12] доказана равномерность всех конечных систем функций из $P_{k,2}$, проекция которых порождает класс M.

В работе автора [13] доказана равномерность любой конечной системы функций A из $P_{k,s}$, такой, что $[\mathrm{pr_s}A]$ содержит мажоритарную функцию. Как следствие получено, что все конечные системы функций $P_{k,2}$, проекция которых целиком не содержится в классах K, D, L, O^∞ или I^∞ равномерны (определения классов см [14]). Для любого замкнутого класса булевых функций B содержащегося целиком в одном из классов K, D, L, O^∞ и I^∞, нетрудно построить пример неравномерной системы функций из $P_{3,2}$, в проекции порождающий класс B, обобщив пример из работы [3].

Введем на множестве E_k частичный порядок следующим образом: $1 > 0$, а остальные элементы несравнимы. Далее под монотонными функциями будем понимать функции, монотонные относительно данного частичного порядка. Заметим, что если $f \in P_{k,2}$ — монотонная функция, то $\mathrm{pr}f \in M$, где M — класс монотонных булевых функций.

Пусть $f(x_1, \ldots, x_n) \in P_{k,2}$, $i \in \{1, \ldots, n\}$. Положим

$$M_f^{x_i} = \bigcup_{\widetilde{\alpha} \in E_k^{n-1}} \{\mathrm{pr}_2 f(\alpha_1, \ldots, \alpha_{i-1}, x, \alpha_i, \ldots, \alpha_{n-1})\},$$

$$V_f^i = \bigcup_{\substack{\widetilde{\alpha} \in E_k^{n-1} \\ \mathrm{pr}f(\alpha_1, \ldots, \alpha_{i-1}, y, \alpha_i, \ldots, \alpha_{n-1}) = x}} \{\widetilde{\alpha}\}.$$

Пусть A — конечная система монотонных функций из $P_{k,2}$, будем говорить, что A обладает свойством $\#$, если существует такое $q = q(A) \geqslant 3$, что для любой функции $f(x_1, \ldots, x_n) \in A$, любой ее переменной x_i и любого $\widetilde{\alpha} \in V_f^{x_i}$, существует функция $g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n, y_1, \ldots, y_q) \in [A]$, такая, что для любого $\widetilde{\beta} \in V_f^{x_i}$, выполнено $\mathrm{pr}g(\widetilde{\beta}, \widetilde{y}) \notin [\{0, 1\}]$ и

1. если $M_f^{x_i} = \{0, 1, x\}$, то выполнено $g(\widetilde{\alpha}, \widetilde{y}) \in M_{01} \setminus (O^\infty \cup I^\infty)$;
2. если $M_f^{x_i} = \{0, x\}$, то выполнено $g(\widetilde{\alpha}, \widetilde{y}) \in M_{01} \setminus O^\infty$;
3. если $M_f^{x_i} = \{1, x\}$, то выполнено $g(\widetilde{\alpha}, \widetilde{y}) \in M_{01} \setminus I^\infty$.

Получены следующие результаты:

Теорема 1. *Конечная система монотонных функций из $P_{k,2}$ равномерна только тогда, когда она обладает свойством $\#$.*

Теорема 2. *Пусть A — конечная система функций из $P_{k,2}$, обладающая свойством $\#$. Пусть функции из A зависят не более чем от n переменных. Тогда $q(A) \leqslant n^{k^n}$.*

Теорема 3. *Пусть A — конечная система функций из $P_{k,2}$, обладающая свойством $\#$. Тогда существуют такие константы c и d, что для любой функции $f \in [A]$ выполнено неравенство $l_A(f) \leqslant c \log_2^2 L_A(f) + d$.*

Автор благодарен А. Б. Угольникову за постановку задачи и обсуждение результатов работы.

Работа выполнена при финансовой поддержке РФФИ (проект № 14-01-00598) и программы фундаментальных исследований ОМН РАН «Алгебра-

ические и комбинаторные методы математической кибернетики и информационные системы нового поколения» (проект «Задачи оптимального синтеза управляющих систем»).

Литература

[1] *Яблонский С. В.* Введение в дискретную математику. М.: Высшая школа, 2001.

[2] *Lau D*, Function Algebras on Finite Sets. Berlin; Heidelberg:Springer-Verlag, 2006.

[3] *Угольников А. Б.* О глубине и полиномиальной эквивалентности формул для замкнутых классов двузначной логики // Матем. заметки. 1987. **42**, вып. 4. 603–612.

[4] *Яблонский С.В., Козырев В.П.* Математические вопросы кибернетики // Информационные материалы Научного совета по комплексной проблеме "Кибернетика" АН СССР. Вып. 19а. М.: 1968. 3–15

[5] *Spira P. M.* On time-hardware complexity tradeoffs for Boolean functions // Proc. 4th Hawai Symp. on System Sciences. North Hollywood: Western Periodicals Company, 1971. 525–527.

[6] *Храпченко В.М.* О соотношении между сложностью и глубиной формул / / Методы дискретного анализа в синтезе управляющих систем. Вып. 32. Новосибирск: ИМ СО АН СССР, 1978. 76–94 .

[7] *Wegener I.* Relating monotone formula size and monotone depth of Boolean functions // Inform. Proces. Let.. 1983. **16**. 41–42.

[8] *Угольников А.Б.* О полиномиальной эквивалентности формул для замкнутых классов двухзначной логики // VII Всесоюз. конф. "Проблемы теоритической кибернетики": тез. докл. Часть 1. Иркутск: Изд-во Иркут. гос. ун-та. 1985, 194-195.

[9] *Ragaz M. E.* Parallelizable algebras // Arch. fur math. Log. und Grundlagenforsch. (1986/87). **26**. 77–99.

[10] *Сафин Р.Ф.* О соотношении между глубиной и сложностью формул для предполных классов k-значной логики // Математические вопросы кибернетики. М: Физматлит, 2004. 223–278.

[11] *Дудакова О.С.* О классах функций k-значной логики, монотонных относительно множеств ширины 2 // Вестн. Моск. ун-та. Матем. Механ. 2008. 1. 31–37.

[12] *Тарасов П. Б.* О равномерности некоторых систем функций многозначной логики // Вестн. Моск. ун-та. Матем. Механ. 2013. 2. 61–64.

[13] *Тарасов П.Б.* О некоторых достаточных условиях равномерности систем функций многозначной логики // Вестн. Моск. ун-та. Матем. Механ. 2013. 5. 41-46.

[14] *Угольников А.Б.* Классы Поста. — М.: Изд-во ЦПИ при мех.-мат. ф.-те МГУ. 2008.

О приближенной билинейной сложности умножения матриц размера 2x2 и 2x6

А. П. Трефилов

s40320@yandex.ru

МГУ им. Ломоносова, Москва

Введение. Задача определения точного и приближенного рангов даже малых матричных тензоров и по сей день не имеет удовлетворительного решения. Во многом это обусловлено тем, что решение данной задачи тесно связано с решением некоторой системы нелинейных уравнений третьей степени, приемлемого алгоритма для решения которой не найдено до сих пор.

Пусть $\|a_{ij}\|_{m\times n}$ обозначает матрицу размера $m \times n$ над некоторым полем. Задача умножения матрицы $\|a_{ij}\|_{m\times n}$ на матрицу $\|b_{kl}\|_{n\times p}$ состоит (по определению) в вычислении mp билинейных форм вида $\sum\limits_{j=1}^{n} a_{ij}b_{jl}$. Если вычислять эти формы непосредственно по указанным формулам, то необходимо сделать mnp умножений и почти столько же сложений. В 1969 г. Штрассен [1] показал, что существуют более быстрые по порядку алгоритмы умножения матриц, построив алгоритм с числом операций над элементами поля $O(n^{\log_2 7})$. К 1986 г. эта оценка трудами многих авторов была понижена до $O(n^{2.38})$ [4] (см. обзор [5]), однако с тех пор эта оценка существенно не улучшена, несмотря на большой интерес к задаче.

Алгоритм Штрассена [1] для умножения матриц порядка n со сложностью $O(n^{\log_2 7})$ основан на найденном им алгоритме умножения двух квадратных матриц порядка 2 с 7 умножениями вместо 8 в обычном алгоритме. Затем этот алгоритм рекурсивно использовался для умножения матриц порядка 2^k с числом умножений 7^k. Для того чтобы можно было применять рекурсию, Штрассен построил специального вида алгоритм умножения матриц порядка 2, а именно так называемый билинейный алгоритм.

О п р е д е л е н и е. Пусть F – некоторое кольцо, и пусть имеется 2 множества переменных $A = \{a_1, a_2, \ldots, a_r\}$ и $B = \{b_1, b_2, \ldots, b_s\}$. *Билинейными алгоритмами* над A и B и кольцом F называются алгоритмы, в которых сначала вычисляются произведения $(\sum\limits_{i=1}^{r} \alpha_i^t a_i) \times (\sum\limits_{j=1}^{s} \beta_j^t b_j)$ некоторых линейных форм (с коэффициентами из F) от первого множества переменных на некоторые линейные формы (с коэффициентами из F) от второго множества переменных, где $t = 1, 2, ..., d$, а на втором этапе вычисляются некоторые линейные комбинации этих d произведений. При этом число умножений d называется *билинейной сложностью алгоритма.*

О п р е д е л е н и е. Будем говорить, что билинейный алгоритм над кольцом F вычисляет систему билинейных форм $C_k = \sum\limits_{i=1}^{r}\sum\limits_{j=1}^{s} c_{ij}^k a_i b_j, k = 1, ..., h$, где c_{ij}^k – произвольные константы из F , если каждая из этих билинейных форм оказывается вычисленной на втором этапе алгоритма. *Билинейной сложностью задачи* вычисления системы билинейных форм над кольцом F называется ми-

нимальная билинейная сложность алгоритмов над F, вычисляющих данную систему билинейных форм.

Требование билинейности при изучении сложности умножения матриц связано с тем, что при рекурсии вместо переменных подставляются матрицы, которые могут не коммутировать. Кроме обычных (точных) билинейных алгоритмов, определенных выше, рассматривают также приближенные билинейные алгоритмы. Интерес к ним связан как с возможностью практического использования, так и с тем фактом, что, имея быстрый приближенный билинейный алгоритм для умножения матриц, можно получать асмптотически быстрый точный алгоритм для умножения матриц (см. обзор [5]).

О п р е д е л е н и е. Приближенным билинейным алгоритмом над двумя множествами переменных A и B и кольцом F называется любой билинейный алгоритм над теми же множествами переменных, в котором в качестве коэффициентов вместо кольца F используется кольцо лорановских многочленов от x с коэффициентами из F. При этом число умножений линейных форм d также называется билинейной сложностью приближенного алгоритма.

О п р е д е л е н и е. Пусть зафиксированы множества переменных A и B и кольцо F. Будем говорить, что приближенный билинейный алгоритм приближенно вычисляет систему билинейных форм C_k над F , $k = 1, \ldots, h$, если на втором этапе этот билинейный алгоритм строит выражения вида $C_k + O(x)$ для всех $k = 1, \ldots, h$, где O(x) – многочлены, содержащие только положительные степени x (коэффициентами многочленов автоматически являются билинейные формы).

О п р е д е л е н и е. Приближенной билинейной сложностью задачи вычисления системы билинейных форм над кольцом F называется минимальная билинейная сложность приближенных билинейных алгоритмов над F, приближенно вычисляющих данную систему билинейных форм. Обозначим через $r(K)$ приближенную билинейную сложность задачи умножения матрицы 2x2 на матрицу 2xK.

Получение приближенного алгоритма для $r(6)$ является важным шагом на пути к решению более общей задачи - вычислению $r(N)$. Первые шаги в этом направлении были сделаны итальянскими математиками [2], они получили верхнюю оценку, равную 10 для $r(3)$. В работе [3] были установлены верхние оценки 13 для $r(4)$ и 16 для $r(5)$. В данной статье получена верхняя оценка 19 для $r(6)$.

Т е о р е м а 1. Величина $r(6)$ не превосходит 19.

С л е д с т в и е. Верхняя оценка для $r(N)$ не превосходит $\lceil \frac{19N}{6} \rceil$.

Заключение. Данный результат получен с помощью разработанных автором новых методов компьютерного поиска, являющихся развитием методов [6].

Работа выполнена при финансовой поддержке РФФИ, проект №12–01–91331-ННИОа.

Литература

[1] *Strassen V.* : Gaussian elimination is not optimal // Numerische Mathemetik. 1969. **13**. S. 354–356.

[2] *Bini D. Capovani M. Lotti G. Romani F.* : $O(n^{2.7799})$ complexity for approximate matrix multiplication // Inform. Process. Lett. 1979. P. 87–97.

[3] *Алексеев В. Б., Смирнов А. В.* : О точной и приближенной билинейных сложностях умножения матриц размеров 4 x 2 и 2 x 2. Современные проблемы математики. Вып. 17. 2013. С. 135–152.

[4] *Coppersmith D. Winograd S.* , Matrix multiplication via arithmetic progressions // J. Symbolic Comput. 1990. **9** №3, P. 251—280.

[5] *Алексеев В. Б.* , Сложность умножения матриц. Обзор. Кибернетический сборник. Вып. 25. 1988. М.: Мир, С. 189—236.

[6] *Смирнов А. В.* , О билинейной сложности и практических алгоритмах умножения матриц // ЖВМ и МФ. 2013. **53** №12. С. 1970–1984

Влияние шума на квантовые блуждания по графам

Л. Е. Федичкин, А. А. Мельников

leonid@phystech.edu, melnikov@phystech.edu

Физико-технологический институт РАН, Москва[1] НИКС, Москва[2] Московский физико-технический институт, Долгопрудный[3] Институт квантовой оптики и квантовой информации ААН, Инсбрук, Австрия[4] Инсбрукский университет, Инсбрук, Австрия[5]

В данной работе изучается обработка квантовой информации в присутствии шумов, которые приводят к появлению ошибок. В настоящее время известно несколько способов манипуляции квантовыми состояними, хранящими информацию. В частности, доказано, что с помощью систем, выполняющих квантовые блуждания по графам, возможно эффективно производить универсальные квантовые вычисления. Ряд практически полезных квантовых алгоритмов имеют в своей основе организацию квантового блуждания по специально построенным графам. Помимо вышеперечисленных достоинств, квантовые блуждания являются наиболее естественным физическим процессом по сравнению с альтернативными методами обработки информации, в том числе с традиционным методом квантовых вычислений посредством построения квантовых схем. В нашем исследовании в качестве примера системы, в которой могут блуждать квантовые частицы, мы рассматриваем кольца из квантовых точек в кремнии [1, 2, 3, 4]. Рассматривается граф из N узлов, образующих замкнутое кольцо. Частица, блуждающий по кольцу, представляет единицу квантовой информации, кудит. В отсутствие шумов вероятность нахождения частицы в 0-ом узле определяется выражением

$$P_0(t) = \frac{1}{N} + \sum_{m,n=0}^{N-1} \frac{1 - \delta_{m+n,0} - \delta_{m+n,N}}{N^2} e^{4i\Omega t \sin\frac{\pi(m+n)}{N} \cos\frac{\pi(m-n)}{N}},$$

где Ω является частотой перескоков между узлами графа.

Блуждания частицы изменяют состояние одного кудита. Для управления состоянием двух кудитов необходимо контролировать степень их запутанности. Гамильтониан

$$H = \Omega \sum_{i,j=0;j\neq i,i+1}^{N-1} |x_{i+1}, y_j\rangle \langle x_i, y_j| + |x_i, y_j\rangle \langle x_{i+1}, y_j| +$$

$$+ |x_j, y_{i+1}\rangle \langle x_j, y_i| + |x_j, y_i\rangle \langle x_j, y_{i+1}|$$

описывает эволюцию двух частиц в кольцевом графе. Отталкивающиеся частицы делают невозможным их нахождение в одной и близлежащих квантовых точках. Отталкивание приводит к квантовым корреляциям между кудитами.

Для считывания квантовой информации необходимо использовать измерители тока. Но данные измерители являются источниками шумов в системе. Матрица плотности, описывающая состояние двух блуждающих частиц, изменяется ввиду гамильтониана прыжков H и шума, вызванного измерителями, с параметром Γ:

$$\rho' = e^{-iHt/\hbar} \left(e^{-\Gamma t}\rho + (1 - e^{-\Gamma t})\rho_M\right) e^{iHt/\hbar}. \quad (1)$$

Решение уравнения (1) и использование мер квантовой запутанности позволяют продемонстрировать аннигиляцию перепутанности при определенных уровнях шума. Тем не менее, мы обнаружили положительную роль шума. Для области параметров наблюдается повторное появление перепутанности, которое следовало за аннигиляцией. Данный механизм можно использовать для создания квантовых операций.

Литература

[1] *Solenov D., Fedichkin L.* Continuous-time quantum walks on a cycle graph // Physical Review A 73, 1 (2006)

[2] *Fedichkin L., Solenov D., Tamon C.* Mixing and Decoherence in Continuous-Time Quantum Walks on Cycles // Quantum Information and Computation 6:3 (2006)

[3] *Melnikov A. A., Fedichkin L. E.* Two-particle fermionic quantum walks on a cycle graph // arXiv:1310.0420 (2013)

[4] *Мельников А. А., Федичкин Л. Е.* Квантовые блуждания идентичных частиц // Труды ФТИАН, подготовлена к печати

Подход Ляпунова-Яблонского при построении и исследовании модели управляющих систем обслуживания конфликтных потоков

М. А. Федоткин, М. А. Рачинская

fma5@rambler.ru, rachinskaya.maria@gmail.com

Нижегородский государственный университет им. Н.И. Лобачевского, Нижний Новгород

Общее описание системы на содержательном уровне

В работе рассматривается процесс управления конфликтными неординарными пуассоновскими потоками в классе циклических алгоритмов. Предполагается, что на обслуживание поступает m статистически независимых потоков $\Pi_1, \Pi_2, \ldots, \Pi_m$. В каждый вызывающий момент по потоку Π_j, где $1 \leqslant j \leqslant m$, поступает с интенсивностью λ_j пачка из одного, двух или трех требований с вероятностями p_j, q_j и s_j соответственно. Кроме того, входные потоки являются конфликтными, т. е. одновременное обслуживание требований различных потоков невозможно. Выбран циклический алгоритм смены фаз (состояний) $\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(2m)}$ обслуживающего устройства с длительностями $T_1, T_2, \ldots, T_{2m}$ соответственно. В этом случае последовательность переходов состояний обслуживающего устройства имеет вид $\Gamma^{(1)} \to \Gamma^{(2)} \to \ldots \to \Gamma^{(2m-1)} \to \Gamma^{(2m)} \to \Gamma^{(1)} \to \ldots$ При этом в каждой фазе $\Gamma^{(2j-1)}$ происходит обслуживание только потока Π_j, а последующая фаза $\Gamma^{(2j)}$ служит для переналадки обслуживающего устройства и обслуживание никаких требований в ней не ведется. Рассматривается система обслуживания с ожиданием и без потерь. За время T_{2j-1} в состоянии $\Gamma^{(2j-1)}$ может быть обслужено не более l_j заявок потока Π_j. Результаты исследований интерпретируются на задаче регулирования движения транспортных потоков на локальном перекрестке. В частности, в работе [1] было показано, что транспортные потоки, двигающиеся по магистралям в условиях, препятствующих свободному обгону, хорошо аппроксимируются неординарными пуассоновскими потоками.

Построение и исследование вероятностной модели

Следуя кибернетическому подходу Ляпунова-Яблонского в данной системе обслуживания требований и управления потоками были выделены схема, информация, координаты и функции. Схема системы, описывающая ее конструкцию, содержит следующие блоки: 1) входные полюса первого и второго типов: m неординарных пуассоновских потоков $\Pi_1, \Pi_2, \ldots, \Pi_m$ и соответственно m потоков насыщения $\Pi_1^*, \Pi_2^*, \ldots, \Pi_m^*$, которые являются выходными потоками системы при ее максимальной загрузке и экстремальном обслуживании требований; 2) внешняя память: очереди $O_1, O_2, \ldots, O_m$ соответственно по потокам $\Pi_1, \Pi_2, \ldots, \Pi_m$; 3) блок по переработке внешней памяти: экстремальная стратегия обслуживания, при которой из очереди O_j на обслуживание выбирается максимально возможное количество заявок, но не превышающее

величины l_j; 4) внутренняя память: обслуживающее устройство с $2m$ вышеуказанными состояниями; 5) блок по переработке внутренней памяти: циклический алгоритм смены состояний обслуживающего устройства; 6) выходные полюса: выходные потоки $\Pi'_1, \Pi'_2, \ldots, \Pi'_m$ управляющей системы. Информация системы есть математическое описание или кодирование всех ее блоков. Например, информация о входном полюсе первого типа задается неординарным пуассоновским потоком. Таким образом, набор состояний всех блоков системы и механизмы их смены образуют информацию исследуемой кибернетической системы. Координатами являются номер заявки по каждому потоку, номер входного потока и потока насыщения, номер заявки на обслуживание в очереди, номер очереди, номер состояния обслуживающего устройства, номер обслуженной заявки, номер выходного потока. Функция изучаемой системы содержательно есть управление входными потоками и непосредственно обслуживание заявок.

Положим, что наблюдение за системой начинается в некоторый момент τ_0 переключения состояния обслуживающего устройства. Особенность предлагаемого подхода заключается в отказе от рассмотрения системы в непрерывном времени в пользу отслеживания ее состояний в случайные дискретные моменты $\tau_i, i = 0, 1, \ldots$, смены фазы обслуживающего устройства. Для дальнейшей формализации описания системы по потоку Π_j необходимо ввести следующие случайные элементы: 1) $æ_{j,i} \in \{0, 1, \ldots\}$ — число машин, находящихся в очереди в момент времени τ_i; 2) $\Gamma_i \in \{\Gamma^{(1)}, \Gamma^{(2)}, \ldots, \Gamma^{(2m)}\}$ — состояние обслуживающего устройства на промежутке $[\tau_i, \tau_{i+1})$; 3) $\eta_{j,i}$ — число машин, поступивших в систему за промежуток $[\tau_i, \tau_{i+1})$; 4) $\xi_{j,i}$ — максимальное число требований, которые могут быть обслужены за промежуток $[\tau_i, \tau_{i+1})$; 5) $\xi'_{j,i}$ — реальное число требований, которые были обслужены за промежуток $[\tau_i, \tau_{i+1})$; 6) $\xi'_{j,-1}$ — реальное число требований, которые были обслужены за промежуток $[0, \tau_0)$. Поскольку входные потоки независимы и алгоритм управления потоками циклический, то исследование процесса обслуживания можно проводить по каждому из m потоков отдельно. Применение кибернетического подхода позволило установить, что динамика изменения состояния системы обслуживания по потоку Π_j определяется рекуррентным соотношением $(\Gamma_{i+1}, æ_{j,i+1}, \xi'_{j,i}) = (u(\Gamma_i), \max\{0, æ_{j,i} + \eta_{j,i} - \xi_{j,i}\}, \min\{æ_{j,i} + \eta_{j,i}, \xi_{j,i}\})$ для трехмерной случайной последовательности $\{(\Gamma_i, æ_{j,i}, \xi'_{j,i-1}); i = 0, 1, \ldots\}$. Здесь функция $u(\Gamma^{(k)}) = \Gamma^{(k+1)}$ при $k = 1, 2, \ldots, 2m-1$ и $u(\Gamma^{(2m)}) = \Gamma^{(1)}$. Относительно этой векторной последовательности, являющейся вероятностной моделью рассматриваемой системы, было доказано следующее.

Теорема 1. *Для фиксированного $1 \leqslant j \leqslant m$ случайная векторная последовательность $\{(\Gamma_i, æ_{j,i}, \xi'_{j,i-1}); i = 0, 1, \ldots\}$ с заданным начальным распределением вектора $(\Gamma_0, æ_{j,0}, \xi'_{j,-1})$ является управляемой однородной марковской цепью со счетным числом состояний. Причем ее пространство состояний разбито на два класса: незамкнутое подмножество несущественных состояний и замкнутое подмножество существенных состояний с периодом $2m$.*

Были получены рекуррентные соотношения для одномерных распределений процесса обслуживания вида $\{(\Gamma_i, æ_{j,i}, \xi'_{j,i-1}); i = 0, 1, \ldots\}$, а также для

их производящих функций. Для указанного процесса обслуживания по потоку Π_j были доказаны следующие утверждения.

Теорема 2. *Необходимым и достаточным условием существования единственного стационарного режима в системе по потоку* Π_j *является выполнение неравенства* $\lambda_j T(2s_j + q_j + 1) - l_j < 0$.

Теорема 3. *При* $\lambda_j T(2s_j + q_j + 1) - l_j < 0$ *стационарные вероятности* $Q_j(\Gamma^{(2j)}, 0, y) = \lim\limits_{i\to\infty} \mathbf{P}(\Gamma_i = \Gamma^{(2j)}, æ_{j,i} = 0, \xi'_{j,i-1} = y), y = 0, 1, \ldots, l_j$, *управляемой марковской последовательности* $\{(\Gamma_i, æ_{j,i}, \xi'_{j,i-1}); i = 0, 1, \ldots\}$ *являются единственным решением системы линейных уравнений вида*

$$Q_j(\Gamma^{(2j)}, 0, 0) - \exp\{-\lambda_j T\}/(1 - \exp\{-\lambda_j T\}) \sum_{y=1}^{l_j} Q_j(\Gamma^{(2j)}, 0, y) = 0,$$

$$2m \sum_{y=0}^{l_j-1} Q_j(\Gamma^{(2j)}, 0, y)(l_j - y) = l_j - \lambda_j T(1 + q_j + 2s_j),$$

$$\sum_{y=0}^{l_j-1} Q_j(\Gamma^{(2j)}, 0, y)(z_k^{l_j} - z_k^y) = 0, k \in \{1, 2, \ldots, l_j - 1\}.$$

Частный случай системы обслуживания

Отметим, что применение подхода Ляпунова-Яблонского помогло построить модель системы обслуживания, представляющую исследователю хорошую аналитическую базу. Например, для случая $l_j = 1$ были найдены явные выражения для стационарного распределения процесса обслуживания. Кроме того, в случае существования стационарного режима при $l_j = 1$ получено равенство для математического ожидания количества заявок в очереди для стационарного режима:

$$\begin{aligned} \lim_{i\to\infty} \mathbf{M} æ_{j,i} = \frac{1}{2}(1 - \lambda_j T(1 + q_j + 2s_j))^{-1} \lambda_j T \times \\ \times(\lambda_j T(1 + q_j + 2s_j)^2 + 2q_j + 6s_j) + \\ +(2m)^{-1}\lambda_j(1 + q_j + 2s_j)((2m - 1)T_{2j} + (2m - 2)T_{2j+1} + \ldots + T_{2j-2}). \end{aligned} \tag{1}$$

Средняя длина очереди в стационарном режиме, которая вычисляется по формуле (1), является важным показателем качества работы системы обслуживания. Эта формула позволяет также исследовать параметры системы с целью ее оптимизации по критерию минимума величины очереди по потоку Π_j.

Работа выполнена в ННГУ по госбюджетной теме №01201456585 "Математическое моделирование и анализ стохастических эволюционных систем и процессов принятия решений".

Литература

[1] *Федоткин М. А., Рачинская М. А.* Исследование математической модели трафика автомобилей на основе подхода Ляпунова-Яблонского // Проблемы теоретическо кибернетики. Материалы XVI Международной конференции. — Нижний Новгород: Изд-во Нижегородского госуниверситета, 2011. — С. 508–512.

Поиск объектов как задача распознавания образов

В. Б. Фофанов, А. Н. Жизневский

Viatcheslav.Fofanov@ksu.ru, write2aracon@gmail.com

Казанский Федеральный Университет, г.Казань

Введение

Предлагаются формализация и решение задачи поиска объектов по изображению сцены. При этом моделью сцены служит локально однородное случайное поле, а ее изображением — выборочная поверхность. Поиск объектов выполняется в три этапа. Вначале выявляются зоны интереса, представляющие квадратные фрагменты сцены, содержащие по одному объекту и его окружению. Для этого строится семейство квадратных фрагментов, гарантированно содержащих зону интереса для каждого объекта. Затем проводится их классификация на зоны и не зоны. Целью второго этапа является получение проекций объектов, оказавшихся в зонах. Это достигается за счет классификации пикселей каждой зоны интереса на два класса. Форма полученной проекции используется на завершающем этапе для классификации самого объекта.

Работа является обобщением ранее полученных результатов [1]– [3].

Модель сцены

Каждый объект определяется конечным подмножеством A целочисленной решетки Z^2, называемым его проекцией, и совокупностью случайных величин $\xi_A = (\xi_a)_{a\in A}$ со значениями из $Y = \{0, 1, \ldots, |Y| - 1\}$. Семейство $x_A = (x_a)_{a\in A}$, $x_A \in Y$, называется изображением объекта ξ_A. Предполагается, что объект ξ_A является фрагментом однородного случайного поля со средним значением m_A и ковариационной функцией K_A, для которого выполняются условия эргодической теоремы Слуцкого. Это позволяет использовать среднее арифметическое значение $x_A = \left(\sum_{a\in A} x_a\right)/|A|$, подсчитанное по изображению, в качестве состоятельной оценки среднего значения m_A . Множество всех объектов с непересекающимися проекциями, сумма которых равняется Z^2, называется локально однородной сценой.

Пусть d — евклидово расстояние на Z^2. Точки z и t из Z^2 назовем соседними, если $d(z, t) = 1$. Назовем точку z из $A \subset Z^2$ граничной, если у нее есть сосед t из $Z^2 \setminus A$. Совокупность $Fr(A)$ граничных точек множества A будет называться его границей.

Объект ξ_A назовем светлым (темным) пятном, если существует такой квадрат Q на Z^2, что $A \subset Q \setminus Fr(Q)$, если $\xi_{Q\setminus A}$ является фрагментом однородного случайного поля со средним $m_{Q\setminus A}$ и если $m_A > m_{Q\setminus A}$ $(m_A < m_{Q\setminus A})$. Семейство $\xi_Q = (\xi_z)_{z\in Q}$ называется зоной интереса для ξ_A. Предметом поиска являются пятна с известной формой и размерами.

Этап 1 - поиск зон интереса

Пусть ξ_A — пятно с проекцией A, $d(A)$ — его диаметр, а l — целое число, удовлетворяющее неравенству $d(A) + 2 \leqslant l$. Выберем натуральное число Δ и построим на Z^2 семейство $Q(l, \Delta)$ квадратов со стороной l и левой верхней вершиной z вида $z = z_0 + i\Delta e_1 + j\Delta e_2$, $i \in Z$, $j \in Z$. Тогда справедливо следующее утверждение.

Теорема 1. *Если Δ удовлетворяет неравенству $1 \leqslant \Delta \leqslant l - d(A) - 1$, то существует квадрат Q из $Q(l, \Delta)$ такой, что $A \subset Q \setminus Fr(Q)$.*

Пусть a — центр квадрата Q, $B(a, r) \subset Q$ — квадратная окрестность с центром a и радиусом r, $n = |B(a, r)|$. Разделим $Fr(Q)$ на s не пересекающихся связных частей Fr_j, $1 \leqslant j \leqslant s$, по n точек в каждой, и вычислим средние арифметические значения $\bar{x}_j = \left(\sum_{z \in Fr_j} x_z\right)/n$, $1 \leqslant j \leqslant s$, и $\bar{x}_a = \left(\sum_{z \in B(a,r)} x_z\right)/n$. Сопоставим каждому квадрату Q со стороной l и центром $a \in Z^2$ признак f_a, определенный равенством $f_a(x) = \sum_{j=1}^{z} I_{]0,+\infty[}(\bar{x}_a - \bar{x}_j)$. Пусть Θ_1 — множество зон интереса, а Θ_2 — множество квадратов, содержащих только фон. Справедливо следующее утверждение.

Теорема 2. *Пусть $P(\Theta_1)$ и $P(\Theta_2)$ — априорные вероятности классов Θ_1 и Θ_2, Fr_j, $1 \leqslant j \leqslant s$, — связные фрагменты границы $Fr(Q)$, содержащие по $n = |B(a, r)|$ пикселей и такие, что $d(Fr_i, Fr_j) > 2\hat{r}$, а f_a — признак квадрата ξ_Q, соответствующий r и s. Тогда для любого $\varepsilon > 0$ существуют $s(\varepsilon)$ и $n(\varepsilon)$, что вероятность $e(h^{s(\varepsilon)}_{n(\varepsilon)})$ ошибки решающего правила $h^{s(\varepsilon)}_{n(\varepsilon)} : Y_f \to \{1, 2\}$ вида*

$$h^{s(\varepsilon)}_{n(\varepsilon)}\left(f_{\{a\}}(x)\right) = \begin{cases} 1, & f_a(x) = s(\varepsilon) \\ 2, & f_a(x) < s(\varepsilon) \end{cases}.$$

классификации квадратов из Θ не превосходит ε.

Этап 2 - Сегментация

Далее символом A° обозначается подмножество из A, содержащее вместе с каждой точкой z и ее окрестность $B(z, r)$.

Теорема 3. *Пусть ξ_Q — зона интереса на сцене, построенной скользящим суммированием по окрестности с радиусом $\hat{r}$, $P\left((Q \setminus A \setminus Fr(Q))^\circ\right)$ и $P(A^\circ)$ — априорные вероятности классов, Fr_j, $1 \leqslant j \leqslant s$, — связные фрагменты границы $Fr(Q)$, состоящие из $n = |B(a, r)|$ пикселей каждый, и такие, что $d(Fr_i, Fr_j) > 2\hat{r}$, f_z — признак, соответствующий r и s. Тогда для любого $\varepsilon > 0$ существуют такие $s(\varepsilon)$ и $n(\varepsilon)$, что вероятность $e(h^{s(\varepsilon)}_{n(\varepsilon)})$ ошибки классификации пикселей из $A^\circ + (Q \setminus A \setminus Fr(Q))^\circ$ не превосходит ε.*

Этап 3 - классификация объектов

На заключительном этапе форма проекции объекта применяется для определения его принадлежности к одному из заранее заданных классов. В качества формального описания (определения) формы используется распределе-

ние длины хорды, вырезаемой проекцией из случайной прямой. Поэтому сравнение двух проекций по форме сводится к проверке равенства соответствующих распределений. При решении прикладных задач ее заменяют проверкой гипотезы однородности для выборок. Для этого по проекции неизвестного объекта, полученной по изображению, строится выборка длин хорд. Проверяется гипотеза однородности для этой выборки и выборки, построенной для каждого объекта из базы. Подсчитывается количество случаев принятия гипотезы для каждого класса. Объект относят к тому классу, для которого число подобных случаев оказалось наибольшим.

Литература

[1] *Фофанов В. Б.* Формализация сцены в задаче дешифрирования многозональных изображений // Оптический журнал. — 2007. — Т. 74, № 3. — С. 51–54.

[2] *Aleev R. M., Martynov S. A., Fofanov V.. B.* Remarks on Searching Zones of Interest in Locally Uniform Scene // Pattern Recognition and Image Analysis. — 2011. — V. 21, № 2. — P. 212–215.

[3] *Fofanov V. B., Zhiznevskii A. N.* Segmentation of Regions of Interest on Locally Homogeneous Scenes // Pattern Recognition and Image Analysis. — 2012. — V. 22, № 2. — P. 257–264.

Иерархия классов булевых функций, представимых в детерминированных и недетерминированный моделях OBDD ветвящихся программ по параметру ширины.

К. Р. Хадиев

kamilhadi@gmail.com

Казанский федеральный университет, Казань

В этой работе рассматриваются известные модели ветвящихся программ — $OBDD$, недетерминированная $OBDD$ ($NOBDD$) и k–$OBDD$. Ветвящиеся программы и их модификации $OBDD$ и $k-OBDD$ определены в книге [11]. $OBDD$ $P(X)$ на множестве переменных $X = \{x_1, x_2, \ldots, x_n\}$ — это ветвящаяся программа, обладающая следующими свойствами. Вершины $P(X)$ разбиты на n уровней $1, \ldots, n$ таким образом, что для каждого $i \in \{1, \ldots, n-1\}$ ребра из вершин уровня i ведут только в вершины уровня $(i+1)$. На каждом уровне i считывается значение только одной переменной x_{j_i}. На любом пути вычисления каждая переменная считывается один раз. $P(X)$ задает порядок $\theta(X) = (x_{j_1}, \ldots, x_{j_n})$ считывания переменных. Через $\theta = (j_1, \ldots, j_n)$ обозначим перестановку индексов, задаваемую порядком $\theta(X)$. Различные $OBDD$ могут использовать различные порядки $\theta(X)$ считывания переменных.

Недетерминированная $OBDD$ или $NOBDD$ — это $OBDD$, в которой из одной вершины может исходить несколько дуг, помеченных одинаковыми значениями.

k–$OBDD$ — это ветвящаяся программа $P(X)$, состоящая из k слоев, каждый из которых является $OBDD$, причем порядок $\theta(X)$ чтения переменных во всех слоях программы $P(X)$ одинаковый.

Говорят, что $OBDD$ ($NOBDD$ или $k-OBDD$) $P(X)$ вычисляет булеву функцию $f(X)$ ($f : \{0,1\}^n \to \{0,1\}$), если для любого набора $\nu \in \{0,1\}^n$, такого что $f(\nu) = 1$ В $P(X)$ есть путь из начальной вершины в ту из двух финальных вершин, которая помечена символом 1. В противном случае такого пути нет. Вершины последнего уровня называются финальными и помечаются символами из $\{0,1\}$.

Ширина $w(P)$ $OBDD$ ($NOBDD$ или $k-OBDD$) P — это максимум от количества вершин на уровне, взятый по всем уровням P. Сложность $S(P)$ — это число ее внутренних вершин.

Заметим, что непосредственно из определения для $OBDD$, $NOBDD$ и $k-OBDD$ P имеем $w(P) \leqslant S(P) \leqslant k \cdot w(P) \cdot n$.

В большинстве работ, которые рассматривали ширину или размер в качестве параметра сложности, определялось различие между классами булевых функций, представимых моделями экспоненциальной и полиномиальной ширины или размера. Рассматривались как модели с меньшими ограничениями, так и недетерминированные, вероятностные и другие расширения $OBDD$ и $k-OBDD$. Примерами таких работ являются [6, 2, 1, 3, 5, 8, 9, 10]

В этой работе была построена более тонкая иерархия по ширине.

Через $\mathbf{OBDD_w}$, $\mathbf{NOBDD_w}$ и $\mathbf{k-OBDD_w}$ обозначим класс булевых функций, вычислимых $OBDD$, $NOBDD$ и $k-OBDD$ ширины w, соответственно.

Для моделей $OBDD$ и $NOBDD$ была получена следующая иерархия:

Теорема 1. *Для любых целых n, $w = w(n)$, $16 \leqslant w \leqslant 2^{n/4}$, где n — длина входного набора, справедливо:*

$$\mathbf{OBDD}_{\lfloor \mathbf{w}/\mathbf{8} \rfloor - \mathbf{1}} \subsetneq \mathbf{OBDD_w}, \qquad \mathbf{NOBDD}_{\lfloor \mathbf{w}/\mathbf{8} \rfloor - \mathbf{1}} \subsetneq \mathbf{NOBDD_w},$$

Доказательство. Доказательство основано на сложности функции перемешанного равенства EQS_d, которая является аналогом введенной в работе [1] функции, но зависящей существенным образом только от первых w переменных. Для нее можно построить $OBDD$ ширины $8 \cdot 2^{d/4} - 5$. При этом используя принцип Дирихле можно показать, что для EQS_d нельзя построить $NOBDD$ ширины $2^{d/4} - 1$. ∎

Для различных моделей классы сложности соотносятся между собой следующим образом:

Теорема 2. *Для любых целых n, $w = w(n)$ и $w' = w'(n)$ таких, что $w \leqslant 2^{n/4}$, $O(\log^4(w+1)\log\log(w+1)) < w' < w/8 - 1$, n — длина входного набора, :*

$\mathbf{NOBDD}_{\lfloor \log(\mathbf{w}) \rfloor} \subsetneq \mathbf{OBDD_w}$, $\mathbf{OBDD_w}$ *и* $\mathbf{NOBDD_{d'}}$ *не сравнимы.*

Доказательство. Первое утверждение доказывается моделированием $NOBDD$ детерминированной моделью.

Доказательство второго основано на сложности отрицания EQS_d. Для нее можно построить $NOBDD$ ширины $O(d^4 \log d)$, используя метод "finger printing" [7]. При этом используя принцип Дирихле можно показать, что ее нельзя построить $OBDD$ ширины $2^{d/4} - 1$. ∎

Более точная иерархия получена для ширины, не превышающей $n/2$:

Теорема 3. *Для любых целых n, $w = w(n)$, $1 < w \leqslant n/2$, где n — длина входного набора, справедливо:*

$$\mathbf{OBDD_{w-1}} \subsetneq \mathbf{OBDD_w}, \qquad \mathbf{NOBDD_{w-1}} \subsetneq \mathbf{NOBDD_w},$$

Доказательство. Доказательство основано на сложности известной функции MOD_d, проверяющей равно ли 0 число единиц по модулю d. Для нее можно построить $OBDD$ ширины d. При этом, используя принцип Дирихле, можно показать, что для MOD_d нельзя построить $NOBDD$ ширины $d-1$. ∎

При этом для различных моделей классы сложности соотносятся между собой следующим образом:

Теорема 4. *Для любых целых n, $w = w(n)$ и $w' = w'(n)$ таких, что $w \leqslant n/2$, $O(\log^2 w \log\log w) < w' \leqslant w-1$, n — длина входного набора, справедливо:*

$\mathbf{NOBDD_{\lfloor \log(w) \rfloor}} \subsetneq \mathbf{OBDD_w}$, $\mathbf{OBDD_w}$ *и* $\mathbf{NOBDD_{w'}}$ *не сравнимы*

Доказательство. Первое утверждение доказывается аналогично теореме 2.

Доказательство второго основано на сложности функции $NotO_d$, проверяющей, не равно ли число нулей и единиц в первых d переменных. Для нее можно построить $NOBDD$ ширины $O(\log^2 d \log\log d)$, используя метод "finger printing". При этом, используя принцип Дирихле, можно показать, что ее нельзя построить $OBDD$ ширины $d-1$. ∎

Для модели $k-OBDD$ была получена следующая иерархия:

Теорема 5. *Для целых чисел $k = k(n), w = w(n)$ таких, что $2kw(2w + \lceil \log k \rceil + \lceil \log 2w \rceil) < n, k \geqslant 2, w \geqslant 84$ выполняется следующее собственное включение:*

$$\mathbf{k-OBDD_{\lfloor w/21 \rfloor - 3}} \subsetneq \mathbf{k-OBDD_w}$$

Доказательство. Доказательство основано на сложности перемешанной адресной функции $SAF_{t,d}$, являющейся модификацией функции PJ из работы [5] и использующий метод перемешивания из работы [1]. Для $SAF_{\lceil k/3 \rceil, \lceil w/5 \rceil}$ можно построить $k-OBDD$ ширины w. При этом, используя нижнюю оценку для $k-OBDD$, представленную в статье [4], можно показать, что для $SAF_{\lceil k/3 \rceil, \lceil w/5 \rceil}$ нельзя построить $k-OBDD$ ширины $\lfloor w/21 \rfloor - 3$.

∎

Работа выполнена при поддержке РФФИ, проект № 14-07-00557.

Литература

[1] *Ablayev F.* Randomization and nondeterminsm are incomparable for ordered read-once branching programs. // Electronic Colloquium on Computational Complexity (ECCC). — 1997. — V. 21, № 4.

[2] *Ablayev F., Gainutdinova A., Karpinski M., Moore C., Pollette C.* On the computational power of probabilistic and quantum branching program. // Information and Computation. — 2005. — V. 203, № 2.— P. 145–162.

[3] *Ablayev F., Karpinski M.* On the Power of Randomized Ordered Branching Programs. // ICALP'96 Lecture Notes in Computer Science. — 1998. — V. 1099, — P. 348–356.

[4] *Ablayev F., Khadiev K.* Extension of the hierarchy for k-OBDDs of small width. // Russian Mathematics. — 2013. — V. 57, № 3.— P. 46–50.

[5] *Bollig B., Sauerhoff M., Sieling D., Wegener I.* Hierarchy theorems for kOBDDs and kIBDDs. // Theoretical Computer Science. — 1998. — V. 205, № 1-2.— P. 45–60.

[6] *Borodin A., Razborov A., Smolensky R.* On lower bounds for read-k-times branching programs. // Computational Complexity. — 1993. — V. 3, № 1— P. 1–18.

[7] *Freivalds R.* A Classical Introduction to Modern Number Theory, 2nd ed. // Mathematical Foundations of Computer Science LNCS Springer-Verlag. — 1979. — V. 74— P. 57–69.

[8] *Hromkovic J., Sauerhoff M.* Tradeoffs between Nondeterminism and Complexity for Communication Protocols and Branching Programs. // 17th STACS, LNCS. — 2000. — V. 1770,— P. 145–156.

[9] *Hromkovic J., Sauerhoff M.* On the Power of Nondeterminism and Randomness for Oblivious Branching Programs. // Theory of Computing Systems — 2003. — V. 36,— P. 159–182.

[10] *Thathachar J. S.* On separating the read-k-times branching program hierarchy. // 30th ACM STOC — 1998. — P. 653–662.

[11] *Wegener I.* Branching Programs and Binary Decision Diagrams: Theory and Applications // Society for Industrial and Applied Mathematics, Philadelphia — 2000.

Уточнение иерархии классов булевых функций, представимых в моделях k-OBDD ветвящихся программ.

К. Р. Хадиев

kamilhadi@gmail.com

Казанский федеральный университет, Казань

В этой работе рассматриваются известные модели ветвящихся программ $k-OBDD$. Ветвящиеся программы и их модификации $OBDD$ и $k-OBDD$ определены в книге [8]. $OBDD$ $P(X)$ на множестве переменных $X = \{x_1, x_2, \ldots, x_n\}$ — это ветвящаяся программа, обладающая следующими свойствами. Вершины $P(X)$ разбиты на n уровней $1, \ldots, n$ таким образом, что для каждого $i \in \{1, \ldots, n-1\}$ ребра из вершин уровня i ведут только в вершины уровня $(i+1)$. На каждом уровне i считывается значение только одной переменной x_{j_i}. На любом пути вычисления каждая переменная считывается один раз. $P(X)$ задает порядок $\theta(X) = (x_{j_1}, \ldots, x_{j_n})$ считывания переменных. Через $\theta = (j_1, \ldots, j_n)$ обозначим перестановку индексов, задаваемую порядком $\theta(X)$. Различные $OBDD$ могут использовать различные порядки $\theta(X)$ считывания переменных.

k–$OBDD$ — это ветвящаяся программа $P(X)$, состоящая из k слоев, каждый из которых является $OBDD$, причем порядок $\theta(X)$ чтения переменных во всех слоях программы $P(X)$ одинаковый.

Говорят, что $k-OBDD$ $P(X)$ вычисляет булеву функцию $f(X)$ ($f : \{0,1\}^n \to \{0,1\}$), если для любого набора $\nu \in \{0,1\}^n$, такого что $f(\nu) = 1$ В $P(X)$ есть путь из начальной вершины в ту из двух финальных вершин,

которая помечена символом 1. В противном случае такого пути нет. Вершины последнего уровня называются финальными и помечаются символами из $\{0,1\}$.

Через $\mathbf{k-OBDD}$ обозначим класс булевых функций, которые вычислимы $k-OBDD$ полиномиальной сложности.

Вопросы нижних оценок и иерархий для k раз читающих программ и $OBDD$ были рассмотрены в различных работах [4, 1, 3, 5, 6, 7].

В работе [3] доказана следующая иерархия (иерархия Bolling-Sauerhoff-Sieling-Wegener): для $k = o(n^{1/2}/\log^{3/2} n)$ выполняется собственное включение

$$(\mathbf{k-1})-\mathbf{OBDD} \subset \mathbf{k}-\mathbf{OBDD}.$$

Улучшение этой оценки уже было представлено в работе [2], однако с некоторыми ограничениями на ширину.

В данной работе был разработан метод представления процесса вычисления в $k-OBDD$ в виде булевой формулы специального вида, по аналогии с представлением k раз читающих программ в работе [4]. Этот метод позволяет продолжить иерархию Bolling-Sauerhoff-Sieling-Wegener для $k-OBDD$ (Теорема об иерархии).

Введем необходимые определения.

Ширина $w(P)$ программы $k-OBDD$ P — это максимум от количества вершин на уровне, взятый по всем уровням P.

Заметим, что непосредственно из определения $k-OBDD$ P имеем

$$w(P) \leqslant S(P) \leqslant k \cdot w(P) \cdot n.$$

Через $\mathbf{k-OBDD}_{\mathcal{W}}$ обозначим класс булевых функций, которые вычислимы $k-OBDD$ ширины $w \in \mathcal{W}$, для некоторого множества $\mathcal{W}$.

В этой работе был получен следующий результат

Теорема 1 (Теорема об иерархии). *Для $k = k(n)$ и $w = w(n)$ таких, что $k \log w = o(n/\log n)$, $w \in \mathcal{W}$ справедливо следующее собственное включение:*

$$(\mathbf{k-1})-\mathbf{OBDD}_{\mathcal{W}} \subsetneq \mathbf{k}-\mathbf{OBDD}_{\mathcal{W}}$$

В частности, если выбрать в качестве множества $\mathcal{W}$ множество $POLY$ всех полиномов относительно n, то мы получим следующую теорему:

Теорема 2. *Для $k = k(n)$ таких, что $k = o(n/\log^2 n)$, справедливо следующее собственное включение:*

$$(\mathbf{k-1})-\mathbf{OBDD} \subsetneq \mathbf{k}-\mathbf{OBDD}$$

Для доказательства теоремы 1 используются сложностные свойства булевой функции F_t, которая является модификацией функции PJ_k^{bool}, введенной в работе [3]. Отличие состоит в том, что в рамках одного прохода по переменным производиться не два прыжка, а пять.

Для функции F_k удалось построить $k-OBDD$ ширины $w \in \mathcal{W}$, при этом доказать, что она не может быть представлена ни в какой $(k-1)-OBDD$ ширины $w \in \mathcal{W}$.

Доказательство непредставимости основывается на следующей нижней оценке:

Теорема 3. *Пусть $f(X)$ — некоторая булева функция. Ее распознает $k-OBDD$ P ширины w, тогда f может быть представлена в следующем виде:*

$$f(X) = \bigvee_{j=1}^{d} \bigwedge_{i=1}^{k} g_{j,i}(X), \tag{1}$$

где $g_{j,i}$ представима в виде $OBDD$ ширины w при $1 \leqslant j \leqslant q, 1 \leqslant i \leqslant k$ и $d \leqslant w^{k-1}$.

Удается доказать, что F_k не представима в виде 1, а значит и не представимо ни в какой $(k-1)-OBDD$

Работа выполнена при поддержке РФФИ, проект № 14-07-00557.

Литература

[1] *Ablayev F., Karpinski M.* On the Power of Randomized Ordered Branching Programs. // ICALP'96 Lecture Notes in Computer Science. — 1998. — V. 1099, — P. 348–356.

[2] *Ablayev F., Khadiev K.* Extension of the hierarchy for k-OBDDs of small width. // Russian Mathematics. — 2013. — V. 57, № 3.— P. 46–50.

[3] *Bollig B., Sauerhoff M., Sieling D., Wegener I.* Hierarchy theorems for kOBDDs and kIBDDs. // Theoretical Computer Science. — 1998. — V. 205, № 1-2.— P. 45–60.

[4] *Borodin A., Razborov A., Smolensky R.* On lower bounds for read-k-times branching programs. // Computational Complexity. — 1993. — V. 3, № 1— P. 1–18.

[5] *Hromkovic J., Sauerhoff M.* Tradeoffs between Nondeterminism and Complexity for Communication Protocols and Branching Programs. // 17th STACS, LNCS. — 2000. — V. 1770,— P. 145–156.

[6] *Hromkovic J., Sauerhoff M.* On the Power of Nondeterminism and Randomness for Oblivious Branching Programs. // Theory of Computing Systems — 2003. — V. 36,— P. 159–182.

[7] *Thathachar J. S.* On separating the read-k-times branching program hierarchy. // 30th ACM STOC — 1998. — P. 653–662.

[8] *Wegener I.* Branching Programs and Binary Decision Diagrams: Theory and Applications // Society for Industrial and Applied Mathematics, Philadelphia — 2000.

О средней сложности булевых функций с распределением Бернулли на области определения

А. В. Чашкин

chashkin@inbox.ru

МГУ, Москва

Ниже изучается среднее время вычисления значений булевых функций неветвящимися программами с условной остановкой [1, 2], с распределением

Бернулли на n-мерном булевом кубе таким, что $\Pr(x) = p^{n-\|x\|}(1-p)^{\|x\|}$, где $p > 1/2$ и x — булев набор длины n. Число команд, выполненных программой P на наборе переменных x, назовем временем работы P на x и обозначим через $t_P(x)$. Средним временем работы программы P назовем величину $T(P) = \sum_x t_P(x)\Pr(x)$, где суммирование производится по всем булевым наборам длины n. Величину $T(f) = \min T(P)$, где минимум берется по всем программам, вычисляющим f, назовем средней сложностью функции f, а программу, на которой достигается минимум — минимальной.

Пусть f — булева функция, P — программа, вычисляющая f. Каждому двоичному набору x длины n, рассматриваемому как двоичная запись натурального числа, поставим в соответствие его номер $N_P(x)$ такой, что $1 \leqslant N_P(x) \leqslant 2^n$; $N_P(x) < N_P(y)$, если $T_P(x) < T_P(y)$; $N_P(x) < N_P(y)$, если $T_P(x) = T_P(y)$ и $x < y$. Аналогичным образом определим целочисленную функцию $M(x)$: $1 \leqslant M(x) \leqslant 2^n$; $M(x) < M(y)$, если $\Pr(x) > \Pr(y)$; $M(x) < M(y)$, если $\Pr(x) = \Pr(y)$ и $x < y$.

Через B_i и S_i будем обозначать шар и сферу радиуса i с центром в нулевом наборе, а через $L(f)$ — сложность реализации функции f схемами.

Теорема 1. *Для почти всех n-местных булевых функций f справедливо равенство*

$$T(f) = \Theta\Big(\sum_{i=1}^{n-3} \frac{|S_{i+1}|}{\log_2 |B_{i+1}|}(1 - \Pr(|B_i|))\Big),$$

причем для каждой такой функции f

$$T(f) = O\Big(\sum_{i=1}^{n-3} \frac{|S_{i+1}|}{\log_2 |B_{i+1}|}(1 - \Pr(|B_i|))\Big).$$

Доказательство. **1.** Введем множества $D_1 = B_2$, $D_2 = S_3, \ldots, D_{n-4} = S_{n-3}$, $D_{n-3} = S_{n-2} \cup S_{n-1} \cup S_n$. Каждое множество D_i разобьем на m равномощных подмножеств $R_{(i-1)m+j}$ так, чтобы сложность характеристической функции каждого такого подмножества была $O(n)$. Программу P, вычисляющую значение n-местной функции f построим на основе следующего алгоритма. Для каждого i от 1 до $(n-3)m$ последовательно выполняются следующие действия. Вычисляются характеристическая функция χ_{R_i} множества R_i и определенная на этом множестве частичная функция f_i, которая совпадает на нем с функцией f. Если значение χ_{R_i} равно единице, то $f = f_i$ и вычисления прекращаются, если значение χ_{R_i} равно нулю, то значение i увеличивается на единицу. Нетрудно показать [2, 3], что если $m = o(n/\log_2 n)$, то последовательное вычисление функций χ_{R_i} и f_i можно организовать так, что

$$L(\chi_{R_i}, f_i) = O\Big(\frac{|R_i|}{\log_2 |\cup_{j=1}^{i} R_j|}\Big).$$

Среднее время работы такой программы будет удовлетворять неравенству

$$T(P) = O\Big(\sum_{i=1}^{(n-3)m} \frac{|R_i|}{\log_2 |\cup_{j=1}^{i} R_j|}(1 - \Pr(\cup_{j=1}^{i-1} R_j))\Big).$$

Так как $\Pr(R_k) = \Pr(R_l)$ при $(i-1)m < k, l \leqslant im$, то при растущем параметре m аргумент функции "O" асимптотически равен сумме

$$\sum_{i=1}^{(n-3)m} \frac{|R_i|}{\log_2 |\cup_{j=1}^{i} R_j|}(1 - \Pr(\cup_{j=1}^{i} R_j)).$$

Введем множества $Q_i = \cup_{j=1}^{i} R_j$. Тогда

$$T(P) = O\Big(\sum_{i=1}^{(n-3)m} \frac{|R_i|}{\log_2 |Q_i|}(1 - \Pr(Q_i))\Big). \tag{1}$$

2. Пусть $1 = M_0 < M_1 < \cdots < M_k = 2^n$ — целые, $M_1 \geqslant 2n$, $k = o(2^n)$, $Y_i = \{x \mid M(x) \leqslant M_i\}$, $Z_i = Y_i \setminus Y_{i-1}$, f — n-местная булева функция, P — минимальная программа, вычисляющая f. Пусть x_i такое, что $N_P(x_i) = M_i$. Оценим число булевых функций, у минимальных программ которых для $i = 1, 2, \ldots, k$ найдется x_i такое, что

$$t_P(x_i) \leqslant \frac{M_i}{6 \log_2 M_i}.$$

Каждая такая функция однозначно определяется первыми $t_P(x_i)$ командами своей минимальной программы и двоичным набором длины не более чем $2^n - N_P(x_i)$ — значениями на тех аргументах, время работы на которых больше времени работы на x_i. Для числа N_i, равного числу различных программ, сложность которых не превосходит $T_P(x_i)$, справедливо неравенство [2]

$$N_i \leqslant \left(\frac{M_i}{6 \log_2 M_i} + n + 1\right)^{3 \cdot \frac{M_i}{6 \log_2 M_i}} \leqslant M_i^{\frac{M_i}{2 \log_2 M_i}} \leqslant 2^{M_i/2}.$$

Следовательно, число рассматриваемых функций, не превосходит величины

$$\sum_{i=1}^{k} 2^{M_i/2} \cdot 2^{2^n - M_i} \leqslant k 2^{2^n - M_i/2} = o\big(2^{2^n}\big).$$

Сравнивая полученную оценку с числом всех n-местных булевых функций, видим, что все минимальные программы почти всех таких функций удовлетворяют условию: если x_i такое, что $N_P(x_i) = M_i$, где $i = 1, 2, \ldots, k$, то

$$t_P(x_i) > \frac{M_i}{6 \log_2 M_i}.$$

Положим $X_0 = \{x_0\}$, $X_i = \{x \mid N_P(x_{i-1}) < N_P(x) \leqslant N_P(x_i)\}$, $T_0 = 0$ и $T_i = \frac{M_i}{6 \log_2 M_i}$ для $i = 1, 2, \ldots, k$. Тогда $\Pr(\bigcup_{i=0}^{j} X_i) \leqslant \Pr(Y_j)$ и для среднего

времени работы каждой такой программы имеем

$$T(P) = \sum_x t_P(x)\Pr(x) \geqslant \sum_{i=1}^{k}\sum_{x\in X_i} t_P(x)\Pr(x) \geqslant$$
$$\geqslant \sum_{i=1}^{k} t_P(x_{i-1})\Pr(X_i) > \sum_{i=1}^{k} T_{i-1}\Pr(X_i) =$$
$$= \sum_{i=1}^{k-1}(T_i - T_{i-1})\sum_{j=i+1}^{k}\Pr(X_j) = \sum_{i=1}^{k-1}(T_i - T_{i-1})\Pr(\bigcup_{j=i+1}^{k} X_j) =$$
$$= \sum_{i=1}^{k-1}(T_i - T_{i-1})(1 - \Pr(\bigcup_{j=0}^{i} X_j)) \geqslant \sum_{i=1}^{k-1}(T_i - T_{i-1})(1 - \Pr(Y_i)).$$

Так как

$$\frac{M_i}{\log_2 M_i} - \frac{M_{i-1}}{\log_2 M_{i-1}} > \frac{M_i - M_{i-1}}{\log_2 M_i + \log_2 M_{i-1}} > \frac{M_i - M_{i-1}}{2\log_2 M_i} = \frac{|Z_i|}{2\log_2 |Y_i|},$$

то

$$T(P) \geqslant \sum_{i=1}^{k-1} \frac{|Z_i|}{12\log_2 |Y_i|}(1 - \Pr(|Y_i|)). \tag{2}$$

3. Отождествляя множества R_i, Q_j в неравенстве (1) с множествами Z_i, Y_j в неравенстве (2) и учитывая параметры множеств R_i, нетрудно показать, что для почти всех n-местных булевых функций f

$$T(f) = \Theta\Big(\sum_{i=1}^{(n-3)m} \frac{|R_i|}{\log_2 |Q_i|}(1 - \Pr(Q_i)\Big) = \Theta\Big(\sum_{i=1}^{n-3} \frac{|S_{i+1}|}{\log_2 |B_{i+1}|}(1 - \Pr(|B_i|))\Big),$$

а для каждой такой функции f

$$T(f) = O\Big(\sum_{i=1}^{(n-3)m} \frac{|R_i|}{\log_2 |Q_i|}(1 - \Pr(Q_i)\Big) = O\Big(\sum_{i=1}^{n-3} \frac{|S_{i+1}|}{\log_2 |B_{i+1}|}(1 - \Pr(|B_i|))\Big),$$

Теорема доказана. ∎

Работа выполнена при поддержке РФФИ, проект № 14-01-00598.

Литература

[1] *Чашкин А. В.* О среднем времени вычисления значений булевых функций // Дискретный анализ и исследование операций. Серия 1.— 1997. — Т. 4, № 1. — С. 60–78.

[2] *Чашкин А. В.* Дискретная математика. — М.: Академия, 2012. — 352 с.

[3] *Шоломов Л. А.* О реализации недоопределенных булевых функций схемами из функциональных элементов // Проблемы кибернетики. — Вып. 21. — М.: Наука, 1969 — С. 215–226.

Классы изоморфизмов коммутативных локальных алгебр специального вида над конечным полем

Б. В. Чокаев

chokaev@cs.msu.ru

МГУ им. Ломоносова, Москва

Обозначим через F_q конечное поле характеристики p, мощности q. Хорошо известно, что для любых неприводимых над полем F_q многочленов $f(X)$ и $g(X)$ одинаковой степени n алгебры $F_q[X]/f(X)$ и $F_q[X]/g(X)$ изоморфны и образуют конечное поле мощности q^n (см. [3]). В данной работе доказывается, что для произвольного натурального m алгебры $F_q[X]/(f(X))^m$ и $F_q[X]/(g(X))^m$ также изоморфны. Это означает, что все возможные алгебры вида $F_q[X]/(f(X))^m$ не зависят от многочлена $f(X)$ и параметризуются двумя параметрами: n и m. Обозначим алгебру $F_q[X]/(f(X))^m$ через $F_q\langle n,m\rangle$. Алгебры такого вида являются интересными в связи с тем, что они возникают в разложении коммутативных групповых алгебр в прямое произведение неразложимых алгебр (см. [5], [6]). В свою очередь, класс групповых алгебр играют важную роль при изучении алгебраической структуры и сложности умножения матричных алгебр (см. [1], [2], [4]).

Теорема 1. *Для любых неприводимых над полем F_q многочленов $f(X)$ и $g(X)$ одинаковой степени n и для произвольного натурального m алгебры $F_q[X]/(f(X))^m$ и $F_q[X]/(g(X))^m$ являются изоморфными.*

Доказательство. Обозначим $A \cong F_q[X]/(f(X))^m$ и покажем, что A — локальная алгебра, то есть $A/\mathrm{rad}A \cong D$ — алгебра с делением.

Лемма 1. *Для любых n и m алгебра $F_q[X]/(f(X))^m$ имеет ровно один максимальный идеал, где $f(X)$ — произвольный неприводимый над F_q многочлен степени n.*

Доказательство. Пусть $I = (f(X))$ — идеал алгебры $F_q[X]/(f(X))^m$, порожденный многочленом $f(X)$, и пусть $a(X), b(X)$ — два произвольных элемента этой алгебры. Докажем, что, если $a(X) \notin I$, то $b(X) \in (a(X)) + I$. Заметим, что $a(X)$ и $b(X)$ можно представить в виде:
$a(X) = \alpha_{m-1}(X)(f(X))^{m-1} + \alpha_{m-2}(X)(f(X))^{m-2} + \ldots + \alpha_1(X)f(X) + \alpha_0(X)$,
$b(X) = \beta_{m-1}(X)(f(X))^{m-1} + \beta_{m-2}(X)(f(X))^{m-2} + \ldots + \beta_1(X)f(X) + \beta_0(X)$,
где $\deg\alpha_j(X) < \deg f(X)$, $\deg\beta_j(X) < \deg f(X)$, $j = 0, \ldots, m-1$. Так как $a(X) \notin I$, то $\alpha_0(X) \neq 0$. Домножим $a(X)$ на элемент $(\alpha_0(X))^{-1}\beta_0(X)$, где обратный элемент берется по модулю $f(X)$, получим:
$a(X)(\alpha_0(X))^{-1}\beta_0(X) = \gamma_{m-1}(X)(f(X))^{m-1} + \ldots + \gamma_1(X)f(X) + \beta_0(X)$. Значит,
$b(X) = a(X)(\alpha_0(X))^{-1}\beta_0(X) +$
$+(\beta_{m-1}(X) - \gamma_{m-1}(X))(f(X))^{m-1} + \ldots + (\beta_{m-1}(X) - \gamma_{m-1}(X))f(X) \in (a(X)) + I$. Следовательно, I — максимальный идеал.

Заметим, что $I^m = \{0\}$, поэтому I — нильпотентный идеал. Так как радикал алгебры содержится в любом максимальном идеале, и любой нильпотент-

ный идеал содержится в радикале, то $I = \mathrm{rad}F_q[X]/(f(X))^m$. Следовательно, I — единственный максимальный идеал. ∎

Из леммы 1 следует, что $A/\mathrm{rad}A \cong D$, где $D \cong F_q[X]/f(X)$, то есть $A/\mathrm{rad}A$ является полем, и поэтому A является локальной алгеброй. Кроме того, из доказательства 1 леммы следует, что $A \cong D \oplus \mathrm{rad}A$ и элемент $a \in A, a = d \oplus b, d \in D, b \in \mathrm{rad}A$, является обратимым в A тогда и только тогда, когда $d \neq 0$.

Перейдем к доказательству того, что алгебры $F_q[X]/(f(X))^m$ и $F_q[Y]/(g(Y))^m$ являются изоморфными. Для этого достаточно найти элемент алгебры $F_q[X]/(f(X))^m$, минимальный многочлен которого равен $(g(Z))^m$. Как было отмечено выше, алгебры (поля) $F_q[X]/f(X)$ и $F_q[Y]/g(Y)$ являются изоморфными. Пусть под действием некоторого изоморфизма элемент Y алгебры $F_q[Y]/g(Y)$ отображается в элемент $h(X)$ алгебры $F_q[X]/f(X)$. Рассмотрим разложение многочлена $g(Z)$ над полем $F_q[X]/f(X)$: $g(Z) = v(Z)(Z - h(X))$, $v(Z)$ — некоторый многочлен над полем $F_q[X]/f(X)$, $v(h(X)) \neq 0$ так как многочлен $g(Z)$ не имеет кратных корней (или другими словами поле $F_q[Y]/g(Y)$ является сепарабельным расширением поля F_q). Условие $v(h(X)) \neq 0$ означает, что многочлен $v(h(X))$ не делится на $f(X)$. Учитывая замечание к лемме 1, получаем, что $v(h(X)) \notin \mathrm{rad}F_q[X]/(f(X))^m$ и поэтому является обратимым элементом алгебры $F_q[X]/(f(X))^m$. Обозначим $w(X) = h(X) + f(X) \in F_q[X]/(f(X))^m$. Так как многочлен $v(h(X))$ не делится на $f(X)$, то $v(w(X))$ также не делится на $f(X)$, и поэтому является обратимым элементом алгебры $F_q[X]/(f(X))^m$. Имеем $q(w(X)) = q(h(X) + f(X)) = v(h(X) + f(X))(h(X) + f(X) - h(X)) = v(h(X) + f(X))f(X)$. Поэтому $(g(Z))^m$ — аннулирующий многочлен для $w(X)$, так как $(g(w(X)))^m = v(h(X) + f(X))^m f(X)^m = 0$. Но для любого $k \leqslant m$, $(g(Z))^k = v(h(X)+f(X))^k f(X)^k \neq 0$. Следлвательно, $(g(Z))^m$ — минимальный многочлен для $w(X)$. ∎

Работа выполнена при поддержке гранта РФФИ № 12-01-91331-ННИО-а.

Литература

[1] *H. Cohn, C. Umans.* A Group-Theoretic Approach to Fast Matrix Multiplication. // FOCS 2003: 438–449 (2003).

[2] *H. Cohn, R. D. Kleinberg, B. Szegedy, C. Umans.* Group-theoretic Algorithms for Matrix Multiplication. // FOCS 2005: 379–388 (2005).

[3] *Б. Л. Ван дер Варден.* Алгебра. М.: Наука, 1979.

[4] *Поспелов А. Д.* Сложность умножения в ассоциативных алгебрах. Диссертация. //Московский государственный университет им. М. В. Ломоносова, 2008.

[5] *Чокаев Б. В.* Сложность умножения в коммутативных групповых алгебрах над полями характеристики 0. // Вестник МГУ, серия 15, № 4, стр. 30-40. Изд-во МГУ, Москва, 2010.

[6] *Чокаев Б. В.* Сложность умножения в коммутативных групповых алгебрах над полями простой характеристики. // Дискретная математика, т. 22, вып. 4, стр. 121-137. “Наука”, Москва, 2010.

О минимизации типичных булевых функций для аддитивных мер сложности

И. П. Чухров

chip@icad.org.ru

Институт автоматизации проектирования РАН, Москва

При минимизации для представления булевых функций применяются две эквивалентные модели. В аналитической модели используются понятия булевой функции, импликанты, дизъюнктивной нормальной формы (ДНФ) и т.д., зависящие от n переменных. В геометрической модели эквивалентными понятиями являются подмножество вершин, грань, комплекс граней и т.д. в n-мерном единичном кубе.

Функционал, определенный на множестве всех комплексов граней, является мерой сложности, если он удовлетворяет аксиомам неотрицательности, монотонности, выпуклости (субаддитивности) и инвариантности относительно изоморфизма. Мера сложности называется аддитивной, если функционал сложности является аддитивным, при этом сложность комплекса равна сумме сложностей граней комплекса.

Аддитивными мерами сложности являются функционалы: l — число граней комплекса (длина комплекса), L — сумма рангов граней комплекса, L_σ — число координат в гранях комплекса равных σ, где $\sigma \in \{0,1\}$.

Теорема 1. *Для меры сложности $\mathcal{L} \not\equiv 0$ и любой грани I выполняется:*

(i) если $\mathcal{L}(I) = 0$, то $\min\{L_0(I), L_1(I)\} \leqslant R_{\mathcal{L}}^{\min}$,

(ii) если $\mathcal{L}(I) > 0$, то $\mathcal{L}(I) \geqslant C_{\mathcal{L}}^{\min} > 0$,

где $R_{\mathcal{L}}^{\min}$ — минимальный ранг грани, которая имеет положительную $\mathcal{L}$-сложность, $C_{\mathcal{L}}^{\min}$ — положительная константа, зависящая только от меры сложности.

Утверждение теоремы 1 означает, во-первых, что нулевую сложность могут иметь только грани, в которых число координат либо равных 0, либо равных 1, является конечным числом не превосходящим $R_{\mathcal{L}}^{\min}$. И, во-вторых, для граней ненулевой сложности существует минимальное положительное значение.

Для единичного куба B^n и целых i, k, где $0 \leqslant i \leqslant i+k \leqslant n$, обозначим:

$S_{i,i+k}^n$ — пояс куба, т.е. вершины слоёв куба B_j^n с номерами $j = i, \ldots, i+k$.

$\mathcal{L}^n(i, i+k)$ — $\mathcal{L}$-сложность граней размерности k, которые содержатся в поясе $S_{i,i+k}^n$ единичного куба B^n. Такие грани являются изоморфными, следовательно, имеют одинаковую сложность для любой меры сложности.

Следствие 1. *В единичном кубе B^n для меры сложности $\mathcal{L} \not\equiv 0$ все грани сложности равной 0 и размерности не более $k_0 = \lceil \log n \rceil$ содержатся в множестве $S_{0,p}^n \cup S_{n-p,n}^n$, где $p = R_{\mathcal{L}}^{\min} + k_0$, и их число не превосходит $2^{\log^2 n(1+o(1))}$ при $n \to \infty$.*

Комплексом граней булевой функции $f \in P_n$ называется любой комплекс граней, который содержит множество вершин совпадающее с множеством единичных вершин функции f в кубе B^n. Два комплекса граней называются *эквивалентными*, если они являются комплексами граней одной функции.

Комплекс граней называется *$\mathcal{L}$-минимальным*, если он имеет наименьшую меру сложности $\mathcal{L}$ среди всех эквивалентных комплексов граней. *l-минимальный* комплекс граней называется *кратчайшим*.

$\mathcal{L}$-сложность и максимальную длину $\mathcal{L}$-минимальных комплексов граней булевой функции f обозначим через $\mathcal{L}(f)$ и $l_{\mathcal{L}}(f)$ соответственно.

Очевидно, что $l(f) = l_l(f) \leqslant l_{\mathcal{L}}(f)$ для любой меры сложности $\mathcal{L}$.

Исследованию свойств почти всех булевых функций были посвящены работы Ю. И. Журавлева, Ю. Л. Васильева, В. В. Глаголева, Р. Г. Нигматуллина, А. А. Сапоженко, А. Д. Коршунова, С. Е. Кузнецова, А. Е. Андреева и др. Для типичных булевых функций были изучены свойства различных видов граней, например, установлено отсутствие допустимых граней размерности более $\lceil \log n \rceil$, были доказаны асимптотические равенства $l(f) \sim l_L(f)$ и

$$l(f) \sim \bar{l}(n) = \frac{\bar{c}_n \cdot 2^n}{\log n \log\log n},$$

где $\bar{l}(n)$ — среднее значение $l(f)$ для функций n переменных, $1 \leqslant \bar{c}_n$ [1] и $\bar{c}_n \leqslant 1.5$ [2] или $\bar{c}_n \leqslant \omega(n)$ [3], при этом функция $\omega(n)$ колеблется между $1.38826\ldots$ и $1.54169\ldots$ в зависимости от от *дробной части* $\log\log n + \log\log\log n$.

В следующей теореме сформулированы достаточные условия для асимптотического равенства максимальной длины минимальных и длины кратчайших комплексов граней почти всех булевых функций.

Теорема 2. *Пусть для аддитивной меры сложности $\mathcal{L} \not\equiv 0$ в единичном кубе B^n при $n \to \infty$ выполняются условия:*

(i) максимальная $\mathcal{L}$-сложность граней ограничена полиномом от n,

(ii) грани размерности не более $k_0 = \lceil \log n \rceil$, которые содержатся в поясе $S^n_{r,n-r}$ для $r = \left\lfloor \frac{n}{2} - \Theta\left(\sqrt{n \log n}\right) \right\rfloor$, имеют асимптотически одинаковую $\mathcal{L}$-сложность.

Тогда для почти всех функций $l_{\mathcal{L}}(f) \sim l(f)$ при $n \to \infty$.

Для любой меры сложности среди граней, которые имеют размерность не более k_0 и расположены в поясе куба ширины большей k_0, минимальная сложность достигается на гранях размерности k_0, а максимальная — на гранях размерности 0. Поэтому условие (ii) теоремы 2 выполняется, если

$$\min_{i=r,\ldots,n-r-k_0} \mathcal{L}^n(i, i+k_0) \sim \max_{i=r,\ldots,n-r} \mathcal{L}^n(i,i)$$

для $r = \left\lfloor \frac{n}{2} - \Theta\left(\sqrt{n \log n}\right) \right\rfloor$ и $k_0 = \lceil \log n \rceil$ при $n \to \infty$.

Верхние мощностные оценки для максимальных и типичных значений числа минимальных ДНФ булевой функции считались тривиальными и ставилась задача получения нетривиальных верхних оценок [4, стр. 102]. Однако для максимального числа минимальных ДНФ верхняя оценка оказалась достижима по порядку логарифма [5].

Для числа $\mathcal{L}$-минимальных комплексов граней функции f, которое обозначается через $\mu_{\mathcal{L}}(f)$, верхняя оценка может быть получена из очевидного соотношения:

$$\mu_{\mathcal{L}}(f) \leqslant \sum\nolimits_{i=l(f)}^{l_{\mathcal{L}}(f)} \binom{g(f)}{i}$$

где $g(f)$ — число допустимых граней булевой функции f. Для почти всех функций из этого соотношения следует, что $\log \mu_l(f) \leqslant \bar{c}_n \cdot 2^n (1 + o(1))$, где $1 \leqslant \bar{c}_n$, т.е. верхняя оценка превосходит число булевых функций n переменных.

Теорема 3. *Если $l_{\mathcal{L}}(f) \sim l(f)$ для почти всех функций при $n \to \infty$, то*

$$\log \mu_{\mathcal{L}}(f) \lesssim (\bar{c}_n - 1) \cdot 2^n.$$

Из теоремы 3 и оценки $\bar{c}_n \leqslant 1.5$ [2] следует, что при $n \to \infty$ для почти всех функций $\log \mu_l(f) \lesssim 2^{n-1}$ и $\log \mu_{\mathcal{L}}(f) \lesssim 2^{n-1}$, если $l_{\mathcal{L}}(f) \sim l(f)$.

Следствие 2. *Пусть для аддитивной меры сложности функционал имеет вид $L_Q(I) = Q(L_0(I), L_1(I))$, где $Q(x, y)$ — произвольный полином двух переменных и I — грань куба. Тогда для почти всех функций $l_{L_Q}(f) \sim l(f)$ и, соответственно, для числа L_Q-минимальных комплексов граней почти всех функций справедлива оценка теоремы 3.*

Если сложность грани может быть величиной порядка c^n в единичном кубе B^n, то сложность комплекса граней для случайной функции может определяться отдельными гранями. Например, аддитивная мера сложности $\mathcal{L}$, определяется соотношением $\mathcal{L}(I) = c^{L_1(I)}$ для любой грани I, где $c > 4$. Для почти всех функций любой неприводимый комплекса граней M содержит грань размерности не более $\lceil \log n \rceil$ и «близкую» к вершине $\tilde{1}$, т.е. L_1-сложность такой грани асимптотически равна n. При этом почти все единичные вершины функции расположены в средних слоях куба B^n и содержатся в гранях, для которых L_1-сложность асимптотически равна $\frac{n}{2}$. Так как длина любого неприводимого комплекса граней меньше 2^n, то при $c > 4$ и $n \to \infty$ выполняется:

$$\mathcal{L}(M) \gtrsim l(M)(1 - o(1))\, c^{n/2(1-o(1))} + c^{n(1-o(1))} \sim c^{n(1-o(1))}.$$

Представляется интересным вопрос о существовании аддитивной меры сложности $\mathcal{L}$, для которой выполняется условие (i) и не выполняется условие (ii) теоремы 2. Существование такой меры сложности будет означать, что выполнения только условия (i) не достаточно для справедливости теоремы 2.

Работа выполнена при поддержке РФФИ, проект № 13-01-00958 А.

Литература

[1] *Кузнецов С. Е.* О нижней оценке длины кратчайшей ДНФ почти всех булевых функций // Вероятностные методы и кибернетика. —Казань: Изд-во Казанского ун-та. — 1983. — № 19. — С. 44–47.

[2] *Андреев А. Е.* Об одной модификации градиентного алгоритма // Вестник МГУ. Мат. Мех. — 1985. — № 3. — С. 29–35.

[3] *Pippenger N.* The shortest disjunctive normal form of a random Boolean function // Random Structures & Algorithms. — 2003. — V. 22, № 2. — P. 161–186.

[4] *Васильев Ю. Л., Глаголев В. В.* Метрические свойства дизъюнктивных нормальных форм // Дискретная математика и математические вопросы кибернетики. Т. 1. — М.: Наука. — С. 99–148.

[5] *Чухров И. П.* О минимальных комплексах граней в единичном кубе // Дискрет. анализ и исслед. операций. — 2012. — Т. 19, № 3. — С. 79–99.

Мебиусовы алгебры, связанные с задачами линейного программирования

В. Н. Шевченко

shev@uic.nnov.ru

Нижегородский государственный университет им. Н. И. Лобачевского

1. С целочисленной $(m+1)\times(d+1)$-матрицей $A=(a_{ij})$, $i=0,1,\ldots,m$; $j=0,1,\ldots,d$, свяжем полиэдр P, то есть множество решений системы линейных неравенств

$$a_{i0}+\sum_{k=1}^{d}a_{ik}x_k\geqslant 0,\quad i=1,\ldots,m, \tag{1}$$

и соответствующий ему конус $C(P)$

$$x_0\geqslant 0,\quad a_{i0}x_0+\sum_{k=1}^{d}a_{ik}x_k\geqslant 0,\quad i=1,\ldots,m, \tag{2}$$

их граневые решетки и решетки их триангуляций. Соответствующие определения можно посмотреть в [1, 2].

Цель доклада — изложить имеющиеся и вновь полученные результаты о мебиусовых алгебрах, связанных с этими решетками. Под мебиусовой алгеброй здесь понимается алгебра инцидентности частично упорядоченного множества (poset) [3, 4].

2. Для $(d\times n)$-матрицы B со столбцами b_j $(j=1,2,\ldots,n)$ обозначим через $B^{\angle}=\{\sum_{j=1}^{n}b_jy_j:\ y_j\geqslant 0\}$ множество неотрицательных линейных комбинаций ее столбцов и предположим, что $B^{\angle}$ совпадает со множеством решений системы

$$\sum_{k=1}^{d}a_{ik}x_k\geqslant 0,\quad (i=1,\ldots,m) \tag{3}$$

Триангуляцией конуса K с узлами из множества B назовем множество $T(B)=\{S_1,\ldots,S_t\}$ таких S_τ, для которых выполнены следующие условия:

1) $S_\tau\subseteq\{1,\ldots,n\}$,
2) $|S_\tau|=r=\operatorname{rank}B(S_\tau)$,
3) $B^{\angle}=\bigcup_{\tau=1}^{t}B^{\angle}(S_\tau)$,
4) $B^{\angle}(S_\tau)\cap B^{\angle}(S_\sigma)=B^{\angle}(S_\tau\cap S_\sigma)$.

Множество $\Delta(T(B))=\bigcup_{\tau=1}^{t}\Gamma(S_\tau)$ дает пример геометрической реализации d-мерного однородного симплициального комплекса (с. к.). При $k=0,\ldots,d$ обозначим через $\Delta_k=\bigcup_{\tau=1}^{t}\Gamma_k(S_\tau)$ множество k-мерных граней с. к. Δ, положим $f_k(\Delta)=|\Delta_k|$, $f(\Delta)=(f_0(\Delta),\ldots,f_d(\Delta))$ и

$$f(\lambda,\Delta)=\sum_{k=0}^{d}f_k(\Delta)\lambda^k.$$

Следуя [2, 5–8], представим многочлен $f(\lambda, \Delta)$ в виде

$$f(\lambda, \Delta) = \sum_{k \in \mathbf{Z}_+} \gamma_k(\Delta) \lambda^k (1+\lambda)^{d-k}$$

и назовем целочисленную последовательность $\gamma = (\gamma_0, \gamma_1, \dots)$ (d, n)-реализуемой, если $\gamma_k = \gamma_k(\Delta)$ при $k = 0, 1, \dots, d$ и $\gamma_k = 0$ при $k > d$.

Для любых натуральных чисел a и i существует единственное *биномиальное i-разложение числа*

$$a = \binom{a_i}{i} + \binom{a_{i-1}}{i-1} + \dots + \binom{a_j}{j},$$

где

$$a_i > a_{i-1} > \dots > a_j \geqslant j \geqslant 1.$$

Тогда число

$$a^{<i>} = \binom{1+a_i}{1+i} + \dots + \binom{1+a_j}{1+j}$$

называется *i-й псевдостепенью числа a*.

Теорема 1.

1. *Если найдется такое k, при котором $\gamma_{k+1} > \gamma_k^{<k>}$, то последовательность γ не реализуема ни при каком d.*
2. *Если $\gamma_{k+1} \leqslant \gamma_k^{<k>}$ при $k = 1, \dots, d-1$ (условия Маколея), то последовательность γ — $(2d)$-реализуема.*

Следующая теорема позволяет находить минимальное d, при котором последовательность γ d-реализуема.

Теорема 2. *Для d-реализуемости целочисленной последовательности $\gamma = (\gamma_0, \gamma_1, \dots)$ необходимо и достаточно, чтобы выполнялись следующие условия:*

1. $\gamma_0 = 1$, $\gamma_i \geqslant 0$ *при* $i = 1, \dots, d$ *и* $\gamma_k = 0$ *при целых* $k \geqslant d$,
2. $\gamma_i \geqslant \gamma_{d-i} \leqslant \gamma_{d-i-1}$ *при* $i = 1, \dots, \lfloor \frac{d}{2} \rfloor$
3. $\gamma_{i+1} - \gamma_{j-i} \leqslant (\gamma_i - \gamma_{j+1-i})^{<i>}$ *при* $j = d, \dots, 2d$ *и* $i = 1, \dots, \lfloor \frac{j}{2} \rfloor$.

Все ли перечисленные в теореме 2 условия необходимо проверять, автору не известно.

3. Для решения аналогичного вопроса о триангуляции конуса $C(P)$ достаточно найти (в линейном пространстве Q^{d+1}) матрицу $B(P)$, аналогичную матрице B, и, заменив d на $d+1$, воспользоваться теоремами 1 и 2.

Полученный критерий позволяет поставить вопрос о «наиболее экономной» реализации f-вектора (точнее говоря, γ-вектора). Предполагается ознакомить слушателей с результатами, полученными на этом пути.

Литература

[1] *Бухштабер В. М., Панов Т. Е.* Торические действия в топологии и комбинаторике. М.: МЦНМО, 2004.

[2] *Шевченко В. Н.* О разбиении выпуклого политопа на симплексы без новых вершин // Известия ВУЗ. Математика. 1997, №,12. С. 89–99.

[3] *Стенли Р.* Перечислительная комбинаторика. М.: Мир, 1990.

[4] *Холл М.* Комбинаторика. М.: Мир, 1970

[5] *Шевченко В. Н.* Триангуляции выпуклых многогранников и реализация их f-векторов // Российская конференция «Дискретная оптимизация и исследование операций»: Материалы конференции (Алтай, 27 июня – 3 июля 2010). Новосибирск: Изд-во Ин-та математики, 2010. С. 75–81.

[6] *Шевченко В. Н.* Триангуляции многогранных конусов и булевы функции // Информационный бюллетень Ассоциации математического программирования. № 12. Екатеринбург: УрО РАН, 2011, С. 221–222.

[7] *Шевченко В. Н.* Триангуляции многогранных конусов и реализация их f-векторов // Алгебра и линейная оптимизация: Междунар. конф., посв. 100-лет. С. Н. Черникова (2012; Екатеринбург): тез. докл. ИММ УрО РАН. Екатеринбург: УМЦ-УПИ, 2012, С. 182–185

[8] *Шевченко В. Н.* Триангуляции выпуклых многогранников и их булевы функции // Математические вопросы кибернетики. Вып. 16. М.: Физматлит, 2007. С. 43–56

Сводимость и равносильность недоопределенных алфавитов

Л. А. Шоломов

sholomov@isa.ru

Институт системного анализа РАН, Москва

Задан конечный алфавит $A_0 = \{a_i \mid i \in M\}$ *основных* символов. Каждому непустому $T \subseteq M$ сопоставлен символ a_T, называемый *недоопределенным*. *Доопределением символа* a_T считается всякий основной символ a_i, $i \in T$. Выделена система $\mathcal{T} \subseteq 2^M$ некоторых непустых подмножеств T множества M и с ней связан *недоопределенный алфавит* $A = \{a_T \mid T \in \mathcal{T}\}$. Через A^* будем обозначать множество всех слов в алфавите A, через $\mathbf{a}^n$ — слово $\mathbf{a} \in A^n$.

В отличие от обычной постановки задачи сжатия данных [1], когда по коду требуется полностью восстановить исходную информацию, в случае недоопределенных данных кодирование должно обеспечить восстановление лишь какого-либо доопределения данных. Такая более мягкая постановка предоставляет некоторые дополнительные возможности, одной из которых является возможность перехода к новому алфавиту. Преобразования недоопределенных данных, сохраняющие информационные свойства, изучались в [2]. В настоящей работе нас больше интересуют алгоритмические свойства.

Пусть наряду с A задан недоопределенный алфавит B. Для него основным алфавитом является $B_0 = \{b_j \mid j \in L\}$ и символы $b_U \in B$ соответствуют множествам U некоторой системы $\mathcal{U} \subseteq 2^L$. Считаем, что взаимоотношение алфавитов A и B представлено соответствием $R_{AB} \subseteq A \times B$ общего вида, в котором символы алфавита A могут иметь несколько образов, символы алфавита B — несколько прообразов. Назовем алфавиты A и B с заданным для них соответствием R_{AB} *соответственными алфавитами*; символы a_T и b_U, такие что $(a_T, b_U) \in R_{AB}$, *соответственными символами*; последовательно-

сти $\mathbf{a} = a_{T_1} \dots a_{T_n}$ и $\mathbf{b} = b_{U_1} \dots b_{U_n}$, для которых $(a_{T_i}, b_{U_i}) \in R_{AB}$, $i = 1, \dots, n$, *соответственными последовательностями*.

При сжатии недоопределенных данных в алфавите A каждой последовательности $\mathbf{a} \in A^*$ сопоставляется код $c(\mathbf{a}) \in \{0,1\}^*$, позволяющий восстановить какое-либо ее доопределение $\mathbf{a}^{(0)}$. Аналогичные обозначения используются в случае алфавита B. Скажем, что *сжатие в алфавите B сводится к сжатию в алфавите A*, если для любой пары соответственных последовательностей $(\mathbf{a}^n, \mathbf{b}^n)$ существует двоичная последовательность (инструкция) $\tilde{\nu}$ длины $l(\tilde{\nu}) = o(n)$, позволяющая по любому коду $c(\mathbf{a}^n)$ построить некоторый код $c(\mathbf{b}^n)$. В этом случае будем говорить, что алфавит A *сильнее* алфавита B и записывать $A \succsim B$. Соответственные алфавиты A и B будем называть *равносильными* и записывать $A \eqsim B$, если $A \succsim B$ и $B \succsim A$. Равносильность содержательно означает, что асимптотически оптимальное кодирование последовательностей $\mathbf{b}^n$ можно получить, сведя сжатие в алфавите B к сжатию в алфавите A и используя для последовательностей $\mathbf{a}^n$ асимптотически оптимальное кодирование.

Всякую функцию $F : M \to L$ можно распространить на A, положив $F(a_T) = b_{F(T)}$, где $F(T) = \{F(i) \mid i \in T\}$. Скажем, что *алфавит B функционально выразим через A*, если существует функция F такая, что для всех пар $(a_T, b_U) \in R_{AB}$ имеет место $F(a_T) \subseteq b_U$.

Справедлив следующий факт.

Теорема 1. *Для соответственных алфавитов A и B соотношение $A \succsim B$ имеет место тогда и только тогда, когда алфавит B функционально выразим через A.*

Рассмотрим вопрос распознавания соотношений $A \succsim B$ и $A \eqsim B$ для соответственных алфавитов A и B. Скажем, что символ $a_i \in A_0$ *мажорирует* в алфавите A символ $a_j \in A_0$, если для всякого $T \in \mathcal{T}$ принадлежность $j \in T$ влечет $i \in T$. Операция *исключения мажорируемого символа* a_j из алфавита A состоит в том, что каждый символ $a_T \in A$ заменяется символом $a_{T \setminus j}$. Последовательным исключением из алфавита A мажорируемых символов можно построить алфавит $\hat{A}$, называемый *приведенным*, к которому эта операция не применима. Можно доказать, что приведенный алфавит единствен с точностью до изоморфизма (переименования символов). Аналогичным образом по алфавиту B можно построить приведенный алфавит $\hat{B}$. Если $\hat{M}$ и $\hat{L}$ означают множества индексов символов, не исключенных из $\hat{A}$ и $\hat{B}$, то $\hat{A} = \{a_{T \cap \hat{M}} \mid a_T \in A\}$, $\hat{B} = \{b_{U \cap \hat{L}} \mid b_U \in B\}$. Алфавиты $\hat{A}$ и $\hat{B}$ связаны соответствием $R_{\hat{A}\hat{B}} = \{(a_{T \cap \hat{M}}, b_{U \cap \hat{L}} \mid (a_T, b_U) \in R_{AB}\}$.

Теорема 2.
1. Соответственные алфавиты A и B равносильны тогда и только тогда, когда построенные по ним приведенные алфавиты $\hat{A}$ и $\hat{B}$ изоморфны.
2. Существует эффективный (полиномиальный) алгоритм распознавания равносильности соответственных алфавитов.

По эффективному методу распознавания соотношения $A \eqsim B$ может быть построен эффективный метод распознавания $A \succsim B$ в согласно следующему утверждению.

Утверждение 3. *Соотношение* $A \gtrsim B$ *выполнено тогда и только тогда, когда* $AB \approx A$, *где* $AB = \{a_T b_U \mid (a_T, b_U) \in R_{AB}\}$, $R_{AB,A} = \{(a_T b_U, a_T) \mid (a_T, b_U) \in R_{AB}\}$.

Опишем план доказательства теоремы 1.

1°. Рассматриваются *недоопределенные источники* X в алфавите A, порождающие независимо символы $a_T \in A$ с некоторыми вероятностями p_T. *Энтропией* источника X называется величина

$$\mathcal{H}(X) = \min_Q \left\{ -\sum_{T \in \mathcal{T}} p_T \log \sum_{i \in T} q_i \right\},$$

где минимум берется по наборам $Q = (q_i,\ i \in M)$ вероятностей символов a_i. Подробнее об энтропии недоопределенных данных см. в [3]. *Источники* X и Y в алфавитах A и B, заданные совместным распределением $\{p(a_T, b_U)\}$, называются *соответственными*, если $p(a_T, b_U) > 0 \Rightarrow (a_T, b_U) \in R_{AB}$. В случае $A \gtrsim B$ доопределение любой пары $(\mathbf{a}^n, \mathbf{b}^n)$ соответственных последовательностей может быть восстановлено по двоичной информации $(c(\mathbf{a}^n), \tilde{\nu})$, длина которой асимптотически совпадает с длиной кода $c(\mathbf{a}^n)$. Отсюда и из теоремы кодирования недоопределенных данных [3] выводится, что соответственные пары источников (X, Y) удовлетворяют соотношению $\mathcal{H}(XY) = \mathcal{H}(X)$.

2°. Доказывается, что если алфавит A не содержит мажорируемых символов, то для любого $a_i \in A$ существует источник X с алфавитом A, энтропия $\mathcal{H}(X)$ которого достигается на наборе Q, в котором $q_i > 0$.

3°. В случае, когда алфавит A не содержит мажорируемых символов и соответственные источники X и Y в алфавитах A и B удовлетворяют условию $\mathcal{H}(XY) = \mathcal{H}(X)$, строится функция F, присутствующая в определении выразимости алфавита B через A. Построение использует результат из п. 2°.

4°. По произвольному источнику X путем устранения из его алфавита A всех мажорируемых символов и пересчета вероятностей строится *приведенный источник* $\hat{X}$. Доказывается, что $\mathcal{H}(\hat{X}) = \mathcal{H}(X)$ и для любого источника Y выполнено $\mathcal{H}(\hat{X}Y) = \mathcal{H}(XY)$.

5°. Если источники X и Y удовлетворяют условию $\mathcal{H}(XY) = \mathcal{H}(X)$, то в силу 4° выполнено $\mathcal{H}(\hat{X}Y) = \mathcal{H}(\hat{X})$. Это позволяет построить в соответствии с 3° функцию F, отображающую $\hat{A}$ в B, а затем распространить ее на A. Отсюда следует, что если выполнено $A \gtrsim B$, то алфавит B выразим через A.

6°. Обратно, если алфавит B выразим через A посредством функции F, то для любых соответственных последовательностей $\mathbf{a}$ и $\mathbf{b}$ и любого доопределения $\mathbf{a}^{(0)} = a_{i_1} \dots a_{i_n}$ для $\mathbf{a}$ последовательность $F(\mathbf{a}^{(0)}) = F(a_{i_1}) \dots F(a_{i_n})$ доопределяет $\mathbf{b}$. Поэтому в качестве $c(\mathbf{b})$ можно использовать $c(\mathbf{a})$. Доопределение последовательности $\mathbf{b}$ можно плучить как $F(\mathbf{a}^{(0)})$, где $\mathbf{a}^{(0)}$ — доопределение, найденное по $c(\mathbf{a})$.

Возможность использования кода $c(\mathbf{a}^n)$ вместо $c(\mathbf{b}^n)$ показывает, что дополнительная информации $\tilde{\nu}$ не расширяет область применимости данного подхода.

Работа выполнена при поддержке ОНИТ РАН, проект 1-1 программы "Интеллектуальные информационные технологии, системный анализ и автоматизация".

Литература

[1] *Галлагер Р.* Теория информации и надежная связь. — М.: Советское радио, 1974. — 720 с.

[2] *Шоломов Л. А.* Преобразование нечетких данных с сохранением информационных свойств // Дискретный анализ и исследование операций. — 2005. — Сер. 1, Т. 12, № 3. — С. 85–104.

[3] *Шоломов Л. А.* Элементы теории недоопределенной информации // Прикладная дискретная математика. Приложение №2. — 2009. — С. 18–42.

Представление автоматных марковских моделей на основе укрупнения цепей Маркова

Б.Ф. Эминов, В.М. Захаров

bulfami@mail.ru

КНИТУ-КАИ им.А.Н.Туполева, Казань

Представлено решение задачи алгоритмического синтеза автоматных марковских моделей на основе предложенного алгоритма укрупнения конечных цепей Маркова. Дана сравнительная оценка алгоритмической сложности рассматриваемых автоматных моделей. Определена зависимость сложности от размера стохастической матрицы, описывающей закон полученной укрупненной цепи Маркова, и длины имплицирующего вектора этой матрицы.

Введение

Конечный автономный вероятностный автомат (АВА) с детерминированной функцией выхода и со стохастической матрицей P переходов состояний можно преобразовать в композицию [1], состоящую из источника случайности Бернулли, выдающего с вероятностью p_i букву x_i из некоторого алфавита X, и детерминированного автомата. Методы синтеза [1, 2] АВА базируются на разложении стохастической матрицы P цепи Маркова (ЦМ) на множество стохастических булевых матриц и имплицирующий вектор (ИВ) [2] размера l. Величина l определяет комбинационную сложность соответствующего автомата.

В [3] предложен подход укрупнения ЦМ на основе заданной исходной простой ЦМ, удовлетворяющей определенным условиям укрупнения. В [4] представлен алгоритм данного укрупнения (алгоритм, сохраняющий марковское свойство укрупненного процесса).

Целью данной работы является алгоритмическое представление автоматной модели, реализующей последовательность состояний укрупненной цепи Маркова на основе заданной исходной простой ЦМ и получение оценки сложности модели.

Решение задачи

Пусть задана простая цепь Маркова системой вида

$$(P, S), \tag{1}$$

где $P = (p_{ij})$, $i, j = \overline{0, n-1}$ – стохастическая матрица ЦМ,

$S = (s_i)$ – множество состояний ЦМ.

Автоматной марковской моделью ЦМ, эквивалентной системе (1), является автономный вероятностный автомат вида [2]

$$(S, \hat{X}, \Delta(x, s)), \tag{2}$$

где S – то же самое, что и в (1), $\hat{X}$ – дискретная конечная случайная величина вида $\hat{X} = \begin{pmatrix} x_0 & x_1 & \dots & x_{l-1} \\ p_0 & p_1 & \dots & p_{l-1} \end{pmatrix}$, $0 \leqslant p_i \leqslant 1$, $\sum\limits_{i=0}^{l-1} p_i = 1$, а $\Delta(x, s)$ – функция переходов АВА.

Рассмотрим АВА (автоматную модель марковской функции[2])

$$(S, P, Y, \lambda(s) = y), \tag{3}$$

где элементы S и P – те же, что и в (1), $Y = \{y_0, y_1, \dots, y_{t-1}\}$ - выходной алфавит, $\lambda(s) = y$ - функция выхода, определенная на множестве Y.

Автомату (3) можно поставить в соответствие эквивалентный вероятностный автомат вида [2]

$$(S, \hat{X}, \Delta(x, s) = \ s, Y, \lambda(s) = y), \tag{4}$$

где S, Y, $\lambda(s)$ - те же элементы, что и в (3), а элементы $\hat{X}$, $\Delta(x, s)$ - те же, что и в (2).

Пару $(\hat{X}, \Delta(x, s))$ определим на основе разложения [1] матрицы P автомата (1):

$$P = \sum_{i=0}^{l-1} p_i M_i, \tag{5}$$

где p_i, $i = \overline{0, l-1}$ - элементы стохастического вектора $\overline{P}$ (элементы ИВ матрицы P), M_i, $i = \overline{0, l-1}$- стохастическая булева матрица размера $n \times n$, переменная l удовлетворяет соотношению

$$l \leqslant n^2 - n + 1. \tag{6}$$

Функцию $\lambda(s)$ зададим путем разбиения множества S на непересекающиеся подмножества

$$\{A_0, A_1, \dots, A_{t-1}\}, \bigcup_{i=0}^{t-1} A_i = S, A_j \cap A_k = 0 \text{ при } \forall j, \forall k = \overline{0, t-1} \text{ и } j \neq k. \tag{7}$$

Т.е. $\lambda(s)$ реализует отображение вида $\lambda(s)$: S $\rightarrow$ $Y = \{y_0, y_1, \dots, y_{t-1}\}$.

Функции $\Delta(x, s)$ и $\lambda(s)$ автомата (4) можно реализовать программно табличным способом, храня данные в оперативной памяти (ОЗУ). В этом случае

сложность реализации, измеряемая количеством ячеек ОЗУ, функций $\Delta(x,s)$ и $\lambda(s)$ определяются величинами соответственно

$$v_1 \;=\; l \times n \text{ и } v_2 \;=\; n \times t, \tag{8}$$

где l определяется из (6). Тогда сложность автомата (4) оценивается величиной $v = v_1 + v_2$.

Пусть в автоматах (2) и (3) цепь Маркова задана матрицей P и, при заданном разбиении (7) множества S, матрица P удовлетворяет условию укрупнения [3]. Тогда матрице P можно поставить в соответствие по алгоритму укрупнения [4] матрицу $\hat{P}$ размера $t \times t$, $t \leqslant n$, которая описывает укрупненную цепь Маркова со множеством состояний $S^{(y)} = \{s_0^{(y)}, s_1^{(y)}, ..., s_{t-1}^{(y)}\}$. Размер ИВ для матрицы $\hat{P}$, определяемого по разложению вида (5), удовлетворяет соотношению $l \leqslant t^2 - t + 1$.

Для этого случая сложность автоматов (2) и (4) можно оценить по аналогии с (8) величиной $v_y = l_1 \times t$, $v_y < v$.

Литература

[1] Davis A.S. Markov chains as random input automata. Amer. Math. Monthly, 1961, 68, 3, pp. 264-267.

[2] Бухараев Р.Г. Основы теории вероятностных автоматов. М.: Наука, 1985. 287 с.

[3] Кемени Дж., Снелл Дж. Конечные цепи Маркова. М.: Наука, 1970. 272 с.

[4] Захаров В.М., Эминов Б.Ф. Алгоритмы укрупнения цепей Маркова // Вестник Казанского государственного технического университета им. А.Н. Туполева, №2 (выпуск 1), 2013. С.125-133.

Подписано в печать 05.06.2014 г.
Форм. бум. 60х84 1/16. Печ. л. 19,25.
Тираж 200. Заказ № 5/06.

Издательство «Отечество».
420032, РТ, г. Казань, ул. Шоссейная, 22а.

Отпечатано в ООО «ОрионПлюс».
420000, РТ, г. Казань,
ул. Г.Камала, д. 7, офис 201.