Московский государственный университет имени М. В. Ломоносова
Пензенский государственный технологический университет
Вычислительный центр имени А. А. Дородницына РАН
Институт прикладной математики имени М. В. Келдыша РАН

Проблемы теоретической кибернетики

Материалы XVIII международной конференции
(Пенза, 19–23 июня 2017 г.)

Пенза, 2017

УДК 519.7
ББК 22.18
П 78

П78 **Проблемы теоретической кибернетики**: XVIII международная конференция (Пенза, 19–23 июня 2017 г.) : Материалы : Под редакцией Ю. И. Журавлева. — М. : МАКС Пресс, 2017. — 274 с.

Problems of theoretical cybernetics: XVIII International Conference (Penza, 19–23 June, 2017) : Proceedings : Ed. Yu. I. Zhuravlev. — M. : MAKS Press, 2017. — 274 p.

ISBN 978-5-317-05577-6

Сборник содержит доклады XVIII международной конференции «Проблемы теоретической кибернетики» (Пенза, 19–23 июня 2017 г.), организованной при поддержке Российского фонда фундаментальных исследований (проект № 17-01-20217-г). Тематика конференции включает следующие направления: синтез и сложность управляющих систем, надежность, контроль и диагностика управляющих систем, автоматы, языки и программирование, теория графов, комбинаторика, теория кодирования, теория распознавания образов, математическое программирование и исследование операций, математическая теория интеллектуальных систем, прикладная математическая логика, теория функциональных систем, теория оптимального управления, приложения кибернетики в естествознании и технике.

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

Ключевые слова: синтез и сложность управляющих систем, надежность управляющих систем, контроль и диагностика управляющих систем, автоматы, языки программирования, теория графов, комбинаторика, теория кодирования, распознавание образов, интеллектуальные системы, прикладная математическая логика, функциональные системы, оптимальное управление.

The collection represents proceedings of the XVIII international conference “Problems of Theoretical Cybernetics” (Penza, 19–23 June, 2017), that is sponsored by Russian Foundation for Basic Research (project N 17-01-20217-г). The conference subject area includes: control systems synthesis, complexity, reliability, and diagnostics; automata; computer languages and programming; graph theory; combinatorics; coding theory; theory of pattern recognition; mathematical programming and operations research, mathematical theory of intelligence systems; applied mathematical logic; functional systems theory; optimal control theory; applications of cybernetics in natural science and technology.

For scientists and specialists in areas of mathematical cybernetics, discrete mathematics, computer science and their applications.

Keywords: control systems synthesis and complexity, control systems reliability and diagnostics, automata, computer languages, graph theory, combinatorics, coding theory, pattern recognition, intelligence systems, applied mathematical logic, functional systems, optimal control.

Под общей редакцией академика РАН Ю. И. Журавлева.

Ответственные редакторы: Д. С. Романов, Б. Р. Данилов

M.E.: D. S. Romanov, B. R. Danilov

ISBN 978-5-317-05577-6

Содержание

The quadrangular genus of complete graphs

Serge Lawrencenko

Russian State University of Tourism and Service, e-mail: lawrencenko@hotmail.com

The genus of the complete graph was established by Ringel and Youngs [6] and was mainly concerned with *triangulations* of surfaces. Nonetheless, since then a great deal of interest has also been generated in *quadrangulations* of surfaces. In particular, Hartsfield and Ringel [2, 3], the author [4, 5], and the author et al. [1] have considered minimal quadrangulations of surfaces. The purpose of the proposed talk is to identify and clarify copyright issues around the quadrangular genus of complete graphs and related topics.

Define the quadrangular genus of a graph G, denoted $\gamma_4(G)$, to be the minimum value of h for which G has a 2-cell embedding in S_h (the sphere with h handles) such that the smallest face in the embedding is a quadrangle. Similarly, define the nonorientable quadrangular genus of G, $\gamma_4^*(G)$, to be the minimum value of k for which G has a 2-cell embedding in N_k (the sphere with k crosscaps) such that the smallest face in the embedding is a quadrangle.

The author et al. [1] established the following result for the orientable case while Nora Hartsfield managed to do the nonorientable case.

Theorem 1. *The quadrangular genus of the complete graph on n vertices is given by*

$$\gamma_4(K_n) = \left\lceil \frac{n^2 - 5n + 8}{8} \right\rceil \tag{1}$$

for orientable surfaces and

$$\gamma_4^*(K_n) = \left\lceil \frac{n^2 - 5n + 8}{4} \right\rceil \tag{2}$$

for nonorientable surfaces.

In fact, paper [1] was written in 1998. It was submitted to Discrete Mathematics in June 1998 but shortly after that the main author (Lawrencenko) withdrew it from the journal. The reason for the withdrawal was that the author learned from Nora Hartsfield that she got formula (2).

Therefore, Nora and the author agreed to merge their papers and produce one joint paper with both formulae (1) and (2) instead of two complementary papers drafted separately. Unfortunately, the author was too slow in fulfilling the project, and as a result no journal paper has been published as yet. Sadly, Nora passed away in 2011. The author learned about her death only in winter 2016.

It must be emphasized that the main result of paper [5] (Liu W., Ellingham M. N., Ye D., and Zha X.—the author's former co-workers) immediately follows from our Theorem 1 (formulae (1) and (2)).

In addition, paper [1] has been circulating over the internet around the mathematics community. For instance, paper [1] was used by Vladimir Korzhik, Yusuke Suzuki, etc.: Citations of [1] are included in the reference list of Suzuki's important paper [7].

Moreover, Theorem 1 has been used by Yusuke Suzuki [7], with proper referencing to [1], for Yusuke's proofs of the following two results. The real significance of these results is that they perfectly show the relationship between triangulations and quadrangulations of closed surfaces.

Theorem 2 (Suzuki [7]). *There exists an integer h_0 such that for any two surfaces S_{h_1} and S_{h_2} satisfying the inequalities $h_1 \geqslant h_0$ and*

$$2h_2 - 3h_1 - \left\lceil \frac{-331 + 19\sqrt{1 + 48h_1}}{12} \right\rceil \geqslant -1,$$

there exists a triangulation of S_{h_1} whose graph quadrangulates S_{h_2}.

Theorem 3 (Suzuki [7]). *There exists an integer k_0 such that for any two surfaces N_{k_1} and S_{h_2} satisfying the inequalities $k_1 \geqslant k_0$ and*

$$2h_2 - k_1 - \left\lceil \frac{-331 + 19\sqrt{1 + 24k_1}}{12} \right\rceil \geqslant -1,$$

there exists a triangulation of N_{k_1} whose graph quadrangulates S_{h_2}.

However, Suzuki [7] mistakenly attributes formula (2) to Hartsfield and Ringel [2]: Paper [2] covers only the case $n \equiv 1 \mod 4$ and the proof of the following theorem should be built upon formula (2).

Theorem 4 (Suzuki [7]). *There exists an integer h_0 such that for any two surfaces S_{h_1} and N_{k_2} satisfying the inequalities $h_1 \geqslant h_0$ and*

$$k_2 - 3h_1 - \left\lceil \frac{-91 + 11\sqrt{1 + 48h_1}}{12} \right\rceil \geqslant -1,$$

there exists a triangulation of S_{h_1} whose graph quadrangulates N_{k_2}.

References

[1] Chen B., Lawrencenko S., Yang H. Determination of the 4-genus of a complete graph // Submitted to Discrete Mathematics and withdrawn by S. Lawrencenko, June 1998. // URL: `https://t.co/cUg6R9Jwyw` (Accessed on March 17, 2017).

[2] Hartsfield N., Ringel G. Minimal quadrangulations of orientable surfaces // J. Combin. Theory Series A. — 1989. Vol. 50. — P. 186–195.

[3] Hartsfield N., Ringel G. Minimal quadrangulations of orientable surfaces // J. Combin. Theory Series B. — 1989. Vol. 46. — P. 84–95.

[4] Lawrencenko S. Realizing the chromatic numbers and orders of spinal quadrangulations of surfaces // J. Combin. Math. Combin. Comput. — 2013. Vol. 87. — P. 303–308.

[5] Liu W., Ellingham M. N., Ye D., Zha X. Quadrangular embeddings of complete graphs // Cornell University Library, eprint arXiv:1606.00948. 3 Jun 2016. URL: https://t.co/1bBGI8QbcI (Accessed on March 18, 2017).

[6] Ringel G. Map Color Theorem. — New York, Heidelberg, Berlin : Springer-Verlag, 1974. — 191 p.

[7] Suzuki Y. Triangulations on closed surfaces which quadrangulate other surfaces II // Discrete Math. — 2005. Vol. 303. — P. 234–242. URL: https://t.co/l25n4pzv21 (Accessed on March 18, 2017).

[8] Лавренченко С. А. Спектр хроматических чисел спинальных квадрангуляций замкнутой поверхности / Материалы XI Международного семинара «Дискретная математика и ее приложения», посвящ. 80-летию со дня рождения акад. О. Б. Лупанова (Москва, МГУ, 18–23 июня 2012 г.) / Под ред. О. М. Касим-Заде. — М.: Изд-во мех.-мат. ф-та МГУ, 2012. — С. 295–298. URL: https://t.co/VROkvMyukJ (Accessed on March 18, 2017).

Метод квантового хеширования

Аблаев Фарид Мансурович, Аблаев Марат Фаридович, Васильев Александр Валерьевич, Зиятдинов Мансур Тагирович

Казанский федеральный университет, e-mail: fablayev@gmail.com, mablayev@gmail.com, alexander.ksu@gmail.com, gltronred@gmail.com

В работе [1] нами проведена формализация понятия квантового хеширования и положено начало исследований свойств квантового хеширования. В нескольких публикациях нашей группы были предложены уточнения понятия квантового криптографического хеширования, обобщающего в рамках квантовой информатики классические понятия односторонней (one-way) функции и свойства устойчивости к коллизиям [2].

Важным свойством квантовой криптографической функции является то, что такая функция является физически односторонней (physical one-way). Напомним, что в классической информатике понятие односторонности (one-way) функции является условным: односторонние функции существуют, если $P \neq NP$.

Оказалось, что квантовая односторонность и квантовая устойчивость к коллизиям противоречат друг другу: чем более односторонней является квантовая функция, тем менее она устойчива к коллизиям, и наоборот. Но оказалось, что можно хорошо сбалансировать эти свойства [3]. Отметим, что сбалансированные квантовые хеш-функции требуют высокой степени запутанности между кубитами. В связи с этим применяется техника фазовой трансформации для

создания квантовых хеш-конструкций, которая может быть реализована на основе существующих оптических технологий.

Важным результатом наших исследований является разработка метода конструирования квантовой хеш-функции на основе множеств с малым отклонением (small-bias sets), определяемых для конечных групп [4]. Техника множеств с малым отклонением развивается в теоретической информатике с 1990-ых. Известно, что в двоичном случае такие множества, по сути, определяются двоичными кодами, исправляющими ошибки. Поэтому, в частности, двоичные квантовые хеш-функции полностью описываются классом двоичных кодов, исправляющих ошибки. Отметим, что в общем (не двоичном) случае такой связи нет.

Для построения квантовых хеш-функций на основе классических семейств нами было введено понятие квантового хеш-генератора [5]. Под квантовым хеш-генератором понимается конечное множество функций, на основе которых можно конструировать квантовые хеш-функции. В этих терминах множества с малым отклонением задают квантовый хеш-генератор.

Важным результатом было доказательство того, что комбинация заданного квантового хеш-генератора (а следовательно и соответствующей квантовой хеш-функции) с семейством функций, задаваемых кодом, исправляющих ошибки, задают новый квантовый хеш-генератор, а следовательно и новую квантовую хеш-функцию.

В частности, используя связь между универсальными хеш-семействами и методом отпечатков Фрейвалдса, продемонстрирована явная квантовая хеш-функция, и доказано, что эта конструкция является оптимальной в смысле необходимого числа кубит.

Таким образом, в рамках наших исследований предложены методы построения квантовых хеш-функций на основе множеств с малым отклонением и на основе комбинаций таких конструкций и конструкций кодов, исправляющих ошибки.

На основе квантовых хеш-функций предложен ряд различных квантовых коммуникационных протоколов, в том числе протокол квантовой цифровой подписи [1]. В частности, рассмотрены подходы, на которых основаны многие эффективные и защищенные квантовые коммуникационные протоколы, такие как известный протокол однобитовой квантовой цифровой подписи.

Нами рассмотрены некоторые применения квантового хеширования. Среди них — приложение данной техники для вычисления булевых функций в квантовой коммуникационной модели [6]. Комбинация двоичных функций с q-ичной квантовой хеш-функцией осуществлена здесь на основе предложенного нами ранее полиномиального представления булевых функций, которое мы назвали характеристическим. Используя такую комбинацию представлений, мы предложили метод вычисления булевых функций в квантовой односторонней

коммуникационной модели, где один из участников выполняет свою часть вычислений и отправляет сообщение другому участнику, который после своей части вычислений должен выдать значение функции. Некоторые результаты также справедливы и для более ограниченной трехсторонней модели с одновременной отправкой сообщений, в которой участники могут взаимодействовать только через третью сторону, называемую рефери. Нами приведены некоторые явные примеры булевых функций, чьи полиномиальные представления обладают нужными свойствами для применения предложенного подхода. При этом получаемые квантовые трехсторонние протоколы экспоненциально превосходят известные классические аналоги.

Также нами предложены адаптации метода квантового хеширования для построения эффективных квантовых алгоритмов в различных квантовых моделях вычислений: в конечных квантовых автоматах и один раз читающих квантовых ветвящихся программах [7].

Работа выполнена при поддержке РФФИ (проект № 17-07-01606-а).

Список литературы

[1] Ablayev F. M., Vasiliev A. V. Cryptographic quantum hashing // Laser Phys. Lett. — 2014. — V. 11, No 2. — P. 025202.

[2] Ablayev F. M., Ablayev M. F. On the concept of cryptographic quantum hashing // Laser Physics Letters. — 2015. — V. 12, No. 12. — P. 125204.

[3] Ablayev F., Ablayev M., Vasiliev A. On the balanced quantum hashing // Journal of Physics: Conference Series. — 2016. — V. 681, No. 1. — P. 012019.

[4] Vasiliev A. Quantum hashing for finite abelian groups // Lobachevskii Journal of Mathematics. — 2016. — V. 37, No. 6. — P. 751–754.

[5] Ablayev F. M., Ablayev M. F. Quantum hashing via e-universal hashing constructions and classical fingerprinting // Lobachevskii Journal of Mathematics. — 2015. — V. 36, No. 2. — P. 89–96.

[6] Vasiliev A. Quantum communications based on quantum hashing // International Journal of Applied Engineering Research. — 2015. — V. 10, No. 12. — P. 31415–31426.

[7] Ablayev F., Vasiliev A. Computing boolean functions via quantum hashing // Computing with New Resources, C.S. Calude et al. (Eds.): Gruska Festschrift, Lecture Notes in Computer Science. — 2014. — V. 8808. — P. 149–160.

Эффективная реализация квантовых хеш-функций в квантовых ветвящихся программах на основе техники Фурье преобразований

Аблаев Фарид Мансурович, Аблаев Марат Фаридович[1], Корольков Андрей Вячеславович[2]

[1] Казанский федеральный университет, e-mail: fablayev@kpfu.ru,mablayev@kpfu.ru
[2] Московский технологический университет, Академия Криптографии РФ, e-mail: ankor11111@mail.ru

Квантовое хеширование является обобщением понятия хеширования для квантовых вычислительных моделей. Перспективы криптографического использования квантовых хеш функций представлены в [1]. Данная работа продолжает исследования работ [1–3].

Следующая система понятий и обозначений используется в работах [1–3]. Там же приводятся формулировки и доказательства цитируемых результатов.

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

— Пусть $\mathbb{X}$ – конечное множество. Функция $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ называется односторонне (one-way) ϵ-устойчивой функцией, если ψ эффективно вычисляется и вероятность ее обращения не превосходит ϵ.

— Классически-квантовая функция $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ является односторонне ϵ-устойчивой с $\epsilon \leqslant 2^s/|\mathbb{X}|$.

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

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

Это условие обеспечивает вероятность коллизии не более δ.

— **Свойство 1.** *Если функция $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ коллизия δ-устойчива, то верна следующая нижняя оценка на s*

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

— Односторонне δ-устойчивая и коллизия ϵ-устойчивая квантовую функцию $\psi : \mathbb{X} \to (\mathcal{H}^2)^{\otimes s}$ называют (δ, ϵ)-устойчивой квантовой хеш-функцией (δ, ϵ)-квантовой хеш-функцией.

Для $a \in \mathbb{Z}_q$ характер χ_a элемента x группы $\mathbb{Z}_q$ определяется как $\chi_a(x) = \omega^{ax}$, где $\omega = e^{\frac{2\pi i}{q}}$. Характер $\chi_0 \equiv 1$ называют тривиальным.

— Множество $S \subseteq \mathbb{Z}_q$ называется ϵ-biased, если для каждого нетривиального характера $\chi \in \{\chi_a : a \in \mathbb{Z}_q\}$ выполняется

$$\frac{1}{|S|}\left|\sum_{x\in S}\chi(x)\right| \leqslant \epsilon.$$

ϵ-biased множество S интересно, если $|S| \ll |\mathbb{Z}_q|$ (т. к. $S = \mathbb{Z}_q$ является 0-biased множеством). Первые результаты о конструкциях ϵ-biased множеств были доказаны в 1990-ых. В частности, доказано, что ϵ-biased множества S мощности $O(\log q/\epsilon^2)$ существуют. В последнее десятилетие предложены явные конструкции ϵ-biased множеств близкие по мощности к $O(\log q/\epsilon^2)$.

Теорема 1. *Пусть* $0 < \epsilon < 1$, *пусть* $S \subseteq \mathbb{Z}_q$ *является* ϵ*-biased множеством Пусть*

$$H_S = \{h_a(x) = ax \pmod q, \quad a \in S\}$$

— множество функций ($h_a : \mathbb{Z}_q \to \mathbb{Z}_q$*), определяемых множеством* S*. Тогда квантовая функция* $\psi_S : \mathbb{Z}_q \to (\mathcal{H}^2)^{\otimes \log |S|}$

$$|\psi_S(x)\rangle = \frac{1}{\sqrt{|S|}}\sum_{a\in S}\omega^{h_a(x)}|a\rangle$$

является (δ, ϵ) *устойчивой квантовой хеш функцией с* $\delta \leqslant |S|/(q \log q)$.

Отметим, что при выборе ϵ-biased множества $S \subseteq \mathbb{Z}_q$ минимально возможным (мощности $O(\log q/\epsilon^2)$) конструкция квантовой хеш функции ψ_S является оптимальной в силу свойства 1. В этом случае квантовая функция $\psi_S : \mathbb{Z}_q \to (\mathcal{H}^2)^{\otimes \log |S|}$ является (δ, ϵ) устойчивой квантовой хеш функцией с $\delta = O(1/(q\epsilon^2))$.

Основным результатом данной работы является теорема

Теорема 2. *Пусть* $S \subset \mathbb{Z}_q$ *является* ϵ*-biased множеством,* $T = |S|$. *Тогда для каждого* $x \in \mathbb{Z}_q$ *его квантовый хеш образ* $|\psi_S(x)\rangle$ *(квантовое хеш состояние)*

$$|\psi_S(x)\rangle = \frac{1}{\sqrt{T}}\sum_{j=0}^{T-1}\omega^{a_j x}|j\rangle$$

может быть вычислено квантовой ветвящейся программой Q *со следующими характеристиками:* Q *построена на* $s = \log T$ *кубитах, вычисление* $|\psi_S(x)\rangle$ *происходит за* $\log q$ *шагов.*

Этот результат показывает, что при выборе ϵ-biased множества S близкого по мощности $O(\log q/\epsilon^2)$ элементы группы $\mathbb{Z}_q$ хешируются весьма эффективно с использованием порядка $\log \log q$ кубит и за $\log q$ шагов.

При этом отметим, что наша конструкция требует максимальной “сцепленности” этих $\log \log q$ кубит. Задача построения устойчиво работающих квантовых

регистров, основанных на квантовой памяти с максимально возможным сцеплением, образующих его кубит, является на сегодняшний день центральной задачей квантовых технологий. Полученные, в частности, в этой работе результаты демонстрируют высокую эффективность возможностей квантового хеширования в случае успехов в разработке квантовой памяти, устойчиво оперирующей уже с несколькими десятками кубит.

Список литературы

[1] Корольков -А. О некоторых прикладных аспектах квантовой криптографии в контексте развития квантовых вычислений и появления квантовых компьютеров //Вопросы кибербезопасности. — 2015. — Т. 9, № 1. — С. 6–13.

[2] Ablayev M. On quantum (δ, ϵ)-resistant hashing // Lobachevskii Journal of Mathematics. — 2016. — V. 37, № 6. — P. 755–764.

[3] Ablayev F., Ablayev M., Vasiliev A., Ziatdinov M. Quantum fingerprinting and quantum hashing. Computational and cryptographical aspects // Baltic J. Modern Computing. — 2016. — V. 4, No. 4. — P. 860–875.

О примитивных однородных графах с экспонентом, равным 2

Абросимов Михаил Борисович[1], **Костин Сергей Вячеславович**[2]

[1] Саратовский национальный исследовательский государственный университет имени Н. Г. Чернышевского, e-mail: mic@rambler.ru

[2] Московский технологический университет (МИРЭА), e-mail: kostinsv77@mail.ru

Неотрицательная квадратная матрица A называется *примитивной*, если существует натуральное k, такое что A^k положительна. Минимальное такое значение k называется *экспонентом* матрицы A [1]. Понятие примитивности легко переносится на графы. Мы будем рассматривать простые графы, то есть графы без петель и кратных ребер.

Вершина v достижима из вершины u за $k \geqslant 1$ шагов, если существует последовательность рёбер (маршрут) $\{u, w_1\}, \{w_1, w_2\}, \ldots, \{w_{k-1}, v\}$. Если A — матрица смежности графа $G = (V, \alpha)$, то есть двоичная булева матрица отношения смежности α, то достижимость вершины v из вершины u за k шагов означает, что на пересечении строки и столбца, соответствующих вершинам u и v соответственно, в матрице A^k стоит 1 (сложение и умножение элементов матриц выполняется в булевой алгебре).

Граф $G = (V, \alpha)$ называется *примитивным*, если существует натуральное k, такое что между любой парой вершин графа G существует маршрут длины k (иначе говоря, в матрице A^k все элементы равны 1). Минимальное такое значение k называется экспонентом графа G и обозначается $\exp(G)$. Ряд работ посвящён исследованию экспонентов однородных примитивных матриц [2, 3]. С точки зрения графов рассматриваемые в этих работах матрицы соответствуют

диграфам. В данной работе мы будем рассматривать экспоненты неориентированных однородных графов.

Однородным или *регулярным* n-вершинным графом порядка p называется простой неориентированный n-вершинный граф, все вершины которого имеют степень p. Множество n-вершинных однородных графов порядка p будем обозначать $R_{n,p}$.

Очевидно, что любой примитивный граф является связным. Цикл длины 3 будем называть треугольником. Через $g(G)$ будем обозначать обхват графа G, то есть наименьшую из длин циклов графа G. Так как в неориентированных графах нет петель, то примитивных графов с экспонентом, равным 1, не существует, то есть $\exp(G) > 1$. Нас будут интересовать однородные графы с $\exp(G) = 2$. Очевидно, что диаметр таких графов $d \leqslant 2$, однако это условие не является достаточным.

Теорема 1 (критерий). *Граф G является примитивным с $\exp(G) = 2$ тогда и только тогда, когда $d \leqslant 2$, и каждое ребро графа G входит в треугольник.*

Второе условие теоремы отдельно также не является достаточным. На рисунке 1 представлен 10-вершинный регулярный граф порядка 4. Можно заметить, что каждое ребро этого графа входит в треугольник, граф является примитивным, однако его экспонент равен 3.

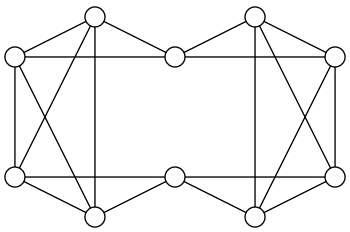


Рис. 1. 10-вершинный регулярный граф порядка 4 с $\exp(G) = 3$.

Если рассматривать произвольные графы, то можно найти пример с меньшим числом вершин. На рисунке 2 представлен 7-вершинный граф с $\exp(G) = 3$, каждое ребро которого входит в треугольник:

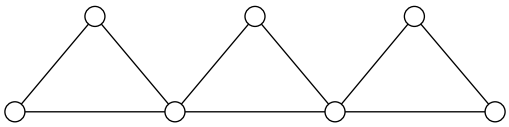


Рис. 2. 7-вершинный граф с $\exp(G) = 3$.

Следствие. *Пусть G — примитивный граф с $\exp(G) = 2$. Тогда его обхват $g(G) = 3$.*

Легко заметить, что любой полный граф K_n при $n > 2$ является примитивным, и $\exp(K_n) = 2$. Так как каждое ребро примитивного графа G с $\exp(G) = 2$

входит в треугольник, то степень всех вершин графа G не ниже 2. Оказывается, оценку минимальной степени вершин графов с экспонентом, равным 2, можно повысить.

Теорема 2. *Среди регулярных графов $R_{n,2}$ только граф K_3 имеет экспонент, равный 2.*

Теорема 3. *Примитивных кубических графов с экспонентом, равным 2, при $n > 4$ не существует.*

Теорема 4. *Если $p > n/2$, то любой n-вершинный регулярный граф порядка p является примитивным с экспонентом, равным 2.*

Следующий результат показывает, что для заданного p число p-регулярных графов с экспонентом, равным 2, является конечным.

Теорема 5. *Если p — чётное и $n > p^2 - p + 1$ или p — нечётное и $n > p^2 - p - 1$, то не существует примитивных n-вершинных регулярных графов порядка p с экспонентом, равным 2.*

Был произведён вычислительный эксперимент с использованием кластера высокопроизводительных вычислений ПРЦ НИТ СГУ по подсчёту регулярных графов с экспонентом, равным 2, и числом вершин до 16. Результаты для 4- и 5-регулярных графов представлены в таблице. Вычисления показывают, что оценка из теоремы 5 может быть улучшена.

Таблица 1: Количество n-вершинных p-регулярных графов с экспонентом, равным 2

n	$p = 4$	$p = 5$
4	0	0
5	1	0
6	1	1
7	2	0
8	2	3
9	3	0
10	0	24
11	1	0
12	0	210
13	0	0
14	0	116
15	0	0
16	0	2

Изображение 11-вершинного 4-регулярного графа с экспонентом 2 можно найти в статье [4].

Список литературы

[1] Wielandt H. Unzerlegbare nicht negative Matrizen // Math. Zeitschr. — 1950. — V. 52. — P. 642–648.

[2] Jin M., Lee S. G., Seol H. G. Exponents of r-regular primitive matrices // Trends in Mathematics Information Center for Mathematical Sciences. — 2003. — V. 6, № 2. — P. 51–57.

[3] Bueno M. I., Furtado S. On the exponent of r-regular primitive matrices // ELA. The Electronic Journal of Linear Algebra. — 2008. — V. 17. — P. 28–47.

[4] Костин С. В. Об использовании задач по теории графов для интеллектуального развития учащихся // Математика в образовании: Сб. статей. Вып. 10 / под ред. И.С. Емельяновой. Чебоксары: Изд-во Чувашского ун-та. — 2014. — С. 68–74.

Об оценке числа дополнительных дуг минимальных рёберных 1-расширений ориентаций цепи

Абросимов Михаил Борисович[1], **Моденова Ольга Владимировна**[2]

[1] Саратовский национальный исследовательский государственный университет, e-mail: mic@rambler.ru
[2] Научно-образовательный центр «Эрудит», e-mail: oginiel@rambler.ru

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

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

2) граф G^* содержит $n + k$ вершин, то есть $|V^*| = |V| + k$;

3) α^* имеет минимальную мощность при выполнении условий 1) и 2).

Понятие минимального вершинного k-расширения появилось в работе John P. Hayes [1] как модель для исследования отказоустойчивости элементов дискретных систем. Позднее в работе [2] была введена модель для исследования отказов связей между элементами.

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

1) граф G^* является рёберным k-расширением графа G, то есть граф G вкладывается в каждый граф, получающийся из G^* удалением любых его k рёбер (дуг);

2) граф G^* содержит n вершин, то есть $|V^*| = |V|$;

3) α^* имеет минимальную мощность при выполнении условий 1) и 2).

В работе [1] доказывается, что минимальным вершинным 1-расширением n-вершинной цепи является $(n+1)$-вершинный цикл, в работе [2] доказывается, что минимальным рёберным 1-расширением n-вершинной цепи является n-вершинный цикл. Легко показать [3], что эти расширения являются единственным с точностью до изоморфизма. Задача поиска минимального вершинного или рёберного k-расширения для произвольного графа является вычислительно сложной [4], и в общем виде решение удалось получить лишь для некоторых классов графов. Обзор основных результатов можно найти в работе [3].

Рассмотрим ориентации цепи. Очевидно, что ориентация цепи, являющая гамильтоновым графом, имеет единственное с точностью до изоморфизма минимальное рёберное 1-расширение — контур с тем же числом вершин. Другой интересной ориентацией является орцепь, состоящая только из источников и стоков, минимальным рёберным 1-расширением которой при чётном числе вершин будет ориентация цикла, состоящая только из источников и стоков.

Теорема 1. *Среди всех ориентаций цепей при чётном числе вершин MP-1P с 1 дополнительной дугой имеют только гамильтоновы цепи и цепи, состоящие только из источников и стоков. Среди всех ориентаций цепей при нечётном числе вершин только гамильтоновы цепи имеют MP-1P с 1 дополнительной дугой.*

Теорема 2. *Не существует ориентаций цепей с числом вершин n > 5 таких, что MP-1P отличается на 2 дополнительные дуги.*

Общих схем, которые бы позволили построить минимальные рёберные 1-расширения для каких-либо ориентаций цепей, отличных от гамильтоновой и цепи, состоящей из стоков и источников, пока получить не удается. В связи с этим большой интерес представляет оценка на число дополнительных дуг минимальных рёберных 1-расширений. Для минимальных вершинных 1-расширений ориентаций цепи удалось получить некоторые оценки на число дополнительных дуг (см. [4, 5]). В данной работе исследуется случай рёберных расширений. Обозначим число дополнительных дуг минимального рёберного 1-расширения графа G через $ec(G)$. Тогда оценка из последней теоремы может быть записана следующим образом:

$$3 \leqslant ec(P_n), \forall n > 5.$$

Полученный результат удалось улучшить. Очевидно, что концевая вершина ориентации цепи может быть или источником, или стоком. Будем говорить, что концевые вершины цепи одного типа, если они одновременно являются источниками или стоками. В противном случае будем говорить, что концевые вершины имеют разный тип.

Рёберное 1-расширение графа G называется *неприводимым*, если никакая его собственная часть не является рёберным 1-расширением графа G. Заметим достаточно очевидный факт, который можно использовать для получения верхней оценки числа дополнительных дуг:

Теорема 3. *Неориентированный цикл C_n при $n > 2$ является неприводимым рёберным 1-расширением для любой ориентации цепи P_n, отличной от гамильтоновой и от ориентации, состоящей из источников и стоков при чётном числе вершин.*

В неориентированном цикле подразумевается, что ребро представляет собой пару встречных дуг. Из теоремы получается оценка

$$ec(P_n) \leqslant n + 1.$$

Теорема 4. *Число дополнительных дуг MP-1P ориентации цепи P_n, имеющей концы разного типа и отличной от гамильтоновой и от ориентации, состоящей из чередующихся источников и стоков:*

$$\left\lceil \frac{n}{6} \right\rceil + 1 \leqslant ec(P_n) \leqslant n + 1.$$

Теорема 5. *Число дополнительных дуг MP-1P ориентации цепи P_n, имеющей концы одинакового типа:*

$$\left\lceil \frac{n}{4} \right\rceil + 1 \leqslant ec(P_n) \leqslant n + 1.$$

Список литературы

[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. О сложности некоторых задач, связанных с расширениями графов // Матем. заметки. — 2010. — Т. 88, вып. 5. — С. 643–650.

[4] Абросимов М. Б., Моденова О. В. Характеризация орграфов с малым числом дополнительных дуг минимального вершинного 1-расширения // Изв. Сарат. ун-та. Нов. сер. — 2013. — Т. 13. Сер. Математика. Механика. Информатика, вып. 2, ч. 2. — С. 3–9.

[5] Абросимов М. Б., Моденова О. В. Характеризация орграфов с тремя дополнительными дугами в минимальном вершинном 1-расширении // Прикладная дискретная математика. — 2013. — № 3. — С. 68–75.

О замкнутых классах в частичной k-значной логике, содержащих все функции, доопределимые до монотонных

Алексеев Валерий Борисович

МГУ имени М. В. Ломоносова, e-mail: vbalekseev@rambler.ru

Пусть $E_k = \{0, 1, \ldots, k-1\}$ и E_k^n — множество всех наборов длины n с элементами из E_k. Множество всех функций $f(x_1, x_2, \ldots, x_n)$, отображающих E_k^n в E_k, называют множеством k-значных функций и обозначают P_k, а множество всех функций $f(x_1, x_2, \ldots, x_n)$, отображающих E_k^n в $E_k \cup \{*\}$, называют множеством частичных k-значных функций и обозначают P_k^*. При этом $*$ трактуется как неопределенность. Алгебры таких функций относительно обычной операции суперпозиции, которая включает подстановку функций в функции, а также добавление и изъятие фиктивных переменных, называют, соответственно, k-значной логикой и частичной k-значной логикой. В работе рассматриваются классы функций в частичной k-значной логике, замкнутые относительно суперпозиции. В работе [1] автором совместно с Вороненко А. А. было показано, что для любого предполного класса A из P_2, отличного от класса линейных функций, в P_2^* существует конечное число замкнутых классов, содержащих A. Для класса линейных функций было показано, что множество таких классов имеет мощность континуум. Обобщение этой задачи для P_k^* при $k \geqslant 3$ рассмотрели Haddad L., Lau D., Rosenberg I. G. [2, 3]. Вопрос о мощности соответствующих семейств замкнутых классов был решен для всех предполных классов из P_k кроме предполных классов монотонных функций. Для этих классов было показано только, что все замкнутые классы в P_k^*, содержащие предполный (в P_k) класс монотонных функций, образуют такую же структуру из 9 классов, как в P_2^*, если частичный порядок S на E_k, относительно которого рассматривается монотонность, имеет один минимальный и один максимальный элемент и обладает свойством: если частичная функция не нарушает монотонность относительно S ни на какой паре наборов, то она доопределима до всюду определенной функции, монотонной относительно S.

В данной работе представлен следующий новый результат.

Теорема 1. *Для любого натурального t существуют k и частичный порядок S на E_k с одним минимальным и одним максимальным элементами такие, что между классом частичных функций, доопределимых до монотонных относительно S, и классом частичных функций без нарушения монотонности существует цепочка (по вложению) из t различных замкнутых классов.*

Пусть S — произвольный фиксированный частичный порядок на E_k. Обычным образом (покоординатное сравнение) он порождает частичный порядок S^n на E_k^n. Будем говорить, что у функции $f(x_1, x_2, \ldots, x_n)$ из P_k^* не нарушается

монотонность на множестве наборов D, если для любых двух наборов α и β из D, на которых функция определена (то есть принимает значение, отличное от $*$), выполняется импликация:

$$\alpha \leqslant \beta \Longrightarrow f(\alpha) \leqslant f(\beta)$$

(слева сравнимость относительно S^n, справа относительно S). Будем говорить, что функция $f(x_1, x_2, \ldots, x_n)$ из P_k^* монотонна на множестве наборов D, если она на D всюду определена и у нее на D не нарушается монотонность. Доопределением функции $f(x_1, x_2, \ldots, x_n)$ из P_k^* на множестве наборов D будем называть замену всех значений $*$ на наборах из D на значения из E_k. Будем говорить, что функция $f(x_1, x_2, \ldots, x_n)$ из P_k^* доопределима до монотонной на множестве наборов D, если существует такое ее доопределение на D, которое монотонно на D.

Пусть r — натуральное число и $r \geqslant 2$. Обозначим через M_S^r множество всех функций из P_k^*, которые доопределимы до монотонных (относительно частичного порядка S) на любом множестве наборов мощности r. Следующая теорема несложно доказывается.

Теорема 2. *Класс M_S^r замкнут при любых S и r.*

Нетрудно видеть, что M_S^2 — это класс всех функций из P_k^*, у которых не нарушается монотонность (на множестве всех наборов). Через M_S^∞ обозначим множество всех функций из P_k^*, которые доопределимы до монотонных (на множестве всех наборов). Нетрудно показать, что класс M_S^∞ также замкнут. Из определения вытекает, что:

$$M_S^2 \supseteq M_S^3 \supseteq M_S^4 \supseteq \ldots \supseteq M_S^\infty.$$

Если S — линейный порядок, то все классы в этой последовательности совпадают. Однако существуют частичные порядки, для которых $M_S^2 \neq M_S^\infty$, и возникает вопрос о количестве замкнутых классов между M_S^∞ и M_S^2. Теорема 1 показывает, что это количество для различных S не ограничено сверху.

Основой для доказательства теоремы 1 является следующая теорема. Пусть A, B — два подмножества из S. Будем писать $A \leqslant B$, если $a \leqslant b$ для любых двух элементов $a \in A$, $b \in B$. Будем говорить, что пара множеств A, B со свойством $A \leqslant B$ отделима, если в S существует элемент d такой, что $A \leqslant \{d\} \leqslant B$. Пару A_1, B_1 будем называть собственной подпарой пары A, B, если $A_1 \subseteq A$, $B_1 \subseteq B$ и либо $A_1 \neq A$, либо $B_1 \neq B$.

Теорема 3. *Пусть S — частично упорядоченное множество с одним минимальным и одним максимальным элементами, и пусть в S есть пара неотделимых подмножеств (A, B) такая, что любая ее собственная подпара отделима. Пусть $|A| + |B| = r$. Тогда $M_S^r \neq M_S^{r+1}$.*

Покажем, как теорема 1 выводится из теорем 3 и 2.

Доказательство теоремы 1. Рассмотрим частично упорядоченное множество с $2r - 2$ элементами $\{\min, \max, a_1, a_2, b_1, b_2, \ldots, b_{r-2}, c_1, c_2, \ldots, c_{r-2}\}$, где $r \geqslant 4$,

$\min$ — единственный минимальный элемент, $\max$ — единственный максимальный элемент, $a_i < c_j$ и $a_i < b_j$ при всех i, j; $c_j < b_m$ при всех $m \neq j$, c_j и b_j — не сравнимы. Пусть $A = \{a_1, a_2\}$, $B = \{b_1, b_2, \ldots, b_{r-2}\}$. Тогда пара множеств (A, B) удовлетворяет условиям теоремы 3. Полученное частично упорядоченное множество обозначим F_r. Рассмотрим частично упорядоченные множества F_4, F_5, ..., F_{t+2} с различными элементами, отождествим все их минимальные, а также все максимальные элементы. Полученное частично упорядоченное множество обозначим G_t. Тогда в G_t для любого r, где $4 \leqslant r \leqslant t+2$, есть пара подмножеств, удовлетворяющая условиям теоремы 3 для этого r. Пусть k — число элементов в G_t и пусть все элементы в G_t занумерованы от 0 до $k-1$. Тогда можно считать G_t частичным порядком на E_k. Из теоремы 3 получаем:

$$M^{\infty}_{G_t} \subseteq M^{t+3}_{G_t} \subset M^{t+2}_{G_t} \subset \ldots \subset M^{4}_{G_t} \subseteq M^{2}_{G_t},$$

причем все классы замкнуты согласно теореме 2. **Теорема 1 доказана.**

Вопрос о существовании бесконечной цепочки замкнутых классов между M^{∞}_S и M^2_S остается открытым.

Работа выполнена при поддержке РФФИ (проект № 17-01-00782-а).

Список литературы

[1] Алексеев В. Б., Вороненко А. А. О некоторых замкнутых классах в частичной двузначной логике // Дискретная математика. — 1994. — Т. 6, вып. 4. — С. 58-79.

[2] Haddad L., Lau D., Rosenberg I. G. Intervals of partial clones containing maximal clones // J. Autom. Lang. Comb. — 2006. — V. 11, № 4. — P. 399–421.

[3] Haddad L., Lau D. Uncountable families of partial clones containing maximal clones // Beiträge zur Algebra und Geometrie. — 2007. — V. 48, № 1. — P. 257–280.

О надежности двойственных схем в P_k

Алехина Марина Анатольевна

Пензенский государственный технологический университет, e-mail: ama@sura.ru

Пусть $n \in N$, $k \geqslant 2$, $E_k = \{0, 1, ..., k-1\}$, а P_k — множество всех функций k-значной логики, т. е. функций $f(x_1, ..., x_n) : \{E_k\}^n \to E_k$. Рассмотрим реализацию функций из P_k схемами из ненадежных функциональных элементов в полном конечном базисе $B = \{e_1, ..., e_q\}$ ($q \geqslant 1$).

Пусть $f(x_1, ..., x_n)$ — произвольная функция. Обозначим $\tilde{x}^n = (x_1, ..., x_n)$. Считаем, что схема из ненадежных функциональных элементов реализует функцию $f(\tilde{x}^n)$, если она реализует ее при отсутствии неисправностей в схеме. Предполагается, что неисправности элементов произвольные и статистически независимые, т. е. все элементы схемы переходят в неисправные состояния

независимо друг от друга. Как и в [1], вводятся понятия вероятности ошибки и правильного значения на выходе схемы, надежности и ненадежности схемы.

Ранее при $k = 2$ [2] доказано, что ненадежность схемы S, реализующей булеву функцию $f(\tilde{x}^n)$, равна ненадежности двойственной схемы S^*, построенной из элементов базиса $B^* = \{e_1^*, \ldots, e_q^*\}$ и реализующей функцию $f^*(x_1, ..., x_n) = \bar{f}(\bar{x}_1, ..., \bar{x}_n)$, двойственную функции $f(\tilde{x}^n)$. (Здесь e_j^* — булева функция, двойственная булевой функции e_j.)

Это свойство дает возможность переносить результаты о ненадежности схемы S, реализующей булеву функцию f, в базисе B при заданных неисправностях элементов, в другой, двойственный базис B^* для схемы S^*, реализующей двойственную функцию f^* при определенных неисправностях [2]. Например (см. [3, 4]), результаты о ненадежности, доказанные для схемы, реализующей булеву функцию f в базисе B, при однотипных константных неисправностях типа 0 на выходах элементов справедливы для двойственной схемы, реализующей функцию f^* в базисе B^*, при однотипных константных неисправностях типа 1 на выходах элементов; а результаты о ненадежности, доказанные для схемы, реализующей булеву функцию f в базисе B, при инверсных неисправностях элементов (см. [5–7]) справедливы для двойственной схемы, реализующей функцию f^* в базисе B^*, при тех же неисправностях элементов.

Возникают вопросы: «Имеет ли место подобное свойство в P_k ($k \geqslant 3$)?», «Если “да”, то для каких базисов, функций и неисправностей? А главное, относительно какой перестановки рассматривать двойственность?»

Ответы на эти вопросы для функций k-значной логики получены в этой работе. Но сначала введем понятие двойственной функции в P_k.

Функцию $f^*(x_1, ..., x_n) = N(f(Nx_1, ..., Nx_n))$ назовем функцией, *двойственной* функции $f(x_1, ..., x_n)$ относительно перестановки на множестве E_k, которую задает функция Nx, где $Nx = k - 1 - x$ — отрицание Лукашевича.

Замечание 1. *В P_k понятие двойственной функции вводится относительно перестановки на множестве E_k (например, см. [8, с. 41]). Но поскольку в этой статье мы рассматриваем только одну перестановку, которую задает функция Nx, далее будем говорить о двойственности не упоминая эту перестановку.*

Ненадежный функциональный элемент E^* с приписанной ему функцией $e^*(x_1, ..., x_m)$ $(m \geqslant 1)$ назовем *двойственным* элементу E с функцией $e(x_1, ..., x_m)$, если функция $e^*(x_1, ..., x_m)$ двойственна функции $e(x_1, ..., x_m)$ и для любого $i \in E_k$ и любого входного набора $\tilde{a}^m$ элемента E верно равенство $P_i(E, \tilde{a}^m) = P_{Ni}(E^*, N(\tilde{a}^m))$.

Две схемы S и S^* назовем *двойственными*, если одна получается из другой заменой всех элементов на двойственные им элементы соответственно. Нетрудно проверить, что если схема S реализует функцию $f(x_1, ..., x_n)$, то схема S^*

реализует двойственную функцию ей функцию $f^*(x_1, ..., x_n) = N(f(Nx_1, ..., Nx_n))$.

Теорема 1. *Пусть S — любая схема с n входами и одним выходом, $f(x_1, ..., x_n)$ – функция, которую реализует схема S. Тогда любого $i \in E_k$ и для любого входного набора $\tilde{a}^n$ схемы S верно равенство:*

$$P_i(S, \tilde{a}^n) = P_{Ni}(S^*, N(\tilde{a}^n)),$$

где S^* — схема, двойственная схеме S, $N(\tilde{a}^n) = (Na_1, ..., Na_n)$.

Доказательство теоремы 1 нетрудно провести методом математической индукции по числу элементов в схеме S (как и в [2]).

Следствие 1. *Ненадежности двойственных схем S и S^* равны, т. е. верно равенство $P(S) = P(S^*)$.*

Таким образом, ненадежности двойственных (относительно перестановки, которую задает функция Nx) схем равны для функций k-значной логики. В частности, например, утверждение о ненадежности схемы, реализующей k-значную функцию f, в базисе B

– при однотипных константных неисправностях типа 0 на выходах элементов справедливо для ненадежности двойственной схемы, реализующей функцию f^*, в базисе B^* при однотипных константных неисправностях типа $k-1$ на выходах элементов;

– при инверсных неисправностях на выходах элементов [9] справедливо для ненадежности двойственной схемы, реализующей функцию f^*, в базисе B^* при инверсных неисправностях на выходах элементов.

Работа выполнена при поддержке РФФИ (проект № 17-01-00451-а).

Список литературы

[1] Алехина М. А. Синтез схем из ненадежных элементов в P_k // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. — 2015. — № 3. — С. 3–10.

[2] Алехина М. А. Синтез, надежность и сложность схем из ненадежных функциональных элементов : дис. . . . докт. физ.-мат. наук : 01.01.09 : защищена 01.10.2004 : утв. 24.11.2004 / Алехина Марина Анатольевна. — Пенза, 2004. — 169 с. — Библиогр. : с. 167–169.

[3] Алехина М. А. Синтез и сложность надежных схем из ненадежных элементов // Математические вопросы кибернетики. — 2002. — № 11. — С. 193–218.

[4] Алехина М.А. О надежности схем в произвольном полном конечном базисе при однотипных константных неисправностях на выходах элементов // Дискретная математика. — 2012. — Т. 24, № 3. — С. 17–24.

[5] Алехина М. А. О надежности и сложности схем в базисе $\{x|y\}$ при инверсных неисправностях элементов // Дискретный анализ и исследование операций. Серия 1. — 2005. — Т. 12, № 2. — С. 3–11.

[6] Алехина М. А., Шилов А. В. Верхние оценки ненадежности схем в некоторых базисах при инверсных неисправностях на выходах элементов // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. — 2006. — № 5. — С. 4–12.

[7] Алехина М. А., Чугунова В. В. Об асимптотически наилучших по надежности схемах в базисе $\{\&, \vee, -\}$ при инверсных неисправностях на входах элементов // Дискретный анализ и исследование операций. Серия 1. — 2006. — Т. 13, № 4. — С. 3–17.

[8] Марченков С. С. Функциональные системы. — М. : МАКС Пресс, 2012. — 47 с.

[9] Алехина М. А., Барсукова О. Ю. Оценки ненадежности схем в базисе Россера–Туркетта // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. — 2014. — № 1. — С. 5–19.

Расширение функции гильотинного размещения

Андрианова Анастасия Александровна, Мухтарова Татьяна Маратовна, Фазылов Валерий Рауфович

Казанский (Приволжский) федеральный университет, e-mail: Anastasiya.Andrianova@kpfu.ru, Tatyana.Moukhtarova@kpfu.ru, vfazylov@gmail.com

Для задачи прямоугольной ортогональной упаковки известен аппарат функций гильотинного размещения (ФГР) [1], который дает точное решение задачи определения всевозможных минимальных размеров листов, достаточных для размещения заданного набора прямоугольников D, который включает m типов прямоугольников (деталей) в количестве k_i при $i = 1 \dots m$ ($a_i \times b_i$ — размеры детали i-го типа).

Определение 1 [1]. *Функцией гильотинного размещения для набора деталей D называется функция $f(\cdot; D) : R \to R$, значение которой в точке $x \in R$ равно минимальной длине листа шириной x, достаточной для гильотинного размещения набора D. Положим, что $f(+\infty; D) = \max\limits_{1 \leqslant i \leqslant m} b_i$ и, если $x < \max\limits_{1 \leqslant i \leqslant m} b_i$, то $f(x; D) = +\infty$.*

ФГР является ступенчатой полунепрерывной справа монотонной невозрастающей функцией с конечным числом ступеней и может быть представлена в табличной форме:

$$\begin{pmatrix} x_1 & x_2 & \dots & x_n \\ y_1 & y_2 & \dots & y_n \end{pmatrix},$$

где n — количество ступеней ФГР.

Если набор D состоит из одной детали, то ФГР строится в явном виде ([1]). Если же в наборе D две и более деталей, $f(\cdot; D)$ вычисляется по рекуррентной

формуле:

$$f(x;D) = \min\{f_1(x;D), f_2(x;D)\}, \forall x \in R,$$

где $f_1(\cdot;D)$ — ФГР при условии, что первый разрез будет сделан поперек листа, а $f_2(\cdot;D)$— при условии, что первый разрез будет сделан вдоль листа. Функция $f_2(\cdot;D)$ является квазиобратной к $f_1(\cdot;D)$ (см. [2]) и ее можно легко получить, зная $f_1(\cdot;D)$. Функция $f_1(\cdot;D)$ вычисляется по формуле:

$$f_1(x;D) = \min_{\{D_1,D_2\}\in R_D} \{f(x;D_1) + f(x;D_2)\}, \forall x \in R, \qquad (1)$$

где R_D — множество всевозможных разбиений набора D на два поднабора. Видно, что основная сложность алгоритма вычисления ФГР заключается в реализации вычисления суммы и минимума двух ФГР.

Для решения задачи раскроя на практике требуется знать не только, можно ли разместить набор деталей на листе, но и иметь информацию о раскрое листа и размещении на нем деталей. *Картой размещения набора D на листе* будем называть набор троек (x_{ij}, y_{ij}, z_{ij}), $j = 1, \ldots, k_i$, $i = 1, \ldots, m$, где (x_{ij}, y_{ij}) — координаты левого нижнего угла j-ой детали i-го типа на листе, z_{ij} — ее ориентация:

$$z_{ij} = \begin{cases} 0, \text{ если деталь расположена вдоль листа,} \\ 1, \text{ если деталь расположена поперек листа.} \end{cases}$$

Картой гильотинного раскроя листа для получения набора D будем называть последовательность гильотинных (от края до края) разрезов листа и его кусков, реализация которых дает все прямоугольники набора D. *Картой гильотинного размещения набора деталей D на листе* называется карта размещения, соответствующая карте гильотинного раскроя.

Для построения карт гильотинного раскроя листа и гильотинного размещения деталей на листе в ФГР не хватает информации. Поэтому мы вводим понятие расширения функции гильотинного размещения (РФГР), которое представляет собой ФГР, дополненную данными о первом разрезе листа и о соответствующем разбиении набора деталей на два поднабора, размещаемые на левом и правом (или нижнем и верхнем) кусках листа.

Определение 2. *Расширением функции гильотинного размещения для ФГР $f(\cdot;D)$ является таблица*

$$\begin{pmatrix} x_1 & x_2 & \ldots & x_n \\ y_1 & y_2 & \ldots & y_n \\ r_1 & r_2 & \ldots & r_n \\ N_1 & N_2 & \ldots & N_n \end{pmatrix},$$

где $x_k \times y_k$, $k = 1, \ldots, n$ — размеры минимальных листов, достаточных для гильотинного размещения деталей набора D, n — их количество, r_k — параметр первого разреза $k-$го листа, N_k — номер первого поднабора, соответствующего оптимальному разбиению набора D.

Параметры первого разреза листа и номер левого/нижнего поднабора определяются следующим образом:

$$r_k = \begin{cases} 0, & \text{если } D \text{ состоит из одной детали,} \\ r, & \text{если лист разрезается на куски } x_k \times r \text{ и } x_k \times (y_k - r), \\ -r, & \text{если лист разрезается на куски } r \times y_k \text{ и } (x_k - r) \times y_k, \end{cases}$$

$$N_k = \begin{cases} \text{номер типа детали, если набор состоит из одной детали,} \\ Num(D_1; D), \text{ если набор включает несколько деталей,} \end{cases}$$

где D_1 — поднабор, размещаемый на левом/нижнем куске листа, его номер вычисляется по формуле $Num(D_1; D) = \sum\limits_{i=1}^{m} l_i \prod\limits_{j=i+1}^{m} (k_j+1)$, $(l_1, l_2, \ldots, l_m)$ — состав поднабора D_1, $(k_1, k_2, \ldots, k_m)$ — состав исходного набора D. Состав дополняющего поднабора $D_2 = D \setminus D_1$ определяется однозначно и он размещается на втором (правом/верхнем) куске листа.

Заметим, что для построения карт гильотинного раскроя листа и гильотинного размещения деталей набора D на заданном листе требуется наличие РФГР для D и всех его поднаборов. Из формулы (1) видно, что в алгоритм вычисления ФГР нетрудно включить определение значений параметров r_k и N_k.

В заключение заметим, что карта гильотинного размещения деталей набора D может служить как эвристическим решением задачи негильотинного размещения набора на листе, так и начальным рекордом при решении задачи негильотинного размещения методом ветвей и границ (см. [3, 4]).

Список литературы

[1] Лернер Э. Ю., Фазылов В. Р. Функция гильотинного размещения // Исслед. по приклад. матем. — Вып. 21. — Казань: Унипресс, 1999. — С. 187–196.

[2] Лернер Э. Ю., Фазылов В. Р. Квазиобратные функции и их свойства // Исслед. по приклад. матем. — Казань: Казан. матем. общество, 1997. — Вып. 22. — С. 63–74.

[3] Андрианова А. А., Мухтарова Т. М., Фазылов В. Р. Модели негильотинного размещения набора прямоугольных деталей на листе и полуполосе// Учен. зап. Казан. ун-та. Сер. Физ.-матем. науки. — 2013.— Т. 155, кн. 2. — С. 5–18.

[4] Андрианова А. А., Мухтарова Т. М., Фазылов В. Р. Модель компактного размещения набора прямоугольников на листе// Тез. докл. XVI Байкальской междунар. шк.-сем. «Методы оптимизации и их приложения» (о. Ольхон, Байкал, 30 июня – 6 июля 2014 г.). — Иркутск: ИСЭМ СО РАН, 2014. — С. 33.

О РЕШЕНИИ ОДНОЙ ИГРЫ $2 \times n$ И $m \times 2$ С ИСПОЛЬЗОВАНИЕМ СИСТЕМ КОМПЬЮТЕРНОЙ МАТЕМАТИКИ

Анищенкова Надежда Геннадьевна[1], **Болотин Иван Борисович**[2]

[1] Смоленский государственный университет, e-mail: nadezhddaadhzedan@gmail.com

[2] Смоленский государственный университет, e-mail: IBBolotin@smolgu.ru

1. Постановка задачи

Рассмотрим следующую игру. *В игру играют двое игроков. Первый игрок называет любое натуральное число от* 1 *до* m *(*$m \geqslant 2$*), второй — любое натуральное число от* 1 *до* n *(*$n \geqslant 2$*). Если сумма названных чисел четна, то первый игрок получает количество очков, равное сумме названных чисел. В противном случае второй игрок получает количество очков, равное сумме названных чисел.*

В данной игре стратегическое множество первого игрока имеет вид $S_1 = \{1, 2, ..., m\}$, т. е. он имеет m чистых стратегий — назвать одно из натуральных чисел от 1 до m, при этом стратегическое множество второго игрока имеет вид $S_2 = \{1, 2, ..., n\}$ и он имеет n чистых стратегий — назвать одно из натуральных чисел от 1 до n.

Ради краткости будем обозначать сформулированную игру $\Pi_{m,n}$.

Платежная матрица рассматриваемой игры имеет вид:

$$A = \begin{pmatrix} 2 & -3 & ... & (-1)^{1+n}(1+n) \\ -3 & 4 & ... & (-1)^{2+n}(2+n) \\ ... & ... & ... & ... \\ (-1)^{m+1}(m+1) & (-1)^{m+2}(m+2) & ... & (-1)^{m+n}(m+n) \end{pmatrix}$$

Пусть $p = (p_1, p_2, ..., p_m)$ — смешанная стратегия первого игрока и v — цена игры, тогда математическая модель данной игры для первого игрока имеет вид (см., например, [1]):

$$\begin{cases} \sum\limits_{i=1}^{m} (-1)^{i+j}(i+j)p_i \geqslant v, \\ \sum\limits_{i=1}^{m} p_i = 1, \\ p_i \geqslant 0, \\ j = 1, 2, ..., n, \end{cases} \tag{1}$$

$$z = v \to max. \tag{2}$$

Таким образом, требуется определить значения переменных p_1, p_2, ..., p_m, v, удовлетворяющих системе ограничений (1) и обращающих в максимум функцию (2).

Аналогично, пусть $q = (q_1, q_2, ..., q_n)$ — смешанная стратегия второго игрока и v — цена игры, тогда математическая модель данной игры для второго игрока имеет вид (см., например, [1]):

$$\begin{cases} \sum\limits_{j=1}^{n}(-1)^{i+j}(i+j)q_j \leqslant v, \\ \sum\limits_{j=1}^{n} q_j = 1, \\ q_j \geqslant 0, \\ i = 1, 2, ..., m, \end{cases} \tag{3}$$

$$\tilde{z} = v \to min. \tag{4}$$

Таким образом, требуется определить значения переменных q_1, q_2, ..., q_n, v, удовлетворяющих системе ограничений (3) и обращающих в минимум функцию (4).

Замечание 1. *В случае, когда стратегические множества игроков совпадают, т. е. $m = n$, сформулированная задача рассмотрена в работе авторов* [*2].*

2. О решении задачи $\Pi_{2,n}$.

Рассмотрим задачу $\Pi_{m,n}$ в случае, когда $m = 2$. Используя компьютерное моделирование с применением системы компьютерной математика Wolfram Mathematica (см., например, [3]), аналогичное рассмотренному в [2], получим следующий результат.

Теорема 1. *Пусть $n = 2k$ или $n = 2k + 1$, $k \in N$. Тогда цена v игры $\Pi_{2,n}$ равна $-\dfrac{2k-1}{2(2k+4)}$. Активными чистыми стратегиями первого игрока будет выбор чисел 1 и 2 с вероятностями $\dfrac{2k+5}{2(2k+4)}$, $\dfrac{2k+3}{2(2k+4)}$ соответственно. Активными чистыми стратегиями второго игрока будет выбор чисел 1 и $2k$ с вероятностями $\dfrac{4k+3}{2(2k+4)}$ и $\dfrac{5}{2(2k+4)}$ соответственно.*

Замечание 2. Из теоремы 1 следует, что в традиционной постановке игры $\Pi_{2,2}$ (см., например, [4]) цена v игры равна $-\dfrac{1}{12}$, а вероятности использования активных чистых стратегий 1 и 2 каждым из игроков равны $\dfrac{7}{12}$ и $\dfrac{5}{12}$ соответственно.

3. О решении задачи $\Pi_{m,2}$.

Рассмотрим задачу $\Pi_{m,n}$ в случае, когда $n = 2$. Используя компьютерное моделирование [2], получим следующий результат.

Теорема 2. *Пусть* $m = 2k + 1$ *или* $m = 2k + 2$, $k \in N$. *Тогда цена* v *игры* $\Pi_{m,2}$ *равна* $\dfrac{2k-1}{2(2k+6)}$. *Активными чистыми стратегиями первого игрока будет выбор чисел* 2 *и* $2k+1$ *с вероятностями* $\dfrac{4k+5}{2(2k+6)}$, $\dfrac{7}{2(2k+6)}$ *соответственно. Активными чистыми стратегиями второго игрока будет выбор чисел* 1 *и* 2 *с вероятностями* $\dfrac{4k+5}{2(2k+6)}$ *и* $\dfrac{7}{2(2k+6)}$ *соответственно.*

Замечание 3. Из сформулированных теорем 1 и 2 следует, что цена рассматриваемых игр отлична от нуля, т. е. данные игры являются нечестными.

Список литературы

[1] Аллен Р. Математическая экономия. — М.: Издательство иностранной литературы, 1963. – 999 с.

[2] Анищенкова Н. Г., Болотин И. Б. Об одном подходе к решению антагонистических игр с большим количеством чистых стратегий игроков // International Journal of Open Information Technologies. — 2016.— Vol. 4, No. 9. — С. 8–12.

[3] Wolfram S. The Mathematica Book, Fifth Edition. — Cambridge: Cambridge University Press, 2003. — 1488 p.

[4] Коковин С. Г., Савватеев А. В., Тонис А. С. Большой задачник игр для РЭШ-НГУ-ВШЭ-ЕУСПб. — 2003. — 60 с.

О функциональной системе автоматов с операцией суперпозиции

Бабин Дмитрий Николаевич

МГУ имени М. В. Ломоносова, e-mail: d.n.babin@mail.ru

В задаче полноты автоматов с операцией сурерпозиции, найден пример замкнутого класса автоматов, нерасширяющегося до предполного класса.

Известно, что решение задач полноты и выразимости для систем автоматных функций наталкивается на существенные трудности [1]. Так, в работе [2] установлена континуальность множества предполных классов для систем автоматных функций, а в работе М. И. Кратко [3] установлена алгоритмическая неразрешимость задачи полноты относительно суперпозиции с обратной связью для конечных систем автоматных функций. Несмотря на то, что имеет место неполнота любой конечной системы автоматов относительно суперпозиции, система, состоящая из автоматов с двумя входами, является полной [4]. Более того, полна система, состоящая из одноместных автоматов и булевых функций. Для автоматов с операцией суперпозиции оставался открытым вопрос о расширяемости замкнутых классов до предполных. Автор показал, что этот вопрос решается отрицательно.

Общий обзор результатов в теории автоматов дан в работах [6–8].

Обозначим $E_2 = \{0, 1\}$, множество булевых функций вида $g : E_2^n \to E_2$ обозначим через $\mathbf{P_2}$. Обозначим через E_2^∞ множество всех сверхслов из нулей и единиц $a(1)a(2)...$, где $a(j) \in E_2$, $j = 1, 2, ...$

Функция вида

$$f : (E_2^\infty)^n \to (E_2^\infty)^m$$

называется *автоматной функцией (a-функцией)*, она задается рекуррентно соотношениями (1).

$$\begin{cases} q_1(1) = q0_1, \\ ... \\ q_s(1) = q0_s \\ q_1(t+1) = \phi_1(q_1(t), ..., q_s(t), a_1, ..., a_n), \\ ... \\ q_s(t+1) = \phi_s(q_1(t), ..., q_s(t), a_1, ..., a_n) \\ b_1(t) = \psi_1(q_1(t), ..., q_s(t), a_1, ..., a_n) \\ ... \\ b_m(t) = \psi_m(q_1(t), ..., q_s(t), a_1, ..., a_n) \end{cases} \qquad (1)$$

Вектор $q = (q_1, ..., q_s)$ задает *состояние* a-функции f, $q0$ её *начальное состояние*, буквы

$$a = (a_1, a_2..., a_n) \text{ и } b = (b_1, ..., b_m)$$

называются *входной и выходной буквами*, а сверхслова

$$a(1)a(2)\dots \text{ и } b(1)b(2)\dots \quad —$$

входными и выходными сверхсловами соответственно.

Вектор-функции ϕ и ψ называются функциями *переходов и выходной* функцией соответственно, а шестерка

$$(E_2^n, E_2^s, E_2^m, \phi, \psi, q_0) \quad —$$

автоматом, порождающим функцию f. Будем считать, что все состояния автомата достижимы из начального.

Класс всех a-функций обозначим через $\mathbf{P}$. В этом классе обычным образом введем операции суперпозиции автоматных функций. Для суперпозиции будем использовать модификации операций из [5].

Для автоматов, задающих автоматные функции, при операциях суперпозиции естественным образом выбирается множество состояний, достижимых из начального. Автоматы, имеющие одинаковые автоматные функции, называются эквивалентными.

Автоматную функцию G_0, задаваемую уравнениями

$$\begin{cases} q(1) = 0, \\ q(t+1) = a(t), \\ b(t) = q(t), \end{cases}$$

назовём автоматной функцией «задержки».

Автоматную функцию T_0, задаваемую уравнениями

$$\begin{cases} (q_1(1), q_2(1)) = (0,0) \\ (q_1(t+1), q_2(t+1)) = (a_1(t), a_2(t)), a_1(t)a_2(t) = 00 || a_1(t)a_2(t) = 11 \\ (q_1(t+1), q_2(t+1)) = (q_1(t), q_2(t)), a_1(t)a_2(t) = 01 \\ (b_1(t), b_2(t)) = (q_1(t), q_2(t)), \end{cases}$$

назовём автоматной функцией «триггер».

Константной автоматной функцией назовем автоматную функцию, выдающую одно и тоже периодическое выходное сверхслово на всех входных сверхсловах. Класс константных автоматных функций обозначим через $\mathbf{K}$.

Имеет место

Теорема. *Не существует предполного класса автоматных функций, содержащего замкнутый класс* $[\mathbf{K} \cup \mathbf{P_2} \cup \{G_0, T_0\}]$.

Список литературы

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

[2] Кудрявцев В. Б. О мощностях множеств предполных классов некоторых функциональных систем, связанных с автоматами // ДАН СССР. — 1963. — Т. 151, № 3. С. 493–496.

[3] Кратко М. И. Алгоритмическая неразрешимость проблемы распознавания полноты для конечных автоматов // ДАН СССР. — 1964. — Т. 155, № 1. — С. 35–37.

[4] Бабин Д. Н. О полноте двухместных автоматных функций относительно суперпозиции // Дискретная матетматика. — 1989. — Т. 1, вып. 4. — С. 423–431.

[5] Мальцев А. И. Итеративные алгебры и многообразие Поста // Алгебра и логика. — 1966. — Т. 5, № 2. — С. 5–24.

[6] Алёшин С. В. Об одном следствии теоремы Крона-Роудза // Дискретная математика. — 1999. — Т. 11, вып. 4. — С. 101–109.

[7] Алёшин С. В. Алгебраические системы автоматов. — М.: МАКС Пресс, 2016.

[8] Алёшин С. В. Кафедра математической теории интеллектуальных систем (МаТИС) // Интеллектуальные системы. — 2014. — Т. 18, вып. 2. — С. 5–30.

Программная структура модифицируемой системы искусственного интеллекта автономного робота

Бабич Андрей Михайлович

АО «НПП Рубин» (Пенза), e-mail: FieryEye@yandex.ru

Несмотря на то, что современные интеллектуальные робототехнические системы зачастую служат сходным целям, с программной точки зрения они, как правило, не похожи друг на друга. Элементы программного обеспечения (ПО), реализованные для робота одного производителя, невозможно перенести напрямую на робота другого производителя без передачи исходного кода. При этом существующие системы искусственного интеллекта (ИИ), такие как ROS, Microsoft Robotics Developer Studio, URBI, и др. рассчитаны на использование разработчиками ПО и сложны для понимания пользователями, не имеющими знаний в области программирования и робототехники [1–3]. Это приводит к тому, что разработанные системы ИИ либо не предполагают внесения существенных изменений со стороны пользователя или стороннего разработчика, либо они ориентированы на использование упрощённых (например графических) языков программирования, ограничивающих возможности системы в целом.

В работе [4] был рассмотрен общий подход к разработке и особенности системы ИИ, позволяющей преодолеть данные проблемы. Такая система предполагает наличие простого в использовании пользовательского интерфейса, а также механизма расширения её возможностей за счёт установки дополнительных программных компонентов. Данные возможности достигаются путём широкого использования динамически подключаемых библиотек или общих объектов, обладающих стандартным интерфейсом (для удобства обозначим их как «программные модули»), а также использованием связки «Режим-Задание-Задача» для поэтапной конкретизации абстрактных команд пользователя.

Следующий шаг в реализации данной системы ИИ — определение структуры её программных модулей, их взаимодействие и общий алгоритм работы. На рисунке 1 представлена общая схема обмена информации программных модулей. Система состоит из следующих модулей: «Монитор датчиков», «Картограф», «База данных модели мира», «Управление актуаторами», «Сторожевой модуль», «Решатель», «Корректор», «Контроллер», а также «Построение решения». Каждый из этих программных модулей выполняется в своём собственном потоке. На рисунке также обозначены непрограммные элементы: «Очередь задач», «Набор низкоуровневых команд», «Журнал» и технические средства робота — «Датчики» и «Актуаторы, индикация и сигнализация (АИС)».

«Монитор датчиков» производит опрос датчиков и обработку полученных данных. Данный модуль реализует аппаратную абстракцию средств ввода информации для остальной системы. Полученная информация передаётся модулю «Картограф», который осуществляет общий анализ данных и заносит результат в базу данных (БД), представляющую модель мира, с которой в дальнейшем работает система ИИ.

Сторожевой модуль осуществляет постоянный поиск признака команды в модели мира и оповещение модуля «Построение решения» в случае его обнаружения. Сам признак команды определяется пользователем. «Управление актуаторами» производит преобразование поступающих к нему низкоуровневых команд в электрические сигналы. Модуль реализует аппаратную абстракцию для АИС.

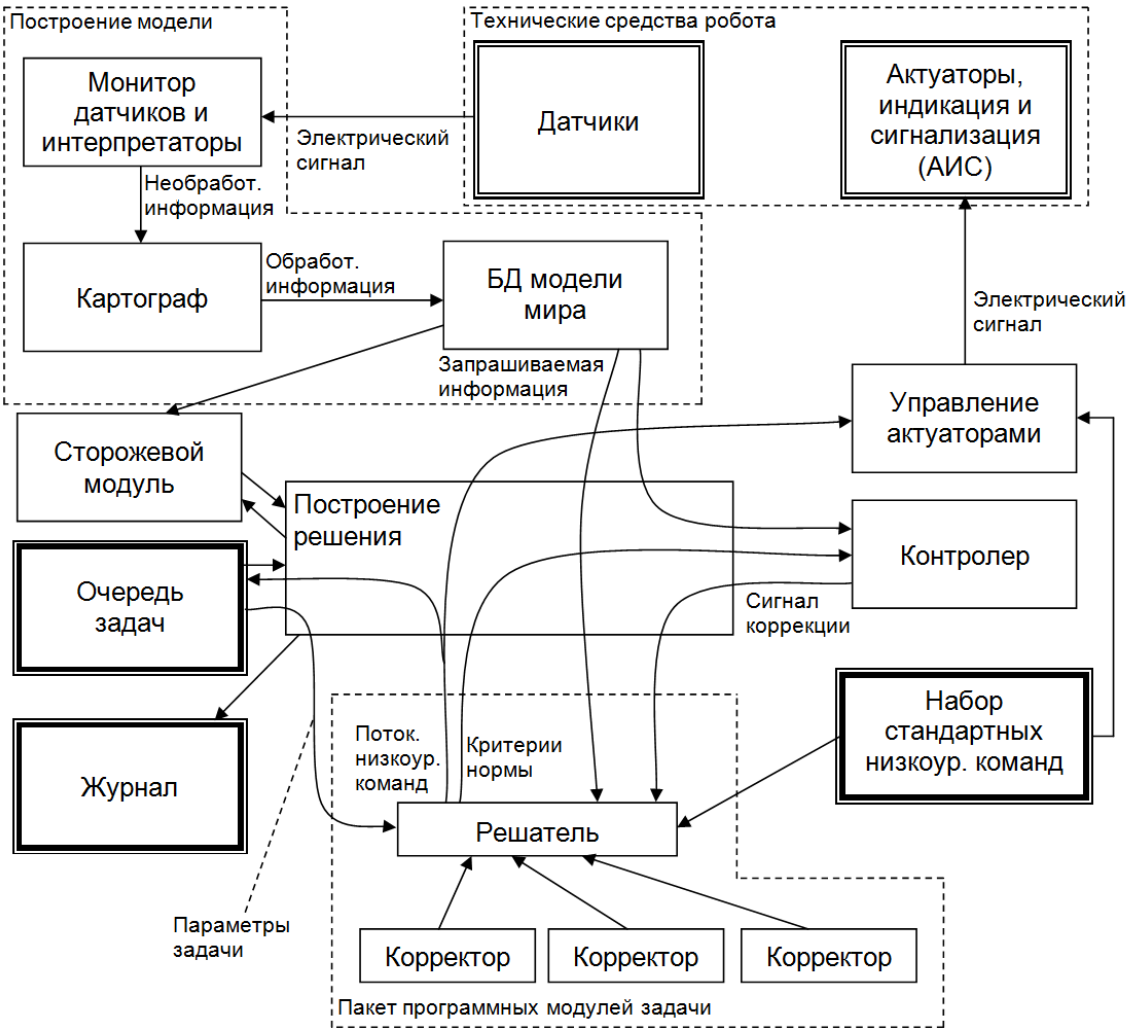


Рис. 1. Схема обмена информацией программных модулей системы.

«Решатель» предназначен для решения минимальной задачи, которую может поставить пользователь. На его вход поступают параметры этой задачи из очереди и информация о модели мира. На выходе «Решатель» представляет

последовательность низкоуровневых команд, необходимых для решения, а также набор критериев нормы, которые определяют допустимую величину различия между ожидаемой и фактической моделями мира в ходе выполнения этих команд. В зависимости от задачи команды могут передаваться либо модулю «Управление актуаторами», если необходимо воздействие на внешнюю среду, либо модулю «Очередь задач», если необходимо изменение в поведении робота. Критерии нормы передаются модулю «Контроллер».

«Контроллер» производит проверку корректности выполнения текущей задачи на основе информации, полученной из БД и набора критериев нормы, переданных модулем «Решатель». В случае, если состояние внешней среды некорректно, «Контроллер» посылает соответствующий сигнал модулю «Решатель». После этого «Решатель» самостоятельно, либо при помощи модуля «Корректор» формирует и передаёт модулю «Управление актуаторами» низкоуровневые команды для коррекции действий в случае получения соответствующего сигнала от модуля «Контроллер».

«Построение решения» является главным потоком, который осуществляет связь остальных потоков между собой, а также формирует очередь задач через анализ связки «Режим-Задание-Задача», сформированной пользователем.

Список литературы

[1] The Robot Operating System (ROS) [Электронный ресурс]. — адрес доступа: http://www.ros.org/ (дата обращения 1.03.2017).

[2] Microsoft Robotics Developer Studio [Электронный ресурс]. — адрес доступа: http://www.microsoft.com/robotics/ (дата обращения 1.03.2017).

[3] GOSTAI [Электронный ресурс]. — адрес доступа: http://www.gostai.com/ (дата обращения 1.03.2017).

[4] Бабич А. М., Акимов М. В. Особенности реализации модифицируемой системы искусственного интеллекта автономного робота // Вопросы радиоэлектроники. Серия СОИУ. — 2016. — Вып. 1. — С. 11–17.

Условия отсутствия адекватных решений некоторых задач интеллектуальной поддержки управления в процессе функционирования многоагентных систем

Бабич Михаил Юрьевич

АО «НПП Рубин» (Пенза), e-mail: babichmj@mail.ru

Рассмотрим многоагентные системы S. Агентами системы S являются лица (или группа лиц), управляющие техническими устройствами. Агенты системы S подчинены подсистеме управления (ПУ). Вычислительные средства ПУ,

автоматизируя обработку поступающих входных данных, информируют должностных лиц (ДЛ) о состоянии объектов управления, выдают рекомендации по их управлению, но не оказывают непосредственного управляющего воздействия на объекты управления. Управляющие воздействия ПУ осуществляются через команды ДЛ. Наблюдается мягкое управление людскими и техническими ресурсами в том смысле, что при четко поставленных целях жестко не регламентируются их пути достижения. Входную информацию каждая ПУ воспринимает через средства наблюдения (СН). В их число входят сейсмические датчики, устройства визуального наблюдения, средства навигации, ДЛ и др. Возможно наличие противоречивой, неточной, нечеткой информации, поступающей от различных и независимых СН.

К классу подобных многоагентных систем можно отнести достаточно большое число систем: системы управления транспортом, системы МЧС, системы управления силовыми операциями и т. д. Вследствие растущей сложности отслеживаемых процессов необходима организация решений задач интеллектуальной поддержки управления. Однако эти задачи не находят широкого применения в ПУ рассматриваемых систем. Причины этого были приведены в [1]. Однако, зачастую, складывается впечатление, что при дальнейшем развитии научно-технических разработок, усовершенствовании алгоритмов и технических характеристик средств вычислительной техники решения задач интеллектуальной поддержки управления ДЛ в ПУ найдут свое место в специфических процессах функционирования рассматриваемых систем.

Проанализируем более тонкие ограничения на их использование.

1. Под реальной информацией будем понимать информацию о состоянии системы — $S(t)$, о состоянии внешней среды системы — $C(t)$, а также о состоянии множества агентов — $A(t)$ в некоторый момент времени t. Реальная информация является объективной и не зависит от таких факторов, как неточность передачи данных, ошибки в программном обеспечении (ПО), неточность СН и т. д. Под виртуальной информацией будем понимать информацию, в поле которой работают ДЛ в ПУ. В виртуальной информации все приведенные факторы возможного ее искажения могут присутствовать. Виртуальная информация создается на основе полученных данных от СН. В ПУ ее обрабатывают средства вычислительной техники и ПО. ДЛ вынуждены принимать решения на основе виртуальной информации, хранящейся в БД системы и обрабатываемой ПО.

Виртуальная информация создает некоторую модель состояния системы $S(t)$, внешней среды $C(t)$ и множества агентов $A(t)$. Как для всякой модели, встает вопрос об ее адекватности. Ответ на вопрос об адекватности модели может существовать только в том случае, если реальная информация о состоянии системы, внешней среды и множеств агентов периодически повторяется, то

есть существует некоторый период времени $[\tau_1, \tau_2]$, для которого выполняется:

$$\begin{array}{l} \forall t' > \tau_2, \exists t \in [\tau_1, \tau_2]: \\ \quad ||S(t) - S(t')|| < \varepsilon, ||C(t) - C(t')|| < \varepsilon, ||A(t) - A(t')|| < \varepsilon. \end{array} \quad (1)$$

В (1) ε означает, что отличия между рассматриваемыми состояниями для решаемых задач являются несущественными.

Если соотношение (1) не выполняется, то есть повторяемость событий в рассматриваемых системах является незначительной, то вопрос об адекватности модели остается открытой. Вместе с тем, некоторые системы при определенных условиях достаточно часто могут входить в область странного аттрактора. Область странного аттрактора характеризуется незначительным горизонтом прогноза. Выбор траектории изменения состояний системы в бифуркационных точках зависит от непредсказуемых значений большого количества параметров, условий, что делает невозможным прогнозирование будущего состояния с помощью используемых моделей.

2. ДЛ в ПУ находится в виртуальном информационном пространстве. ДЛ может либо самостоятельно принять решение на основе полученной информации, либо получить обработанную информацию ПО, которое реализует какой-либо алгоритм решения задач интеллектуальной поддержки. В последнем случае мы будем находиться в рамках логики компьютера. Однако заметим, что в настоящее время многие исследователя в своих работах доказывают, что логика работы компьютера принципиально отличается от мышления человека. Наиболее характерны два круга задач, где человек имеет преимущество и решение которых занимает особое место среди когнитивных процессов: быстрое распознавание человеком не только отдельных объектов, но и сложных ситуаций; умение человека быстро рассуждать, получая логически верные следствия, притом, что он не всегда способен выстроить корректное доказательство.

Если решение задач интеллектуальной поддержки относится к описанным выше процессам, то ДЛ на основе своего опыта и интуиции значительно быстрее и точнее получает необходимое решение.

3. Существенным фактором, усложняющим моделирование и управление систем рассмотренного класса, является то, что как ДЛ, принимающие решения в ПУ, так и ДЛ, относящиеся к объектам управления, являются интеллектуальными агентами. Одним из свойств интеллектуальных агентов, влияющим в некоторых случаях на функционирование системы S, является следующее: любое живой организм всегда принадлежит не одной, а нескольким системам с полностью не совпадающими целями функционирования, возможность его перехода из одной системы в другую определяется его состоянием, состоянием внешней среды и состоянием систем, которым он принадлежит. Будем считать, что агент принадлежит системе S, если агент, взаимодействуя с другими элементами или подсистемами, входящими в систему S, должен функционировать в плане достижения цели системы S под управлением ПУ.

Таким образом, выполняется следующая аксиома:

$$\forall a \in A, \exists S_1, S_2: \\ (a \in S_1) \wedge (a \in S_2) \wedge (S_1 \neq S_2) \wedge (S_1 \not\subset S_2) \wedge (S_2 \not\subset S_1) \wedge (P_{S_1} \neq P_{S_2}). \quad (2)$$

В (2) P_{S_1} и P_{S_2} обозначают цели систем S_1 и S_2, то есть любой агент принадлежит не только одной системе, но и некоторой другой системе, причем цели систем не совпадают.

Не всегда, но при соблюдении определенных условий, возникает скрытое взаимодействие систем. Под скрытым взаимодействием понимается взаимодействие, которое осуществляется без управления ПУ. ПУ систем не знают о скрытом взаимодействии или не могут на него повлиять (в действительности, влияние возможно, но только опосредованное). При таком взаимовлиянии меняется алгоритм функционирования рассматриваемых систем.

Каким образом можно нивелировать наличие «нерешаемых задач»?

Для решения «нерешаемых задач» необходимо искать другие подходы. Одним из таких подходов является реализация метода прецедентов [2] в сочетании с методом сценирования и ролевых игр (МСИ). В отличие от обычного метода прецедентов, метод прецедентов совместно с МСИ не выдает готовое решение, а стимулирует мыслительную деятельность ДЛ. Примером имитационной модели, реализующей МСИ, является разработанная модель обнаружения нарушителей границы охраняемой области [3].

Список литературы

[1] Бабич М. Ю. Решение задач системы поддержки принятия решений в процессе управления распределенными, динамическими ресурсами // Информационно-измерительные и управляющие системы. — 2014. — Т. 12, № 4. — С. 12–18.

[2] Карпов Л. Е., Юдин В. Н. Адаптивное управление по прецедентам, основанное на классификации состояний управляемых объектов // Труды Института системного программирования РАН. — 2007. — Т. 13, № 2. — С. 37–57.

[3] Бабич М. Ю., Ковалева В. С., Нисенбаум Е. Э., Ползунов Н. В., Рыбин С. А. Имитационная модель обнаружения нарушителей границы охраняемой области // Вопросы радиоэлектроники. Серия ЭВТ. — 2008. — Вып. 5. — С. 112–120.

Приложения теории графов в геометрии 6-мерных почти эрмитовых многообразий

Банару Михаил Борисович[1], **Банару Галина Анатольевна**[2]

[1] Смоленский государственный университет, e-mail: mihail.banaru@yahoo.com
[2] Смоленский государственный университет, e-mail: mihail.banaru@yahoo.com

1. Эрмитова геометрия (или геометрия почти эрмитовых многообразий) имеет тесные и многоплановые связи со многими другими областями математики, в

частности, с различными разделами дискретной математики. Одно из наиболее содержательных и интересных приложений дискретной математики в теории почти эрмитовых многообразий — характеризация гиперповерхностей этих многообразий в терминах типовых чисел (характеризация Такаджи–Курихары). В этом направлении авторами опубликовано более 20 работ (см., например, [1–5]). Относительно недавно Карриазо, Фернандес и Родригес-Идальго установили интереснейшую связь между дифференциальной геометрией и теорией графов [6]. Они показали, что (в локальном смысле) всякое подмногообразие почти эрмитова многообразия допускает ассоциацию с некоторым графом. Конструкция такой ассоциации установлена Карриазо и Фернандесом ранее [7].

2. Напомним, что почти эрмитовой (almost Hermitian, AH-) структурой на четномерном многообразии M^{2n} называется пара $\{J,\ g = \langle\cdot,\ \cdot\rangle\}$, где J — почти комплексная структура, $g = \langle\cdot,\ \cdot\rangle$ — риманова метрика на этом многообразии [8]. При этом J и $g = \langle\cdot,\ \cdot\rangle$ должны быть согласованы условием

$$\langle JX,\ JY\rangle = \langle X,\ Y\rangle\,, \quad X, Y \in \aleph(M^{2n}),$$

где $\aleph(M^{2n})$ — модуль C^∞-гладких векторных полей на многообразии M^{2n}. Многообразие с фиксированной на нем почти эрмитовой структурой называется почти эрмитовым (AH-) многообразием.

3. Обратим внимание на то, что результаты Карриазо, Фернандеса и Родригеса-Идальго относятся, в основном, к подмногообразиям 4-мерных почти эрмитовых многообразий, и в гораздо меньшей степени — к подмногообразиям 6-мерных почти эрмитовых многообразий [6].

В работах [1–5] авторами изучались 6-мерные почти эрмитовы подмногообразия алгебры Кэли. Известно, что каждое из так называемых 3-векторных произведений в алгебре октав порождает на 6-мерном ориентируемом подмногообразии почти эрмитову структуру. В основном рассматривались эрмитовы 6-мерные подмногообразия алгебры Кэли, то есть многообразия с интегрируемой почти эрмитовой структурой, а кроме этого — приближенно келеровы многообразия и многообразия со структурами классов G_1 и G_2 (см., например, [9] и [10]) в общепринятой классификации Грея–Хервеллы [8].

Как оказалось, результаты Карриазо, Фернандеса и Родригеса-Идальго можно применить к изучению гиперповерхностей 6-мерных почти эрмитовых подмногообразий алгебры Кэли. Основной результат содержит

Теорема. *Граф Карриазо–Фернандеса ассоциируется (в локальном смысле) со всякой гиперповерхностью 6-мерного келерова, приближенно келерова, локально конформно келерова, специального эрмитова и эрмитова подмногообразий алгебры Кэли.*

Список литературы

[1] Банару М. Б., Банару Г. А. О спектре некоторых тензоров уплощающихся 6-мерных эрмитовых подмногообразий алгебры Кэли // Материалы VII

Международного семинара «Дискретная математика и ее приложения». — Ч. 2. М.: Изд-во МГУ, 2001. — С. 250–253.

[2] Banaru M. B., Banaru G. A. A note on six-dimensional planar Hermitian submanifolds of Cayley algebra // Buletinul Academiei de Ştiinte a Republicii Moldova. Matematica. — N 1(74). — 2014. — P. 23–32.

[3] Банару М. Б. Почти контактные метрические гиперповерхности с типовым числом 1 или 0 в приближенно келеровых многообразиях // Вестник Московского университета. — Сер. 1. Математика. Механика. — 2014. — № 3. — С. 60–62.

[4] Banaru M. B., Banaru G. A. A note on almost contact metric hypersurfaces of nearly Kählerian 6-sphere // Bulletin of the Transilvania University of Braşov. Series III. Mathematics, Informatics, Physics. — V. 8(57), N 2. — 2015. — P. 21–28.

[5] Banaru M. B., Banaru G. A. 1-cosymplectic hypersurfaces axiom and six-dimensional planar Hermitian submanifolds of the Octonian // SUT Journal of Mathematics. — V. 51, N. 1. — 2015. — P. 1–9.

[6] Carriazo A., Fernandez L. M., Rodriguez-Hidalgo A. Submanifolds weakly associated with graphs // Proc. Indian Acad. Sci. (Math. Sci.). — V. 119, N. 3. — 2009. — P. 297–318.

[7] Carriazo A., Fernandez L. M. Submanifolds associated with graphs // Proc. Amer. Math. Soc. — 2004. — V. 132(11). — P. 3327–3336.

[8] Gray A., Hervella L. M. The sixteen classes of almost Hermitian manifolds and their linear invariants // Ann. Mat. Pura Appl. — V. 123, N. 4. — 1980. — P. 35–58.

[9] Banaru M. B. A note on six-dimensional G1-submanifolds of the octave algebra // Taiwanese J. Math. — V. 6, N. 3. — 2002. — P. 383–388.

[10] Банару М. Б. О 6-мерных G2-подмногообразиях алгебры Кэли // Математические заметки. — Т. 74, № 3. — 2003. — С. 323-328.

О ДИАГОНАЛЬНЫХ РАНГАХ КОНЕЧНЫХ ПОЛУГРУПП

Барков Илья Викторович

НИУ МИЭТ, e-mail: barkovian@gmail.ru

Правым полигоном [1] над полугруппой S называется множество X, на котором действует полугруппа S, то есть определено отображение $X \times S \to X$, $(x, s) \mapsto xs$, такое, что выполняется тождество $(xs)s' = x(ss')$ при всех $x \in X$, $s, s' \in S$. Если S — полугруппа, то множество $S \times S$ будет являться правым полигоном над S относительно действия $(x, y)s = (xs, ys)$ при всех $x, y \in X = S \times S$, $s \in S$. Такой полигон называется *диагональным правым полигоном над* S.

Порождающее множество G полигона $(S \times S)_S$ называется *неприводимым*, если никакое его собственное подмножество $G' \subset G$ не является порождающим для этого полигона. Очевидно, любое конечное порождающее множество может быть уменьшено до неприводимого. Порождающее множество называется *минимальным*, если оно минимально по мощности среди всех конечных порождающих множеств.

Так как полигон над полугруппой является унарной алгеброй, то по теореме 1 из [2] мы получаем, что в любом (правом, левом, би-) полигоне неприводимое множество образующих является минимальным.

Односторонний полигон является алгебраической моделью автомата без выхода. Таким образом диагональный полигон можно считать параллельным соединением автоматов, у которых множество состояний совпадает с входным алфавитом, а функция перехода совпадает с умножением в данной полугруппе. Корректность определения функции перехода гарантируется ассоциативностью полугрупповой операции.

Правым диагональным рангом полугруппы S будем называть *мощность минимального порождающего множество полигона* $(S \times S)_S$. Обозначать ранг будем $\mathrm{rdr}S$.

Односторонний диагональный ранг полугруппы порядка n может принимать значения от n до n^2. Полугрупп, имеющих крайние значения диагональных рангов, немного. Они описываются в следующих теоремах.

Теорема 1 ([3], теорема 4). *Пусть S — полугруппа из n элементов. Равенство $\mathrm{rdr}\, S = n$ имеет место в том и только в том случае, если выполняется хотя бы одно из следующих условий:*

1. *S — группа;*
2. *S — двухэлементная полугруппа правых нулей;*
3. *S — полугруппа со следующей таблицей Кэли:*

$\cdot$	e	a	b
e	e	a	b
a	b	b	b
b	b	b	b

Теорема 2. *Пусть S — n-элементная полугруппа и $\mathrm{rdr}\, S = n^2$. Тогда S — полугруппа левых нулей.*

Аналитические выражения диагональных рангов известны так же для следующих полугрупп (далее n — порядок полугруппы):

— полугруппа с нулевым умножениеем: $n^2 - 1$;

— полугруппа правых нулей: $n^2 - n$;

— вполне (0)-простая полугруппа: описанием довольно громоздко, поэтому ограничимся ссылкой на [3], теоремы 2 и 3.

Из [4] так же известно, что для $n > 2$ всегда существует 3-нильпотентная полугруппа ранга $n^2 - 4$. Однако ни одно из перечисленных выше значений рангов не определяет полугруппу. Например, существует 4-элементная полугруппа с правым диагональным рангом равным 15 со следующей таблицей Кэли:

$\cdot$	e	a	b	c
e	e	e	e	e
a	e	a	a	a
b	e	b	b	b
c	e	c	c	c

Порождающее множество диагонального полигона определяется, вообще говоря, не единственным образом. Существование порождающего множества удобного для анализа вида гарантируется следующей теоремой. **Теорема 3.** *Пусть S — полугруппа. Тогда существует минимальное порождающее множество диагонального полигона $(S \times S)_S$, состоящее из элементов следующего вида:*

— (a, a)*;*

— (a, b) *такие, что найдётся $s \in S$ такой что $(a, b)s = (b, a)$;*

— (a, b) *и* (b, c)

Из данной теоремы следует, например, что диагональный ранг нильпотентной полугруппы имеет ту же чётность, что и её порядок.

Список литературы

[1] Kilp M., Knauer U., Mikhalev A. V. Monoids, acts and categories. — Berlin, New York: W. de Gruyter, 2000. — 546 p.

[2] Карташов В. К. Независимые системы порождающих и свойство Хопфа для унарных алгебр // Дискретная математика. — 2008. — Т. 20, № 4. — С. 79–84.

[3] Барков И. В., Шакиров Р. Р. Конечные полугруппы минимального ранга // Математический вестник педвузов и университетов Волго-Вятского региона. — 2014. — № 16. — С. 55–63.

[4] Барков И. В. Оценки диагональных рангов 3-нильпотентных полугрупп // Сборник научных трудов МИЭТ. Посвящается 70-летию профессора Алексея Сергеевича ППоспелова. — 2016. — С. 83–88.

Оценки ненадежности схем в базисе Россера–Туркетта при неисправностях типа 1 на выходах элементов

Барсукова Оксана Юрьевна[1], **Пичугина Полина Григорьевна**[2]

[1] Пензенский государственный университет, e-mail: kuzya_7@mail.ru

[2] Пензенский государственный университет, e-mail: polinapichugina@yandex.ru

Пусть $n \in \mathbf{N}$, $E_3 = \{0, 1, 2\}$, а P_3 — множество всех функций трехзначной логики, т. е. функций $f(x_1, ..., x_n) : (E_3)^n \to E_3$. Рассмотрим реализацию функций из множества P_3 схемами из ненадежных функциональных элементов в базисе Россера–Туркетта $B = \{0, 1, 2, J_0(x_1), J_1(x_1), J_2(x_1), \min\{x_1, x_2\}, \max\{x_1, x_2\}\}$. Обозначим $x_1 \& x_2 = \min\{x_1, x_2\}$, $x_1 \vee x_2 = \max\{x_1, x_2\}$, $\tilde{x}^n = (x_1, ..., x_n)$. Также, как в работах [1–5] вводятся определения вероятности ошибки на выходе схемы, ненадежность и надежность схемы.

Предполагается, что базисные элементы подвержены неисправностям типа 1 на выходах, причем переходят в неисправные состояния независимо друг от друга с вероятностью ε ($0 < \varepsilon < 1/2$). Эти неисправности характеризуются тем, что на единичных входных наборах функциональный элемент выдает правильное значение 1 с вероятностью 1, а на остальных наборах — значение 1 с вероятностью ε, а правильное значение с вероятностью $1 - \varepsilon$.

Обозначим базисный элемент с функцией $\&$ через $E_\&$, а базисный элемент с функцией $\vee$ — через $E_\vee$.

Например, функционирование базисного элемента $E_\&$ при неисправностях типа 1 можно описать таблицей 1.

Пусть f — произвольная функция из P_3, а S — любая схема, реализующая функцию f. По схеме S построим схему, которая реализует ту же функцию f, но, возможно (при некоторых условиях на $P(S)$), более надежно. Для этого возьмем два экземпляра схемы S и один элемент $E_\&$, соединим выходы схем S со входами элемента $E_\&$. Построенную схему назовем D. Затем возьмем два экземпляра схемы D и один элемент $E_\vee$. Соединим соединим выходы схем D со входами элемента $E_\vee$. Построенную схему назовем $\psi(S)$.

Табл. 1

x_1	x_2	$x_1 \& x_2$	$P_0(E_\&, \tilde{x}^2)$	$P_1(E_\&, \tilde{x}^2)$	$P_2(E_\&, \tilde{x}^2)$
0	0	0	$1-\varepsilon$	ε	0
0	1	0	$1-\varepsilon$	ε	0
0	2	0	$1-\varepsilon$	ε	0
1	0	0	$1-\varepsilon$	ε	0
1	1	1	0	1	0
1	2	1	0	1	0
2	0	0	$1-\varepsilon$	ε	0
2	1	1	0	1	0
2	2	2	0	ε	$1-\varepsilon$

В теореме 1 найдено рекуррентное соотношение для ненадежностей схем S и $\psi(S)$.

Теорема 1. *Пусть f — произвольная функция из P_3, S — любая схема, реализующая f, а $P(S)$ — ненадежность схемы S. Тогда схема $\psi(S)$ реализует функцию f с ненадежностью*

$$P(\psi(S)) \leqslant \max\{3\varepsilon + 2P^2(S), \varepsilon + (\varepsilon + 2P(S))^2\}.$$

Доказательство теоремы 1 такое же, как в случае инверсных неисправностей на выходах элементов [3].

С помощью теоремы 1 получим верхнюю оценку ненадежности схемы, реализующей произвольную функцию f, при неисправностях типа 1 на выходах элементов. Справедлива теорема 2.

Теорема 2. *Любую функцию $f \in P_3$ можно реализовать такой схемой S, что $P(S) \leqslant 3\varepsilon + 32\varepsilon^2$ при всех $\varepsilon \in (0, 1/300]$.*

Доказательство теоремы 2 проводится методом математической индукции по числу переменных функции f [3].

Обозначим через $K_1(n)$ множество таких функций трехзначной логики, зависящих от переменных $x_1, ..., x_n$ $(n \geqslant 3)$, что каждая из этих функций принимает все три значения $0, 1, 2$ и не представима ни в виде $x_k \vee h(\tilde{x}^n)$, ни в виде $x_k \& h(\tilde{x}^n)$ $(k \in \{1, 2, ..., n\},\ h(\tilde{x}^n)$ — произвольная функция трехзначной логики). Пусть $K_1 = \bigcup\limits_{n=3}^{\infty} K_1(n)$.

В работе [1] доказано, что $|K_1(n)| \geqslant 3^{3^n} - 2n3^{2 \cdot 3^{n-1}} - 3 \cdot 2^{3^n}$. Поэтому класс $K_1(n)$ содержит почти все функции из $P_3(n)$.

Справедлива теорема о нижней оценке ненадежности схем, реализующих функции из класса K_1.

Теорема 3. *Пусть функция $f \in K_1$. Тогда для любой схемы S, реализующей f, при $\varepsilon \in (0, 1/300]$ верно неравенство $P(S) \geqslant 3\varepsilon - 3\varepsilon^2 + \varepsilon^3$.*

Доказательство этой теоремы основано на следующей идее: в схеме S, реализующей функцию из K_1, выделяется подсхема из трех элементов и доказывается, что ненадежность всей схемы не меньше минимальной вероятности ошибки подсхемы на определенных наборах. [3].

Таким образом, в базисе Россера–Туркетта (в P_3) при неисправностях типа 1 на выходах элементов:

1) любую функцию трехзначной логики можно реализовать схемой, ненадежность которой асимптотически (при $\varepsilon \to 0$) не больше 3ε;

2) для почти любой функции такая схема является асимптотически оптимальной по надежности и функционирует с ненадежностью, асимптотически равной 3ε при $\varepsilon \to 0$.

Работа выполнена при поддержке РФФИ (проект № 17-01-00451-а).

Список литературы

[1] Алехина М. А., Барсукова О. Ю. Оценки ненадежности схем в базисе Россера–Туркетта // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. — 2014. — № 1 (29). — С. 5–19.

[2] Алехина М. А., Барсукова О. Ю. Ненадежность схем в базисе Россера–Туркетта // Прикладная дискретная математика. Приложение. № 7. — 2014. — С. 109–110.

[3] Барсукова О. Ю. Синтез надежных схем, реализующих функции двузначной и трехзначной логик : дис. . . . канд. физ.-мат. наук : 01.01.09 : защищена 06.06.2014 : утв. 24.01.2015 / Барсукова Оксана Юрьевна. — Пенза, 2014. — 87 с. — Библиогр. : с. 86–87.

[4] Алехина М. А., Барсукова О. Ю. О надежности схем, реализующих функции трехзначной логики // Дискретный анализ и исследование операций. — 2014. — Т. 21. — № 4 (118). — С. 12–24.

[5] Алехина М. А., Барсукова О. Ю. Нижняя оценка ненадежности схем в базисе, состоящем из функции Вебба // Прикладная дискретная математика. Приложение. № 8. — 2015. — С. 102–103.

Обращение экспоненциальных производящих функций многочленов Эйлера целого положительного порядка

Бондаренко Леонид Николаевич[1], **Шарапова Марина Леонидовна**[2]

[1] Московский университет им. С. Ю. Витте, e-mail: leobond5@mail.ru

[2] Московский государственный университет имени М. В. Ломоносова, e-mail: msharapova@list.ru

В [1] приведены основные свойства чисел Эйлера I и II порядка. Их обобщения на порядок $r \geqslant 1$ связаны с перестановками мультимножества $\{1^r, \ldots, n^r\}$,

введенными И. Гесселем и Р. Стенли в [2] (ГС-перестановками порядка r). ГС-перестановкой порядка r называется слово $\sigma = \sigma_1 \dots \sigma_{rn}$ над алфавитом $[n] = \{1, \dots, n\}$, буквы которого, стоящие между любыми двумя вхождениями символа $s \in [n]$, не меньше s.

Свойства множеств $\mathfrak{S}_n^{(r)}$ всех таких перестановок изучались в [2, 3], в частности, $|\mathfrak{S}_n^{(r)}| = (r(n-1)+1)!^{(r)} = 1 \cdot (r+1) \cdot \ldots \cdot (r(n-1)+1)$, где $(r(n-1)+1)!^{(r)}$ — обобщение символа факториала, введенное Ё. Ониши [4].

С помощью числа подъемов $\mathrm{rise}(\sigma) = \#\{i \in [rn] : \sigma_{i-1} < \sigma_i,\ \sigma_0 = 0\}$, $\sigma \in \mathfrak{S}_n^{(r)}$ можно определить числа Эйлера $A_{n,k}^{(r)} = \#\{\sigma \in \mathfrak{S}_n^{(r)} : \mathrm{rise}(\sigma) = k\}$ порядка r, а также доказать рекуррентное соотношение [3]

$$A_{0,k}^{(r)} = \delta_{0k},\ A_{n+1,k}^{(r)} = kA_{n,k}^{(r)} + (rn - k + 2)A_{n,k-1}^{(r)},\ k \in \mathbb{Z},\ n \geqslant 0, \qquad (1)$$

где символ Кронекера δ_{ij} равен 1 при $i = j$ и 0 при $i \neq j$. Применение (1) для многочленов Эйлера $A_n^{(r)}(t) = \sum_{k=1}^{n} A_{n,k}^{(r)} t^k$ порядка r дает формулу

$$A_0^{(r)}(t) = 1,\ A_{n+1}^{(r)}(t) = (rn+1)tA_n^{(r)}(t) + t(1-t)D_tA_n^{(r)}(t),\ n \geqslant 0,$$

где $D_t = d/dt$ — оператор дифференцирования, причем $A_n^{(r)}(1) = |\mathfrak{S}_n^{(r)}|$.

Число спусков $\mathrm{des}(\sigma) = \#\{i \in [rn] : \sigma_i > \sigma_{i+1},\ \sigma_{rn+1} = 0\}$, $\sigma \in \mathfrak{S}_n^{(r)}$ совпадает с числом подъемов реверсированной перестановки $\mathbf{r}\sigma$, т. е. имеем $A_{n,k}^{(r)} = \#\{\sigma \in \mathfrak{S}_n^{(r)} : \mathrm{des}(\sigma) = k\}$. Пусть $\mathrm{lev}_i(\sigma) = \#\{s \in [n] : l_s(\sigma) = i\}$ — число уровней ранга $i \in [r-1]$ при $r > 1$, а $l_s(\sigma) = 0$, если для подслова $s^{j_1}w_1 \dots s^{j_{m-1}}w_{m-1}s^{j_m},\ j_1 + \ldots + j_m = r$ (подслова $w_1, \dots, w_{m-1}$ могут быть пустыми) при $k \in [m-1]$ сумма $j_1 + \ldots + j_k = i$, иначе $l_s(\sigma) = i$. Можно показать, что $\mathrm{rise}(\sigma) + \mathrm{des}(\sigma) + \sum_{i=1}^{r-1} \mathrm{lev}_i(\sigma) = rn + 1$ и $\mathrm{lev}_i(\mathbf{r}\sigma) = \mathrm{lev}_{r-i}(\sigma)$.

Методом математической индукции по n с помощью соотношения (1) доказывается эйлеровость числа уровней ранга i.

Теорема 1. $A_{n,k}^{(r)} = \#\{\sigma \in \mathfrak{S}_n^{(r)} : \mathrm{lev}_i(\sigma) = k\},\ i = 1, \dots, r-1.$

Так как m-кратная мультиномиальная свертка

$$\langle P_n(t) \rangle^m = \sum_{\substack{k_1 + \ldots + k_m = n \\ k_i \geqslant 0}} \binom{n}{k_1 \dots k_m} P_{k_1}(t) \dots P_{k_m}(t)$$

последовательности $\{P_k(t)\}_{k=0}^n$ многочленов $P_k(t)$ степени k является коэффициентом при $u^n/n!$ в m-й степени экспоненциальной производящей функции $\sum_{n=0}^{\infty} P_n(t)\, u^n/n!$ [1], то дифференцированием экспоненциальной производящей функции $1 + \sum_{n=1}^{\infty} (r(n-1)+1)!^{(r)} u^n/n! = (1-ru)^{-1/r}$, где $(r(n-1)+1)!^{(r)} = A_n^{(r)}(1)$, находится формула

$$A_{n+1}^{(r)}(1) = \sum_{\substack{k_1 + \ldots + k_{r+1} = n \\ k_i \geqslant 0}} \binom{n}{k_1 \dots k_{r+1}} A_{k_1}^{(r)}(1) \dots A_{k_{r+1}}^{(r)}(1), \qquad (2)$$

служащая, как и теорема 1, базой вывода следующего утверждения.

Теорема 2. *Для многочленов Эйлера порядка r справедливо соотношение*

$$A_0^{(r)}(t) = 1,\ A_{n+1}^{(r)}(t) = (t-1)\langle A_n^{(r)}(t)\rangle^r + \langle A_n^{(r)}(t)\rangle^{r+1},\ n \geqslant 0. \tag{3}$$

Доказательство. В [5] дано комбинаторное доказательство q-аналога формулы (3) при $r = 1$. Соотношение (2) показывает, что можно аналогично [5] положить $M = \{1^{k_1}, \ldots, (r+1)^{k_{r+1}}\}$ при $r \geqslant 1$ и рассмотреть декартово произведение $\mathfrak{S}(M) \times \mathfrak{S}_{k_1}^{(r)} \ldots \times \mathfrak{S}_{k_r}^{(r)}$. Затем, применяя алгоритм генерации ГС-перестановок порядка r [3], нетрудно получить при $r \geqslant 1$ соответствующие равенства для числа спусков. При этом рассмотрение числа инверсий не производится, т. е. в [5] полагается $q = 1$. Тогда с учетом (2) доказательство в [5] при $q = 1$ переносится на любое $r \geqslant 1$ и приводит к соотношению (3). Отметим, что при $r > 1$ комбинаторное доказательство для q-аналога формулы (3) не проходит. **Теорема 2 доказана.**

Теорема 3. *Обратные функции $F^{(r)}(t,\, v)^{-1}$ и $G^{(r)}(t,\, w)^{-1}$ для экспоненциальных производящих функций $v = F^{(r)}(t,\, u) = \sum_{n=1}^{\infty} A_n^{(r)}(t)u^n/n!$ и, соответственно, $w = v/t = G^{(r)}(t,\, u)$ записываются в следующем виде:*

$$u = F^{(r)}(t,\, v)^{-1} = \int_0^v \frac{dz}{(z+t)(z+1)^r}, \tag{4}$$

$$u = G^{(r)}(t,\, w)^{-1} = \frac{1}{(r-1)!} D_t^{r-1}\left(\frac{t^{r-1}}{1-t}\ln\left(\frac{w+1}{tw+1}\right)\right), \tag{5}$$

а еще одно представление многочленов Эйлера порядка r дает формула

$$A_n^{(r)}(t) = tD_w^{n-1}\left(\frac{w}{G^{(r)}(t,\, w)^{-1}}\right)^n\Bigg|_{w=0}. \tag{6}$$

Доказательство. Из равенства $A_1^{(r)}(t) = t$ следует, что коэффициент при u в ряде $w = G^{(r)}(t,\, u)$ равен 1, т. е. существуют обратные функции $F^{(r)}(t,\, v)^{-1}$ и $G^{(r)}(t,\, w)^{-1}$. Так как $1 + v = \sum_{n=0}^{\infty} A_n^{(r)}(t)u^n/n!$, то соотношению (3) отвечает дифференциальное уравнение

$$\frac{dv}{du} = (v+t)(v+1)^r,$$

которое с учетом равенства $F^{(r)}(t,\, 0) = 0$ и приводит к интегралу (4).

Формула (5) находится из выражения

$$u = G^{(r)}(t,\, w)^{-1} = \int_0^w \frac{dz}{(z+1)(tz+1)^r},$$

полученного из (4) заменой $v = tw$, с помощью простых тождеств

$$G^{(1)}(t,\, w)^{-1} = \frac{1}{1-t}\ln\left(\frac{w+1}{tw+1}\right),\ \ \frac{1}{(r-1)!}D_t^{r-1}\left(\frac{t^{r-1}}{tw+1}\right) = \frac{1}{(tw+1)^r},$$

а теорема Бюрмана–Лагранжа дает равенство (6). **Теорема 3 доказана.**

Найденные несложные выражения (4) и (5) описывают обратные функции для экспоненциальных производящих функций многочленов Эйлера порядка r, простое явное представление которых при $r > 1$ отсутствует.

Список литературы

[1] Грэхем Р., Кнут Д., Паташник О. Конкретная математика. Основание информатики. — М. : Мир, 1998. — 704 с.

[2] Gessel I., Stanly R. P. Stirling polynomials. // Journal of combinatorial theory. Series A. — 1978. — Vol. 24.— No. 1.— P. 24–33.

[3] Бондаренко Л. Н., Шарапова М. Л. Параметрические комбинаторные задачи и методы их исследования. // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. — 2010. — № 4 (16). — С. 50–63.

[4] Ониши Ё. Обобщенные числа Бернулли–Гурвица и универсальные числа Бернулли. // Успехи математических наук. — 2011. — Т. 66. — Вып. 5. — С. 47–108.

[5] Chow Chak–On. A recurrence relation for the "inv"analogue of q-Eulerian polynomials. // The electronic journal of combinatorics. — 2010. — 17. — #N22.

О сложности проверки инвариантности булевых полиномов, относительно одного класса линейных преобразований переменных

Бухман Антон Владимирович

Московский государственный университет имени М. В. Ломоносова, e-mail: antvbx@gmail.com

В данной работе будем рассматривать булевы функции, их задание полиномами Жегалкина [1]. Будем считать, что функции зависят от переменных $x_1, \ldots, x_n$. Рассмотрим следующую задачу. Пусть у нас есть полином Жегалкина $P(x_1, \ldots, x_n)$ и некоторое преобразование переменных: $x_i \to f_i(x_1, \ldots, x_n)$. Вопрос: верно ли, что после применения преобразований $f_1, \ldots, f_n$ к переменным полинома P, снова получится тот же полином P? И если это верно, то будем говорить, что полином P инвариантен относительно этих преобразований.

Немного о том, как появилась эта задача. В работе [2] был построен полиномиальный алгоритм для проверки чётности булевой функции, заданной полиномом. Чётность полинома равносильна его инвариантности относительно преобразований $x_1 \to x_1 \oplus 1, \ldots, x_n \to x_n \oplus 1$. Проверка чётности булевой функции нужна для решения вопроса о полноте системы булевых функций. И естественным образом возникает вопрос об обобщении этой задачи на класс

произвольных линейных преобразований. В данной статье будет описан один класс линейных преобразований, и для этого класса построен полиномиальный алгоритм, который, получив на вход полином и систему преобразований, выдаёт ответ ДА, если полином инвариантен, и НЕТ — иначе.

Введём алгоритмическую модель. В качестве исполнителя алгоритма возьмём RAM машину [3]. Под сложностью работы алгоритма будем понимать сложность в худшем случае на словах заданной длины. Входной полином будем кодировать словом длины $O(ln)$, где l — длина полинома (количество слагаемых), а n — количество переменных, от которых зависит полином.

Описание класса преобразований

Нам задан полином от переменных $x_1, \ldots, x_n$. Пусть переменные разделены на две части: первая $x_1, \ldots, x_s$ и вторая $x_{s+1}, \ldots, x_n$. Будем рассматривать преобразования вида (назовём их преобразованиями типа $(*)$):

$$\begin{cases} x_1 & \to & x_1 \oplus x_{q_1} \\ \ldots & & \\ x_s & \to & x_s \oplus x_{q_s} \\ x_{s+1} & \to & x_{s+1} \\ x_n & \to & x_n, \end{cases} \qquad (*)$$

где $q_i > s$.

Особенность этих преобразований в том, что часть переменных x_{s+1} и выше не меняются, но они могут входить в выражения для первых s переменных. Такое свойство преобразований позволяет построить полиномиальный алгоритм, применяя технику аналогичную той, что использовалась в работе [2].

Стоит отметить, что если число s ограничить некоторой константой, то мы могли бы за полиномиальное время подставить вместо переменных выражения из правых частей преобразования $(*)$ и привести подобные слагаемые. Если бы получили тот же полином, что и исходный, то это значит, что он инвариантен. Данная задача представляет интерес для случая, когда s большое и по порядку сравнимо с n, потому что если в полиноме есть слагаемые ранга близкого к n, то замена переменных на правые части соотношений $(*)$ с последующим раскрытием скобок приведёт к экспоненциальному числу слагаемых.

Построение полиномиального алгоритма

Идея полиномиального алгоритма состоит в ограничении перебора. Далее будет показано, что в случае экспонециального разрастания полинома после применения преобразований к каждому слагаемому, можно говорить о том, что полином не инвариантен относительно этих преобразований.

Введём обозначение. Пусть K слагаемое полинома P. Через $I(K)$ обозначим множество переменных из $x_1, \ldots, x_s$, которые входят в K.

Лемма 1. *Пусть полином P инвариантен относительно системы преобразований переменных описанного выше вида. Тогда для любой элементарной конъюнкции K, для которой существует слагаемое K' полинома P такое, что $K \leqslant K'$, справедливо, что существуют два слагаемых K_1, K_2 полинома P таких, что $I(K_1) \cap I(K_2) = I(K)$.*

Доказательство. Здесь будет приведена общая идея доказательства. Не ограничивая общности, будем считать, что между (в смысле частичного порядка на множестве мономов) K и K' не лежит других слагаемых полинома P. Тогда индукцию проводим по разности $|I(K')| - |I(K)| = t$. Для $t = 0$ берём $K_1 = K_2 = K'$, получаем утверждение леммы.

Пусть для t доказано, докажем для $t + 1$. Заметим, что мономов K^* таких, что $K < K^* \leqslant K'$ ровно $2^{t+1} - 1$. Для каждого такого K^* существует нечётное число слагаемых полинома P, которые в пересечении с K' дают это слагаемое. Но каждое такое слагаемое после раскрытия скобок и приведения подобных даёт K. Поэтому если нет слагаемого K'', которое $I(K') \cap I(K'') = I(K)$, то после раскрытия скобок и приведения подобных слагаемых мы получили бы в конечном полиноме слагаемое K, что невозможно, так как полином инвариантен относительно преобразований. **Лемма 1 доказана.**

Из леммы 1 непосредственно следует, что для каждого слагаемого K' полинома P инвариантного относительно преобразований $(*)$ верно, что $2^{|I(K')|} \leqslant l^2$. Поэтому если после применения к некоторому слагаемому полинома P преобразований $(*)$ получится полином с более, чем l^2 слагаемыми, то P не может быть инвариантен относительно преобразований $(*)$. На основании этого факта можно строить полиномиальный алгоритм для проверки инвариантности.

Теорема 1. *Существует полиномиальная детерминированная RAM машина, которая получив на вход полином Жегалкина и набор преобразований типа $(*)$, выдаёт ответ ДА, если полином инвариантен относительно системы и НЕТ иначе.*

Доказательство. Опишем явно алгоритм. Для каждого слагаемого полинома проводим следующую процедуру. Вместо переменных подставляем правые части преобразования переменных, раскрываем скобки. Если окажется, что в результате раскрытия скобок получилось слагаемых больше, чем l^2, то выдаём ответ НЕТ (так как по лемме 1 этот полином не может быть инвариантным). Складываем полученные полиномы и приводим подобные слагаемые. Если получился исходный полином выдаём ответ ДА, так как он инвариантен, ответ НЕТ иначе. **Теорема 1 доказана.**

Работа выполнена при поддержке РФФИ (проект № 17-01-00782-а).

Список литературы

[1] Яблонский С. В. Введение в дискретную математику. — М. : Высшая школа, 2003. — 384 с.

[2] Селезнева С. Н. О сложности распознавания полноты множеств булевых функций, реализованных полиномами Жегалкина // Дискрет. матем. — 1997. — Т. 9, № 4. — С. 24–31.

[3] Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М. : Мир, 1982. — 416 с.

Явная формула для числа помеченных хордальных графов бех P_4-подграфов

Воблый Виталий Антониевич

МГТУ им. Н. Э. Баумана, e-mail: vitvobl@yandex.ru

Обозначим через a_n число помеченных связных хордальных графов без P_4-подграфов с n вершинами. В [1] для a_n найдено рекуррентное соотношение, а также получена асимптотика при $n \to \infty$.

Пусть $S(n,k)$ — числа Стирлинга 2-го рода, которые могут быть определены с помощью производящей функции [2, с. 43, 54, 55]:

$$\frac{1}{k!}(e^z-1)^k = \sum_{n=0}^{\infty} S(n,k)\frac{z^n}{n!}.$$

Пусть еще $T(z) = \sum\limits_{k=1}^{\infty} \frac{k^{k-1}}{k!} z^k$ — древесная функция, удовлетворяющая уравнению $T(z)\exp(-T(z)) = z$ [3].

Теорема. *Для a_n при $n \geqslant 1$ верна формула*

$$a_n = \sum_{k=1}^{n} (-1)^{n+k} k^{k-1} S(n,k).$$

Доказательство. Введем производящую функцию $f(x) = \sum\limits_{n=1}^{\infty} a_n \frac{x^n}{n!}$. В [1] получено уравнение $f(x) = (1-e^{-x})e^{f(x)}$.

Следовательно, имеем $f(x)e^{-f(x)} = 1 - e^{-x}$,

$$f(x) = T(1-e^{-x}) = \sum_{k=1}^{\infty} \frac{k^{k-1}}{k!}(1-e^{-x})^k = \sum_{k=1}^{\infty} k^{k-1}(-1)^k \sum_{n=0}^{\infty} S(n,k)\frac{(-x)^n}{n!} =$$

$$\sum_{n=1}^{\infty} \frac{x^n}{n!} \sum_{k=1}^{\infty} k^{k-1}(-1)^{n+k} S(n,k) = \sum_{n=1}^{\infty} a_n \frac{x^n}{n!}.$$

Теорема 1 доказана.

Список литературы

[1] Castelo R. Wormald N. Enumeration of P_4-free chordal graphs // Graphs and Combinatorics, — 2003. — Vol. 19. — P. 467–474.

[2] Риордан Дж. Введение в комбинаторный анализ. — М.: ИЛ, 1962. — 288 с.

[3] Knuth D. E., Pittel B. A recurrence related to trees // Proc. Amer. Math. Soc. — 1989. — Vol. 105:2. — P. 335–349.

Перечисление помеченных геодезических k-циклических кактусов

Воблый Виталий Антониевич[1], **Мелешко Анна Константиновна**[2]

[1] МГТУ им. Н. Э. Баумана, e-mail: vitvobl@yandex.ru

[2] МГТУ им. Н. Э. Баумана, e-mail: akmeleshko@gmail.com

Геодезический граф — это связный граф, у которого любая пара вершин связана единственной кратчайшей цепью (геодезической) [1]. Кактусом называется связный граф, в котором нет ребер, лежащих более чем на одном простом цикле [2, с. 93]. Все блоки кактуса — ребра или простые циклы. Цикломатическим числом связного графа называется увеличенная на единицу разность между числом ребер графа и числом его вершин. Под k-циклическим графом понимается связный граф с цикломатическим числом, равным k.

Геодезические графы используются при проектировании топологической структуры компьютерных сетей [3].

Теорема. *Число помеченных геодезических k-циклических кактусов с n вершинами $GC(n,k)$ равно*

$$GC(n,k) = \frac{(n-1)!}{n2^k k!} \sum_{m=0}^{[\frac{n-2k-1}{2}]} \binom{m+k-1}{k-1} \frac{n^{n-2m-k-1}}{(n-2m-2k-1)!}.$$

Доказательство. Пусть $S(n,k)$ — число помеченных связных графов с n вершинами и цикломатическим числом k, а $B_k(z)$ — экспоненциальная производящая функция для числа помеченных блоков с цикломатическим числом k.

В работе [4] автором было получено соотношение

$$S(n,k) = \frac{(n-1)!}{nk!}[z^{-1}] \exp(nz) Y_k(nB_1'(z), ..., nk!B_k'(z)) z^{-n},$$

где $[z^{-1}]$ — оператор формального вычета [5, с. 25], а $Y_k(x_1, ..., x_k)$ — многочлен разбиений.

Многочлены разбиений (многочлены Белла) $Y_m(x_1, ..., x_m)$ могут быть определены с помощью формулы [6]:

$$Y_m(x_1, ..., x_m) = \sum_{\pi(m)} \frac{m!}{k_1! \cdots k_m!} \left(\frac{x_1}{1!}\right)^{k_1} \cdots \left(\frac{x_m}{m!}\right)^{k_m},$$

где суммирование проводится по всем разбиениям $\pi(m)$ числа m:
$k_1 + 2k_2 + ... + mk_m = m, \quad k_i \geqslant 0, \quad i = 1, ..., m.$

Так как у кактусов нет блоков с цикломатическим числом $k > 1$, то $B_k = 0$ при $k > 1$ и $Y_m(x_1, 0, ..., 0) = x_1^m$. Кроме того, кактусы — планарные графы. Следовательно, у геодезических кактусов все циклы имеют нечетную длину [1] и, поскольку число помеченных циклов с $2n + 1$ вершинами равно $(2n)!/2$, получим

$$B_1(z) = \sum_{n=1}^{\infty} \frac{1}{2}(2n)! \frac{z^{2n+1}}{(2n+1)!}, \quad B_1'(z) = \sum_{n=1}^{\infty} \frac{1}{2} z^{2n} = \frac{z^2}{2(1-z^2)}.$$

После разложения экспоненты в ряд, имеем

$$GC(n,k) = \frac{(n-1)!}{nk!}[z^{-1}] \sum_{l=0}^{\infty} \frac{n^l z^l}{l!} \Big(\frac{nz^2}{2(1-z^2)}\Big)^k z^{-n} =$$

$$= \frac{(n-1)!}{nk!}[z^{-1}] \sum_{l=0}^{\infty} \frac{n^{l+k}}{l!} \frac{z^{l+2k-n}}{2^k(1-z^2)^k}.$$

Используя известный ряд [6, с. 141]: $(1-z)^{-k} = \sum_{m=0}^{\infty} \binom{m+k-1}{k-1} z^m$, найдем

$$GC(n,k) = \frac{(n-1)!}{n2^k k!}[z^{-1}] \sum_{l=0}^{\infty} \sum_{m=0}^{\infty} \binom{m+k-1}{k-1} \frac{n^{l+k}}{l!} z^{l+2k-n+2m} =$$

$$= \frac{(n-1)!}{n2^k k!} \sum_{m=0}^{\infty} \binom{m+k-1}{k-1} \frac{n^{n-2m-k-1}}{(n-2m-2k-1)!}.$$

Учитывая, что биномиальный коэффициент обращается в нуль при $n - 2m - 2k - 1 < 0$, получим утверждение теоремы. **Теорема 1 доказана.**

Список литературы

[1] Stemple J. G., Watkins M. E. On planar geodetic graphs // J. Combin. Theory. — 1968. — Vol. 4. — P. 101–117.

[2] Харари Ф., Палмер Э. Перечисление графов. — М. : Мир, 1977. — 324 с.

[3] Frasser C.E. k-geodetic graphs and their application to the topological design of computer networks.// Proc. Argentinian Workshop on Theoretical Computer Science — 1999. 28 JAIIO-WAIT‘99. — 1999. — P. 187–203.

[4] Воблый В. А. О перечислении помеченных связных графов с заданными числами вершин и ребер.// Дискретный анализ и исследование операций. — 2016. — Т. 23, № 2.— С. 5–20.

[5] Гульден Я., Джексон Д. Перечислительная комбинаторика. — М. : Наука, 1990. — 504 с.

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

Задача построения универсальных функций

Вороненко Андрей Анатольевич
Московский государственный университет имени М. В. Ломоносова, e-mail: dm6@cs.msu.ru

В настоящей статье описываются результаты цикла работ автора по поставленной им проблеме существования и синтеза универсальных функций [1].

Введем необходимые определения. Пусть нам доступен выбор произвольной функции $f(x_1, \ldots, x_n)$. Пусть у противника есть ложная информация о принадлежности функции $f(x_1, \ldots, x_n)$ некоторому классу K. Будем говорить, что функция $f(x_1, \ldots, x_n)$ порождает функцию $g(x_1, \ldots, x_n)$, если существует подмножество X области определения функции $f(x_1, \ldots, x_n)$ такое, что $g(x_1, \ldots, x_n)$ является единственной функцией из класса K, совпадающей с $f(x_1, \ldots, x_n)$ на X. Функция $f(x_1, \ldots, x_n)$ называется универсальной для класса K, если она порождает любую функцию из класса K. Универсальная функция может быть частичной. В этом случае для дискретных универсальных функций имеет смысл решать задачу о минимизации размера области определения.

Теорема 1. *Для класса линейных булевых функций универсальные функции существуют для числа переменных $n \geqslant 4$. При этом минимальный размер области определения универсальной функции асимптотически ограничен сверху* [1] *функцией $3\frac{1}{5}n$, а снизу* [2] *— функцией $2\frac{1}{6}n$.*

Теорема 2 [3]. *Для любых k и n универсальная функция для класса линейных k-значных содержит не менее $k(n+1)$ наборов на области определения.*

Для доказательства теоремы 2 используется тот факт, что универсальная функция должна принимать значение каждой константы не менее, чем в $n+1$ точке в общем положении. Доказательство верхней оценки теоремы 1 получается при помощи конструктивного построения универсальных функций четырех и пяти переменных, а также лемм о переходе к кратным размерностям числа аргументов и увеличении размерности на единицу. Нижняя оценка теоремы 1 получена анализом количества наборов, требуемых на порождение некоторых функций, не порождаемых подфункцией, порождающей ноль.

Теорема 3 [3, 4, 5, 6]. *Универсальные функции для класса линейных k-значных существуют при $k \geqslant 5$ для любого $n \geqslant 2$, а при $k = 3$ и $k = 4$ — для любого $n \geqslant 3$.*

При решении задач об универсальных k-значных функциях использовалась техника сведения задачи к проблеме покрытия матрицы со столбцами — упорядоченными парами функций, а строками — всевозможными значениями функций. Для больших k применялся градиентный алгоритм. При этом скорость сходимости падала за счет потери несовместимых по значению в одной точке строк. Для малых k использоалась генерация случайных функций и полный перебор с ограничениями.

Справедливы следующие утверждения.

Теорема 4 [7]. *Для класса монотонных булевых функций универсальных функций не существует.*

Данный результат следует из необходимости иметь все верхние нули и нижние единицы для порождения монотонной функции.

Теорема 5 [7]. *Существует последовательность* $\{f_n(x_1, \dots x_n)\}$ *функций, порождающих* $2^{\Omega(\frac{2^n}{n^{4.5}})}$ *монотонных булевых функций.*

Использовалась функция голосования с инвертированными средними слоями.

Теорема 6. *Для класса булевых полиномов степени не выше* s*, начиная с некоторого* n*, существуют универсальные функции. Минимальный размер области определения универсальной функции является величиной* $\Theta(n^s)$.

Здесь применялись все техники, использованные как для порождения булевых, так и для k-значных линейных функций.

Работа выполнена при поддержке гранта РНФ № 16-11-10014.

Список литературы

[1] Вороненко А. А. Об универсальных частичных функциях для класса линейных // Дискретная математика. 2012. № 3. С. 62–65.

[2] Вороненко А. А., Вялый М. Н. Нижняя оценка мощности области определения универсальных функций для класса линейных булевых функций // Дискретная математика. 2016. № 4. С. 50–57.

[3] Вороненко А. А. О порождении ложных образов линейных k-значных функций // Прикладная математика и информатика. № 48. М: МАКС Пресс, 2015. С. 85–92.

[4] Вороненко А. А. О порождении ложных образов линейных k-значных функций для составных k при растущем числе переменных // Вестн. Моск. ун-та. Сер. 15. Вычисл. матем. и киберн. 2016. № 2. С. 28–31.

[5] Вороненко А. А., Воронова Н. К., Ильютко В. П. О существовании универсальных функций для класса линейных k-значных при небольших k // Прикладная математика и информатика. № 51. М: Макс Пресс, 2016. С. 100–108.

[6] Воронова Н. К. Универсальные функции двух переменных. Выпускная квалификационная работа. М: ВМК МГУ. 2016.

[7] Вороненко А. А., Федорова В. С. О порождении булевых функций в предположении монотонности // Вестн. Моск. ун-та. Сер. 15. Вычисл. матем. и киберн. 2013. № 1. С. 46–47.

Отождествление переменных у бесповторных функций

Вороненко Андрей Анатольевич, Малахова Елена Сергеевна

Московский государственный университет им. М. В. Ломоносова, e-mail: dm6@cs.msu.ru, elena777mc@yandex.ru

При изучении вопросов сложности представлений булевых функций формулами возникают такие понятия, как бесповторность и слабоповторность булевых функций. Бесповторные булевы функции, то есть функции, для которых существует представление в виде формулы, в которой каждая переменная встречается не более одного раза, являются классическими объектами исследований, они изучались еще с 40–50-х годов прошлого века. Простая природа таких функций сказывается и в дальнейшем, они находят применение уже для более сложных задач, например, при синтезе настраиваемых модулей для управляющих логических устройств [1].

Повторная функция в базисе B называется слабоповторной, если любая ее подфункция бесповторна в этом базисе. Первым шагом в изучении слабоповторных функций был результат, полученный В. А. Стеценко [2, 3, 4] об описании всех обобщенных типов функций слабоповторных в базисе $B_0 = \{\&, \vee, \neg\}$.

Функциями Стеценко называются функции следующих пяти семейств:

$$\begin{aligned}
f_d^s &= x_1(\bar{x}_2 \vee x_3 \dots x_s) \vee \bar{x}_2\bar{x}_3 \dots \bar{x}_s, && s \geqslant 3,\\
f_t^s &= x_1x_2 \dots x_s \vee \bar{x}_1\bar{x}_2 \dots \bar{x}_s, && s \geqslant 2,\\
f_m^s &= x_1(x_2 \vee \dots \vee x_s) \vee x_2 \dots x_s, && s \geqslant 3,\\
f_4 &= x_1(x_2 \vee x_3) \vee x_3x_4,\\
f_5 &= x_1(x_3x_4 \vee x_5) \vee x_2(x_3 \vee x_4x_5).
\end{aligned}$$

Теорема (Стеценко). *Из любой повторной в базисе B_0 функции можно подстановкой констант получить подфункцию, обобщенно однотипную с одной из функций Стеценко.*

Следующим шагом был результат Н. А. Перязева [5] об описании всех слабоповторных функций в базисе $B_2 = \{\&, \vee, \neg, \oplus\}$.

Функциями Перязева называются следующие четыре функции:

$$\begin{aligned}
p_1 &= x_1x_2x_3 \oplus \bar{x}_2\bar{x}_3,\\
p_2 &= x_1(x_2 \vee x_3) \oplus x_2x_3,\\
p_3 &= x_1x_2x_3x_4 \oplus (x_1 \oplus \bar{x}_2)\bar{x}_3\bar{x}_4,\\
p_4 &= x_1(x_2 \vee x_3x_4) \oplus (x_3 \vee x_2x_4).
\end{aligned}$$

Теорема (Перязев). *Из любой повторной в бинарном базисе функции подстановкой констант можно получить подфункцию, обобщенно однотипную либо с одной из функций Стеценко (для f_t^s $s \geqslant 3$), либо с одной из функций Перязева.*

Аппарат дискретных функций широко применяется при логическом проектировании вычислительных устройств, в синтезе, диагностике и контроле различного рода схем, кодировании информации и передаче данных, в языках программирования и т. д. Для правильной организации практической деятельности необходимо организовать ее наиболее эффективным образом. При этом значимыми являются исследования различных свойств бесповторных и слабоповторных булевых функций.

В настоящей работе было проведно исследование свойств бесповторной функции при отождествлении двух переменных в базисе B_0 и двух переменных в базисе B_2.

В базисе $B_0 = \{\&, \vee, \neg, 0, 1\}$ *каноническим деревом* называется помеченное корневое дерево, удовлетворяющее следующим условиям:

1. Константой 0 или 1 (бесповторные функции) может быть помечена лишь вершина, являющаяся единственной корневой вершиной в дереве.
2. Листья дерева помечены переменными или их отрицаниями (литералами). Разные листья помечены литералами, соответствующими различным переменным.
3. Внутренние вершины помечены символами $\vee$ и $\&$. При этом каждая внутренняя вершина имеет не менее 2-х сыновей.
4. Смежные вершины помечены различными символами.

Было введено понятие *усеченного дерева* и *спиленного дерева*. Корнем усеченного дерева является первая общая внутренняя вершина на пути от отождествленных переменных к корню исходного дерева. При этом усеченное дерево не имеет внутренних вершин, кроме идущих от отождествленных переменных к корню. Все листья усеченного дерева составляют подмножество листьев исходного дерева.

Операция спила — подстановка констант на места переменных, кроме y^δ и y^σ, которую нельзя применять в следующих случаях:

1. Дерево имеет не более одной внутренней вершины (спиленное дерево вида 1).
2. Дерево имеет по одной внутренней вершине на кажом из путей от отождествленных переменных к корню в случае $\delta \neq \sigma$ (спиленное дерево вида 2).
3. Дерево имеет две внутренние вершины на одном пути и одну внутреннюю вершину на втором пути в случае $\delta = \sigma$ (спиленное дерево вида 3).

Спиленное дерево — дерево, полученное из *усеченного дерева*, к которому нельзя применить операцию спила.

Для базиса B_0 была сформулирована и доказана следующая основная теорема:

Теорема 1. *Если у бесповторной в базисе* $B_0 = \{\&, \vee, \neg\}$ *функции отождествить две переменные* (y^δ *и* y^σ)*, то получится либо бесповторная функция, либо повторная функция, из которой с помощью подстановок констант можно получить функцию, обобщенно однотипную функции Стеценко* f_4*, либо функции Стеценко* f_d^3*. При этом, если из усеченного дерева функции можно получить спиленное дерево вида* 1*, функция является бесповторной, вида* $2-$ *обобщенно однотипной функции Стеценко* f_d^3*, вида* $3-$ *обобщенно однотипной функции Стеценко* f_4*.*

Для базиса B_2 была сформулирована и доказана следующая основная теорема:

Теорема 2. *Если у бесповторной в базисе* $B_2 = \{\&, \vee, \neg, \oplus\}$ *функции отождествить две переменные* (y^δ *и* y^σ)*, то получится либо бесповторная функция, либо повторная функция, из которой с помощью подстановок констант можно получить функцию, обобщенно однотипную одной из функций Стеценко* f_4*,* f_d^3 *или* f_t^3*, либо функции Перязева* p_1*.*

Работа выполнена при поддерже гранта РФФИ № 15-01-07474-а.

Список литературы

[1] Артюхов В. Л., Копейкин В. Л., Шалыто А. А. Настраиваемые модули для управляющих логических устройств. — Ленинград: Энергоиздат, 1981. — 166 с.

[2] Стеценко В. А. Об одном необходимом признаке для предмаксимальных базисов в P_2 // Докл. АН СССР. — № 315. — 1990. — С. 1304–1307.

[3] Стеценко В. А. О предплохих базисах в P_2 // Математические вопросы кибернетики. Вып. 4 — М.: Наука, 1992. — С. 139–177.

[4] Stetsenko V. On almost bad Boolean bases // Theoretical Computer Science. — 1994. — V. 136. — P. 419–469.

[5] Перязев Н. А. Слабоповторные булевы функции в бинарном базисе // Дискретная математика и информатика. Вып. 4. — Иркутск: Изд-во Иркут. ун-та, 1998. — С. 12.

Построение асимптотически оптимальных двусторонних вложений полных двоичных деревьев в прямоугольные решётки

Высоцкий Лев Игоревич

Московский государственный университет имени М. В. Ломоносова, e-mail: vysotskylev@yandex.ru

При проектировании различных цифровых схем и аналоговых устройств на определённом этапе встаёт вопрос геометрического размещения элементов и

проводников на кристалле СБИС. При этом одним из важнейших параметров, который производители стремятся оптимизировать, является размер схемы.

В данной работе исследуется задача размещения полных двоичных деревьев в прямоугольной решётке (ПР), т. е. графе, вершинами которого являются точки (x, y) плоскости с целочисленными координатами, заключённые в прямоугольник $[a, a + \ell] \times [b, b + h]$, а рёбра соединяют все пары точек (x_1, y_1) и (x_2, y_2) таких, что $|x_1 - x_2| + |y_1 - y_2| = 1$.

Для формализации самого понятия размещения необходимо ввести следующее определение (см., например, [1]). *Вложением* графа F в граф G называется пара инъективных отображений (φ, ψ):

$$\varphi \ : \ V(F) \to V(G), \quad \psi \ : \ E(F) \to C(G),$$

где $V(F)$ и $V(G)$ — множества вершин графов F и G соответственно, $E(F)$ — множество рёбер графа F, а $C(G)$ — множество цепей графа G. Эти отображения должны обладать тем свойством, что для любого ребра $e = (u, v) \in E(F)$ цепь $\psi(e)$ соединяет $\varphi(u)$ и $\varphi(v)$.

В соответствии с моделируемой реальной задачей рассматриваются лишь те вложения, в которых образы рёбер не пересекаются, образы листьев дерева расположены на нижней и верхней, а образ корня — на боковой стороне решётки.

В работе ставится задача нахождения минимального по площади вложения полного двоичного дерева.

Рассмотрим вложение полного двоичного дерева D_d, построенное индуктивно с базой, изображённой на рисунке 1 а, б для случаев $d = 1$ и $d = 2$, и с переходом, изображённым на рисунке 1 в, г для случаев чётного и нечётного d соответственно (см. [1]). При этом каждую цепь, соединяющую корни поддеревьев, будем проводить так, чтобы каждая её вершина лежала не выше соответствующей вершины любой другой такой цепи. Назовём вложение *стандартным*, если оно получено из построенного таким образом вложения применением нескольких операций горизонтального растяжения, при каждой из которых выбирается некоторая координата x_0, удаляются все транзитные цепи, пересекаемые вертикальной прямой $x = x_0$, получившиеся половины решётки раздвигаются по горизонтали на некоторое целочисленное расстояние, а удалённые цепи начиная с самой нижней «перетрассируются» заново настолько низко, насколько возможно. Если вместо перетрассировки пересекаемые рёбра лишь «размножаются», то такое вложение назовём *упрощённым стандартным* (см. Рис. 2).

Основным путём двустороннего вложения $\mathcal{M}$ назовём такой путь от корня к листу, что при его удалении получаются поддеревья (назовём их *основными*), каждое из которых имеет листья лишь на одной из сторон решётки. Заметим, что у любого вложения такой путь можно выделить.

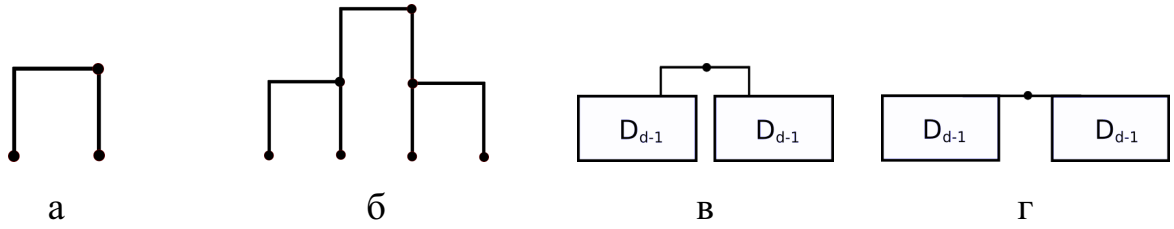


Рис. 1. Вложение полного двоичного дерева.

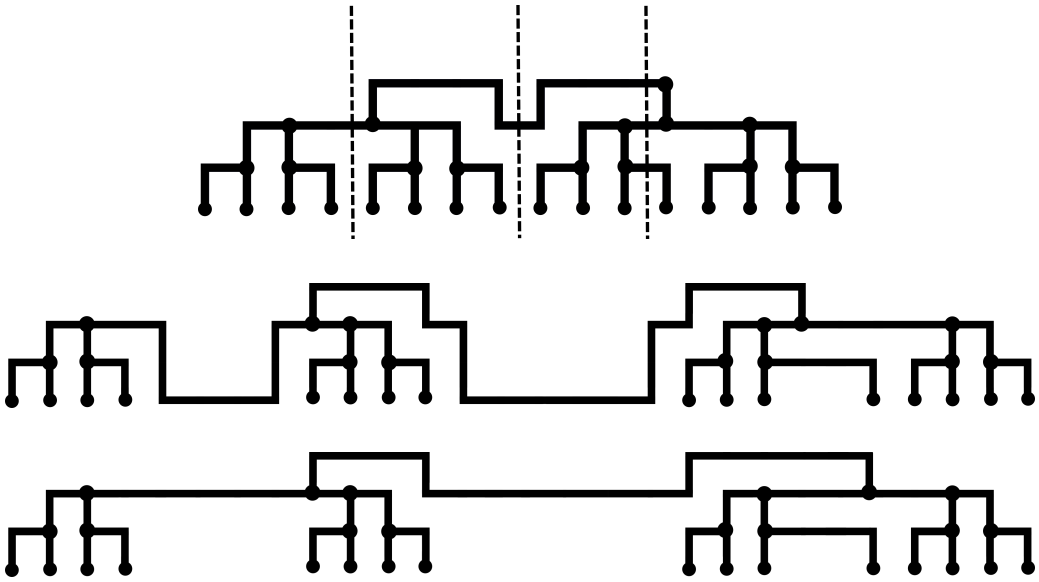


Рис. 2. Стандартное вложение D_4 до (сверху) и после (в середине) операции растяжения, внизу — упрощённое стандартное вложение после тех же растяжений; пунктиром отмечены выбранные вертикальные прямые.

Можно показать, что для любого d существует вложение полного двоичного дерева D_d, в котором все основные поддеревья имеют упрощённые стандартные вложения, а площадь отличается от оптимальной S^* на величину $O(S^* \frac{\log d}{d})$.

Предлагается решить задачу минимизации площади в классе вложений, в которых основные поддеревья имеют упрощённые стандартные вложения. Для этого нужно заметить, что лишь небольшое количество поддеревьев, ответвляющихся на основном пути, вносят значимый вклад в длину решётки: суммарный размер первых k поддеревьев есть $2^d(1 - 1/2^k)$, т. е. при $k = \lceil \log_2 d \rceil$ это число асимптотически равно 2^d. Поэтому можно перебрать все возможные взаимные расположения лишь этих k поддеревьев на обоих сторонах решётки и дальше решать задачу выбора оптимального растяжения каждого из них.

Для этого предлагается использовать следующий подход. Рассмотрим его на примере двух поддеревьев A и B. Пусть $(a_i)_{i=0}^n$ и $(b_j)_{j=0}^m$ — ординаты верхних точек вдоль прямых $x = i$ ($x = j$) для нерастянутых стандартных вложений дерева A и B соответственно. *Растяжением* последовательности $(a_i)_{i=0}^n$

назовём последовательность $(a'_k)_{k=0}^N$, для которой (a_i) является подпоследовательностью, и к тому же все элементы a'_k, находящиеся между a_i и a_{i+1}, равны ординатам верхних рёбер, пересекаемых прямой $x = i + \frac{1}{2}$. Ясно, что два растяжения $(a'_k)_{k=0}^N$ и $(b'_k)_{k=0}^N$ со свойством $a'_k + b'_k \leqslant h$ соответствуют растяжениям деревьев A и B, при которых возможно их вложение в решётку высоты h, первое на одной стороне решётки, а второе — на другой.

Для нахождения минимальной длины N введём величину $F(p, q)$ — минимальная длина растяжения, удовлетворяющего указанному свойству, для последовательностей $(a_i)_{i=0}^p$ и $(b_j)_{j=0}^q$. Используя принцип оптимальности Беллмана, можно выразить $F(p, q)$ через $F(p, q-1)$, $F(p-1, q)$ и $F(p-1, q-1)$. Таким способом можно эффективно (за $O(pq)$ шагов) найти величину $F(m, n)$, равную минимальной длине решётки высоты h, в которую возможно описанное вложение деревьев A и B. Для обобщения рассуждения на случай двух наборов поддеревьев, нужно лишь взять две «склеенные» последовательности высот и применить к ним описанный алгоритм.

Используя модификацию алгоритма Хиршберга [2], можно добиться, чтобы алгоритм построения вложения дерева D_d площади $S^*(1 + O(\frac{\log d}{d}))$ имел временную сложность $O(d^4 2^{2d})$ битовых операций и требовал $O(d2^d)$ бит дополнительной памяти.

Список литературы

[1] Ложкин С. А., Ли Да Мин. О некоторых оптимальных вложениях двоичных и троичных деревьев в плоские прямоугольные решётки // Вестник Московского Университета. Серия 9. Вычислительная математика и киберенетика. — 1995. — № 4. — С. 49–55.

[2] Hirschberg, D. S. A linear space algorithm for computing maximal common subsequences // Communications of the ACM. — 1975. — V. 18, № 5. — P. 341–343.

Вычислительные возможности конечных автоматов со счетчиком для задач отделимости

Гайнутдинова Аида Фаритовна

Казанский федеральный университет, e-mail: aida.ksu@gmail.com

Односторонние конечные являются одной из простейших вычислительных моделей, служащих для распознавания языков. Известно, что классы языков, распознаваемых вероятностными конечными автоматами с ограниченной ошибкой и детерминированными конечными автоматами, совпадают, однако в ряде случаев вероятностные автоматы оказываются экспоненциально экономнее детерминированных. Также известно, что экспоненциальное преимущество

сложности вероятностных автоматов перед детерминированными является максимально возможным. При добавлении в модель конечных автоматов дополнительных ресурсов, таких как дополнительная память, данная ситуация меняется. Так, известно, что вероятностные конечные автоматы со счетчиком являются более мощной моделью, чем детерминированные [1].

Наряду с задачами распознавания языков в последние годы активно исследуются задачи отделимости языков. Задача отделимости является обобщением задачи распознавания и формулируется следующим образом. Пусть даны два языка L и L', заданные над алфавитом Σ ($L \cap L' = \varnothing$, а также $L \cup L' \neq \Sigma^*$ в общем случае). Для входного слова $w \in L \cup L'$ требуется определить, $w \in L$ или $w \in L'$. В ряде случаев результаты, полученные для задач отделимости, отличны от аналогичных результатов для задач распознавания [2–4]. Так, известно, что для задач отделимости преимущество в сложности вероятностных автоматов перед детерминированными может быть более чем экспоненциальным [4]. В данной работе исследуются вероятностные и детерминированные конечные автоматы, снабженные дополнительной память в виде счетчика. Показывается, что для задач отделимости вероятностная модель превосходит по своим вычислительным возможностям детерминированную.

Приведем необходимые определения.

Односторонний вероятностный конечный автомат со счетчиком (1ВКАС) — это пятерка

$$M = (Q, \Sigma, \delta, q_0, Q_a).$$

Здесь $\delta : Q \times \Sigma \times \{Z, NZ\} \times Q \times \{-m, ..., m\} \longrightarrow [0, 1]$ — функция перехода, где $\delta(q, \sigma, z, q', c)$ — это вероятность того, что при считывании входного символа $\sigma \in \Sigma$ при состоянии счетчика z автомат перейдет из состояния $q \in Q$ в состояние $q' \in Q$ и увеличит при этом значение счетчика на $c \in \{-m, ..., m\}$. Состояние счетчика Z (соответственно, NZ) означает, что значение счетчика нулевое (соответственно, ненулевое). Функция перехода должна удовлетворять условию: для каждой тройки ($q \in Q$, $\sigma \in \Sigma$, $z \in \{Z, NZ\}$),

$$\sum_{q' \in Q, c \in \{-m,...,m\}} \delta(q, \sigma, z, q', c) = 1.$$

После считывания всех букв входного слова w автомат завершает вычисление и принимает (отвергает) слово w, если финальное состояние автомата принадлежит множеству Q_a ($Q \setminus Q_a$). Вероятность $Pr^M_{accept}(w)$ принятия входного слова w вычисляется суммированием вероятностей всех принимающих вычислительных путей автомата M. Вероятность того, что автомат M отвергнет слово, равна $Pr^M_{reject}(w) = 1 - Pr^M_{accept}(w)$.

Если функция перехода принимает значения из множества $\{0, 1\}$, мы имеем модель одностороннего детерминированного конечного автомата со счетчиком (1ДКАС).

Пусть имеются два языка L_{yes}, L_{no} ($L_{yes}, L_{no} \subseteq \Sigma^*, L_{yes} \cap L_{no} = \varnothing$). Будем говорить, что 1ДКАС решает задачу отделимости $L = (L_{yes}, L_{no})$, если он принимает все слова из L_{yes} и отвергает все слова из L_{no}.

Las Vegas 1BKAC — это 1BKAC, который, кроме ответа «accept» или «reject», может выдавать третий тип ответа — «?» («не знаю»). В этом случае множество Q состояний автомата разбивается на три непересекающихся подмножества $Q_{acc}, Q_{rej}, Q_?$ ($Q_{acc} \cup Q_{rej} \cup Q_? = Q$). Вероятность ответа «accept» (соответственно, «reject» или «?») определяется суммированием вероятностей вычислительных путей, завершающихся в состояниях из множества Q_{acc} (соответственно, Q_{rej} или $Q_?$). Будем говорить, что Las Vegas 1BKAC решает задачу отделимости $L = (L_{yes}, L_{no})$, если существует $\varepsilon \in [0, 1)$ такое, что для любого $w \in L_{yes}$ выполняется $Pr^M_{accept}(w) \geqslant 1 - \varepsilon$, $Pr^M_{reject}(w) = 0$, и для любого $w \in L_{no}$ выполняется $Pr^M_{reject}(w) \geqslant 1 - \varepsilon$, $Pr^M_{accept}(w) = 0$. То есть для любого $w \in L_{yes} \cup L_{no}$ автомат M никогда не выдает неверный ответ, но с вероятностью $\leqslant \varepsilon$ может выдать ответ «?».

Рассмотрим задачу отделимости $ONE\text{-}NONE = (ONE, NONE)$, где два непересекающихся языка ONE, $NONE$, заданные в алфавите $\{a, b, c, d\}$ определены следующим образом. Множество $ONE \cup NONE$ состоит из слов вида $\sigma\gamma$, где $\sigma \in \{a, b, c\}^*$, $\gamma \in \{d\}^*$, $|\gamma| \geqslant |\sigma|$, при этом для слов из языка ONE ровно для одной из пар символов (a, b), (b, c) или (c, a) число вхождений первого и второго символа пары в слово σ совпадают, для слов из языка $NONE$ ни для одной из пар символов (a, b), (b, c) или (c, a) число вхождений первого и второго символа пары в слово σ не совпадают.

В работе [5] показано, что задача отделимости $ONE\text{-}NONE$ разрешима Las Vegas 1BKAC с вероятностью правильного ответа $1/6$ и не разрешима 1DCA. В данной работе мы улучшаем данный результат.

Определим задачу отделимости $ONE\text{-}NONE_t$=$(ONE_t, NONE_t)$, где языки ONE_t (соответственно, $NONE_t$) состоят из слов, являющихся конкатенацией t произвольных слов из языка ONE (соответственно, из языка $NONE$).

Доказаны следующие теоремы.

Теорема 1. *Задача отделимости $ONE\text{-}NONE_t$ разрешима Las Vegas 1BKAC P с вероятностью правильного результата $p = 1 - (\frac{2}{3})^t$.*

Теорема 2. *Не существует 1ДКАС, решающего задачу отделимости $ONE\text{-}NONE_t$.*

Список литературы

[1] Freivalds R. Fast probabilistic algorithms // Mathematical Foundations of Computer Science. — 1979. — V. 74. – P. 57–69.

[2] Gainutdinova A., Yakaryilmaz A. Unary probabilistic and quantum automata on promise problems // Developments in Language Theory, 19th International Conference, DLT 2015, — 2015. — P. 252–263.

[3] Ambainis A., Yakaryilmaz A. Superiority of exact quantum automata for promise problems // Information Processing Letters. — 2012. — No. 112 (7). — P. 289–291.

[4] Geffert V., Yakaryilmaz A. Classical automata on promise problems // Descriptional Complexity of Formal Systems – 16th International Workshop, DCFS 2014. — 2014. — LNCS, V. 8614. — P. 126–137.

[5] Nakanishi M., Yakaryilmaz A. Classical and quantum counter automata on promise problems // Implementation and Application of Automata – 20th International Conference, CIAA. — 2015. — V. 9223. — P. 224–237.

О сложности верификации автоматов-преобразователей над коммутативными полугруппами

Гнатенко Антон Романович[1], **Захаров Владимир Анатольевич**[2]

[1] Московский государственный университет имени М. В. Ломоносова, e-mail: teawithcakes_yandex.ru

[2] Национальный исследовательский университет «Высшая школа экономики», e-mail: zakh_cs.msu.su

Метод верификации моделей программ [1] широко применяется для проверки правильности поведения реагирующих программ. Обычно в качестве модели программы обычно используются модели Крипке — размеченные системы переходов. Вычислениями в такой модели являются бесконечные последовательности состояний, связанные отношением переходов. Формальные описания требований правильного поведения модели задаются формулами темпоральных логик PLTL, CTL, PDL, μ-исчисление и др. Модель M удовлетворяет спецификации φ, если формула φ выполняется во всех начальных состояниях модели M.

Обычно элементарные свойства вычислений в моделях Крипке зависят только от состояний модели, но не от последовательности переходов, ведущей в достигнутое состояние. Однако поведение некоторых типов реагирующих систем проявляется соответствием (отношением) между последовательностями управляющих сигналов, поступающих на вход системы извне, и совокупным эффектом (композицией) откликов или действий, которые вырабатывает или исполняет система в ответ на внешние воздействия. При верификации таких реагирующих систем элементарные свойства поведения — это свойства конечных последовательностей действий, а не состояний модели. Это обстоятельство существенно влияет на устройство формального языка спецификаций и сложность алгоритмов верификации.

В статье [2] в качестве формальной модели последовательных реагирующих систем была предложена модель вычислений конечных автоматов-преобразователей, работающих над полугруппами действий. Для спецификации поведений таких автоматов в статье [3] был предложен специальный вариант темпоральной логики линейного времени LTL-FL (LTL with Formal Languages).

Формальные языки (множества конечных слов фиксированных алфавитов) в формулах LTL-FL используются для параметризации темпоральных операторов. В этой же статье было показано, что задача проверки выполнимости формул регулярного фрагмента FL-LTL на конечных автоматах преобразователях, работающих над свободными полугруппами, разрешима со сложностью по времени, оцениваемой двойной экспонентой от размера проверяемой формулы.

В данной заметке мы предприняли попытку распространить полученный в статье [3] результат на класс автоматов-преобразователей, работающих над коммутативными полугруппами. В таком автомате выполняемые действия могут мыслиться как движения робота в многомерном пространстве. Нам удалось установить, что алгоритмически неразрешимая задача проверки достижимости конфигураций в сетях Петри с ингибиторными дугами [4] сводима к задаче проверки выполнимости FL-LTL на конечных автоматах преобразователях, работающих над вполне коммутативными полугруппами и абелевыми группами. Показано также, что даже задача верификации простейших свойств происшествия, которые выражаются формулами с одним темпоральным оператором, столь же сложна, как и задача проверки выполнимости формул арифметики Пресбургера [5].

Пусть заданы конечные алфавиты *событий* $\mathcal{C}$ и *действий* $\mathcal{A}$. Слова и ω-слова в алфавите $\mathcal{C}$ называются *потоками событий*. Действия из множества $\mathcal{A}$ являются образующими моноида $(S, \circ, e)$. Элементы моноида S называются *историями*. В данной заметке рассматриваются вполне коммутативные моноиды с n образующими, которые будем обозначать $\mathbf{N}^n$, и абелевы группы ранга n, которые будем обозначать $\mathbf{Z}^n$.

Конечный *автомат-преобразователь* $\pi = \langle \mathcal{C}, \mathcal{A}, Q, Q_0, T \rangle$ — это система переходов, состоящая из конечного множества *состояний* Q, подмножества *начальных состояний* Q_0 и *отношения переходов* $T \subseteq Q \times \mathcal{C} \times Q \times \mathcal{A}^*$. Отношение переходов T тотально, т.е. из любого состояния q в множестве T есть некоторый переход (q, c, q', h). *Вычислением* преобразователя π называется всякая бесконечная последовательность четверок

$$(q_0, c_1, q_1, h_1), (q_1, c_2, q_1, h_2), \ldots, (q_{i-1}, c_i, q_i, h_i), (q_i, c_{i+1}, q_{i+1}, h_{i+1}), \ldots$$

в которой $q_0 \in Q_0$ и $(q_{i-1}, c_i, q_i, h_i) \in T$ для любого $i, i \geqslant 1$. Последовательность $\alpha = (c_1, h_1), (c_2, h_2), \ldots, (c_i, h_i), \ldots$ называется *траекторией* данного вычисления. Множество траекторий всех вычислений преобразователя π обозначим записью $Tr(\pi)$.

Формулы темпоральной логики LTL-FL строятся из базовых предикатов, выражающих отношения в моноиде S, при помощи булевых связок и темпоральных операторов X_{exp}, Y_{exp}, F_{exp} и G_{exp}, параметризованных регулярными выражениями exp над множеством сигналов $\mathcal{C}$. В статье [3] полностью определено отношение выполнимости LTL-FL-формул на заданной траектории $\alpha \models \varphi$. Например, выполнимость $\alpha \models F_{exp}P$ означает, что для некоторого $i, i \geqslant 0$,

имеет место включение $c_1 c_2 \dots c_i \in L(exp)$ и при этом элемент $h_1 \circ h_2 \circ \dots \circ h_i$ полугруппы S удовлетворяет предикату P. Автомат-преобразователь π удовлетворяет LTL-FL-формуле φ (обозначается записью $\pi \models \varphi$), если соотношение $\alpha \models \varphi$ имеет место для любой траектории α из множества $Tr(\pi)$.

Для вполне коммутативных моноидов и абелевых групп, элементы которых могут быть представлены целочисленными векторами, базовыми предикатами являются арифметические отношения вида $\vec{x} = \vec{c}$ или $\vec{x} \leqslant \vec{c}$.

Теорема 1. *Существует алгоритм, который для любой сети Петри PN с ингибиторными дугами, имеющей k позиций, и любой разметки M для этой сети, строит такой автомат-преобразователь π_{PN} над абелевой группой $\mathbf{Z}^k$ и такую LTL-FL формулу φ_M, что разметка M достижима в сети PN тогда и только тогда, когда $\pi_{PN} \models \varphi_M$.*

Следствие 1. *Задача верификации автоматов-преобразователей, работающих над абелевыми группами, относительно LTL-FL формул алгоритмически неразрешима.*

Теорема 2. *Существует алгоритм, который для любой сети Петри PN с ингибиторными дугами, имеющей k позиций, и разметки M для этой сети, строит такой автомат-преобразователь π'_{PN} над вполне коммутативным моноидом $\mathbf{N}^{2k}$ и такую LTL-FL формулу ψ_M, что разметка M достижима в PN тогда и только тогда, когда $\pi'_{PN} \models \psi_M$.*

Следствие 2. *Задача верификации автоматов-преобразователей, работающих над вполне коммутативными моноидами, относительно LTL-FL формул алгоритмически неразрешима.*

Теорема 3. *Для любого отношения $R, R \subseteq \mathbf{N}^k$, выразимого в арифметике Пресбургера существует такой автомат-преобразователь π_R над вполне коммутативным моноидом $\mathbf{N}^{2k}$, что $R = \{\vec{c} : \pi_R \models F_{\mathcal{C}^*}(\vec{x} = \vec{c})\}$.*

Работа выполнена при поддержке РФФИ (проект № 16-01-00546-а).

Список литературы

[1] Кларк Э. М., Грамберг О., Пелед Д. Верификация моделей программ. Model Checking. — М.: МЦНМО, 2002. — 416 с.

[2] Захаров В. А. Моделирование и анализ поведения последовательных реагирующих программ // Труды Института системного программирования РАН. — 2015. — Т. 27, № 2. — С. 221–250.

[3] Zakharov V., Kozlova D. On the model checking of sequential reactive systems // Proceedings of the 25-th International Workshop on Concurrency, Specification and Programming, Rostock, Germany, September 28-30, 2016. — 2016. — V. 1698. — P. 233–244.

[4] Котов В. Е. Сети Петри. — М.: Мир, 1984. — 160 с.

[5] Рабин М. О. Разрешимые теории // Справочная книга по математической логике. — М.: Наука, 1982. — Т. 3. — С. 77–111.

Об автоматных моделях развития атак в компьютерных сетях и вычислительных алгоритмах их исследования

Горбатенко Дмитрий Евгеньевич[1], **Кочемазов Степан Евгеньевич**[2], **Семенов Александр Анатольевич**[3]

[1] Институт математики, экономики и информатики Иркутского государственного университета, e-mail: gorbadima@gmail.com

[2] Институт динамики систем и теории управления им. В. М. Матросова СО РАН, e-mail: veinamond@gmail.com

[3] Институт динамики систем и теории управления им. В. М. Матросова СО РАН, e-mail: biclop.rambler@yandex.ru

Проблемы моделирования и предотвращения атак в компьютерных сетях образуют актуальное направление современной информационной безопасности. В настоящей работе предлагается один класс моделей для описания развития атак в компьютерных сетях, а также приводятся техники вычислительного исследования таких моделей, базирующиеся на алгоритмах решения проблемы булевой выполнимости (SAT). В отличие от известных подходов к моделированию атак в сетях при помощи специальных графов [1–2], в предлагаемых нами моделях граф атак не представлен явно. Фактически любая трасса атаки может быть выведена как набор, выполняющий построенную специальным образом булеву формулу. В рамках введенных моделей, с нашей точки зрения, весьма естественно интерпретируются некоторые задачи, связанные с анализом атак в компьютерных сетях. В частности, мы можем исследовать с использованием современных алгоритмов решения SAT такие задачи как поиск атак с учетом различных дополнительных условий, а также задачи блокирования развития атак посредством расстановки патчей.

Будем рассматривать компьютерную сеть как ориентированный размеченный граф $G = (V, A)$. Множество вершин V, $|V| = n$, данного графа соответствует множеству хостов исходной компьютерной сети, дуги G отображают взаимодействия между хостами.

Каждому хосту сопоставляется некоторый набор данных, представляемый двоичным вектором, который будем называть состоянием рассматриваемого хоста. Каждый такой вектор содержит информацию относительно прав доступа рассматриваемого хоста на другие хосты сети, а также информацию, касающуюся уязвимостей, доступных злоумышленнику на данном хосте. Мы будем полагать, что в каждый момент дискретного времени t, $t \in \{0, 1, \ldots\}$, на хосте, представленном вершиной v_i, $i \in \{1, \ldots, n\}$, злоумышленник может использовать (говорят «эксплуатировать») одну или несколько доступных уязви-

мостей (если таковые имеются) и получить в результате несанкционированные права доступа на рассматриваемый или некоторый связанный с ним хост. В каждый последующий момент времени злоумышленник может все глубже проникать в сеть, пока не достигнет некоторой поставленной цели – например, получит доступ к информации особой ценности, находящейся на некотором хосте сети. Совокупность изменений в данных сети, происходящих в ходе описанного процесса, определяет атаку злоумышленника на рассматриваемую сеть. Каждая атака является цепью элементарных атак, представляющих собой множества пар вида (предусловие, постусловие) для каждого хоста сети. Будем считать, что в произвольный момент $t \in \{0, 1, \ldots\}$ наблюдаемое состояние произвольного хоста v_i является предусловием элементарной атаки, а ее постусловием является состояние v_i в момент времени $t + 1$. Правила перехода между конкретными предусловиями и постусловиями жестко заданы.

В целом похожий подход (без четкого выделения компоненты дискретного времени) использовался во множестве работ по моделированию атак в компьютерных сетях [3]. Описанная выше интерпретация ближе всего к интерпретации М. Данфорт [4] и именно ее мы и придерживаемся далее.

Заметим, что развитие атак в компьютерных сетях может рассматриваться в контексте довольно интенсивно изучаемых в последнее время процессов активации многоагентных систем. Такого сорта задачи впервые возникли, по-видимому, в биоинформатике [5] при изучении активационной динамики генных сетей. В статье [6] была исследована динамика активации коллективных систем в рамках феномена конформности, в соответствии с которым агент-конформист принимает решение о действии, опираясь на мнение агентов, образующих его окружение (окрестность). Рассматривались задачи скорейшей активации сети за счет удачного расположения специальных агентов-активаторов, а также обратная задача – требовалось привести сеть в неактивное состояние за счет удачного расположения агентов, бездействующих в любой момент времени (т. н. «лоялисты»).

Мы отмечаем схожесть процессов активации коллективов, рассмотренных в [6], и процессов развития атак в компьютерных сетях – активным считается хост, на котором злоумышленник получил в некоторый момент времени несанкционированные права.

На данном этапе в вычислительных экспериментах компьютерные сети интерпретировались графами случайной структуры, сгенерированными в соответствии с моделями Уоттса-Строгатца [7] и Барабаши-Альберт [8]. Для кодирования задач активации сетей в SAT использовались техники, основанные на сортирующих сетях [9].

Были решены следующие задачи:

1. задача нахождения всех возможных действий злоумышленника за заданное количество шагов;

2. задача блокирования небольшого количества уязвимостей таким образом, чтобы злоумышленник не смог добраться до целевого хоста, но сеть продолжала функционировать согласно установленным правилам.

Решение задач осуществлялось на компьютере с процессором Intel Core-i3 2310m (8Gb) оперативной памяти. Удавалось находить решения для задач на нескольких сотнях хостов с четырьмя уязвимостями для обычных хостов и двумя уязвимостями для хостов-файерволлов.

Работа поддержана грантом РНФ № 16-11-10046.

Список литературы

[1] Ammann P., Wijesekera D., Kaushik S. Scalable, graph-based network vulnerability analysis // Proc. of the 9th ACM conference on Computer and Communication Security. — 2002. — P. 217–224.

[2] Barik M.S., Sengupta A., Mazumdar C. Attack Graph Generation and Analysis Techniques. // Defence Science Journal, — 2016. — V. 66, N. 6, — P. 559–567.

[3] Sheyner O., Haines J., Jha S., Lippmann R., Wing J., M. Automated generation and analysis of attack graphs. // Proceedings of the 2002 IEEE Symposium on Security and Privacy. Washington, DC, USA: IEEE Computer Society. — 2002. — P. 273–284.

[4] Danforth M., Models for threat assessment in networks [PhD thesis] // School of Computer Science Computer, Science Department. — 2006.

[5] Kauffman S. Metabolic stability and epigenesis in randomly constructed genetic nets // Journal of Theoretical Biology. — 1969. — V. 22, — P. 437–467.

[6] Kochemazov S., Semenov A. Using Synchronous Boolean Networks to Model Several Phenomena of Collective Behavior // PLOS ONE — 2014. — V. 9, N. 12, e115156. — P. 1–28.

[7] Watts D., Strogatz S. Collective dynamics of ‘small-world’ networks // Nature — 1998. — V. 93, — P. 440–442.

[8] Barabasi A.L., Albert R. Emergence of scaling in random networks. // Science. — 1999. — V. 286, — P. 509–512.

[9] Asin R., Nieuwenhuis R., Oliveras A., Rodriguez-Carbonell E. Cardinality networks: a theoretical and empirical study // Constraints. — 2011. — V. 16, N. 2, — P. 195–221.

О НАДЕЖНОСТИ НЕВЕТВЯЩИХСЯ ПРОГРАММ В БАЗИСАХ, СОДЕРЖАЩИХ ОСОБЕННУЮ ФУНКЦИЮ

Грабовская Светлана Михайловна

Пензенский государственный университет, e-mail: swetazin@mail.ru

Рассматривается реализация булевых функций неветвящимися программами с оператором условной остановки [1] в полном конечном базисе, содержащем особенную функцию, т. е. функцию вида

$$x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus \beta_1x_1 \oplus \beta_2x_2 \oplus \beta_3x_3 \oplus \beta_0 \quad (\beta_i \in \{0,1\}, i \in \{0,1,2,3\}).$$

Программы с оператором условной остановки характеризуются наличием управляющей команды — команды условной остановки, дающей возможность досрочного прекращения работы программы при поступлении единицы на вход оператора условной остановки (который еще называют стоп-оператором).

Будем считать, что операторы условной остановки абсолютно надежны, а все вычислительные операторы независимо друг от друга с вероятностью ε $(\varepsilon \in (0, 1/2))$ подвержены однотипным константным неисправностям на выходах. *Константные неисправности типа* 0 *(типа* 1*)* характеризуются тем, что в исправном состоянии вычислительный оператор реализует приписанную ему булеву функцию φ, а в неисправном — функцию 0 (1).

Ненадежностью $N_\varepsilon(Pr)$ программы Pr назовем максимальную вероятность ошибки на выходе программы Pr при всевозможных входных наборах.

Для особенной функции возможны 6 случаев:

1) $\varphi_1(x_1, x_2, x_3) = x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus x_1 \oplus x_2 \oplus x_3 \oplus 1$;
2) $\varphi_2(x_1, x_2, x_3) = x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus x_1 \oplus 1$;
3) $\varphi_3(x_1, x_2, x_3) = x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus x_1 \oplus x_2 \oplus x_3$;
4) $\varphi_4(x_1, x_2, x_3) = x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus x_1$;
5) $\varphi_5(x_1, x_2, x_3) = x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus x_1 \oplus x_2 \oplus 1$;
6) $\varphi_6(x_1, x_2, x_3) = x_1x_2 \oplus x_2x_3 \oplus x_1x_3 \oplus x_1 \oplus x_2$.

Для повышения надежности исходных схем (программ) будем использовать функцию вида $(x_1^{\sigma_1}x_2^{\sigma_2} \vee x_3^{\sigma_3}x_4^{\sigma_4})^{\sigma_5}$ $(\sigma_i \in \{0,1\},\ i \in \{1,2,3,4,5\})$.

Положим, что вычислительные операторы неветвящейся программы подвержены константным неисправностям типа 0 на выходах.

Теорема 1. *В полном конечном базисе* B*, содержащем одну из функций* $\varphi_1(x_1, x_2, x_3)$, $\varphi_4(x_1, x_2, x_3)$, $\varphi_5(x_1, x_2, x_3)$ *или* $\varphi_6(x_1, x_2, x_3)$*, произвольную булеву функцию* f *можно реализовать сколь угодно надежной неветвящейся программой с оператором условной остановки при всех* $\varepsilon \in (0, 1/960]$.

Теорема 2. *В полном конечном базисе* B*, содержащем одну из функций* $\varphi_2(x_1, x_2, x_3)$ *или* $\varphi_3(x_1, x_2, x_3)$*, произвольную булеву функцию* f *можно реализовать неветвящейся программой с оператором условной остановки, ненадежность которой не больше* $\varepsilon + 9\varepsilon^2$ *при всех* $\varepsilon \in (0, 1/960]$.

Пусть теперь вычислительные операторы неветвящейся программы подвержены константным неисправностям типа 1 на выходах.

Теорема 3. *В полном конечном базисе B, содержащем одну из функций $\varphi_2(x_1, x_2, x_3)$ или $\varphi_3(x_1, x_2, x_3)$, произвольную булеву функцию f можно реализовать сколь угодно надежной неветвящейся программой с оператором условной остановки при всех $\varepsilon \in (0, 1/960]$.*

Теорема 4. *В полном конечном базисе B, содержащем одну из функций $\varphi_1(x_1, x_2, x_3)$, $\varphi_4(x_1, x_2, x_3)$, $\varphi_5(x_1, x_2, x_3)$ или $\varphi_6(x_1, x_2, x_3)$, произвольную булеву функцию f можно реализовать неветвящейся программой с оператором условной остановки, ненадежность которой не больше $\varepsilon + 9\varepsilon^2$ при всех $\varepsilon \in (0, 1/960]$.*

Заметим, что функции $\varphi_5(x_1, x_2, x_3)$ и $\varphi_6(x_1, x_2, x_3)$ представимы в виде $(x_1^{\sigma_1}x_2^{\sigma_2} \vee x_2^{\sigma_2}x_3^{\sigma_3} \vee x_1^{\sigma_1}x_3^{\sigma_3})^{\sigma_4}$ ($i \in \{0, 1\}$, $i \in \{1, 2, 3, 4\}$). Поэтому в базисах, содержащих данные функции, повысить надежность исходных программ можно, используя лишь один вычислительный оператор, до значения $\varepsilon + k\varepsilon^2$ при всех $\varepsilon \in (0, 1/960]$. Однако по теореме 1 в базисах, содержащих функции $\varphi_5(x_1, x_2, x_3)$ и $\varphi_6(x_1, x_2, x_3)$, произвольную булеву функцию f можно реализовать сколь угодно надежной неветвящейся программой с оператором условной остановки при константных неисправностях типа 0 на выходах вычислительных операторов, вообще говоря, при всех $\varepsilon \in (0, 1/960]$.

Ранее было установлено [2], что в базисах, содержащих обобщенную конъюнкцию (дизъюнкцию), произвольную булеву функцию f можно реализовать сколь угодно надежной неветвящейся программой с оператором условной остановки при константных неисправностях типа 0 (1) на выходах вычислительных операторов при всех $\varepsilon \in (0, 1/960]$.

Таким образом, был расширен класс базисов, в которых произвольную булеву функцию f можно реализовать сколь угодно надежной неветвящейся программой с оператором условной остановки.

Работа выполнена при поддержке РФФИ (проект № 17-01-00451-а).

Список литературы

[1] Чашкин А. В. О среднем времени вычисления значений булевых функций // Дискретный анализ и исследование операций. — 1997. — Т. 4, № 1. — С. 60–78.

[2] Грабовская С. М. Реализация булевых функций сколь угодно надежными неветвящимися программами // Сб. науч. ст. XIII Междунар. науч.-техн. конф. (г. Пенза, 23–25 ноября 2016 г.) — Пенза : Изд-во ПГУ, 2016. — С. 339–341.

Об оракульной сложности минимизации квазивыпуклых функций на целочисленной решетке

Грибанов Дмитрий Владимирович[1], **Чирков Александр Юрьевич**, **Веселов Сергей Иванович**[2]

[1] НИУ ВШЭ лаб. ЛАТАС, ННГУ им. Н. И. Лобачевского, e-mail: dimitry.gribanov@gmail.com
[2] ННГУ им. Н. И. Лобачевского, e-mail: chir7@yandex.ru, ves20@yandex.ru

В данной работе рассматриваются вопросы, касающиеся оракульных алгоритмов минимизации вещественнозначных функций, заданных на целочисленной решетке. Под оракульным алгоритмом минимизации функции f понимается алгоритм, задающий вопросы оракулу, сравнивающему значения функции на парах различных точек области определения функции f. Другими словами, оракул отвечает на вопросы вида $f(x) \leqslant f(y)$ для $x, y \in Dom(f) \cap \mathbb{Z}^n$. Под оракульной сложностью алгоритма минимизации понимается количество обращений к оракулу, необходимое для построения точки минимума функции.

В работе [1] построен алгоритм минимизации двумерной строго квазивыпуклой функции на двумерной целочисленной решетке с трудоемкостью, зависящей логарифмически от размера области поиска. Также в данной работе впервые вводится понятие разрешающего множества строго квазивыпуклой функции.

В работе [2] сделано обобщение методов работы [1]. Дана нетривиальная нижняя оценка оракульной сложности минимизации любой строго-квазивыпуклой функции на целочисленной решетке произвольной конечной размерности.

Представленная же работа состоит из двух частей. Первая часть расширяет класс рассматриваемых функций и позволяет перенести результаты работы [2] на этот класс. Во второй части рассматриваются функции с дополнительным условием симметричности. По аналогии с работой [1], мы приводим алгоритм минимизации функций этого класса на двумерной целочисленной решетке. Оракульная сложность указанного алгоритма является асимптотически оптимальной в классе оракульных алгоритмов минимизации и зависит логарифмически от области поиска. Более того, константа при логарифме в оценке не сильно отличается от константы в нижней оценке сложности класса.

Рассматриваемый класс функций

Определение 1. Определим класс $\mathbf{G}^n$ вещественнозначных функций, следующим образом. Функция f принадлежит классу $\mathbf{G}^n$ если для любого набора точек $x^{(1)}, x^{(2)}, \ldots, x^{(k)} \in Dom(f) \cap \mathbb{Z}^n$,
таких что $f(x^{(0)}) \leqslant f(x^{(1)}) \leqslant \cdots \leqslant f(x^{(k-1)}) \leqslant f(x^{(k)})$,
и для $\forall y \in x^{(k)} + cone(x^{(k)} - x^{(0)}, \ldots, x^{(k)} - x^{(k-1)})$ верно, что $f(y) \geqslant f(x^{(k)})$.

Определение класса $\mathbf{G}^n$ выглядит абстрактно, поэтому покажем что $\mathbf{G}^n$ содержит гораздо более естественные классы функций.

Рассмотрим множество вещественнозначных функций, для которых $Dom(f)$ — есть выпуклое множество. Функция f называется *квазивыпуклой* если $\forall z \in (x, y) \subseteq Dom(f)$ верно, что $f(z) \leqslant \max\{f(x), f(y)\}$. Функция f называется *строго квазивыпуклой* если $\forall z \in (x, y) \subseteq Dom(f)$ верно, что $f(z) < \max\{f(x), f(y)\}$.

Определим класс функций $\mathbf{F}_n$ следующими условиями:

1) $\forall y \in Dom(f)$ множество $M_f(y)$ является выпуклым множеством (что эквивалентно условию квазивыпуклости функции f).

2) $\forall y \in Dom(f)$ верно, что $\delta(M_f(y)) = \{x \in Dom(f) : f(x) = f(y)\}$.

Следующее утверждение показывает, что класс функций $\mathbf{F}_n$ достаточно богат и является собственным подклассом класса квазивыпуклых функций. Пусть символ $SCONV_n$ обозначает класс непрерывных, строго квазивыпуклых функций, а символ $CONV_n$ обозначает класс всех квазивыпуклых функций.

Утверждение 1. $SCONV_n \subset \mathbf{F}_n \subset CONV_n$.

Следующая лемма доказывает, что $\mathbf{F}^n \subset \mathbf{G}^n$, а это в свою очередь означает, что класс $\mathbf{G}^n$ содержит класс непрерывных, строго квазивыпуклых функций.

Утверждение 2. *Пусть* $f \in \mathbf{F}_n$, $x^{(1)}, x^{(2)}, \ldots, x^{(k)} \in Dom(f)$ *и пусть* $f(x^{(0)}) \leqslant f(x^{(1)}) \leqslant \cdots \leqslant f(x^{(k-1)}) \leqslant f(x^{(k)})$. *Тогда для любых* $y \in Dom(f)$, *таких что* $y \in x^{(k)} + cone(x^{(k)} - x^{(0)}, \ldots, x^{(k)} - x^{(k-1)})$ *верно, что* $f(y) \geqslant f(x^{(k)})$.

Доказательство утверждения 2 для более узкого класса строго квазивыпуклых функций можно найти, например, в [3].

Все утверждения работы [2] легко переносятся на существенно более общий класс $\mathbf{G}_n$. Далее сформулируем основной результат, получаемый таким образом.

Обозначим за $\tau(A, f)$ количество обращений к оракулу, затрачиваемых алгоритмом A для минимизации функции f. Обозначим за $S(n, r)$ множество функций $f \in \mathbf{G}_n$, заданных на целых точках множества $B(n, r)$, где $B(n, r) = \{x \in \mathbb{R}^n : ||x||_\infty \leqslant r\}$, для некоторого целого $r \geqslant 1$. Пусть $\tau(A, n, r) = \max\limits_{f \in S(n,r)} \tau(A, f)$ и $\tau(n, r) = \min\limits_{A} \tau(A, n, r)$. Величину $\tau(n, r)$ будем называть сложностью класса $\mathbf{G}_n$. Тогда верна следующая теорема:

Теорема 1. $\tau(n, r) \geqslant 3^{n-1} \log_2(2r - 1)$.

Симметричные функции на двумерной решетке

Величина $\tau(n, r)$ изучалась в работе [1], где была получена оценка вида $O(\log^2 r)$. В данной работе предлагается рассмотреть подкласс симметричных функций из $\mathbf{G}_2$. Обозначим сложность данного класса функций символом $\sigma(2, r)$. Верна следующая теорема.

Теорема 2. *Для целых* $r \geqslant 3$ *справедливо неравенство:*

$$2 \log_2 r - \log \log \log_2 r \leqslant \sigma(2, r) \leqslant 4,6 \log_2 r.$$

Заметим, что правое неравенство теоремы 2 строится путем предоставления оракульного алгоритма минимизации с указанной оценкой сложности.

Исследование осуществляется при поддержки грантов РФФИ № 16-31-00109-mol-a и № 15-01-06249-a и гранта Президента РФ МК-4819.2016.1, лаборатория ЛАТАС, НИУ ВШЭ.

Список литературы

[1] Чирков А. Ю. Минимизация квазивыпуклой функции на двумерной целочисленной решетке // Вестник Нижегородского университета им. Н. И. Лобачевского. Серия: Математическое моделирование и оптимальное управление. — 2003. — № 1. — С. 227–238.

[2] Золотых Н. Ю., Чирков А. Ю. Нижняя оценка сложности минимизации строго квазивыпуклой функции на целочисленной решетке // Вестник Нижегородского университета им. Н.И. Лобачевского. Серия: Математическое моделирование и оптимальное управление. — 2012. — Т. 2, № 5. — С. 93–96.

[3] Сухарев А. Г., Тимохов А. В., Федоров В. В. Курс методов оптимизации // М.: Наука, 1986.

О классах частичных монотонных функций шестизначной логики

Дудакова Ольга Сергеевна

МГУ имени М. В. Ломоносова, e-mail: olga.dudakova@gmail.com

В работе исследуются замкнутые классы частично определенных функций многозначной логики. Известно, что множество всех замкнутых классов в частичной k-значной логике имеет мощность континуума при всех $k \geqslant 2$. Поэтому представляет интерес задача об описании отдельных фрагментов решетки замкнутых классов в частичной k-значной логике. Описание замкнутых классов в частичной двузначной логике, содержащих множество P_2 всех булевых функций или какой-нибудь из предполных классов в P_2 получено в работах [1, 2]. Подобные результаты получены и для предполных классов функций k-значной логики (см., например, [3]). Однако окончательный результат не удалось получить для предполных классов функций, монотонных относительно частичного порядка, не являющегося решеткой. В настоящей работе рассматриваются классы частичных функций, монотонных относительно частично упорядоченного множества из 6 элементов с наибольшим и наименьшим элементами и двумя парами несравнимых элементов. Установлено, что существует бесконечное число классов частичных функций, содержащих предполный класс всюду определенных монотонных функций.

Обозначим чрез $\mathcal{E}$ частично упорядоченное множество $\{0, \alpha, \alpha', \beta, \beta', 1\}$ с наименьшим элементом 0, наибольшим элементом 1 и двумя парами несравнимых элементов $\alpha,\ \alpha'$ и $\beta,\ \beta'$, для которых $\alpha, \alpha' < \beta, \beta'$. Через P_6^* будем обозначать семейство всех частичных функций на $\mathcal{E}$, то есть множество отображений $\cup_{n\geqslant 1}\{f \mid f : \mathcal{E}^n \to (\mathcal{E} \cup \{*\}\}$. Через M обозначим класс всюду определенных монотонных функций на $\mathcal{E}$ (из определения частичного порядка следует, что M — предполный класс в P_6, см. [3]). Через $\widehat{M^*}$ будем обозначать класс всех частичных функций, монотонных (относительно указанного частичного порядка) на своей области определения. Через M^* будем обозначать замкнутый класс всех частичных функций из $\widehat{M^*}$, доопределяемых до функций из M. Легко видеть, что выполняются следующие соотношения:

$$M \subset M^* \subset \widehat{M^*}.$$

Обозначим через $\mathcal{I}(M, M^*)$ семейство всех замкнутых подклассов класса M^*, содержащих M, а через $\mathcal{I}(M^*, \widehat{M^*})$ семейство всех замкнутых подклассов класса $\widehat{M^*}$, содержащих M^*. Известно (см. [3]), что число замкнутых классов в интервале $\mathcal{I}(M, M^*)$ конечно. В настоящей работе устанавливается, что число классов в интервале $\mathcal{I}(M^*, \widehat{M^*})$ бесконечно.

Пусть $f(x_1, \ldots, x_n) \in P_6^*$. Пятерку наборов $\widetilde{a}, \widetilde{a}', \widetilde{b}, \widetilde{b}', \widetilde{c} \in \mathcal{E}^n$ назовем *квадратом* для f в $\mathcal{E}^n$, если выполняются следующие условия:

1) $\widetilde{a}, \widetilde{a}' < \widetilde{c} < \widetilde{b}, \widetilde{b}'$,

2) $f(\widetilde{a}) = \alpha, f(\widetilde{a}') = \alpha', f(\widetilde{b}) = \beta, f(\widetilde{b}') = \beta', f(\widetilde{c}) = *$.

Отметим, что понятие квадрата для функции f является частным случаем понятия зигзага, введенного в работе [4].

Нетрудно показать, что справедливо следующее утверждение (см. также [4]).

Утверждение 1. *Пусть $f(x_1, \ldots, x_n) \in \widehat{M^*}$. Включение $f \in M^*$ выполняется тогда и только тогда, когда в $\mathcal{E}^n$ нет квадратов для f.*

Пусть $f \in \widehat{M^*}$, $\widetilde{a}, \widetilde{a}', \widetilde{b}, \widetilde{b}', \widetilde{c}$ — квадрат для f. Последовательность наборов $\widetilde{a}_1, \ldots, \widetilde{a}_k,\ k \geqslant 3$, назовем *нижним путем в квадрате*, если выполняются следующие условия:

1) $\widetilde{a}_1 = \widetilde{a},\ \widetilde{a}_k = \widetilde{a}'$,

2) $\widetilde{a}_i$ и $\widetilde{a}_{i+1}$ сравнимы для всех $i = 1, \ldots, k-1$,

3) $f(\widetilde{a}_i) \neq *$ для всех $i = 2, \ldots, k-1$,

4) $\widetilde{a}_i < \widetilde{b}, \widetilde{b}'$ для всех $i = 2, \ldots, k-1$.

Число $k-2$ будем называть *длиной пути*.

Аналогично определим понятие *верхнего пути в квадрате* $\widetilde{a}, \widetilde{a}', \widetilde{b}, \widetilde{b}',\ \widetilde{c}$ для функции $f \in \widehat{M^*}$: верхним путем назовем последовательность наборов $\widetilde{b}_1, \ldots, \widetilde{b}_m,\ m \geqslant 3$, для которых выполняются следующие условия:

1) $\widetilde{b}_1 = \widetilde{b},\ \widetilde{b}_m = \widetilde{b}'$,

2) $\widetilde{b}_i$ и $\widetilde{b}_{i+1}$ сравнимы для всех $i = 1, \ldots, m-1$,

3) $f(\widetilde{b}_i) \neq *$ для всех $i = 2, \ldots, m-1$,

4) $\widetilde{b}_i > \widetilde{a}, \widetilde{a}'$ для всех $i = 2, \ldots, m-1$.

Определим следующие семейства функций:

F_∞: семейство всех функций $f \in \widehat{M}^*$, таких что для f нет квадратов или ни в каком квадрате для f нет нижнего пути в этом квадрате.

F_k: семейство всех функций $f \in \widehat{M}^*$, таких что для любого квадрата для f, в котором есть нижний путь, длина любого нижнего пути в этом квадрате не меньше k, $k \geqslant 1$.

G_∞: семейство всех функций $f \in \widehat{M}^*$, таких что для f нет квадратов или ни в каком квадрате для f нет верхнего пути в этом квадрате.

G_k: семейство всех функций $f \in \widehat{M}^*$, таких что для любого квадрата для f, в котором есть верхний путь, длина любого верхнего пути в этом квадрате не меньше k, $k \geqslant 1$.

Утверждение 2. *Семейства функций $F_\infty, F_1, F_2, \ldots, G_\infty, G_1, G_2, \ldots$ являются замкнутыми классами в P_6^*.*

Из определения семейств функций следует, что выполняются включения

$$M^* \subseteq F_\infty \subseteq \ldots \subseteq F_{k+1} \subseteq F_k \subseteq \ldots \subseteq F_1 = \widehat{M}^*,$$

$$M^* \subseteq G_\infty \subseteq \ldots \subseteq G_{k+1} \subseteq G_k \subseteq \ldots \subseteq G_1 = \widehat{M}^*.$$

Обозначим через $T_{i,j}$ замкнутый класс $F_i \cap G_j$, $i, j = 1, 2, \ldots, \infty$. В этих обозначениях $\widehat{M}^* = T_{1,1}$, $F_i = T_{i,1}$, $G_i = T_{1,i}$ $(i = 2, 3, \ldots, \infty)$.

Основным результатом работы является следующее утверждение.

Теорема. *Интервал $\mathcal{I}(M^*, \widehat{M}^*)$ содержит классы $T_{i,j}$ для всех значений $i = 1, 2, \ldots, \infty$ и $j = 1, 2, \ldots, \infty$. Все указанные классы различны.*

Утверждение 3. *Имеют место следующие включения:*

– $M^* \subset T_{\infty,\infty} \subset T_{\infty,i}, T_{i,\infty}$ *для всех* $i \geqslant 1$;

– $T_{\infty,i} \subset T_{\infty,i-1}, T_{m,i}$ *для всех* $i, m \geqslant 1$;

– $T_{i,\infty} \subset T_{i-1,\infty}, T_{i,m}$ *для всех* $i, m \geqslant 1$;

– $T_{ij} \subset T_{i-1,j}, T_{i,j-1}$ *для всех* $i, j \geqslant 2$;

– $T_{i,1} \subset T_{i-1,1}$; $T_{1,i} \subset T_{1,i-1}$ *для всех* $i \geqslant 1$.

Список литературы

[1] Фрейвалд Р. В. Критерий полноты для частичных функций алгебры логики и многозначных логик // ДАН СССР. — 1966. — Т. 167, № 6. — С. 1249–1250.

[2] Алексеев В. Б., Вороненко А. А. О некоторых замкнутых классах в частичной двузначной логике // Дискретная математика. — 1994. — Т. 6, вып. 4. — С. 58–79.

[3] Lau D. Function algebras on finite sets: a basic course on many-valued logic and clone theory. — Springer Monographs in Mathematics. — Berlin. Springer, 2006. — 668 p.

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

О некоторых направлениях исследований по дискретному анализу в Институте математики СО РАН

Евдокимов Александр Андреевич

Институт математики СО РАН им. С. Л. Соболева, Новосибирский государственный университет, e-mail: evdok@math.nsc.ru

В этом году Институт математики СО РАН им. С. Л. Соболева отмечает своё 60-летие. Этому событию посвящена Международная конференция «Математика в современном мире», которая пройдёт 14–19 августа 2017 г. Будет работать секция «Дискретная математика, информатика и математическая кибернетика».

Немного истории. Начало формированию в институте направления «Дискретный анализ» было положено образованием отдела теоретической кибернетики и двух лабораторий: дискретного анализа (зав. лаб. Ю. И. Журавлёв) и лаборатории теории алгоритмов с оценками (Ю. Л. Васильев). Тематика теоретических и прикладных исследований складывалась и развивалась под влиянием задач кибернетики, перспектив создания вычислительной техники и работ по оборонной тематике. Основной круг математических вопросов и задач общей направленности был очерчен в ряде публикаций А. А. Ляпунова и С. В. Яблонского в сборниках «Проблемы кибернетики». Большое значение для дальнейших исследований по направлению «Дискретный анализ» имело всестороннее изучение дискретных оптимизационных задач минимизации булевых функций, особенно их формульного представления в виде дизъюнктивных нормальных форм. Интерпретируемые как задачи покрытия заданного множества вершин n-мерного гиперкуба его подкубами, они заложили основу всестороннего исследования геометрии, комбинаторики, алгебраической структуры булевых гиперкубов и их связи с задачами теории кодирования, комбинаторными конфигурациями, алгоритмическими и сложностными вопросами представления и вычисления дискретных функций, общими вопросами теории сложности и дискретной оптимизации.

В настоящее время в исследованиях по базовому направлению «Дискретный анализ, коды, комбинаторика» принимают участие в основном коллективы двух

лабораторий: дискретного анализа (зав. лаб. А. А. Евдокимов) и совершенных комбинаторных структур (С. В. Августинович). Последняя сейчас называться лабораторией алгебраической комбинаторики. Институтом издаётся известный в математическом сообществе журнал «Дискретный анализ и исследование операций», в котором представлена значительная часть результатов по дискретному анализу. Все статьи переводятся на английский язык.

Выделим четыре направления исследований, хотя это разбиение, конечно, в значительной степени условно, поскольку исследуемые объекты, задачи и методы тесно взаимосвязаны.

1. Дискретные метрические пространства и графы. Геометрия и комбинаторика n-мерного булева и q-значного кубов. Вложения графов в гиперкубы.
2. Существование, построение и свойства кодов, дистанционно регулярных и совершенных структур, квазигрупп конечных порядков и других комбинаторных конфигураций.
3. Криптографические свойства булевых функций. Бент-функции. Комбинаторика слов и символьных последовательностей.
4. Задачи анализа, синтеза и сложности дискретных функций, включая различные модели вычисления булевых функций и дискретные математические модели функционирования генных сетей.

В докладе будет рассказано о вложениях дискретных метрических пространств и графов [1-3], результатах по криптографии [4-6] и о задачах анализа функционирования дискретных математических моделей генных сетей [7-9]. По последнему направлению с обзором задач и результатов А. А. Евдокимов выступал на XI Международном семинаре «Дискретная математика и её приложения», посвящённом 80-летию со дня рождения академика О. Б. Лупанова (МГУ, 2012, пленарный доклад «Динамические системы дискретных моделей регуляторных контуров генных сетей: анализ и сложность функционирования, восстановление структуры»). Можно посмотреть и материалы выступления на Международной конференции в Алма-Ате [7].

Перейдём к вложениям. Рассматриваемые классы отображений, определяющих вложения, это изометрические или локально изометрические отображения, вложения «с растяжением», параметрическое семейство отображений ограниченного искажения, дискретный аналог гомеоморфных вложений. Последнее предполагает введение аналога непрерывного отображения для дискретного случая, когда отображение «близкие элементы (точки метрического пространства) не разрывает, а далёкие точки не переводит в близкие». Будет дано определение этих понятий «на языке эпсилон-дельта» и топологическое определение в терминах дискретных окрестностей.

Базой рассмотрения вопросов вложения явился целый ряд решений конкретных задач вложения и кодирования структурированной информации. В част-

ности, это ставшие уже классическими задачи вложения графов в гиперкубы, задачи конструирования помехоустойчивых кодов, построение гамильтоновых циклов с различными ограничениями, различного типа вложений целочисленных решёток в гиперкубы. Эти задачи полезно интерпретировать как такие кодирования рассматриваемых объектов, которые сохраняют в образе-коде определённые структурные свойства исходных объектов (структурированное кодирование). Отметим, что при этом речь идёт о сохранении отображением, определяющем вложение, не обязательно метрических или топологических свойств, но и других свойств, например, свойств частичного порядка и алгебраических свойств [2].

По направлению 3) над задачами исследования криптографических свойств булевых функций с большим увлечением и отдачей работает группа молодых сотрудников лаборатории дискретного анализа, аспиранты и студенты [4-6]. Известно, что эта новая и важная для приложений область исследования очень популярна и интересна открывающимися всё новыми связями с различными разделами дискретной математики. Начало было положено организацией в 2011 году Н. Н. Токаревой нового семинара «Криптография и криптоанализ», рассчитанного в основном на студентов НГУ. Исследуются «сильно нелинейные» бент-функции, алгебраически иммунные булевы функции и ряд других криптографических функций, используемых в приложениях при шифровании и защите от атак по каналам связи, цифровой сотовой связи и др. Всё больше расширяются международные связи, в том числе и по реальным прикладным задачам.

В докладе будет рассказано о некоторых результатах по проблеме характеризации класса булевых функций, удовлетворяющих метрическому условию наибольшей удалённости от класса всех аффинных булевых функций на максимально возможное расстояние Хэмминга. Это свойство определяет класс бент-функций, которые представляют большой интерес для приложений. Расскажем и о некоторых других свойствах метрических пространств булевых функций [6] и гипотезах по трудной проблеме нахождения асимптотики числа бент-функций от $2n$ переменных. В целом, в этой области исследования большое число нерешенных задач, в том числе и проблемного уровня [4]. Полученные результаты уже требуют отдельного доклада. Будем надеяться, что такой доклад с обзором полученных результатов будет сделан на одной из ближайших российских Международных конференций по дискретной математике.

Работа выполнена при финансовой поддержке РФФИ (проект № 14-01-00507) и проекта РАН № 0314-2015-0011.

Список литературы

[1] Евдокимов А. А. Метрические свойства вложений и коды, сохраняющие расстояния // Модели и методы оптимизации. — Новосибирск: Наука, 1988.

Тр. АН СССР. Сибирское отделение. Институт математики. — Т. 10. — С. 116–132.

[2] Евдокимов А. А., Кодирование структурированной информации и вложения дискретных пространств // Дискретн. анализ и исслед. опер., сер. 1. — 2000. — Т. 7, № 4. — С. 48–58.

[3] Евдокимов А. А. Вложения графов в n-мерный булев куб и интервальное кодирование табло // Вестник Томского государственного университета. Приложение. — 2006. — № 17. — С. 15–19.

[4] Tokareva N. Bent functions: results and applications to cryptography — Acad. Press. Elsevier, 2016. — 220 p.

[5] Городилова А. А. От криптоанализа шифра к криптографическому свойству булевой функции // ПДМ. — 2016. — № 3 (33). — С. 16–44.

[6] Kolomeec N. A. A graph of minimal distances between bent functions // Матем. вопр. криптогр. — 2016. — Т. 7, № 2÷– С. 103–110.

[7] Евдокимов А. А. Дискретные модели генных сетей: анализ и сложность функционирования // Совместный выпуск журналов Вычислительные технологии, 13:3 и Вестник Казахского национального университета имени Аль-Фараби. Серия математика, механика, информатика, 3 (58). — 2008. — С. 31–37.

[8] Евдокимов А. А., Кочемазов С. Е., Отпущенников И. В., Семёнов А. А., Исследование дискретно-автоматных моделей генных сетей нерегулярной структуры методами символьных вычислений // Дискретн. анализ и исслед. опер. — 2014. — Т. 21, № 3. — С. 25–40.

[9] Батуева Ц. Ч.-Д. Дискретные динамические системы циркулянтного типа с пороговыми функциями в вершинах // Дискретный анализ и исследование операций. — 2014. — Т. 21, № 4. — С. 25–32.

О минимизации схем программ относительно логико-термальной эквивалентности

Жайлауова Шынар Рустембековна[1], **Захаров Владимир Анатольевич**[2]

[1] Московский государственный университет имени М.В. Ломоносова, e-mail: shomiykiy@gmail.com

[2] Московский государственный университет имени М. В. Ломоносова, e-mail: zakh_cs.msu.su

Логико-термальная (л-т) эквивалентность стандартных схем программ была введена в статье [1]. Эффективная разрешимость проблемы л-т эквивалентности, установленная в статьях [2, 3], дает возможность приступить к решению задачи минимизации — построения схемы программ π' наименьшего размера, л-т эквивалентной заданной схеме π. Чтобы отыскать ее решение, заметим, что модель вычислений стандартных схем программ сходна модели вычислений автоматов-преобразователей, работающих над полугруппами (см. [4]). В

статье [5] был предложен метод минимизации автоматов-преобразователей, работающих над упорядоченными левосократимыми полугруппами. В данной заметке мы покажем, что этим методом можно воспользоваться для минимизации стандартных схем программ относительно л-т эквивалентности в случае, когда эти схемы определены над ортогональными консервативными подстановками.

Пусть X — некоторое множество переменных, а σ — некоторая сигнатура первого порядка; над ними обычным образом определяются множество термов $Term$ и атомарных формул $Atom$. Подстановкой называется всякое отображение $\theta : X \to Term$. Множество всех подстановок с операцией композиции и тождественной подстановкой ε обозначим записью $Subst$.

Схема программ представима в виде конечной размеченной системы переходов $\pi(X) = \langle V, entry, exit, B, T\rangle$, в которых V — множество *точек программы*, $entry$ и $exit$ — точки *входа* и *выхода*, $B : V \to Atom$ — *функция разметки*, а $T : (V \setminus \{exit\}) \times \{0,1\} \to (V \setminus \{entry\}) \times Subst$ — *функция переходов*. Если $B(v) = A$, $T(v,0) = (u_0, \theta_0)$ и $T(v,1) = (u_1, \theta_1)$, то это означает, что в точке v реализован оператор ветвления **if** A **then** { θ_1; **goto** u_1 } **else** { θ_0; **goto** u_0 }. Для обозначения атома $B(v)$, приписанного точке v, будем использовать запись A_v, а для перехода из точки v будет использоваться запись $v \xrightarrow{\delta,\theta} u$, в случае $T(v,\delta) = (u,\theta)$.

Любой маршрут $tr = entry \xrightarrow{\delta_0,\theta_0} v_1 \xrightarrow{\delta_1,\theta_1} \cdots \xrightarrow{\delta_{n-1},\theta_{n-1}} v_n \xrightarrow{\delta_n,\theta_n} exit$ назовем *трассой* в программе $\pi(X)$. История трассы tr — это последовательность пар $lth(tr) = (A_{entry}, \delta_0), (A_{v_1}\eta_1, \delta_1), \ldots, (A_{v_n}\eta_n, \delta_n), (A_{exit}\eta_{n+1}, 0)$, где $\eta_i = \theta_{i-1} \cdots \theta_1\theta_0$, $1 \leqslant i \leqslant n+1$; каждая пара состоит из примера атома $A_{v_i}\eta_i$, приписанного точке v_i, и пометки δ_i дуги, по которой совершен переход. *Детерминантом* $det(\pi(X))$ программы $\pi(X)$ называется множество историй всех трасс этой программы. Две программы $\pi_1(X)$ и $\pi_2(X)$ считаются *л-т эквивалентными*, если $det(\pi_1(X)) = det(\pi_2(X))$.

Автомат-преобразователь $A = \langle V, v_{in}, V_{out}, T, h_0\rangle$ над множеством *событий* $\mathcal{C}$ и полугруппой *действий* $(S, \circ, e)$ — это система переходов, состоящая из множества *состояний* Q, *начального состояния* q_0, множества *финальных состояний* F, *функции переходов* $T : Q \times \mathcal{C} \to Q \times S$ и *инициализатора* $h_0, h_0 \in S$. Всякое слово α в алфавите $\mathcal{C}$ (включая пустое слово λ) будем называть *потоком событий*. Функцию переходов T распространим на потоки событий, полагая $T^*(q,\lambda) = (q,e)$, и $T^*(q,c\alpha) = (q'', h\circ g)$, если $T(q,c) = (q',h)$ и $T^*(q',\alpha) = (q'',g)$. Автомат A вычисляет функцию $\Phi_A : C^* \to S$, значения которая определяются так: если $T^*(q_0,\alpha) = (q,h)$ и $q \in F$, то $\Phi_A(\alpha) = h_0 \circ h$; в противном случае значение $\Phi_A(\alpha)$ неопределено. Автоматы A_1 и A_2 называются *S-эквивалентными*, если для любого потока событий α выполняется равенство $\Phi_{A_1}(\alpha) =_S \Phi_{A_2}(\alpha)$.

В статье [5] описан метод минимизации автоматов-преобразователей над полугруппой S, удовлетворяющей следующим требованиям R1–R3.

R1: $gh_1 =_S gh_2 \Rightarrow h_1 =_S h_2$ для любых g, h_1, h_2 из S.

R2: Бинарное отношение $\preceq_S$ в полугруппе S, заданное соотношением

$$h_1 \preceq_S h_2 \Leftrightarrow \exists g : h_1 g =_S h_2,$$

определяет фундированную решетку $(S, \preceq_S)$, в которой точная нижняя грань любой пары элементов эффективно вычислима.

R3: Существует алгоритм решения уравнений $hY =_S g$ в полугруппе S.

Для минимизации стандартных схем программ 1) выделим подполугруппу ортогональных консервативных подстановок $OrtConSubst$, удовлетворяющую требованиям R1–R3, а затем 2) построим сохраняющий эквивалентность транслятор схемы программ в автоматы-преобразователи. Тогда решение задачи минимизации схем программ над классом ортогональных консервативных подстановок можно решить при помощи метода минимизации автоматов-преобразователей, описанного в статье [5].

Термы t_1 и t_2 называются *ортогональными*, если ни один из них не является подтермом другого. Подстановка θ называется *ортогональной*, если для любой переменной x терм $\theta(x)$ не является основным, и для любых двух различных переменных x_1, x_2 термы $\theta(x_1)$ и $\theta(x_2)$ ортогональны. Подстановка θ называется *консервативной*, если для любой переменной x из множества X существует такая переменная y, что терм $\theta(y)$ содержит переменную x. Класс подстановок, являющихся одновременно ортогональными и консервативными, обозначим записью $OrtConSubst$.

Теорема 1. *Множество подстановок $OrtConSubst$ с операцией композиции является полугруппой, обладающей свойствами R1–R3.*

Трансляцию схем программ в автоматы-преобразователи определим следующим образом. Пусть $X = \{x_1, \ldots, x_n\}$. Воспользуемся символом $\perp$, отличным от 0 и 1, двухместным функциональным символом F, и расширим функцию переходов T схемы программ $\pi(X) = \langle V, entry, exit, B, T\rangle$, добавив для каждой точки $v, v \in V$, переход $v \xrightarrow{\perp,\eta_v} exit$, где

$$\eta_v = \{x_1/F(B(v), 1), x_2/F(B(v), 2), \ldots, x_n/F(B(v), n)\}.$$

Полученную в результате расширенную функцию переходов обозначим $\widehat{T}$. Результатом трансляции схемы программ π объявляется автомат-преобразователь $A_\pi = \langle V, entry, \{exit\}, \widehat{T}, \varepsilon\rangle$ над множеством сигналов $\{0, 1, \perp\}$ и полугруппой действий $Subst$. Заметим, что если все переходы схемы π помечены ортогональными консервативными подстановками, то автомат A_π работает над полугруппой подстановок $OrtConSubst$.

Теорема 2. *Каковы бы ни были схемы программ π_1 и π_2, они являются л-т эквивалентными в том и только том случае, когда автоматы-преобразователи A_{π_1} и A_{π_2} являются $Subst$-эквивалентными.*

Следствие. *Схема программ π над классом подстановок $OrtConSubst$ является минимальной тогда и только тогда, когда автомат A_π является минимальным.*

Работа выполнена при поддержке РФФИ (проект № 15-01-05742-а).

Список литературы

[1] Иткин В. Э. Логико-термальная эквивалентность схем программ // Кибернетика. — 1972. — № 1. — С. 5–27.

[2] Сабельфельд В. К. Полиномиальная оценка сложности распознавания логико-термальной эквивалентности // ДАН СССР. — 1979. — Т. 249, № 4. — С. 793–796.

[3] Захаров В. А., Новикова Т. А. Полиномиальный по времени алгоритм проверки логико-термальной эквивалентности программ // Труды Института системного программирования РАН – 2012. – Т. 22. – С. 435–455.

[4] Zakharov V. A. Equivalence checking problem for finite state transducers // Proceedings of the 6th International Conference on Algebraic Informatics, CAI 2015, Stuttgart, Germany, September 1-4, 2015. — Vol. 9270 of Lecture Notes in Computer Science. — 2015. — P. 208–221.

[5] Захаров В. А., Темербекова Г. Г. О минимизации конечных автоматов-преобразователей над полугруппами // Моделирование и анализ информационных систем. — 2016. — Т. 23, №. 6. — С. 741–753.

Квазиоптимальные локально-префиксные коды

Жильцова Лариса Павловна, Смирнова Татьяна Геннадьевна

Нижегородский государственный университет им. Н.И. Лобачевского, e-mail:
larisa.zhiltsova@itmm.unn.ru, tatyana.smirnova@itmm.unn.ru

В работе рассматривается модель алфавитного кодирования, учитывающего синтаксические свойства кодируемых слов языка $L \subset B^*$ (B — алфавит языка). Алфавитное кодирование для алфавита $B = \{b_1, \ldots, b_m\}$ можно задать схемой: $b_i \to v_i$, $(i = 1, \ldots, m)$, где $v_i \in A^*$ (A — алфавит канала связи). Слова v_i называются элементарными кодами, а их набор $V = (v_1, \ldots, v_m)$ — кодом. Через d_i будем обозначать длину элементарного кода буквы b_i, т. е. $d_i = |v_i|$.

Для описания синтаксиса L используется локальная модель $\mathfrak{M}(L)$ языка сообщений [1], которая представляется множеством окрестностей. Окрестностью слова α в L назовем множество

$$\varepsilon(\alpha) = \{b : b \in B, \alpha b B^* \cap L \neq \varnothing\},$$

т. е. окрестность — это множество всех букв алфавита B, которые могут быть непосредственным продолжением слова α в словах из L. Множество всех различных окрестностей в L образует локальную модель

$$\mathfrak{M}(L) = \{\varepsilon(\alpha) : \varepsilon(\alpha) \neq \oslash,\ \alpha \in B^*\} = \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_k\}.$$

Очевидно, локальная модель для любого языка L существует, но сложность ее построения существенно зависит от способа задания языка. Для регулярных языков имеется простой алгоритм построения локальной модели. В [2] показано, что в классе контекстно-свободных языков (КС-языков) проблема построения локальной модели алгоритмически неразрешима, но она становится разрешимой для его подкласса детерминированных КС-языков. Отметим, что проблема оптимального алфавитного кодирования уже в классе детерминированных КС-языков алгоритмически неразрешима. Использование локальной модели позволяет решать задачу оптимального кодирования приближенно, на подклассах кодов с конечной задержкой.

Локальной модели $\mathfrak{M}(L)$ соответствует граф антипрефиксности G, в котором вершинам поставлены в соответствие буквы алфавита B, и две вершины этого графа смежны, если они принадлежат одной окрестности локальной модели. Представим граф антипрефиксности в виде

$$G = K(\varepsilon_1) \cup K(\varepsilon_2) \cup \ldots \cup K(\varepsilon_r),$$

где $K(\varepsilon)$ — полный граф на множестве вершин ε.

Код V называется локально-префиксным относительно графа антипрефиксности G, если для любой пары букв b_i и b_j, связанных ребром, элементарный код v_i не является префиксом v_j, и наоборот.

Пусть $P = (p_1, \ldots, p_m)$, где $0 < p_i < 1$, $\sum_{i=1}^{m} p_i = 1$, — распределение вероятностей на множестве букв алфавита $B = \{b_1, \ldots, b_m\}$. Стоимость кода V определяется по формуле

$$C(V, P) = \sum_{i=1}^{m} p_i \cdot d_i.$$

Код V^*, минимизирующий $C(V, P)$, называется оптимальным для заданного распределения P.

Если граф антипрефиксности $G = K_m$ — полный граф с m вершинами, то локально-префиксный код относительно K_m является префиксным, и вопросы, связанные с оптимизацией алфавитного кодирования, эффективно разрешимы. Известен алгоритм Хаффмана, который строит оптимальный код в классе префиксных кодов.

В настоящей работе рассматривается задача оптимального кодирования в классе локально-префиксных кодов для языков, граф антипрефиксности которых состоит из двух клик.

Локально-префиксные коды, допускающие отождествление элементарных кодов букв алфавита, принадлежащих разным кликам, будем называть кодами с отождествлением.

Квазиоптимальным назовем код, оптимальный в классе локально-префиксных кодов с отождествлением.

Обозначим через R_1 и R_2 — множества букв алфавита B, образующих клики $K(R_1)$ и $K(R_2)$ графа антипрефиксности.

Положим $A = R_1 - R_2 = \{a_1, \ldots, a_{k_1}\}$, $C = R_2 - R_1 = \{c_1, \ldots, c_{k_2}\}$, $E = R_1 \cap R_2 = \{e_1, \ldots, e_{k_3}\}$ и $k_1 \geqslant k_2$.

Зададим на множестве букв алфавита B распределение вероятностей P, $\sum_{a \in B} p(a) = 1$.

Теорема 1. *Пусть* $(p(a_1), p(a_2), \ldots, p(a_{k_1}))$ *и* $(p(c_1), p(c_2), \ldots, p(c_{k_2}))$ *упорядочены по невозрастанию. Тогда существует квазиоптимальный локально-префиксный код, в котором отождествлены элементарные коды пар букв* (a_1, c_1), (a_2, c_2), $\ldots$, (a_{k_2}, c_{k_2}).

Доказательство теоремы основано на следующих леммах.

Лемма 1. *Пусть* $(p(a_1), p(a_2), \ldots, p(a_{k_1}))$ *и* $(p(c_1), p(c_2), \ldots, p(c_{k_2}))$ *упорядочены по невозрастанию. Тогда существует квазиоптимальный код, в котором* $d(a_1) \leqslant d(a_2) \leqslant \ldots \leqslant d(a_{k_1})$ *и* $d(c_1) \leqslant d(c_2) \leqslant \ldots \leqslant d(c_{k_2})$. *Здесь* $d(a)$ — *длина элементарного кода буквы* a.

Лемма 2. *Существует квазиоптимальный код со свойствами из леммы 1, в котором каждая буква* $c_j \in C$ *отождествлена с буквой из множества* A.

Лемма 3. *Существует квазиоптимальный код со свойствами из лемм 1 и 2, в котором каждая буква* $a_i \in A$ *при* $i = 1, \ldots, k_2$ *отождествлена с буквой из множества* C.

Лемма 4. *Существует квазиоптимальный код со свойствами из лемм 1, 2 и 3, в котором для любых двух пар отождествляемых букв* (a_i, c_l), (a_j, c_s) *при* $i < j$ *выполняется также условие* $l < s$.

Рассмотрим распределение $P^* = (p(a_1) + p(c_1),\ p(a_2) + p(c_2),\ \ldots,\ p(a_{k_2}) + p(c_{k_2}), p(a_{k_2+1}), \ldots, p(a_{k_1}), p(e_1), \ldots, p(e_{k_3}))$.

Теорема 2. *Алгоритм Хаффмана, примененный к* P^*, *строит квазиоптимальный локально-префиксный код при отождествлении элементарных кодов пар букв* (a_1, c_1), (a_2, c_2), $\ldots$, (a_{k_2}, c_{k_2}).

Таким образом, алгоритм построения квазиоптимального кода для графа антипрефиксности с двумя кликами состоит из *следующих этапов*:

Этап 1. Сортировка по невозрастанию вероятностей в множествах букв A и C.

Этап 2. Построение нового распределения вероятностей P^* путем сложения вероятностей $p(a_i) + p(c_i)$ для $i = 1, \ldots, k_2$ (соответствует отождествлению элементарных кодов букв a_i и c_i).

Этап 3. Для распределения вероятностей P^* применение алгоритма Хаффмана.

Оценивая трудоемкость отдельных этапов алгоритма, находим оценку $T(m)$ временной сложности построения квазиоптимального кода: $T(m) = O(m \cdot \log m)$.

Список литературы

[1] Марков Ал. А., Смирнова Т. Г. Алгоритмические основания обобщенно-префиксного кодирования // ДАН СССР. — 1984. — Т. 274, № 4. — С. 790–793.

[2] Жильцова Л. П. Об алфавитном кодировании контекстно-свободных языков // Комбинаторно-алгебраические методы в прикладной математике: Межвуз. тематич. сб. научн. трудов. — Горький. — 1983. — С. 106–123.

Асимптотически наилучший метод синтеза булевых рекурсивных схем ограниченной глубины в произвольном базисе

Жуков Владимир Владимирович

Московский государственный университет, e-mail: zhvv117@gmail.com

Введение

Задача синтеза, которая впервые была рассмотрена К. Э. Шенноном, состоит в построении оптимального метода синтеза схем из определенного класса для произвольной функции или системы функций. Для оценки оптимальности метода вводится функция Шеннона, которая для заданного n равна максимальной сложности булевой функции от n переменных. Под сложностью обычно понимается число элементов в схеме или их суммарный вес. Также можно определять функцию Шеннона для задержки.

О. Б. Лупановым [1] был предложен асимптотически наилучший метод синтеза схем в произвольных полных конечных базисах. А именно, асимптотика функции Шеннона для схем из функциональных элементов в стандартном базисе равна $2^n/n$, а в произвольном базисе Б — $\rho_{\text{Б}} \cdot 2^n/n$, где $\rho_{\text{Б}}$ — константа, зависящая от базиса. В работе [2] описан метод синтеза схем из блоков — многовыходных функциональных элементов.

В [3] рассмотрена модель рекурсивных схем из функциональных элементов и получена асимптотика функции Шеннона для данного класса схем при неограниченной глубине рекурсии. Этот результат справедлив как для скалярных рекурсивных схем, при построении которых используются только одновыходные функциональные элементы, так и для схем, построенных из многовыходных элементов. В работе [4] получены верхняя и нижняя оценки функции Шеннона для скалярных рекурсивных схем с ограничением на глубину рекурсии.

В данной статье предложен метод синтеза рекурсивных схем ограниченной глубины, построенных из многовыходных функциональных элементов, в произвольном базисе. На основе данного метода получена асимптотика функции Шеннона для данного класса схем. Случай стандартного базиса рассмотрен в работе [5].

Постановка задачи и полученные результаты

Определим рекурсивную схему Σ из функциональных элементов (РСФЭ) общего вида ограниченной глубины r в базисе Б $= \{\varepsilon_1, ..., \varepsilon_b\}$. Каждый функциональный элемент (ФЭ) базиса $\varepsilon_i \in$ Б определяется тройкой $\langle \varphi_i, L_i, T_i \rangle$, где $\varphi_i(x_1, ..., x_{k_i})$ — функция, которую реализует данный ФЭ, L_i и T_i — его сложность и задержка соответственно. Рекурсивная схема Σ определяется как последовательность схем из функциональных элементов $\Sigma_1, ..., \Sigma_r$ такая, что:

1. Схема Σ_1 — это СФЭ в базисе Б с входами $x_1^1, ..., x_{n_1}^1$ и выходами $y_1^1, ..., y_{m_1}^1$, реализующая систему функций $f_1^1(x_1^1, ..., x_{n_1}^1), ..., f_{m_1}^1(x_1^1, ..., x_{n_1}^1)$;
2. Для любого $i = 2, ..., r$ схема Σ_i — это СФЭ в базисе Б $\cup \{\xi_{i-1}\}$, реализующая систему функций систему функций $f_1^i(x_1^i, ..., x_{n_i}^i), ..., f_{m_i}^i(x_1^i, ..., x_{n_i}^i)$, где ξ_{i-1} — это функциональный элемент с n_{i-1} входами, m_{i-1} выходами и взвешенной сложности c_{i-1}, реализующий систему функций $f_1^{i-1}(x_1^{i-1}, ..., x_{n_{i-1}}^{i-1}), ..., f_{m_{i-1}}^{i-1}(x_1^{i-1}, ..., x_{n_{i-1}}^{i-1})$, то есть систему функций, которую реализует схема Σ_{i-1}.

Полученная РСФЭ Σ реализует ту же систему функций, что и последняя схема Σ_r последовательности $\Sigma_1, ..., \Sigma_r$.

Обозначим класс рекурсивных схем из функциональных элементов ограниченной глубины r в базисе Б как $\mathcal{U}_{\text{Б},r}^{\text{РСФЭ}}$.

Сложность и взвешенная сложность РСФЭ Σ глубины r, задаваемой последовательностью СФЭ $\Sigma_1, ..., \Sigma_r$, определяются через сложности и взвешеные сложности СФЭ последовательности $\Sigma_1, ..., \Sigma_r$:

$$L_{\text{Б},r}^{\text{РСФЭ}}(\Sigma) = L_{\text{Б}}^C(\Sigma_1) + ... + L_{\text{Б}}^C(\Sigma_r) \text{ — сложность РСФЭ;}$$

$$\mathcal{L}_{\text{Б},r}^{\text{РСФЭ}}(\Sigma) = \mathcal{L}_{\text{Б}}^C(\Sigma_1) + ... + \mathcal{L}_{\text{Б}}^C(\Sigma_r) \text{ — взвешенная сложность РСФЭ.}$$

Сложностью (взвешенной сложностью) $L_{\text{Б},r}^{\text{РСФЭ}}(f)$ ($\mathcal{L}_{\text{Б},r}^{\text{РСФЭ}}(f)$) функции $f(x_1, ..., x_n)$ в классе РСФЭ глубины r назовём минимальную сложность (взвешенную сложность) РСФЭ, реализующей функцию $f(x_1, ..., x_n)$. Также определим функции Шеннона $L_{\text{Б},r}^{\text{РСФЭ}}(n)$ и $\mathcal{L}_{\text{Б},r}^{\text{РСФЭ}}(n)$ в классе РСФЭ глубины r, как максимальную сложность и максимальную взешенную сложность функции от n переменных.

В данной работе получена асимптотика функции Шеннона $\mathcal{L}_{\text{Б},r}^{\text{РСФЭ}}(n)$ для РСФЭ ограниченной глубины r в произвольном базисе Б.

Теорема 1. *Функция Шеннона* $\mathcal{L}_{Б,r}^{РСФЭ}(n)$ *для класса* $\mathcal{U}_{Б,r}^{РСФЭ}$ *удовлетворяет следующему асимптотическому равенству :*

$$\mathcal{L}_{Б,r}^{РСФЭ}(n) \sim r\sqrt[r]{c_1 \cdot \ldots \cdot c_{r-1}}\sqrt[r]{\rho_Б}\frac{2^{\frac{n}{r}}}{\sqrt[r]{n}}.$$

Список литературы

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

[2] Редькин Н. П., Марковский А. В. О реализации булевых функций схемами из блоков // Проблемы кибернетики. Вып. 28. — М. : Наука. — 1974. — С. 81–100.

[3] Грибок С. В. Об одной модели рекурсивных схем из функциональных элементов // Вестн. Моск. ун-та. Сер. 15. Вычисл. матем. и киберн. — 2002. — № 4. — С. 31–36.

[4] Блинов С. В., Ложкин С. А. О синтезе рекурсивных схем из функциональных элементов с ограниченной глубиной рекурсии // Материалы XI Международного семинара «Дискретная математика и ее приложения». — М. : Издательство механико-математического факультета МГУ. — 2012. — С. 98–99.

[5] Жуков В. В. Асимптотически наилучший метод синтеза булевых рекурсивных схем ограниченной глубины // Вестн. Моск. ун-та. Сер. 15. Вычисл. матем. и киберн. — 2017. — № 3.

[6] Ложкин С. А. Лекции по основам кибернетики. — М. : Изд. отдел ф-та ВМК МГУ. — 2004.

[7] Яблонский С. В. Элементы математической кибернетики. — М. : Высшая школа. — 2007.

Об одном методе минимизации с погружением надграфиков вспомогательных функций.

Заботин Игорь Ярославич[1], **Казаева Ксения Евгеньевна**[2]

[1] Казанский федеральный университет, e-mail: iyazabotin@mail.ru

[2] Казанский федеральный университет, e-mail: k.e.kazaeva@gmail.com

Предлагается метод минимизации выпуклой недифференцируемой функции при наличии ограничений. Он основан на идеях, заложенных в методах штрафных функций (напр., [1]) и в методах отсечений (напр., [2–4]). На каждом шаге предлагаемого метода строится вспомогательная функция в виде суммы целевой функции и внешней штрафной функции области ограничений. Ее надграфик, а также допустимое множество исходной задачи погружаются в

некоторые многогранные множества, и с их использованием путем решения задачи линейного программирования находится итерационная точка. Оба следующих погружающих множества строятся отсечением от предыдущих множеств найденной точки.

Решается следующая задача:

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

где $f(x)$ — выпуклая в n-мерном евклидовом пространстве R_n функция, множество $D \subset R_n$ выпукло и замкнуто, $\operatorname{int} D \neq \varnothing$.

Пусть $f^* = \min\{f(x) : x \in D\} > -\infty$, $X^* = \{x \in D : f(x) = f^*\}$, $\operatorname{epi}(g, G) = \{(x, \gamma) \in R_{n+1} : x \in G,\ \gamma \geqslant g(x)\}$, где $G \subset R_n$, $g(x)$ — определенная в R_n функция, $W(z, Q)$ — множество нормированных обобщенно-опорных векторов для множества Q в точке z, $K = \{0, 1, \ldots\}$, $x^* \in X^*$.

Предлагаемый метод решения задачи (1) заключается в следующем. Задается последовательность $\{P_k(x)\}$, $k \in K$, выпуклых штрафных для области D функций, удовлетворяющих условиям

$$P_k(x) = 0\ \forall x \in D, \quad P_{k+1}(x) \geqslant P_k(x) > 0, \quad \lim_{k\to\infty} P_k(x) = +\infty\ \forall x \notin D, \tag{2}$$

и полагается $F_k(x) = f(x) + P_k(x)$, $k \in K$. Выбираются точки $v' \in \operatorname{int}\operatorname{epi}(f, D)$, $v'' \in \operatorname{int} D$. Строятся выпуклое ограниченное замкнутое множество $D_0 \subset R_n$, содержащее x^*, и выпуклое замкнутое множество $M_0 \subseteq R_{n+1}$, содержащее $\operatorname{epi}(F_0, R_n)$. Задается число $\bar{\gamma}$ такое, что $\bar{\gamma} \leqslant f_0^*$, где $f_0^* = \min\{f(x) : x \in D_0\}$. Полагается $k = 0$.

1. Отыскивается точка $u_k = (x_k, \gamma_k)$, где $x_k \in R_n$, $\gamma_k \in R_1$, как решение следующей задачи:

$$\gamma - \min, \quad x \in D_k, \quad (x, \gamma) \in M_k, \quad \gamma \geqslant \bar{\gamma}. \tag{3}$$

Если

$$u_k \in \operatorname{epi}(f, D), \tag{4}$$

то $x_k \in X^*$, и процесс решения завершается.

2. В интервале (v', u_k) выбирается точка $v'_k \in R_{n+1}$ так, что бы $v'_k \notin \operatorname{int}\operatorname{epi}(F_k, R_n)$ и для точки $z'_k = u_k + q'_k(v'_k - u_k)$ выполнялось включение $z'_k \in \operatorname{epi}(F_k, R_n)$ при некотором $q'_k \in [1, q]$, $q < +\infty$.

Выбирается конечное множество $A_k \subset W(v'_k, \operatorname{epi}(F_k, R_n))$ и полагается

$$M_{k+1} = M_k \cap \{u \in R_{n+1} : \langle a, u - v'_k\rangle \leqslant 0\ \forall a \in A_k\}. \tag{5}$$

3. Если $x_k \notin D$, то в интервале (v'', x_k) выбирается точка $v''_k \notin \operatorname{int} D$ такая, что $z''_k = x_k + q''_k(v''_k - x_k) \in D$ при некотором $q''_k \in [1, q]$, выбирается конечное множество $B_k \subset W(v''_k, D)$ и полагается

$$D_{k+1} = D_k \cap \{x \in R_n : \langle b, x - v''_k\rangle \leqslant 0\ \forall b \in B_k\}. \tag{6}$$

В противном случае полагается $M_{k+1} = M_k$.

4. Значение k увеличивается на единицу и следует переход к п. 1.

Сделаем некоторые замечания к методу.

Способы задания функций $P_k(x)$ из условий (2) можно найти, например, в [1]. Выбор множеств D_0, M_0 часто не представляет особого труда. Например, допустимо положить $M_0 = R_{n+1}$, и тогда за u_0 можно принять точку $(x_0, \bar{\gamma})$, где $x_0 \in D_0$.

Заметим, что точки v'_k, v''_k в пп. 2, 3 можно выбирать в виде точек пересечения отрезков $[v', u_k]$ и $[v'', x_k]$ с границами множеств $\operatorname{epi}(F_k, R_n)$ и D соответственно, считая $q'_k = q''_k = 1$, $z'_k = v'_k$, $z''_k = v''_k$.

Далее по индукции легко доказывается включение $(x^*, f^*) \in M_k$, $k \in K$. Тем самым обосновывается разрешимость задач (3) при всех $k \in K$.

Теорема 1. *Если имеет место (4), то x_k — решение задачи (1).*

Доказательство утверждения следует из условия (4) и неравенств $\gamma_k \leqslant f^*$, $k \in K$, которые вытекают из ограничений задачи (3).

Перейдем к обсуждению сходимости метода. С учетом выбора точек v''_k, множеств B_k и способа отсечений (6) по методике [2] доказывается

Лемма 1. *Любая предельная точка последовательности $\{x_k\}$, $k \in K$, принадлежит множеству D.*

Лемма 2. *Если $\{u_k\}$, $k \in K_1 \subset K$, — сходящаяся подпоследовательность последовательности $\{u_k\}$, $k \in K$, то справедливо равенство*

$$\lim_{k \in K_1} \|v'_k - u_k\| = 0.$$

Доказательство проводится по схеме обоснования аналогичной леммы 2 из [5], опираясь на способ (5) построения множеств D_{k+1} и выбор точек v'_k, z'_k.

Теорема 2. *Пусть $\{u_k\}$, $k \in K'$, — сходящаяся подпоследовательность последовательности $\{u_k\}$, $k \in K$, и $\tilde{u} = (\tilde{x}, \tilde{\gamma})$ — ее предельная точка. Тогда $\tilde{x} \in X^*$, $\tilde{\gamma} = f^*$.*

Доказательство. Выделим из последовательности $\{z'_k\}$, $k \in K'$, сходящуюся подпоследовательность $\{z'_k\}$, $k \in K'' \subset K'$, и пусть $\tilde{z}$ — ее предельная точка. Используя лемму 1 и способ задания функций $F_k(x)$, нетрудно доказать включение

$$\tilde{z} \in \operatorname{epi}(f, D). \tag{7}$$

С другой стороны, в силу леммы 2 $\lim\limits_{k \in K''} \|v'_k - u_k\| = 0$. Тогда из этого равенства, включения (7), а также равенств $z'_k = u_k + q'_k(v'_k - u_k)$, $k \in K''$, следует, что $\tilde{u} \in \operatorname{epi}(f, D)$. Отсюда имеем $\tilde{x} \in D$, $f(\tilde{x}) \leqslant \tilde{\gamma}$. Кроме того, $\tilde{\gamma} \leqslant f^*$, так как $\gamma_k \leqslant f^*$, $k \in K$. Значит, выполняются неравенства $f(\tilde{x}) \leqslant \tilde{\gamma} \leqslant f^* \leqslant f(\tilde{x})$, из которых вытекает утверждение теоремы.

Теорема 2 доказана.

Список литературы

[1] Васильев Ф. П. Методы оптимизации: в 2 кн. — М.: МЦНМО, 2011. — Кн. 1. — 620 с.

[2] Булатов В. П., Методы погружения в задачах оптимизации. — Новосибирск: Наука, 1977. — 161 с.

[3] Заботин И. Я., Яруллин Р. С. Метод отсечений на основе аппроксимации надграфика с отбрасыванием отсекающих плоскостей // Автоматика и телемеханика. -– 2015. — № 11. — С. 76–88.

[4] Заботин И. Я., Шульгина О. Н., Яруллин Р. С. Метод минимизации с аппроксимацией области ограничений и надграфика целевой функции // Изв. вузов. Математика. — 2016. — № 11. — С. 91–96.

[5] Заботин И. Я. О некоторых алгоритмах погружений-отсечений для задачи математического программирования // Известия Иркутского государственного университета. Сер. Математика. — 2011. — Т. 4, № 2. — С. 91–101.

Двухэтапный метод отсечений для условной минимизации функций

Заботин Игорь Ярославич[1], **Шульгина Оксана Николаевна**[2], **Яруллин Рашид Саматович**[3]

[1] Казанский (Приволжский) федеральный университет, e-mail: IYaZabotin@mail.ru
[2] Казанский (Приволжский) федеральный университет, e-mail: ONShul@mail.ru
[3] Казанский (Приволжский) федеральный университет, e-mail: YarullinRS@gmail.com

Один из классов методов решения задач математического программирования составляют так называемые методы отсечений (напр., [1–3]). Предлагаемый здесь метод относится к названному классу. Он использует при нахождении приближений аппроксимацию многогранными множествами как области ограничений, так и надграфика целевой функции.

Построение каждой точки основной последовательности происходит в два этапа. На первом этапе фиксируется множество, аппроксимирующее область ограничений, и на основе некоторых вспомогательных точек последовательно строятся множества, аппроксимирующие надграфик. Когда качество аппроксимации надграфика становится в определенном смысле приемлемым, первый этап завершается. На втором этапе находится основная итерационная точка, и путем ее отсечения строится очередное множество, аппроксимирующее область ограничений. Отметим, что после нахождения основной итерационной точки предусмотрена возможность обновления аппроксимирующего надграфик множества за счет отбрасывания любого числа ранее построенных секущих плоскостей.

Решается задача

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

где $D = \bigcap\limits_{j \in J = \{1,\ldots,m\}} D_j$, множества D_j, $j \in J$, из n-мерного евклидова пространства $\mathrm{R_n}$ выпуклы и замкнуты, для каждого $j \in J$ выполняется $\mathrm{int}D_j \neq \varnothing$, а $f(x)$ — выпуклая достигающая на D своего минимального значения f^* функция. Отметим, что в (1) допустимо равенство $\mathrm{int}D = \varnothing$.

Пусть $X^* = \{x \in D : f(x) = f^*\}$, $x^* \in X^*$, $\mathrm{epi}(f, \mathrm{R_n}) = \{(x, \gamma) \in \mathrm{R_{n+1}} : x \in \mathrm{R_n}, \gamma \geqslant f(x)\}$, $W(z, Q)$ — множество нормированных обобщенно-опорных в точке z для множества Q векторов, $K = \{0, 1, \ldots\}$.

Предлагаемый метод решения задачи (1) вырабатывает последовательность приближений x_k, $k \in K$, следующим образом. Выбираются точки $v^j \in \mathrm{int}D_j$, $j \in J$, и $v \in \mathrm{int\,epi}(f, \mathrm{R_n})$. Строятся выпуклое ограниченное замкнутое множество $M_0 \subset \mathrm{R_n}$ и выпуклое замкнутое множество $G_0 \subseteq \mathrm{R_{n+1}}$ такие, что $x^* \in M_0$, $\mathrm{epi}(f, \mathrm{R_n}) \subset G_0$. Задаются числа $\bar{\gamma}$, ε_k, τ_k, $k \in K$ такие, что $\bar{\gamma} \leqslant f(x)$ для всех $x \in M_0$, $\varepsilon_k > 0$, $\tau_k \geqslant 0$, $k \in K$,

$$\varepsilon_k \to 0, \quad k \to \infty, \quad \tau_k \to 0, \quad k \to \infty, \tag{2}$$

$1 \leqslant q < +\infty$. Полагается $i = 0$, $k = 0$.

1. Находится решение $u_i = (y_i, \gamma_i)$, где $y_i \in \mathrm{R_n}$, $\gamma_i \in \mathrm{R_1}$, задачи

$$\min\{\gamma : x \in M_k, (x, \gamma) \in G_i, \gamma \geqslant \bar{\gamma}\}. \tag{3}$$

Если $y_i \in D$, $f(y_i) = \gamma_i$, то $y_i \in X^*$, и процесс решения задачи (1) завершается.

2. В интервале $(v, u_i]$ отыскивается точка $\bar{u}_i \notin \mathrm{int\,epi}(f, \mathrm{R_n})$ такая, что существует точка $z_i \in \mathrm{epi}(f, \mathrm{R_n})$, удовлетворяющая неравенству $\|u_i - z_i\| \leqslant q\|u_i - \bar{u}_i\|$. Выбирается конечное множество $A_i \subset W(\bar{u}_i, \mathrm{epi}(f, \mathrm{R_n}))$.

3. Если выполняется неравенство

$$\|u_i - \bar{u}_i\| > \varepsilon_k, \tag{4}$$

то $G_{i+1} = G_i \cap \{u \in \mathrm{R_{n+1}} : \langle a, u - \bar{u}_i\rangle \leqslant 0 \ \forall a \in A_i\}$, и следует переход к п. 1 при i, увеличенном на единицу. В противном случае выполняется п. 4.

4. Выбирается точка $\tilde{y}_i \in M_k$ такая, что $f(\tilde{y}_i) \leqslant f(y_i) + \tau_k$, и полагается $i_k = i$, $x_k = \tilde{y}_i, \sigma_k = \gamma_i$,

$$G_{i+1} = G_{r_i} \cap \{u \in \mathrm{R_{n+1}} : \langle a, u - \bar{u}_i\rangle \leqslant 0 \ \forall a \in A_i\}, \tag{5}$$

где $0 \leqslant r_i \leqslant i$.

5. Если $x_k \in D$, то полагается $M_{k+1} = M_k$, и следует переход к п. 9. Иначе выполняется п. 6.

6. Формируется множество $J_k = \{j \in J : x_k \notin D_j\}$. Для каждого $j \in J_k$ в интервале (v^j, x_k) выбирается точка $\bar{x}_k^j \notin \mathrm{int}D_j$ так, чтобы существовала точка $z_k^j \in D_j$, удовлетворяющая неравенству $\|x_k - z_k^j\| \leqslant q\|x_k - \bar{x}_k^j\|$.

7. Отыскивается номер $j_k \in J_k$ из условия $\|x_k - \bar{x}_k^{j_k}\| = \max\limits_{j \in J_k} \|x_k - \bar{x}_k^j\|$.

8. Выбирается конечное множество $B_k \subset W(\bar{x}_k^{j_k}, D_{j_k})$ и полагается $M_{k+1} = M_k \cap \{x \in \mathrm{R_n} : \langle b, x - \bar{x}_k^{j_k}\rangle \leqslant 0 \ \forall b \in B_k\}$.

9. Значения i и k увеличиваются на единицу, и следует переход к п. 1.

Множества M_0, G_0 естественно выбирать многогранными. Тогда для всех i, $k \in K$ задача (3) будет задачей линейного программирования. Отметим, что она разрешима, т.к. $x^* \in M_k$, $(x^*, f^*) \in G_i$ для всех i, $k \in K$. Из очевидного неравенства $\gamma_i \leqslant f^*$, $i \in K$, следует критерий оптимальности, заложенный в п. 1 метода.

Выбор точки $\bar{u}_i$ и $\bar{x}_k^j$ в пп. 3, 6 возможен. В частности, они могут являться граничными точками множеств $\mathrm{epi}(f, \mathrm{R_n})$ и D_j соответственно. В п. 5 допустимо положить, например, $x_k = \tilde{y}_{i_k} = y_{i_k}$.

Упомянутые выше обновления аппросимирующих надграфик множеств можно проводить на итерациях $i = i_k$ следующим образом. При выполнении (4) качество аппроксимации надграфика считается неудовлетворительным, и G_{i+1} строится на основе G_i. Если же $\|u_i - \bar{u}_i\| \leqslant \varepsilon_k$, то фиксируется основная итерационная точка x_k и множество G_{i_k+1} строится в виде (5) на основе любого из множеств $G_0, \ldots, G_{i_k}$. При выборе $r_{i_k} < i_k$ и происходит отбрасывание отсекающих плоскостей. Доказано, что при каждом $k \in K$ неравенство (4) выполняется лишь для конечного числа номеров $i \in K$. Это означает, что при каждом $k \in K$ номер i_k будет зафиксирован, т.е. наступит возможность обновления множества G_{i_k+1}, и, кроме того, точка x_k будет построена.

Лемма. *Любая предельная точка последовательности* $\{x_k\}$, $k \in K$, *принадлежит* D.

Доказательство леммы проводится по схеме обоснования аналогичного утверждения из [4].

Теорема. *Если* $(\tilde{x}, \tilde{\sigma})$ *— предельная точка последовательности* $\{(x_k, \sigma_k)\}$, $k \in K$, *то* $\tilde{x} \in X^*$, $\tilde{\sigma} = f^*$.

Доказательство. Пусть подмножество номеров $K' \subset K$ таково, что последовательности $\{(x_k, \sigma_k)\}$, $\{(y_{i_k}, \sigma_k)\}$, $\{z_{i_k}\}$, $k \in K'$, являются сходящимися, и, соответственно, $(\tilde{x}, \tilde{\sigma})$, $(\tilde{y}, \tilde{\sigma})$, $\tilde{z}$ — их предельные точки. Отметим, что $\tilde{z} \in \mathrm{epi}(f, \mathrm{R_n})$ в силу замкнутости множества $\mathrm{epi}(f, \mathrm{R_n})$. Так как $\|u_{i_k} - \bar{u}_{i_k}\| \leqslant \varepsilon_k$, $k \in K$, то в силу (2) $\|u_{i_k} - z_{i_k}\| \to 0$, $k \in K'$, а значит, $(\tilde{y}, \tilde{\sigma}) \in \mathrm{epi}(f, \mathrm{R_n})$ или $f(\tilde{y}) \leqslant \tilde{\sigma}$. Но $f(x_k) \leqslant f(y_{i_k}) + \tau_k$, $k \in K$. Тогда с учетом (2) $f(\tilde{x}) \leqslant \tilde{\sigma}$, и ввиду неравенств $\sigma_k \leqslant f^*$ имеем $f(\tilde{x}) \leqslant \tilde{\sigma} \leqslant f^*$. С другой стороны, согласно лемме $\tilde{x} \in D$, т.е. $f(\tilde{x}) \geqslant f^*$. Отсюда и из двух предыдущих неравенств следует утверждение теоремы.

Список литературы

[1] Булатов В. П. Методы погружения в задачах оптимизации. — Новосибирск : Наука, 1977. — 161 с.

[2] Заботин И. Я., Яруллин Р. С. Метод отсечений на основе аппроксимации надграфика с отбрасыванием отсекающих плоскостей // Автоматика и телемеханика. — 2015. — № 2. — С. 76–88.

[3] Нестеров Ю. Е. Введение в выпуклую оптимизацию. — М. : МЦНМО, 2010. — 263 с.

[4] Заботин И. Я. О некоторых алгоритмах погружений-отсечений для задачи математического программирования // Изв. Иркутского государственного ун-та. Сер. «Математика». — 2011. — Т. 4, № 2. — С. 91–101.

Об аутентификации сообщений при помощи квантовых имитовставок на основе графов

Зиятдинов Мансур Тагирович

Казанский (Приволжский) федеральный университет, e-mail: 1mziyatd@kpfu.ru

Квантовое хэширование является одним из криптографических примитивов. В данной статье предлагается способ квантового хэширования, основанный на комбинаторных свойствах графов и основанный на нём способ имитозащиты — обеспечения целостности и аутентификации источника данных.

В этой статье мы будем использовать понятия экспандера и экстрактора, определённых, например, в [1, 2]. Нам также понадобится определение экстрактора, устойчивого к квантовому хранению, данное в [3].

Квантовая хэш-функция отображает строки длины K в квантовые регистры длины s. Если $K \gg s$, никакой атакующий не может получить много информации о входной строке по теореме Холево–Наяка [4]. Формальное определение δ-устойчивой $(K; s)$ квантовой хэш-функции дано в [5]. Равенство двух хэшей может быть проверена, например, с помощью SWAP-теста [6].

Квантовые хэш-функции на основе графов

Зафиксируем группу G с операцией $\odot$ и её характер $\chi : G \to \mathbb{C}$.

Пусть $\Gamma = (V, E) - (d, \lambda)$-экстрактор. Пометим вершины V графа Γ сообщениями (элементами группы G).

Выберем случайным образом одну вершину и случайный путь длины t из неё. Обозначим вершины в этом пути s_j.

Определение 1 (квантовая хэш-функция на основе экспандера). *Квантовая функция на основе экспандера $\Psi_{\Gamma,t}(g)$ отображает элементы G в $(\mathcal{H}^2)^{\otimes m}$:*

$$|\Psi_{\Gamma,t}(g)\rangle = \sum_{k=1}^{t} \chi(g \odot s_k)|k\rangle.$$

Теорема 1. *Для любого $\delta \in (0; 1/2)$ функция $\Psi_{\Gamma,t}$ является δ-устойчивой $(\log |G|; \log t)$ квантовой хэш-функцией, если $t > O(\delta^{-1} \log |G|)$.*

Доказательство. Граница Чернова для экспандеров [7] гласит, что для любого натурального n и любого $\gamma > 0$:

$$\Pr\left[\left|\sum_{i=1}^{n} f(x_i) - n\mathbf{E}_\pi f\right| \geqslant \gamma\right] \leqslant 4N_q \exp\left[-\left(\frac{\gamma}{||f||_\infty}\right)^2 \frac{\epsilon}{20n}\right],$$

где G — взвешенный граф с $\epsilon = 1 - \lambda$ и неравномерностью ν, случайное блуждание на G начинается в распределении q и обладает стационарным распределением π.

Применяя её для нашего случая и ограничив сверху малой вероятностью, например, $1/|G|$, получаем:

$$t \geqslant \frac{20}{(1-\lambda)\delta} \ln(4|G|) = O(\log |G|).$$

Если мы выберем случайный путь длины $t = O(\log |G|)$, с высокой вероятностью мы получим квантовую хэш-функцию. **Теорема 1 доказана.**

Квантовые имитовставки на основе графов

Определение 2 (Квантовая имитовставка). *Квантовая функция S называется (ϵ, δ)-имитовставкой в следующем случае.*

Функция S получает ключ $k \in K$ и сообщение $x \in X$ и возвращает (квантовую) метку для x: $S : K \times X \to T = (\mathcal{H}^2)^{\otimes t}$.

Метки должны быть различными для различных сообщений при одном и том же ключе:

$$\forall k \in K, \forall x \in X, \forall y \neq x : \left|\langle S(k,x)|S(k,y)\rangle\right| < \epsilon. \tag{1}$$

Потребуем также невозможность подделки:

$$\forall k \in K, k \notin \mathrm{Query}(A), (x,t) \leftarrow A(S), \Pr\left[|\langle t|S(k,x)\rangle| \geqslant \epsilon)\right] \leqslant \delta, \tag{2}$$

где A — произвольный атакующий, который может обращаться к S и $\mathrm{Query}(A)$ — множество сделанных им запросов.

Неформально, квантовая имитовставка выводит метку для сообщения. Если кто-либо меняет сообщение, оно с высокой вероятностью не будет верифицировано. Получение доступа к функции имитовставки не может помочь атакующему подделать метку для сообщения с ключом, который не был запрошен.

Теорема 2. *Пусть* $\mathrm{Ext} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *является* (k, b, ϵ) *экстрактором, устойчивым к квантовому хранилищу b и $b > r(d + \log t)$. Тогда*

$$|\Psi_{\mathrm{Ext}}(\mathsf{key}, g)\rangle = \sum_{i=1}^{t} \sum_{j=1}^{2^d} \chi(\mathrm{Ext}(g \circ \mathsf{key} \circ s_i, j))|j\rangle|i\rangle$$

является $(\epsilon; \epsilon + \epsilon^{2^s+1})$ квантовой имитовставкой, безопасной против атакующего A с доступом к r запросам к Ψ_{Ext}.

Доказательство (идея). Доказательство свойства (1) аналогично доказательству теоремы 1.

Для доказательства свойства (2) заметим, что доступ к имитовставке не даёт преимущества атакующему. В самом деле, пусть A — атакующий, который получил преимущество. Тогда мы можем различить $\mathrm{Ext}(X, U_d)$ и U_m, используя $r(\log t + d)$ кубитов. Но $r(\log t + d) < b$, в противоречии с тем, что Ext — экстрактор, устойчивый к хранению b кубит.

Следовательно, атакующий должен вернуть метку без доступа к имитовставке, что эквивалентно получению состояния, близкого к метке. Вероятность такого угадывания p является отношением объёма сферы с радиусом ϵ к объёму всего пространства: $p = c\epsilon^{2^s+1}/(c(1+\epsilon)^{2^s+1}) \leqslant \epsilon^{2^s+1}$. **Теорема 2 доказана.**

Автор выражает благодарность Ф. М. Аблаеву, А. В. Васильеву и М. Кармозино за плодотворные обсуждения.

Часть этой работы выполнена при посещении специального семестра по теории сложности (апрель–июнь 2016), организованного лабораторией им. Чебышева СПбГУ совместно с институтом Сколково и ПОМИ РАН.

Работа выполнена в соответствии с программой повышения конкурентоспособности КФУ.

Список литературы

[1] Hoory S., Linial N., Wigderson A. Expander graphs and their applications // Bulletin of the American Mathematical Society. — 2006. — V. 43, No 4. — Pp. 439–561.

[2] Shaltiel R. An introduction to randomness extractors // Lecture Notes in Computer Science. LNCS 6756. — 2011. — Pp. 21–41.

[3] Ta-Shma A. Short Seed Extractors Against Quantum Storage // Proc. ACM STOC. — 2009. — Pp. 401–408.

[4] Nayak A. V. Lower Bounds for Quantum Computation and Communication : PhD dissertation. University of California, Berkeley, 1999.

[5] Quantum Fingerprinting and Quantum Hashing. Computational and Cryptographical Aspects / F. M. Ablayev, M. F. Ablayev, A. V. Vasiliev, M. T. Ziatdinov // Baltic J. Modern Computing. — 2016. — V. 4, No 4. — Pp. 860–875.

[6] Gottesman D., Chuang I. L. Quantum Digital Signatures // arXiv preprint quant-ph/0105032. — 2001.

[7] Gillman D. A Chernoff bound for random walks on expander graphs // Proc. IEEE FOCS. — 1993. — Pp. 680–691.

Исследование операций обслуживания конфликтных потоков Пуассона по алгоритму с петлей

Зорин Андрей Владимирович

Национальный исследовательский Нижегородский государственный университет им. Н. И. Лобачевского, e-mail: andrei.zorine@itmm.unn.ru

Часто системы массового обслуживания наряду с операциями обслуживания выполняют также и операции управления конфликтными потоками требований. В настоящее время для обслуживания конфликтных потоков предложены

различные управляющие алгоритмы. Однако изучение динамики управляемого объекта для многих из них еще не удается провести в полной мере. В частности, известна проблема определения области существования установившегося режима в пространстве параметров.

Рассмотрим систему обслуживания двух конфликтных пуассоновских независимых потоков Π_1, Π_2. Интенсивность потока Π_j равна λ_j, $j = 1, 2$. Требования потока Π_j помещаются в очередь O_j неограниченной вместимости. Обслуживающее устройство имеет два состояния $\Gamma^{(1)}$ и $\Gamma^{(2)}$. В состоянии $\Gamma^{(j)}$ обслуживаются только требования из очереди O_j. Длительность пребывания T_j обслуживающего устройства в состоянии $\Gamma^{(j)}$ постоянна, после чего наступает момент управляющей операции. Управление конфликтными потоками осуществляется в классе алгоритмов с петлей. В соответствии с этим алгоритмом после состояния $\Gamma^{(1)}$ всегда осуществляется мгновенный переход в состояние $\Gamma^{(2)}$. После состояния $\Gamma^{(2)}$ новое состояние обслуживающего устройства становится $\Gamma^{(1)}$, если в момент принятия решения очередь O_1 не пуста, а в противном случае продлевается пребывание в состоянии $\Gamma^{(2)}$ еще на один такт длительностью T_2. Для задания процесса обслуживания будем использовать потоки насыщения $\Pi_1^{\text{нас}}$ и $\Pi_2^{\text{нас}}$ (см. [1]). При состоянии $\Gamma^{(j)}$ поток насыщения ℓ_j содержит фиксированное число ℓ_j требований за время T_j. Используя методы из работы [2], можно представить данную систему обслуживания конфликтных потоков в виде абстрактной управляющей системы Ляпунова—Яблонского (см. [3]) и задать на некотором вероятностном пространстве $(\Omega, \mathfrak{F}, \mathsf{P})$ многомерную стохастическую последовательность

$$\{(\Gamma_i, \varkappa_{1,i}, \varkappa_{2,i}); i = 0, 1, \ldots\}, \tag{1}$$

элементы которой имеют следующий смысл. Пусть случайные величины $\tau_0 = 0$, τ_1, τ_2, $\ldots$ образуют последовательность моментов осуществления управляющих операций. Тогда Γ_i — состояние обслуживающего устройства на промежутке $(\tau_{i-1}, \tau_i]$, $\varkappa_{j,i}$ — длина очереди O_j в момент τ_i, $i = 1, 2, \ldots$; Γ_0 — состояние прибора в момент τ_0. При принятых в настоящей работе предположениях последовательность (1) образует однородную цепь Маркова. Ее фазовое пространство имеет вид $S = \{\Gamma^{(1)}, \Gamma^{(2)}\} \times \{0, 1, \ldots\} \times \{0, 1, \ldots\}$. Положим $\theta_0 = 0$, $S_0 = \{(\Gamma^{(2)}, 0, x_2) \colon x_2 = 0, 1, \ldots\}$, и введем случайные моменты $\theta_{i+1} = \min\{k \colon k > \theta_i, (\Gamma_k, \varkappa_{1,k}, \varkappa_{2,k}) \in S \setminus S_0\}$ попадания цепи Маркова (1) в множество $S \setminus S_0$. Это марковские моменты. Следовательно, рассматривая случайные объекты $\hat{\Gamma}_i = \Gamma_{\theta_i}$, $\hat{\varkappa}_{j,i} = \varkappa_{j,\theta_i}$, получим снова марковскую последовательность

$$\{(\hat{\Gamma}_i, \hat{\varkappa}_{1,i}, \hat{\varkappa}_{2,i}) \colon i = 0, 1, \ldots\}. \tag{2}$$

Можно показать, что последовательность $\{(\hat{\Gamma}_i, \hat{\varkappa}_{1,i}) \colon i = 0, 1, \ldots\}$ также будет однородной счетной цепью Маркова.

Обозначим $Q_{1,i}(r,x) = \mathsf{P}\{\hat{\Gamma}_i = \Gamma^{(r)}, \hat{\varkappa}_{1,i} = x\}$ вероятность состояния $(\Gamma^{(r)}, x)$ на переходном процессе, $r = 1, 2$, $x = 0, 1, \ldots$ и пусть $Q_1(r,x)$ есть стационарная вероятность этого же состояния в предположении, что стационарное распределение существует.

Теорема 1. *Производящие функции*

$$\Psi_{1,i}(z;1) = \sum_{x=0}^{\infty} Q_{1,i}(r,x)z^x, \quad \Psi_{1,i}(z;2) = \sum_{x=1}^{\infty} Q_{1,i}(2,x)z^x, \quad |z| \leqslant 1$$

удовлетворяют рекуррентным по $i = 0, 1, \ldots$ *уравнениям*

$$\Psi_{1,i+1}(z;1) = z^{-\ell_1} e^{\lambda_1 T_1 (z-1)} \Psi_{1,i}(z;2) + \\ + \sum_{x=1}^{\ell_1 - 1} Q_{1,i}(2,x) \sum_{b=0}^{\ell_1 - 1 - x} \frac{(\lambda_1 T_1)^b}{b!} e^{-\lambda_1 T_1} (1 - z^{x+b-\ell_1}),$$

$$\Psi_{1,i+1}(z;2) = e^{\lambda_1 T_2 (z-1)} \Psi_{1,i}(z;1) + Q_{1,i}(1,0) e^{-\lambda_1 T_2} \cdot \frac{e^{\lambda_1 T_2 (z-1)} - 1}{1 - e^{-\lambda_1 T_2}}.$$

Методами из работы [4] доказываются следующие утверждения.

Теорема 2. *Для существования единственного стационарного распределения марковской цепи* $\{(\hat{\Gamma}_i, \hat{\varkappa}_{1,i}); i = 0, 1, \ldots\}$ *необходимо выполнение неравенства* $\lambda_1(T_1+T_2) - \ell_1 \leqslant 0$ *и достаточно выполнения неравенства* $\lambda_1(T_1+T_2) - \ell_1 < 0$.

Теорема 3. *Пусть* $z_1 = 1$, $z_2, \ldots, z_{\ell_1}$ *суть корни уравнения*

$$z^{\ell_1} - e^{\lambda_1 (T_1+T_2)(z-1)} = 0,$$

лежащие в круге $|z| \leqslant 1$. *Тогда стационарные вероятности* $Q_1(1,0)$, $Q_1(2,1)$, $Q_1(2,2)$, $\ldots$, $Q_1(2,\ell_1 - 1)$ *находятся из системы линейных алгебраических уравнений*

$$\sum_{x=1}^{\ell_1 - 1} Q_1(2,x) \sum_{b=0}^{\ell_1 - 1 - x} \frac{(\lambda_1 T_1)^b}{b!} e^{-\lambda_1 T_1} (z_\alpha^{\ell_1} - z_\alpha^{x+b}) + \\ + Q_1(1,0) e^{-\lambda_1 T_2} \cdot \frac{e^{\lambda_1 (T_1+T_2)(z_\alpha - 1)} - e^{\lambda_1 T_1 (z_\alpha - 1)}}{1 - e^{-\lambda_1 T_2}} = 0, \quad \alpha = 2, 3, \ldots, \ell_1,$$

$$\sum_{x=1}^{\ell_1 - 1} Q_1(2,x) \sum_{b=0}^{\ell_1 - 1 - x} \frac{(\lambda_1 T_1)^b}{b!} e^{-\lambda_1 T_1} (\ell_1 - x - b) + \\ + Q_1(1,0) e^{-\lambda_1 T_2} \frac{\lambda_1 T_2}{1 - e^{-\lambda_1 T_2}} = \frac{\ell_1 - \lambda_1 (T_1 + T_2)}{2}.$$

Пусть $\hat{\lambda}_2 = \lambda_2 \cdot (0{,}5 \cdot T_1 + (0{,}5 - Q_1(1,0) + Q_1(1,0)(1 - e^{-\lambda_1 T_2})^{-1}) \cdot T_2)$ и $\hat{\ell}_2 = 0{,}5 \cdot 0 + (0{,}5 - Q_1(1,0) + Q_1(1,0)(1 - e^{-\lambda_1 T_2})^{-1}) \cdot \ell_2$. Эксперименты на имитационной модели позволяют предполагать, что условием существования стационарного режима в рассматриваемой управляющей системе массового

обслуживания конфликтных потоков является одновременное выполнение неравенств $\lambda_1(T_1 + T_2) - \ell_1 < 0$, $\hat{\lambda}_2 - \hat{\ell}_2 < 0$. Учитывая, что для $Q_1(1,0)$ есть алгоритм вычисления, данные условия следует признать легко проверяемыми. В то же время, величина $Q_1(1,0)$ не выражается элементарным образом через параметры системы.

Список литературы

[1] Федоткин М. А. Оптимальное управление конфликтными потоками и маркированные точечные процессы с выделенной дискретной компонентой. I // Литовский математический сборник. — 1988. — Т. 28, № 4. — С. 784–794.

[2] Зорин А. В. Кибернетическая модель циклического управления конфликтными потоками с последействием // Ученые записки Казанского университета. Серия: Физико-математические науки. — 2014. — Т. 156, № 3. — С. 66–75.

[3] Ляпунов А. А., Яблонский С. В. Теоретические проблемы кибернетики // Проблемы кибернетики: сборник статей. — М.: Физматгиз, 1963. — С. 5–22.

[4] Федоткин М. А. Оптимальное управление конфликтными потоками и маркированные точечные процессы с выделенной дискретной компонентой. II // Литовский математический сборник. — 1989. — Т. 29, № 1. — С. 148–159.

Оптимизация управления предприятием с учетом экологической безопасности производства

Зубков Александр Федорович[1], **Гусынина Юлия Сергеевна**[2]

[1] Пензенский государственный технологический университет, e-mail: zubkova@penzgtu.ru

[2] Пензенский государственный технологический университет, e-mail: gusynina@mail.ru

В основе общественного развития и производственных отношений лежит тесное взаимодействие человека с природой. Человек в своей хозяйственной деятельности отвечает за безопасность производства, организацию оптимального использования природных ресурсов [1].

Установившийся технологический процесс производства не должен нарушать природную среду. Оптимальным условием использования природных ресурсов является минимальное значение их отходов.

В качестве условия проверки минимального значения отходов производства можно принять значение безотходности, равное 75%, при его значении, большем или равным 95%, производство является безотходным.

Основополагающим принципом малоотходности (или полного отсутствия отходов) является системная взаимосвязь всех процессов потребления ресурсов и этапов производства продукции. Системность должна присутствовать и в создании безотходных производств.

Принцип создания ресурсосберегающих производств основывается на использовании замкнутых циклов применения различных производственных ресурсов, входящих в производственную систему [2].

Безотходность связана с оптимизацией одновременно по технологическим, энергетическим, экономическим и экологическим параметрам. Количество и состав отходов определяют экологическую безопасность технологических процессов и производства в целом.

Безопасность экологии производств определим по зависимости:

$$Q_k = \sum_{j=1}^{z} \sum_{i=1}^{n} k_i^z \frac{S_i^z}{\text{ДК}_i^z}, \tag{1}$$

где
k_i^z — количество i-токсичного компонента в отходах для z-сред,
S_i^z — концентрация i-компонента в z-средах,
ДК_i^z — допустимая концентрация i-компонента в z-средах.

Токсичность i-го компонента в твердых отходах определяется по формуле:

$$k_i^{\text{т}} = \frac{m_{\text{т}} p_i}{100\alpha}, \tag{2}$$

где
$m_{\text{т}}$ — количество твердых отходов, т,
p_i — содержание i-токсичного компонента в твердых отходах,
α — выпуск продукции за производственный цикл, т.

Средняя концентрация i-токсичного компонента в газообразных отходах рассчитывается по формуле:

$$S_i^{\text{г}} = \frac{\sum S_i^k V_k}{\sum V_k}, \tag{3}$$

где V_k — общий объём токсичных выбросов.

Задача оптимизации производства приводит к оценке влияния на конечный результат рационального использования природных ресурсов в производстве с их эффективным использованием [3].

Представим модель производства. Необходимо выработать максимум продукции M_{max} при заданных потенциалах природных ресурсов $R = R_{\text{задан}}$, предполагая, что в результате технологического процесса имеются отходы R_1, часть которых R_2 идет на воспроизводство продукции, имея сумму $I_{\text{пр}}$ инвестиций проекта:

$$\begin{cases} R = R_{\text{задан}}, \\ R_1 < R, \\ 0 \leqslant R_2 \leqslant R_1, \\ M(R, R_1, R_2) \to M_{max}, \\ \sum\limits_{i=1}^{n} I_i = I_{\text{пр}}. \end{cases} \tag{4}$$

Количественные оценки модели формируются на основе статистического материала о стоимости ресурсов [4], технологии производства, качестве управления, профессиональной компетенции персонала, увеличении капитальных вложений на i-стадии (I_i). Инвестирование осуществляется нарастающим итогом вплоть до конечной стадии природно-сырьевого-продуктового процесса.

Проведенное исследование показывает, что:

1) комплексная оценка производств позволяет принимать проекты с учетом требований экологической безопасности;

2) своевременное использование инвестиционных средств, их оптимизация является основным условием формирования экологической составляющей производства.

Список литературы

[1] Логвина О. А., Алехина М. А. Оценка значимости воздействия в математическом моделировании экологического состояния объекта // XXI век: итоги прошлого и проблемы настоящего плюс. — 2016. — № 2 (30). — С. 217–221.

[2] Шорникова Т. А. Обобщенная модель формирования хозяйственной политики // Обозрение прикладной и промышленной математики. — 2006. — Т. 13. № 1. — С. 164–165.

[3] Зубков А. Ф., Гусынина Ю. С., Наумов Р. В. Выбор предприятия в системе контрактации заказов // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Экономические науки. — 2008. — № 4 (61). — С. 53–56.

[4] Зубков А. Ф., Бармин М. А., Гусынина Ю. С. Моделирование получения заказа предприятием на конкурсной основе // XXI век: итоги прошлого и проблемы настоящего плюс. — 2015. — № 1 (23). — С. 228–230.

Исследование систем уравнений над обыкновенными графами

Ильев Артем Викторович

Омский государственный технический университет
Институт математики им. С. Л. Соболева СО РАН (Омский филиал), e-mail: artyom_iljev@mail.ru

В монографии Э. Ю. Данияровой, А. Г. Мясникова и В. Н. Ремесленникова «Алгебраическая геометрия над алгебраическими системами» [1] для произвольной алгебраической системы $\mathcal{A}$ языка L доказана универсальная теорема о совместности системы уравнений S над $\mathcal{A}$, и даны общие неалгоритмические процедуры вычисления радикала системы S и построения общего решения системы S, называемого *координатной алгеброй*. При этом язык L тоже является произвольным, т. е. может состоять из любых алгебраических операций, предикатных символов и констант. Изучение решений уравнений и систем уравнений

в фиксированной алгебраической системе $\mathcal{A}$, а также решение следующих отсюда задач называется *алгебраической геометрией над* $\mathcal{A}$.

В рамках совместной работы с В. Н. Ремесленниковым мы рассматриваем в качестве L язык, состоящий из множества C констант, а также иррефлексивного и симметричного предиката смежности $E(x, y)$ и предиката равенства $(x = y)$. Эти предикаты определяют категорию *обыкновенных графов*, т. е. графов без петель и кратных ребер. Данный выбор позволяет превратить общие процедуры из [1] в достаточно хорошие алгоритмические процедуры. Нами были построены алгоритмы для трех классов систем уравнений над произвольными конечными обыкновенными графами:

1) для бескоэффициентных систем уравнений;

2) для систем уравнений с одной переменной;

3) для произвольных систем уравнений *диофантовых языков*, т. е. таких языков, в которых множество констант совпадает с множеством вершин графа.

Основные определения

Так как многие понятия, изложенные выше, пока не являются общепринятыми, то ниже будут приведены формулировки основных определений из монографии [1]. Мы будем рассматривать только предикатные системы, так как язык L обыкновенных графов не содержит функциональных символов и, кроме того, состоит только из одного предиката помимо предиката равенства: $\mathrm{L} = \{E(x, y) \cup (x = y) \cup \mathrm{C}\}$. Поэтому мы адаптируем все определения под предикатный случай.

Пусть X — конечное множество переменных. Множество $T_{\mathrm{L}}(X)$ *термов* языка L от переменных из множества X состоит из всех переменных $x \in X$ и всех константных символов языка L. Множество $At_{\mathrm{L}}(X)$ *атомарных формул* языка L от переменных из множества X состоит из всех формул вида $t_1 = t_2$ и $E(t_1, t_2)$, где $t_1, t_2 \in T_{\mathrm{L}}(X)$. Атомарные форулы называются *уравнениями языка*, а произвольные подмножества $S \subseteq At_{\mathrm{L}}(X)$ — *системами уравнений* языка L. Мы будем рассматривать только конечные системы уравнений.

Любая система уравнений S от k переменных над фиксированным обыкновенным графом $\Gamma = \langle V, \mathrm{L}\rangle$ определяет в аффинном k-мерном пространстве V^k множество своих решений $V_\Gamma(S) = \{\overline{v} \in V^k \mid \Gamma \models S(\overline{v})\}$, которое называется *алгебраическим множеством*. Если $V_\Gamma(S) = \varnothing$, то система уравнений S *несовместна* над Γ; иначе она является *совместной*. Две системы уравнений S_1 и S_2 называются *эквивалентными* над Γ, если $V_\Gamma(S_1) = V_\Gamma(S_2)$. Для любой системы уравнений S над Γ существует единственная эквивалентная ей максимальная система уравнений над Γ, которая называется *радикалом* системы S и обозначается $\mathrm{Rad}_\Gamma(S)$. Если система S несовместна над Γ, то $\mathrm{Rad}_\Gamma(S) = At_{\mathrm{L}}(X)$.

В нашей работе предложены алгоритмы решения следующих задач.

1. Проверка системы уравнений S на совместность.

2. Вычисление радикала системы S.

3. Построение координатного графа системы S.

Отметим, что радикал фиксированной системы уравнений S над графом Γ от переменных из фиксированного множества X однозначно определяет координатный граф $CG_\Gamma(S)$. Его подробное определение приведено в [1]. Роль координатного графа системы S аналогична роли общего решения системы линейных уравнений над полем в линейной алгебре.

Построение информационной базы системы уравнений

Понятие информационной базы системы уравнений играет определяющую роль в наших алгоритмах, которые в данных тезисах будут приведены лишь в виде кратких схем.

Пусть $\Gamma = (V(\Gamma), E(\Gamma))$ — конечный обыкновенный граф и $S(X)$ — конечная система уравнений над Γ. *Информационная база* системы $S(X)$ состоит из набора конечных множеств и натуральных чисел, определяемых по группам:

1) $X = \{x_1, ..., x_k\}$ — множество переменных, $S(X) = \{s_1, ..., s_l\}$ — множество уравнений над X; k, l — числовые параметры.

2) $W_1, ..., W_k$ — подмножества $V(\Gamma)$. W_i состоит из вершин графа Γ, которые содержатся в записи уравнений вида $E(x_i, v)$ из системы $S(X)$; $\alpha_i = |W_i|$ — числовые параметры, где $i = 1, ..., k$.

3) $W_1^\perp, ..., W_k^\perp$ — подмножества $V(\Gamma)$. $W_i^\perp$ состоит из вершин графа Γ, которые смежны с каждой из вершин множества W_i; $\beta_i = |W_i^\perp|$ — числовые параметры, где $i = 1, ..., k$.

4) $W_1^{\perp\perp}, ..., W_k^{\perp\perp}$ — подмножества $V(\Gamma)$. $W_i^{\perp\perp} = (W_i^\perp)^\perp$; $\gamma_i = |W_i^{\perp\perp}|$ — числовые параметры, причем $\gamma_i \geqslant \alpha_i$ для любых $i = 1, ..., k$.

Проверка совместности произвольной системы $S(X)$

Один из важных шагов алгоритма проверки совместности системы уравнений $S(X)$ состоит в определении классов эквивалентности $Y(t_i)$ на $T_{\mathrm{L}}(X)$. В начале работы каждое множество $Y(t_i)$ состоит только из одного терма t_i. Алгоритм строит классы эквивалентности и преобразует систему уравнений $S(X)$ в эквивалентную $\overline{S}(X)$ над графом Γ, не содержащую уравнений вида $t_i = t_j$. Также в процессе работы происходит доопределение множеств $W_1^\perp, ..., W_k^\perp$. В результате алгоритм отвечает на вопрос, совместна ли система $S(X)$ над графом Γ, для любой системы уравнений $S(X)$.

Построение радикала

Если система $S(X)$ оказалась совместной, то процедура рассматривает множества $W_1^\perp, ..., W_{\underline{k}}^\perp$, полученные при выполнении алгоритма проверки совместности системы $\overline{S}(X)$. С их помощью заново определяются множества $W_1^{\perp\perp}, ..., W_k^{\perp\perp}$.

Далее радикал $\mathrm{Rad}_\Gamma(S)$ строится следующим образом.

1) К системе $\overline{S}(X)$ добавляются все уравнения $E(x_i, v_j)$, где $v_j \in W_i^{\perp\perp}$.

2) К полученному множеству атомарных формул добавляются все равенства, следущие из классов эквивалентности $Y(t)$: если $t_i, t_j \in Y(t)$, то $t_i = t_j \in \mathrm{Rad}_\Gamma(S)$.

3) Данное множество атомарных формул дополняется всеми следствиями из него.

Построение координатного графа

Множество вершин координатного графа $\Delta = CG_\Gamma(S(X))$ соответствует множеству классов эквивалентности $Y(t_i)$ и является подмножеством $V(\Gamma) \cup X$, а множество его ребер выглядит следующим образом:

$$E(\Delta) = \{E(\Gamma) \cup E(x_i, x_j) \cup E(x_i, v_m), \text{где } E(x_i, x_j), E(x_i, v_m) \in \mathrm{Rad}_\Gamma(S)\}.$$

Работа выполнена при поддержке РНФ (проект № 17-11-01117).

Список литературы

[1] Э. Ю. Даниярова, А. Г. Мясников, В. Н. Ремесленников. Алгебраическая геометрия над алгебраическими системами. — Новосибирск : Издательство СО РАН, 2016. — 243 с.

Клонирование графов

Иорданский Михаил Анатольевич

Нижегородский государственный педагогический университет им. К. Минина, e-mail: iordanski@mail.ru

Рассматриваются конечные, неориентированные связные графы. Используется конструктор графов [1] с одноместными (унарными) операциями над графами. Применяются операции преобразования графов двух видов. При выполнении одних операций к графу добавляются копии (клоны) некоторых его подграфов, а при выполнении других операций производится удаление клонов.

В работе изучаются условия сохранения некоторых характеристических свойств графов при выполнении над ними операций клонирования и удаления клонов.

Определения и обозначения

Пусть $G'(V', E') -$ связный подграф графа $G(V, E)$, $|V'| < |V|$. Подмножество $V_1 \subset V$ такое, что все вершины $v_i \in V_1$ смежны с вершинами из V', но $v_i \notin V'$, образует *окрестность первого порядка* подграфа G'. Добавляемый к графу G подграф $G''(V'', E'') \cong G'(V', E')$ с такой же окрестностью первого порядка, что и у подграфа G', называется *клоном* подграфа G'. При этом вершины клона G'' соединяются с множеством вершин V_1 окрестности первого порядка подграфа G' так, чтобы подграфы $G(V_1 \cup V')$ и $G(V_1 \cup V'')$ были изоморфны. Операция добавления к графу G клона G'' подграфа G' называется *операцией клонирования* подграфа G'. Обратной к ней является *операция удаления клона* G'' подграфа G' из графа G. Из определения введенных операций следует,

что исходный граф операции клонирования изоморфен подграфу результирующего графа, а при выполнении операции удаления клона результирующий граф изоморфен подграфу исходного графа. Рассмотрим ограничения на операции клонирования и удаления клона, обеспечивающие сохранение некоторых характеристических свойств графов

Планарные графы

Граф называется планарным, если он допускает геометрическую реализацию на плоскости. Для сохранения свойства планарности при выполнении операции клонирования достаточно следующего ограничения.

Лемма 1. *Операция клонирования подграфа $G' \subset G$ сохраняет планарность графа, если все вершины окрестности первого порядка клонируемого подграфа G' принадлежат границе некоторой грани f в плоской укладке графа G.*

Поскольку при выполнении операции удаления клона результирующий граф изоморфен подграфу исходного графа, то справедлива

Лемма 2. *Любая операция удаления клона сохраняет планарность графа.*

Эйлеровы графы

Характеристическим свойством эйлеровых графов является отсутствие вершин нечетной степени. Для сохранения этого свойства при выполнении операций клонирования и удаления клонов достаточно выполнения следующего ограничения.

Лемма 3. *Операция клонирования (удаления клона) сохраняет эйлеровость графа, если каждая вершина окрестности первого порядка соединяется четным числом ребер с вершинами клонируемого подграфа (клона).*

Гамильтоновы графы

Характеристическим свойством гамильтоновых графов является наличие цикла, содержащего все ребра графа. Для сохранения свойства гамильтоновости достаточно ограничить операции клонирования и удаления клонов следующим образом.

Лемма 4. *Операция клонирования подграфа $G' \subset G$ сохраняет гамильтоновость графа, если подграф $G(V \setminus V') \cong K_2$ и ребро $e \in K_2$ входит в гамильтонов цикл графа G.*

Лемма 5. *Операция удаления клона G'' подграфа $G' \subset G$ сохраняет гамильтоновость графа, если:*

1.$G = G' \cup G''$, $G' \cap G'' = K_2$;

2. Подграфы G' и G'' гамильтоновы и общее ребро принадлежит их гамильтоновым циклам.

Двудольные графы

Характеристическим свойством двудольных графов является отсутствие циклов нечетной длины. Для двудольных графов справедливо следующее утверждение.

Лемма 6. *Любые операции клонирования и удаления клона сохраняют двудольность графа.*

Список литературы

[1] Иорданский М. А. Конструктивная теория графов и ее приложения. — Н.Новгород : «Кириллица», 2016. — 172 с.

Алгоритм минимизации частично заданных булевых функций

Казимиров Алексей Сергеевич[1], **Реймеров Сергей Юрьевич**[2]

[1] Иркутский государственный университет, e-mail: a.kazimirov@gmail.com
[2] Иркутский государственный университет, e-mail: sergeyreym@gmail.com

Рассматривается задача нахождения минимальных полиномов частично заданных булевых функций. Для решения данной задачи предлагается генетический алгоритм приближенной минимизации, основанный на спуске к функциям пяти переменных с последующим их доопределением.

Назовем тотальными булевы функции, определенные на всех наборах, и частично заданными — функции, определенные на некотором подмножестве всех двоичных наборов. Частично заданные функции часто возникают на практике в задачах логического синтеза, когда имеет значение поведение функции только на определенных наборах, например, при реализации конечных автоматов.

Полиномиальным представлением булевой функции называется ее представление в виде суммы по модулю 2:

$$f(x_1, \ldots, x_n) = K_i \oplus \ldots \oplus K_s,$$

где K_i — произведение переменных или их отрицаний.

Под сложностью полиномиального представления понимается количество слагаемых в этом представлении. Сложность булевой функции g определяется как количество слагаемых в наименьшем представлении, реализующем данную функцию. Булева функция g является доопределением частично заданной функции f ($g \prec f$), если на множестве определения f значения функций f и g совпадают. Поскольку на наборах, на которых функция f не определена, функция g может принимать любые значения, то для частично заданной функции в общем случае существует несколько различных доопределений. В такой ситуации естественно под сложностью частично заданной функции понимать наименьшую из сложностей ее доопределений: $L(f) = \min_{g \prec f} L(g)$.

Задача минимизации как для тотальных, так и для частично заданных булевых функций заключается в нахождении минимального полинома, реализующего данную функцию. Полином реализует частично заданную функцию, если он реализует одно из ее доопределений. В настоящее время существует алгоритм нахождения минимальных полиномов полностью заданных булевых

функций шести переменных [1], использующий библиотеку представителей классов N-эквивалентности функций пяти переменных. В [2] предложен алгоритм минимизации тотальных функций шести и семи переменных сложности не более 16, основанный на ограниченном переборе функций меньшей размерности. В [3–5] рассматриваются генетические алгоритмы минимизации тотальных булевых функций.

Для минимизации частично заданных булевых функций предлагается следующий генетический алгоритм. Частично заданная функция n аргументов представляется в виде

$$f(x_1, \ldots, x_n) = f_1(x_1, \ldots, x_{n-1}) \oplus x_n f_2(x_1, \ldots, x_{n-1}) \oplus \bar{x}_n f_3(x_1, \ldots, x_{n-1}).$$

Функция f_1 выбирается как произвольная тотальная функция от $(n-1)$ аргумента с помощью генетического алгоритма. Функции f_2 и f_3 находятся из соотношений

$$f_2(x_1, \ldots, x_{n-1}) = f_1(x_1, \ldots, x_{n-1}) \oplus f(x_1, \ldots, x_{n-1}, 1),$$

$$f_3(x_1, \ldots, x_{n-1}) = f_1(x_1, \ldots, x_{n-1}) \oplus f(x_1, \ldots, x_{n-1}, 0).$$

Каждая из функций f_1, f_2, f_3 рекурсивно минимизируется тем же алгоритмом путем разложения на три функции меньшего числа аргументов. Тотальные функции 5 аргументов минимизируются алгоритмом точной минимизации. Для частично заданных функций 5 аргументов с помощью генетического алгоритма выбирается доопределение, дающее минимальную сложность.

Работа выполнена при поддержке РФФИ (проект № 16-31-00280-мол_а).

Список литературы

[1] Gaidukov A. Algorithm to derive minimum ESOP for 6-variable function // 5th International Workshop on Boolean Problems. — Freiberg, Germany, 2002. — P. 141—148.

[2] Sasao T. EXMIN2: A simplification algorithm for exclusive-OR sum-of-products expressions for multiple-valued-input two-valued-output functions // IEEE Trans. Comput.-Aided Des. Integrated Circuits and Syst. — 1993. — V. 12, N. 5. — P. 621–632.

[3] Винокуров С. Ф., Казимиров А. С. Параллельные генетические алгоритмы в задачах минимизации булевых функций // Вестник ТГУ. Приложение. — 2006. — № 17. — С. 226–230.

[4] Винокуров С. Ф., Казимиров А. С. Генетический алгоритм поиска минимальных полиномов булевых функций // Дискретная математика и ее приложения: Материалы X Международного семинара. — М.: Механико-математический факультет МГУ, 2010. — С. 175–177.

[5] Kazimirov A. S., Reymerov S. Yu. On genetic algorithms and neural networks for boolean functions minimization // Proceedings of the XIX International

Conference on Soft Computing and Measurement, SCM 2016, May 25–27, 2016. — P. 260–261.

Управление обслуживанием потока объектов в системе с двумя накопительно-расходными компонентами

Коган Дмитрий Израилевич[1], **Митрошина Анастасия Сергеевна**[2], **Пудов Андрей Семенович**[3], **Федосенко Юрий Семенович**[4]

[1] Московский технологический университет, e-mail: kdi_41@mail.ru
[2] Волжский государственный университет водного транспорта, e-mail: anastasia.kuimova@gmail.com
[3] Волжский государственный университет водного транспорта, e-mail: andrey@andreypudov.com
[4] Волжский государственный университет водного транспорта, e-mail: fds@vgavt-nn.ru

Рассматривается модель управления очередностью однопроцессорного обслуживания конечного детерминированного потока объектов в системе с двумя накопительно-расходными компонентами. Формулируется оптимизационная задача синтеза стратегий обслуживания и предлагается основанный на концепции динамического программирования алгоритм синтеза оптимальных стратегий обслуживания. Модель адекватно описывает процесс управления грузовой обработкой танкерного флота в условиях Северного завоза [1].

1. Рассматривается детерминированный поток $O_n = \{o_1, o_2, ..., o_n\}$ объектов, подлежащих однофазному обслуживанию стационарным процессором P. Процессор оснащен двумя независимыми накопительно-расходными компонентами (резервуарами): компонент Q_1 предназначен для временного хранения жидкого продукта Π_1, компонент Q_2 предназначен для временного хранения жидкого продукта Π_2. Нормативный объем компонента Q_1 равен V_1^*, в начальный момент времени $t = 0$ заполнение Q_1 равно $V_1(0)$. Нормативный объем компонента Q_2 равен V_2^*, в начальный момент времени заполнение Q_2 равно $V_2(0)$.

Для каждого объекта o_i, $i = \overline{1, n}$, определены целочисленные параметры: t_i — момент поступления в очередь на обслуживание, τ_i — норма длительности обслуживания, a_i — штраф за единицу времени пребывания в системе обслуживания, d_i — мягкий директивный срок завершения обслуживания ($d_i \geqslant t_i + \tau_i$), v_i — объемная характеристика. Объекты пронумерованы в порядке их поступления в очередь на обслуживание, т. е. $0 \leqslant t_1 \leqslant \ldots \leqslant t_n$. Поток O_n можно считать состоящим из четырех независимых подпотоков O_1^+, O_1^-, O_2^+, O_2^-, подлежащих обслуживанию процессором P. Объекты подпотока O_1^+ загружены продуктом Π_1, который при обслуживании процессором P должен быть перемещен в компонент Q_1; объекты подпотока O_2^+ загружены продуктом Π_2, который при обслуживании процессором P должен быть перемещен в компонент Q_2; объекты подпотоков O_1^- и O_2^- поступают для обслуживания порожними и процессор P обеспечивает их загрузку соответственно продуктом Π_1, и Π_2 путем пере-

качки из компонентов Q_1 и Q_2. Подпотоки O_1^+, O_1^-, O_2^+, O_2^- удовлетворяют условию $O_1^+ \cup O_1^- \cup O_2^+ \cup O_2^- = O_n$ и попарно не пересекаются. Принадлежность объекта o_i тому или иному подпотоку характеризуется параметром w_i: $w_i = +1$, если $o_i \in O_1^+$; $w_i = -1$, если $o_i \in O_1^-$; $w_i = +2$, если $o_i \in O_2^+$; $w_i = -2$, если $o_i \in O_2^-$.

В результате обслуживания очередного объекта o_i из подпотока O_1^+ (O_2^+) заполнение соответствующего компонента Q_1 (Q_2) увеличивается на величину v_i. По завершению обслуживания очередного объекта o_i из подпотока O_1^- (O_2^-) заполнение соответствующего компонента Q_1 (Q_2) уменьшается на величину v_i. Обслуживание очередного объекта из подпотока O_1^+ (O_2^+) может начаться при наличии достаточного свободного объема в соответствующем компоненте Q_1 (Q_2). Объект подпотока O_1^- (O_2^-) может быть принят процессором P на обслуживание при наличии достаточного количества продукта в Q_1 (Q_2). Обслуживание каждого объекта осуществляется без прерываний; необслуженный объект не может покинуть очередь; непроизводительные простои процессора не предусмотрены; одновременное обслуживание процессором двух и более объектов запрещено. Стратегия S обслуживания объектов потока O_n представляет собой произвольную перестановку $S = \{i_1, i_2, \ldots, i_n\}$ совокупности индексов $N = \{1, 2, \ldots, n\}$; при её реализации объект с индексом i_k обслуживается k-м по очереди, $k = \overline{1,n}$. Стратегию S именуем допустимой, если удовлетворяются отмеченные выше объемные ограничения на обслуживание объектов o_i, $i = \overline{1,n}$. Обозначим через Ω множество допустимых стратегий. Очевидно, что необходимыми условиями непустоты множества Ω является выполнение неравенств:

$$0 \leqslant V_1(0) + \sum_{i:\{w_i=\{+1,-1\}\}} sign(w_i)v_i \leqslant V_1^*,$$

$$0 \leqslant V_2(0) + \sum_{i:\{w_i=\{+2,-2\}\}} sign(w_i)v_i \leqslant V_2^*.$$

Для известной стратегии S арифметически вычисляются обозначаемые через $t^*(i(k), S)$ и $t^{**}(i(k), S)$ соответственно значения моментов начала и завершения обслуживания каждого объекта с индексом $i(k)$, $i = \overline{1,n}$.

2. На практике качество стратегии S в зависимости от складывающейся эксплуатационной ситуации оценивается по значению критерия $K_1(S)$ или $K_2(S)$. При этом критерий $K_1(S)$ представляет собой суммарный штраф по объектам потока O_n за время пребывания в системе обслуживания; критерий $K_2(S)$ оценивает максимальное по продолжительности нарушение директивного срока завершения обслуживания среди всех объектов потока O_n. С учетом введенных выше обозначений, предлагаемые критерии определяются следующим образом:

$$K_1(S) = \sum_{k=1}^{n} a_{i(k)}(t^*(i(k), S) - t_{i(k)}),$$
$$K_2(S) = \max_{1 \leqslant k \leqslant n} (t^{**}(i(k), S) - d_{i(k)}, 0).$$

Изучаемые в данной работе однокритериальные задачи записываются в виде:

$$\min_{S \in \Omega} K_1(S),$$
$$\min_{S \in \Omega} K_2(S).$$

Обе задачи относятся к числу NP-трудных [2]. Для их решения в работе разрабатываются алгоритмы, основанные на концепции дискретного динамического программирования [3, 4, 5].

Работа выполнена при поддержке РФФИ (проект № 15-07-03141).

Список литературы

[1] Северный завоз / Материал из Википедии – свободной энциклопедии. URL : http://ru.wikipedia.org/wiki/Северный_завоз (дата обращения : 10.03.17).

[2] Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М. : Мир, 1982. — 416 с.

[3] Коган Д. И., Куимова А. С., Федосенко Ю. С. Задачи обслуживания бинарного потока объектов в системе с накопительно-расходным компонентом // Автоматика и телемеханика. — 2014. № 7. — С. 122–135.

[4] Беллман Р., Дрейфус С. Прикладные задачи динамического программирования. — М. : Наука, 1965. — 457 с.

[5] Коган Д. И., Федосенко Ю. С. Общая схема реализации алгоритмов динамического программирования в задачах синтеза стратегий однопроцессорного обслуживания потока объектов // Сб. «Информатика и технологии. Инновационные технологии в промышленности и информатике». — М. : МИРЭА, — 2016. — С. 154–157.

Управление двухстадийным обслуживанием конечного детерминированного потока объектов

Коган Дмитрий Израилевич, Ульянов Кирилл Станиславович[1], Федосенко Юрий Семенович[2]

[1] Московский технологический университет, e-mail: kdi_41@mail.ru, kirik516@mail.ru
[2] Волжский государственный университет водного транспорта, e-mail: fds@vgavt-nn.ru

Формулируются задачи синтеза оптимальных дисциплин обслуживания, актуальные для процессов оперативного управления логистическими процессами. Алгоритмы решения задач такого типа конструировались авторами в рамках моделей снабжения дизельным топливом группировки добычных земснарядов [1–4] и предприятий Заполярья [5] в условиях Северного завоза [6]). Исследуемая проблема актуализировалась в связи с созданием математического обеспечения специализированных компьютерных средств поддержки управления диспетчеризацией логистических процессов на внутреннем водном транспорте.

1. Рассматривается следующая математическая модель. Процессор H должен выполнить обслуживание совокупности $O_n = \{1, 2, \ldots, n\}$ поступающих в дискретном времени объектов. Одновременное обслуживание нескольких (более одного) объектов невозможно. Обслуживание каждого объекта реализуется без прерываний. Переналадки процессора не предусмотрены. Полагается, что процессор H готов к обслуживанию объектов совокупности O_n, начиная от момента времени $t = 0$. Для каждого объекта i известны: $t(i)$ — момент поступления (готовности к обслуживанию) и $\tau(i)$ — норма продолжительности обслуживания процессором; считается, что $0 = t(1) \leqslant t(2) \leqslant \ldots \leqslant t(n)$. Объекты, прошедшие обслуживание, далее направляются потребителям, составляющим совокупность $P = \{p_1, p_2, \ldots, p_n\}$. В адрес каждого потребителя должен быть направлен ровно один объект (транспортное средство), каждому объекту должен быть назначен ровно один потребитель. Считается известной $(n\times n)$-матрица $V = \{v(i, j)\}$, где $v(i, j)$ — норма длительности перемещения объекта i от процессора H к потребителю p_j и последующего за этим обслуживание (разгрузки; в случае непригодности объекта i для доставки груза потребителю p_j полагаем $v(i, j) = +\infty$). Считается, что матрица V такова, что каждому потребителю можно взаимно однозначно предписать транспортное средство, способное его обслужить. Для каждого потребителя p_j задана монотонно возрастающая функция индивидуального штрафа $\psi_j(t)$. Если разгрузка прибывшего потребителю p_j транспортного средства заканчивается в момент времени t, то $\psi_j(t)$ — величина индивидуального штрафа (потерь) по данному потребителю, $j = \overline{1, n}$. Все параметры модели считаем принимающими цело-

численные значения. Стратегию S обслуживания процессором H совокупности O_n определим как пару $(\rho; r)$, где $\rho = (i_1, i_2, \ldots, i_n)$ — перестановка элементов множества $\{1, 2, \ldots, n\}$, а r — взаимно однозначное отображение множества $\{1, 2, \ldots, n\}$ в себя. При реализации стратегии S объект i_k в первой компоненте стратегии обслуживается k-м по очереди, $k = \overline{1, n}$, $r(j)$ — индекс потребителя, которому объект j предназначается, $j = \overline{1, n}$. Реализации стратегий считаем компактными [7]. В таком случае при любой стратегии S для каждого потребителя p_j по перестановке ρ арифметически вычисляется обозначаемый через $t^*(S, j)$ момент завершения разгрузки направленного в его адрес транспортного средства. Возникающая оптимизационная *задача* 1 записывается в виде

$$\min \sum_{j=1}^{n} \Psi_j(t^*(S, j)). \tag{1}$$

Данная задача NP-трудна в сильном смысле. Вычислительная сложность её решающего алгоритма, сконструированного по схеме динамического программирования, характеризуется величиной $O(4^n)$.

2. *Задача* 2 — частный случай предшествующей, получаемый в предположении, что все функции индивидуального штрафа линейны: $\psi_j(t) = a(j)t$, $j = \overline{1, n}$. *Задача* 3 — частный случай задачи 2, получаемый при дополнительном предположении $t(i) = 0$, $i = \overline{1, n}$. При любой стратегии штраф по каждому потребителю p_j в задачах 2 и 3 целесообразно разбить на две компоненты: 1) обозначаемый через $\alpha_1(i, j)$ штраф за период до момента завершения обслуживания процессором H направляемого в адрес этого потребителя объекта o_i; 2) обозначаемый через $\alpha_2(i, j)$ штраф за период от момента завершения обслуживания процессором H объекта o_i до момента завершения его разгрузки. Принципиально важным является следующее обстоятельство: каждая из величин $\alpha_2(i, j)$ не зависит от момента завершения обслуживания процессором направляемого в адрес p_j объекта o_i, т. е. $\alpha_2(i, j) = a_j v(i, j)$, $i = \overline{1, n}$, $j = \overline{1, n}$. Решающие задачи 2 и 3 алгоритмы двухэтапны. На первом этапе рассматривается задача о назначениях с аддитивным критерием, определяющая закрепление объектов за пунктами потребления; ее решение обеспечивает минимальную величину суммарных потерь потребителей при доставке грузов от процессора H. На втором этапе минимизируются суммарные потери, связанные с задержками в обслуживании процессором H объектов, направляемых в уже известные для каждого из них пункты потребления. Известно [8], что данная проблема при исходных данных задачи 2 является NP-трудной, а при исходных данных задачи 3 она решается в квадратично зависящем от n времени. Следовательно, задача 2 NP-трудна. Отметим, что конструируемый для второго этапа ее решения основанный на принципе динамического программирования алгоритм имеет оценку вычислительной сложности вида $O(2^n)$. При рассмотрении задачи 3 в случае, когда на первом этапе получается единственное назначение с мини-

мальными суммарными потерями, для построения оптимального решения этой задачи требуется выполнение арифметических операций в количестве $O(n^3)$. Случай множественности числа оптимальных по критерию суммарных потерь назначений нуждается в дополнительном рассмотрении.

Работа выполнена при поддержке РФФИ (проект № 15-07-03141).

Список литературы

[1] Коган Д. И., Федосенко Ю. С. Задачи синтеза оптимальных стратегий обслуживания стационарных объектов в одномерной рабочей зоне процессора // Автоматика и телемеханика. — 2010. — № 10. — С. 50–62.

[2] Коган Д. И., Федосенко Ю. С., Дуничкина Н. А. Бикритериальные задачи обслуживания стационарных объектов в одномерной рабочей зоне процессора // Автоматика и телемеханика. — 2012. — № 10. — С. 93–110.

[3] Вопросы построения стратегий обслуживания стационарных объектов перемещающимся в одномерной рабочей зоне процессором / Д. И. Коган, А. М. Пушкин, Н. А. Дуничкина, Ю. С. Федосенко // Автоматика и телемеханика. — 2016. — № 4. — С. 67–83.

[4] Модели и оптимизационные задачи однопроцессорного обслуживания пакетов объектов / Д. И. Коган, М. А. Трухина, Ю. С. Федосенко, А. В. Шеянов // Автоматика и телемеханика. — 2016. — № 11. — С. 142–157.

[5] Коган Д. И., Куимова А. С., Федосенко Ю.С. Задачи обслуживания бинарного потока объектов в системе с накопительно-расходным компонентом // Автоматика и телемеханика. — 2014. — № 7. — С. 122–135.

[6] Северный завоз / Материал из Википедии — свободной энциклопедии. URL: http://ru.wikipedia.org/wiki/Северный_завоз (дата обращения: 10.03.2017).

[7] Танаев В. С., Шкурба В. В. Введение в теорию расписаний. — М. : Наука, 1975. — 256 с.

[8] Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М. : Мир, 1982. — 416 с.

Задачи о назначениях в приложении к проблемам доформирования грузовых составов

Коган Дмитрий Израилевич[1], **Федосенко Юрий Сергеевич**[2], **Хандурин Дмитрий Константинович**[3]

[1] Московский технологический университет, e-mail: kdi_41@mail.ru

[2] Волжский государственный университет водного транспорта, e-mail: fds@vgavt-nn.ru

[3] Московский технологический университет, e-mail: dmitriy.khandurin@gmail.com

Изучаются диктуемые логистическими приложениями модификации стандартных задач о назначениях, связанные с доформированием грузовых составов:

задача о назначениях с запрещенными элементами и задача о парных назначениях. Вводится новая концепция решения для задач с изначально предписанным максиминным (минимаксным) критерием.

Введение. Многие проблемы принятия решений и, в частности, при планировании перевозок допускают формулировки в рамках стандартной задачи о назначениях [1–3]. Вместе с тем, возникающие на практике дополнительные требования, ограничения и эксплуатационные ситуации обусловливают необходимость построения тех или иных модификаций стандартных моделей о назначениях с последующим их математическим исследованием, включая разработку решающих алгоритмов. Настоящая работа выполнена в русле этой тематики.

1. Каждая задача о назначениях предполагает наличие множества исполнителей $I = \{1, 2, \ldots, n\}$, множества работ $R = \{r_1, r_2, \ldots, r_n\}$ и $(n \times n)$-матрицы численных оценок $A = \{a_{ij}\}$. Назначение — взаимно однозначное отображение π множества исполнителей в множество номеров работ, т.е. множества $\{1, 2, \ldots, n\}$ в себя. Назначение π исполнителю i предписывает работу $r_{\pi(j)}$, $i = \overline{1, n}$. Выделим следующие стандартные однокритериальные задачи о назначениях [3]: классическая задача о назначениях (КЗН) с критерием $\max\limits_{\pi} \sum\limits_{i=1}^{n} a_{i\pi(i)}$; задача о назначениях с минимизируемым аддитивным критерием $\min\limits_{\pi} \sum\limits_{i=1}^{n} a_{i\pi(i)}$; минимаксная задача о назначениях $\min\limits_{\pi} \max\limits_{i} a_{i\pi(i)}$; максиминная задача о назначениях $\max\limits_{\pi} \min\limits_{i} a_{i\pi(i)}$. КЗН именуем простейшей задачей о назначениях (ПЗН), если матрица численных оценок в ней булевозначна. Булевозначную матрицу обозначаем $E = e_{ij}$. Считаем, что в ПЗН исполнитель i способен выполнять работу r_j в том и только том случае, если $e_{ij} = 1$, $i = \overline{1, n}$, $j = \overline{1, n}$. Назначение π называем *абсолютным* решением ПЗН, если $(\forall i)[e_{i\pi(i)} = 1]$.

2. Обобщением КЗН является задача о назначениях с запрещенными элементами (КЗНЗЭ). В задачах такого типа её определяющая матрица A содержит символы запрета — нечисловые элементы z: если $a_{ij} = z$, то исполнитель i не может быть назначен на работу r_j. Назначение π именуем полным допустимым в КЗНЗЭ, если $a_{i\pi(i)}$ отлично от z для всех $i = \overline{1, n}$. Наличие в КЗНЗЭ допустимых полных назначений определяется путем рассмотрения соответствующим образом построенной ПЗН. В случае, когда в рассматриваемой КЗНЗЭ множество допустимых полных назначений непусто, возникает вопрос отыскания на этом множестве оптимального по рассматриваемому критерию решения. Указанная задача легко сводится к КЗН. Неполное допустимое назначение в КЗНЗЭ — это распределение некоторого подмножества M, $M \subseteq \{1, 2, \ldots, n\}$ исполнителей по работам, в котором каждый исполнитель из M получает ровно одну незапрещенную для него работу и каждой работе предписывается

не более одного исполнителя из множества M. Неполное назначение именуем *k-назначением*, если число элементов в множестве M равно k.

Теорема. *Если в КЗНЗЭ, имеющей полные допустимые решения, считаются возможными и неполные допустимые, то при оптимальном значении критерия без работы останутся не более половины исполнителей.*

3. Изучается бикритериальная КЗНЗЭ, в которой первый критерий определяет суммарную производительность, а второй — общее число назначенных на работы исполнителей. Первый критерий считаем максимизируемым, для второго возможны оба варианта. Для обеих постановок конструируются полиномиальные алгоритмы синтеза полных совокупностей эффективных оценок.

4. Вводится новая, заменяющая максиминный (минимаксный) подход, концепция решения задачи о назначениях. При этом используется концепция лексикографического упорядочения критериев многокритериальной задачи. Схема лексикографического упорядочения критериев дает возможность заменить концепции минимаксной и максиминной задач о назначениях более тонкими конструкциями. Пусть $A = \{a_{ij}\} - (n \times n)$-матрица, определяющая максиминную задачу о назначениях; множество значений a_{ij} образует упорядоченную по возрастанию совокупность $\{p_1, p_2, \ldots, p_k\}$. Введем в рассмотрение матрицу ступеней $B(A) = \{b_{ij}\}$, где $b_{ij} = t$ тогда и только тогда, когда $a_{ij} = p_t$. Будем также говорить, что в матрице A число $a_{ij} = p_t$ находится на ступени t. Чем больше число, тем выше номер ступени, на которой оно находится. Решая определяемую матрицей A максиминную задачу о назначениях, мы фактически ищем назначение π, при котором минимальное из чисел $a_{i\pi(i)}$, $i = 1, 2, \ldots, n$ находится на возможно более высокой ступени. Через $W(A, \pi) = (W_1(A, \pi), W_2(A, \pi), \ldots, W_k(A, \pi))$ обозначим k-мерный вектор, t-координата которого с индексом t — количество элементов из списка $\{a_{1\pi(1)}, a_{2\pi(2)}, \ldots, a_{n\pi(n)}\}$, находящихся на ступени номер t матрицы $B(A)$. Назначение π назовем абсолютно maxmin-оптимальным, если оно является решением задачи $\min(W_1(A, \pi), W_2(A, \pi), \ldots,$
$W_k(A, \pi))$ при указанном лексикографическом упорядочении критериев $W_t(A, \pi)$, $t = 1, 2, \ldots, k$. Проблема отыскания абсолютно maxmin-оптимального назначения сводится к решению соответствующим образом построенной КЗН.

5. Рассматривается следующая минимаксная задача о парных назначениях: считаются заданными две $(n \times n)$-матрицы численных оценок $A = \{a_{ij}\}$ и $B = \{b_{ij}\}$. Требуется найти пару назначений π и μ, доставляющих минимум критерию $\max\limits_i (a_{i\pi(i)} + b_{i\mu(i)})$. Излагается основанный на принципе динамического программирования [4–5] алгоритм решения.

Рассмотренные модификации стандартной задачи о назначениях возникли в связи с созданием модельно-алгоритмического обеспечения компьютерных

средств поддержки оперативного управления процессами доформирования грузовых составов.

Работа выполнена при поддержке РФФИ (проект № 15–07–03141).

Список литературы

[1] Гейл Д. Теория линейных экономических моделей. — М. : ИЛ, 1963. — 418 с.

[2] Вагнер Г. Основы исследования операций. — М. : Мир, 1972. — Т. 1. — 335 с.

[3] Коган Д. И. Динамическое программирование и дискретная многокритериальная оптимизация. Н. Новгород : Изд. ННГУ им. Н. И. Лобачевского, 2005. — 260 с.

[4] Беллман Р., Дрейфус С. Прикладные задачи динамического программирования. — М. : Наука, 1965. — 457 с.

[5] Коган Д. И., Федосенко Ю. С. Общая схема реализации алгоритмов динамического программирования в задачах синтеза стратегий однопроцессорного обслуживания потока объектов. Сборник научных трудов международной научно-технической конференции «Информатика и технологии. Инновационные технологии в промышленности и информатике». Московский технологический университет. Физико-технологический институт. — М.: МТУ, 2016. — С. 154–157.

Подпрямо неразложимые полигоны над прямоугольными группами

Кожухов Игорь Борисович, Петриков Александр Олегович

Московский институт электронной техники, e-mail: kozhuhov_i_b@mail.ru, masterpetr@mail.ru

Полигоном (автоматом) над полугруппой S называется множество X, на котором действует полугруппа S, т. е. определено отображение $X \times S \to X$, $(x, s) \mapsto xs$ такое, что $x(st) = (xs)t$ при всех $x \in X$, $s, t \in S$ (см. [1]).

В [2] были описаны полигоны над вполне простой полугруппой $S = \mathcal{M}(G, I, \Lambda, P)$ (определение и обозначения см. в [3]). Приведем это описание. Для группы G и её подгруппы H (не обязательно нормальной) через G/H будем обозначать множество правых смежных классов Hg, рассматриваемое как полигон над группой G относительно операции $Hg \cdot g' = Hgg'$, а символом $\coprod$ — копроизведение полигонов.

Пусть X — множество, $S = \mathcal{M}(G, I, \Lambda, P)$ — вполне простая полугруппа, $Q = \coprod\limits_{\gamma \in \Gamma} Q_\gamma$, где $Q_\gamma \cong G/H_\gamma$ — унитарный полигон над группой G с единицей e, $\kappa_\lambda : Q \to X$, $\pi_i : X \to Q$ — отображения такие, что $q\kappa_\lambda\pi_i = q \cdot p_{\lambda i}$ при любых $q \in Q$, $\lambda \in \Lambda$, $i \in I$. Если положить $x \cdot (g)_{i\lambda} = (x\pi_i \cdot g)\kappa_\lambda$ при $x \in X$, $(g)_{i\lambda} \in S$,

то X будет являться полигоном над S; кроме того, любой полигон над вполне простой полугруппой изоморфен полигону, полученному таким образом.

Подпрямо неразложимая универсальная алгебра — это алгебра, которая не разлагается нетривиальным образом в подпрямое произведение алгебр. Интерес к подпрямо не разложимым алгебрам объясняется классической теоремой Биркгофа, утверждающей, что любая алгебра является подпрямым произведением подпрямо неразложимых. Подпрямо неразложимые полигоны над полугруппами исследовались в [4]; там же были охарактеризованы подпрямо неразложимые полигоны над прямоугольной связкой S, т. е. полугруппой $S = L \times R$, где L — полугруппа левых нулей, а R — полугруппа правых нулей.

Цель данной работы — охарактеризовать подпрямо неразложимые полигоны над прямоугольной группой, т. е. полугруппой $S = L \times G \times R$, где G — группа. Эти полугруппы имеют вид $\mathcal{M}(G, I, \Lambda, P)$, где $p_{\lambda i} = e$ для всех $\lambda \in \Lambda$, $i \in I$.

Прежде чем привести эту характеризацию, сделаем несколько замечаний о полигонах над более широкими классами полугрупп.

Для полигонов над произвольными полугруппами имеет место:

Предложение 1. *Любой подполигон подпрямо неразложимого полигона подпрямо неразложим.*

Пусть X — полигон над вполне простой полугруппой $S = \mathcal{M}(G, I, \Lambda, P)$, задаваемый унитарным G-полигоном Q и отображениями $\kappa_\lambda : Q \to X$, $\pi_i : X \to Q$. Положим $A_\gamma = \{q\kappa_\lambda \mid q \in Q_\gamma, \lambda \in \Lambda\}$, $A = XS$. Нетрудно проверить, что A_γ — подполигоны полигона X и $A = \coprod\limits_{\gamma\in\Gamma} A_\gamma$.

С учётом реузльтатов из [4] нетрудно доказать:

Предложение 2. *Если X — подпрямо неразложимый полигон, то $|\Gamma| \leqslant 2$, а в случае, когда $|\Gamma| = 2$, либо $|A_1| = 1$, либо $|A_2| = 1$.*

Для $x \in X$ определим отображение $\phi_x : I \to Q$ по формуле $\phi_x(i) = x\pi_i$. Характеризация подпрямо неразложимых полигонов над прямоугольными группами может быть сформулирована теперь в теоремах 1, 2.

Теорема 1. *Пусть X — полигон над полугруппой $S = \mathcal{M}(G, I, \Lambda, P)$, где $p_{\lambda i} = e$ для всех $i \in I$, $\lambda \in \Lambda$, и пусть $|\Gamma| = 1$, т. е. $Q = G/H$. Тогда X подпрямо неразложим в том и только том случае, если выполнено одно из следующих условий:*

(i) $q_0\kappa_\lambda \neq q_0\kappa_\mu$ при некоторых $q_0 \in Q$, $\lambda, \mu \in \Lambda$, и если положить $a = q_0\kappa_\mu$, то будут выполнены условия: (a) $q_0\kappa_\nu \in \{a, a'\}$ при всех $\nu \in \Lambda$; (b) $q\kappa_\xi = q\kappa_\eta \notin \{a, a'\}$ при $q \neq q_0$ и любых $\xi, \eta \in \Lambda$; (c) $\phi_x = \phi_y \Leftrightarrow x = y$ или $\{x, y\} = \{a, a'\}$.

(ii) $q\kappa_\lambda = q\kappa_\mu$ при всех $q \in Q$, $\lambda, \mu \in \Lambda$ и выполнены условия: (d) $\phi_x = \phi_y \Leftrightarrow x = y$; (e) существует наименьшая подгруппа H' такая, что $H \subset H'$.

Теорема 2. *Пусть X — полигон над полугруппой $S = \mathcal{M}(G, I, \Lambda, P)$, где $p_{\lambda i} = e$ для всех $i \in I$, $\lambda \in \Lambda$, и пусть $|\Gamma| = 2$ (скажем, $\Gamma = \{1, 2\}$). Тогда X подпрямо неразложим в том и только том случае, если одно из множеств A_1, A_2 одноэлементно (будем считать, что $A_2 = \{x\}$ — одноэлементное множество) и выполняется одно из условий (i), (ii), сформулированных в теореме* 1.

СПИСОК ЛИТЕРАТУРЫ

[1] Kilp M., Knauer U., Mikhalev A. V. Monoids, acts and categories. — Berlin : Walter de Gruyter, 2000. — 529 p.

[2] Avdeyev A. Yu, Kozhukhov I. B. Acts over completely 0-simple semigroups // Acta Cybernetica. — 2000. — V. 14, Iss. 4. — P. 523–531.

[3] Клиффорд А., Престон Г. Алгебраическая теория полугрупп. — М. : Мир, 1972. — Т. 1. — 286 с.

[4] Кожухов И. Б., Халиуллина А. Р. Характеризация подпрямо неразложимых полигонов // Прик. дискр. матем. — 2015. — № 1. — С. 5–16.

Фрагментарные модели в задачах дискретной оптимизации

Козин Игорь Викторович, Полюга Светлана Игоревна

Запорожский национальный университет, e-mail: kozin_ainc00@gmail.com, poluga_veta99@mail.ru

Для некоторых классов дискретных оптимизационных задач существуют эффективные жадные алгоритмы, которые всегда приводят к точному оптимальному решению задачи [1]. Однако в большинстве случаев алгоритмы такого типа приводят лишь к приближенному решению задачи без оценки точности решения. Интересен вопрос, на каких классах задач в принципе возможно применение жадного алгоритма для поиска приближенного оптимального решения.

Поиски классов дискретных задач, допускающих применение жадных алгоритмов, приводят к различным комбинаторным структурам. В частности, это — матроиды [2], гридоиды [3] и наследственные структуры [4]. Однако наиболее общей комбинаторной конфигурацией для подобных задач является фрагментарная структура.

Определение. Фрагментарной структурой (X, E) на конечном множестве X называется такое семейство его подмножеств $E = \{E_1, E_2, ..., E_n\}$, где $\forall\ i = 1, 2, ..., n$, $E_i \subseteq X$, что $\forall E_i \in E,\ E_i \neq \varnothing\ \exists e \in E_i\ E_i \backslash \{e\} \in E$.

Всякое подмножество $A \in E$ (допустимый фрагмент) можно построить из пустого множества путем применения жадного алгоритма, последовательно просматривая элементы множества X и добавляя их таким образом, чтобы на каждом шаге этой процедуры построенное подмножество было допусти-

мым фрагментом. Результат работы алгоритма зависит лишь от начальной перестановки элементов множества X.

Задача поиска оптимального значения функции, заданной на множестве допустимых фрагментов, сводится к оптимизационной задаче на перестановках и, следовательно, может быть решена гибридным алгоритмом, построенным путем комбинации фрагментарного алгоритма и какого-либо комбинаторного алгоритма.

Можно показать, что фрагментарной структурой обладают ряд задач об оптимальном покрытии графа (покрытие звездами, циклами, путями и т. д.), задачи целочисленного прямоугольного раскроя, задачи плоской укладки невыпуклых объектов (тетрамино, пентамино, гексамино). Для всех перечисленных задач удалось построить гибридный алгоритм поиска приближенного оптимального решения на основе комбинации фрагментарного алгоритма и эволюционного алгоритма на перестановках с геометрическим оператором кроссовера. Для рассматриваемых классов задач построены генераторы случайных задач и проведена оценка качества алгоритмов на больших сериях случайно сгенерированных задач.

Список литературы

[1] Whitney H. On the abstract properties of linear dependence // American Journal of Mathematics. — 1935. — Vol. 57, No. 3. — P. 509–533.

[2] Ziegler G. M. Oriented Matroids Today // The Electronic Journal of Combinatorics: dynamic surveys. — http://www.emis.ams.org/journals/EJC/Surveys/ds4.pdf.

[3] Bjorner A. Introduction to greedoids // Matroid Applications. — Cambridge: Cambridge University Press, 1992. — 180 p.

[4] Ильев В. П. Задачи на системах независимости, разрешимые жадным алгоритмом //Дискретная математика. — 2009. — Т. 21, вып. 4. — С. 85–94.

Эффективная стратегия распараллеливания для решения частного случая задачи о сумме подмножеств методом ветвей и границ

Колпаков Роман Максимович[1], **Посыпкин Михаил Анатольевич**[2]

[1] МГУ им М.В. Ломоносова, Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН, e-mail: foroman@mail.ru

[2] Вычислительный центр им. А. А. Дородницына ФИЦ ИУ РАН, e-mail: mposypkin@gmail.com

В данной работе изучается сложность параллельного решения задачи о сумме подмножеств [1] методом ветвей и границ. Описание процедуры решения задачи о сумме подмножеств методом ветвей и границ и базовых понятий, связанных с данной процедурой, можно найти, например, в [2]. Под сложностью

$L[P]$ решения задачи P методом ветвей и границ понимается число подзадач задачи P, исключенных из рассмотрения в процессе ее решения согласно условиям отсева.

Рассматривается классический частный случай [3] задачи о сумме подмножеств:

$$\begin{array}{l}\text{maximize } f(\tilde{x}) = \sum_{i=1}^{n} 2x_i,\\ \text{subject to } f(\tilde{x}) = \sum_{i=1}^{n} 2x_i \leqslant 2k+1,\end{array}$$

который обозначается через $P(n;k)$. Известно [2], что $L[P(n;k)] = \binom{n+1}{k+1}$. Далее для удобства мы будем обозначать через $\hat{P}(n;k)$ задачу $P(n-1;k-1)$, имеющую сложность $L[\hat{P}(n;k)] = \binom{n}{k}$. Отметим, что любая подзадача задачи $\hat{P}(n;k)$ также может быть представлена в виде задачи $\hat{P}(n';k')$, где $n' \leqslant n$ и $k' \leqslant k$.

Пусть для решения задачи $\hat{P}(n;k)$ имеется $p > 1$ процессоров. Рассматривается следующая стратегия распараллеливания решения данной задачи. В процессе решения производится разбиение задачи на подзадачи согласно методу ветвей и границ. Для решения каждой из подзадач выделяется либо ровно один процессор, либо несколько процессоров (в частности, для решения исходной задачи $\hat{P}(n;k)$ выделяются все p процессоров). Если для решения некоторой подзадачи $\hat{P}(n';k')$ выделяется ровно один процессор, то этот процессор решает данную подзадачу методом ветвей и границ. При этом данный процессор может в общем случае получить для решения несколько подзадач, в таком случае он решает все полученные подзадачи, включая подзадачу $\hat{P}(n';k')$, в некоторой заданной очередности. Если для решения некоторой подзадачи $\hat{P}(n';k')$ выделяется несколько процессоров, то далее эти процессоры участвуют только в решении данной подзадачи. Кроме того, из этих процессоров выделяется некоторый *управляющий* процессор u, который после обработки подзадачи $\hat{P}(n';k')$ находит ее оптимальное решение. Обработка подзадачи $\hat{P}(n';k')$ осуществляется в этом случае следующим образом. Пусть $p' > 1$ — число процессоров, выделенных для решения подзадачи $\hat{P}(n';k')$. Без ограничения общности будем полагать, что $k' \leqslant n'/2$ (в противном случае рассмотрим симметричную подзадачу $\hat{P}(n';n'-k')$). Если подзадача $\hat{P}(n';k')$ удовлетворяет условию отсева (т.е. $k' = 0$ в данном случае), то она обрабатывается процессором u согласно процедуре метода ветвей и границ. Пусть теперь $n'/2 \geqslant k' > 0$. Тогда процессор u осуществляет декомпозицию подзадачи $\hat{P}(n';k')$ на две соответствующие подзадачи $\hat{P}(n'-1;k'-1)$ и $\hat{P}(n'-1;k')$. Заметим, что

$$\frac{L[\hat{P}(n'-1;k'-1)]}{L[\hat{P}(n';k')]} = \frac{\binom{n'-1}{k'-1}}{\binom{n'}{k'}} = \frac{k'}{n'}.$$

Выделим два возможных случая:

1 (случай единичной декомпозиции). Пусть $\frac{k'}{n'} \geqslant 1/p'$. Тогда процессор u выделяет $\lfloor k'p'/n' \rfloor$ процессоров на решение задачи $\hat{P}(n'-1; k'-1)$. Если при этом на решение задачи $\hat{P}(n'-1; k'-1)$ выделяется более одного процессора, то среди выделенных процессоров назначается новый управляющий процессор u' (если на решение задачи $\hat{P}(n'-1; k'-1)$ выделяется только один процессор, то этот процессор также будем обозначать через u'). Остальные процессоры, включая управляющий процессор u, выделяются на решение задачи $\hat{P}(n'-1; k')$ (если при этом на решение задачи $\hat{P}(n'-1; k')$ выделяется более одного процессора, то u остается управляющим процессором). Процессор u передает процессору u' для решения задачу $\hat{P}(n'-1; k'-1)$, после чего приступает к решению задачи $\hat{P}(n'-1; k')$. После решения задач $\hat{P}(n'-1; k'-1)$ и $\hat{P}(n'-1; k')$ процессор u' передает решение задачи $\hat{P}(n'-1; k'-1)$ процессору u, который сравнивает решение этой задачи с решением задачи $\hat{P}(n'-1; k')$ и получает в результате оптимальное решение задачи $\hat{P}(n'; k')$.

2 (случай кратной декомпозиции). Пусть $\frac{k'}{n'} < 1/p'$. Тогда процессор u последовательно осуществляет декомпозиции подзадач $\hat{P}(n'-i; k')$ на подзадачи $\hat{P}(n'-i-1; k')$ и $\hat{P}(n'-i-1; k'-1)$ для $i = 1, 2, \ldots, t-1$, где t — наименьшее число такое, что

$$\frac{\sum_{i=1}^{t} L[\hat{P}(n'-i; k'-1)]}{L[\hat{P}(n'; k')]} = \frac{\sum_{i=1}^{t} \binom{n'-i}{k'-1}}{\binom{n'}{k'}} \geqslant \frac{1}{p'}.$$

Отметим, что данное соотношение можно переписать в виде

$$\frac{k'}{n'} \left[1 + \sum_{i=1}^{t-1} \frac{(n'-k')(n'-k'-1)\cdots(n'-k'-(i-1))}{(n'-1)(n'-2)\cdots(n'-i)} \right] \geqslant \frac{1}{p'}.$$

После осуществления последней из этих декомпозиций (т.е. декомпозиции подзадачи $\hat{P}(n'-(t-1); k')$ на подзадачи $\hat{P}(n'-t; k')$ и $\hat{P}(n'-t; k'-1)$) процессор u выбирает из имеющихся процессоров некоторый процессор u'' и передает ему для решения подзадачи $\hat{P}(n'-1; k'-1)$, $\hat{P}(n'-2; k'-1)$,…, $\hat{P}(n'-t; k'-1)$. Остальные имеющиеся процессоры, включая управляющий процессор u, выделяются для решения оставшейся подзадачи $\hat{P}(n'-t; k')$. В результате последовательного решения задач $\hat{P}(n'-1; k'-1)$, $\hat{P}(n'-2; k'-1)$,…, $\hat{P}(n'-t; k'-1)$ процессор u'' находит среди решений этих задач наилучшее решение, которое передается процессору u. После решения задачи $\hat{P}(n'-t; k')$ процессор u сравнивает это решение с наилучшим решением, полученным от процессора u'', и получает в результате оптимальное решение задачи $\hat{P}(n'; k')$ в данном случае.

Будем называть описанную выше процедуру решения задачи $\hat{P}(n; k)$ на p процессорах процедурой *рекурсивного сбалансированного разбиения*. Далее будем полагать, что время $T[\hat{P}(n'; k')]$ решения любой задачи $\hat{P}(n'; k')$ методом

ветвей и границ пропорционально сложности $L[\hat{P}(n';k')]$ этой задачи, т. е.

$$T[\hat{P}(n';k')] = A \cdot L[\hat{P}(n';k')] = A \cdot \binom{n'}{k'}$$

для некоторой константы $A > 0$. В частности, мы предполагаем, что декомпозиция любой подзадачи осуществляется процессором за константное время. Будем также предполагать, что обмен данными между любыми двумя процессорами (в частности, передача от одного процессора к другому произвольной подзадачи) осуществляется за константное время. При этом мы будем придерживаться *синхронизированной* модели передачи данных, т. е. будем предполагать, что во время передачи данных от одного процессора к другому оба эти процессора не могут совершать никаких других операций. Время решения задачи $\hat{P}(n;k)$ на p процессорах посредством процедуры рекурсивного сбалансированного разбиения обозначим через $T_{rbd}[\hat{P}(n;k);p]$.

Положим $\alpha = \prod_{i=1}^{\infty}(1+2^{-i}) \approx 2.384231$.

Теорема 1. $T_{rbd}[\hat{P}(n;k);p] < 2\alpha\frac{T[\hat{P}(n;k)]}{p} + O(n)$.

Работа выполнена при поддержке РФФИ (проект № 15-07-03102-а).

Список литературы

[1] Kellerer H., Pfershy U., Pisinger D. Knapsack Problems. — Springer Verlag, 2004. — 546 с.

[2] Колпаков Р. М., Посыпкин М. А. Верхняя и нижняя оценки трудоемкости метода ветвей и границ для задачи о ранце // Дискретная математика. — 2010. — Т. 22, № 1. — С. 58–73.

[3] Финкельштейн Ю. Ю. Приближенные методы и прикладные задачи дискретного программирования. — М. : Наука, 1976. — 264 с.

Асимптотика числа решений простейших уравнений в подстановках

Колчин Андрей Валентинович

Московский автомобильно-дорожный государственный технический университет (МАДИ), e-mail: akolchin@madi.ru

Памяти Колчина Валентина Федоровича посвящается

Для решения широкого круга комбинаторных задач весьма плодотворным оказывается *вероятностный подход* [1–2]. В частности, в вероятностной комбинаторике находит успешное применение *обобщенная схема размещения*, позволяющая сводить ряд комбинаторных задач к задачам о суммах независимых случайных величин, классическому объекту изучения в теории вероятностей. Обобщенная схема размещения была введена в [3] и заняла заметное место в асимптотических исследованиях в вероятностной комбинаторике. Свое

название эта схема получила в связи с тем, что она является обобщением классической задачи о случайном размещении частиц по ячейкам [2].

Напомним, что в обобщенной схеме размещения частиц распределение заполнений ячеек представимо как условное распределение *независимых* случайных величин при условии, что их сумма принимает фиксированное значение. Пусть $\eta_1, \dots, \eta_N$ — неотрицательные целочисленные случайные величины, рассматриваемые как некоторые числовые характеристики комбинаторной структуры из N компонент, состоящей из n элементов, такие, что $\eta_1 + \dots + \eta_N = n$. Если существуют независимые случайные величины $\xi_1, \dots, \xi_N$ такие, что совместное распределение $\eta_1, \dots, \eta_N$ допускает представление

$$\mathbf{P}\{\eta_1 = k_1, \dots, \eta_N = k_N\} = \mathbf{P}\{\xi_1 = k_1, \dots, \xi_N = k_N \mid \xi_1 + \dots + \xi_N = n\}, \quad (1)$$

где $k_1, \dots, k_N$ — произвольные целые числа, то говорят, что $\eta_1, \dots, \eta_N$ образуют обобщенную схему размещения с параметрами n и N и независимыми случайными величинами $\xi_1, \dots, \xi_N$. Случайные величины $\eta_1, \dots, \eta_N$ интерпретируются как заполнения ячеек.

В силу независимости случайных величин $\xi_1, \dots, \xi_N$, изучение многих характеристик обобщенной схемы размещения сводится к задачам о суммах независимых случайных величин. В случае, когда распределения слагаемых одинаковы и фиксированы (не зависят от числа слагаемых), можно пользоваться хорошо развитой теорией суммирования независимых случайных величин. Однако во многих применениях обобщенной схемы возникает необходимость в *локальных предельных теоремах в схеме серий*.

Как правило (см., например, [4–5]), распределение случайных величин $\xi_1, \dots, \xi_N$ представляется в виде

$$\mathbf{P}\{\xi_1 = k\} = \frac{b_k \theta^k}{k!\, B(\theta)}, \quad (2)$$

где $b_0, b_1, b_2, \dots$ — некоторая последовательность неотрицательных чисел, $B(\theta) = \sum_{k=0}^{\infty} b_k \theta^k / k!$, и θ — параметр, принимающий положительные значения из области сходимости ряда $B(\theta)$. Если соотношение (1) справедливо при некотором θ, то оно остается верным при всех положительных θ из области сходимости ряда $B(\theta)$ (см., например, [4]). Для изучения характеристик обобщенной схемы размещения, как правило, требуются локальные предельные теоремы при всех значениях параметра θ. Основные случаи возможных областей изменения N и θ были рассмотрены в [5–8].

Пусть $S_{n,R}$ есть множество всех подстановок степени n, длины циклов которых лежат в некотором множестве R натуральных чисел. Рассмотрим уравнение

$$X^d = e, \quad (3)$$

где X — подстановка степени n, а e — тождественная подстановка. Нетрудно видеть, что множество всех решений уравнения (3) в симметрической группе

S_n, где d — простое число, совпадает с $S_{n,R}$ с $R = \{1, d\}$; если же d — составное число и $1 = d_0 < d_1 < \cdots < d_r = d$ — все различные делители числа d, то подстановка X является решением уравнения (3) тогда и только тогда, когда длины циклов X принадлежат $R = \{d_0, \ldots, d_r\}$.

Обозначим $T_n^{(d)}$ число решений уравнения (3), и пусть $a_{n,R}$ — число элементов множества $S_{n,R}$. Асимптотика чисел $a_{n,R}$ изучалась, например, в [9–10] с использованием метода перевала. Как показано в [11–12],

$$a_{n,R} = \frac{n!\, e^{B(\theta)}}{\theta^n} \sum_{N=1}^{\infty} \frac{(B(\theta))^N}{N!} e^{-B(\theta)} \mathbf{P}\{\xi_1 + \cdots + \xi_N = n\},$$

где

$$\mathbf{P}\{\xi_1 = k\} = \frac{\theta^k}{kB(\theta)}, \quad k \in R, \quad \text{и} \quad B(\theta) = \sum_{k \in R} \frac{\theta^k}{k}, \tag{4}$$

так что для нахождения асимптотики $T_n^{(d)} = a_{n,R}$ достаточно выбрать подходящее θ и, как в [5–8], получить серию локальных предельных теорем для суммы независимых случайных величин, одинаково распределенных по закону (4).

Используя детали эффекта перехода распределений сумм независимых целочисленных случайных величин, одинаково распределенных по закону (2), с одной решетки на другую, подробно изученные в [6–8], сразу получаем ранее не известные асимптотики для числа решений уравнения (3) при промежуточных порядках стремления параметра θ к нулю. Заметим, что в случае, когда значения параметра θ приближаются к границе сходимости ряда $B(\theta)$, могут появляться и другие предельные распределения.

Список литературы

[1] Гончаров В. Л. Из области комбинаторики // Изв. АН СССР. Сер. матем. — 1944. — Т. 8, № 1. — С. 3–48.

[2] Колчин В. Ф., Чистяков В. П. Комбинаторные задачи теории вероятностей // Итоги науки и техники. Сер. Теория вероятностей, математическая статистика, теоретическая кибернетика. — 1974. — Т. 11. — С. 5–45.

[3] Колчин В. Ф. Один класс предельных теорем для условных распределений // Литовский матем. сборник. — 1968. — Т. 8, № 1. — С. 111–126.

[4] Колчин В. Ф. Случайные отображения. — М. : Наука, 1984. — 207 с.

[5] Колчин А. В. Предельные теоремы для обобщенной схемы размещения // Дискретная математика. — 2003. — Т. 15, № 4. — С. 148–157.

[6] Колчин А. В., Колчин В. Ф. О переходе распределений сумм независимых одинаково распределенных случайных величин с одной решетки на другую в обобщенной схеме размещения // Дискретная математика. — 2006. — Т. 18, № 4. — С. 113–127.

[7] Колчин А. В., Колчин В. Ф. Переход с одной решетки на другую распределений сумм случайных величин, встречающихся в обобщенной схеме размещения // Дискретная математика. — 2007. — Т. 19, № 3. — С. 15–21.

[8] Колчин А. В. Предельные теоремы в обобщенной схеме размещения // Обозрение прикладной и промышленной математики. — 2009. — Т. 16, № 3. — С. 432–435.

[9] Moser L., Wyman M. On solutions of $x^d = I$ in symmetric groups // Canadian J. Math. — 1955. — V. 7, № 2. — P. 159–168.

[10] Волынец Л. М. О числе решений уравнения $x^s = e$ в симметрической группе // Матем. заметки. — 1986. — Т. 40, № 2. — С. 155–160.

[11] Колчин В. Ф. О числе подстановок с ограничениями на длины циклов // Дискретная математика. — 1989. — Т. 1, № 2. — С. 97–109.

[12] Колчин А. В. Уравнения, содержащие неизвестную подстановку // Дискретная математика. — 1994. — Т. 6, № 1. — С. 100—115.

О схемах глубины два для функции голосования

Комбаров Юрий Анатольевич

МГУ имени М. В. Ломоносова, e-mail: yuri.kombarov@gmail.com

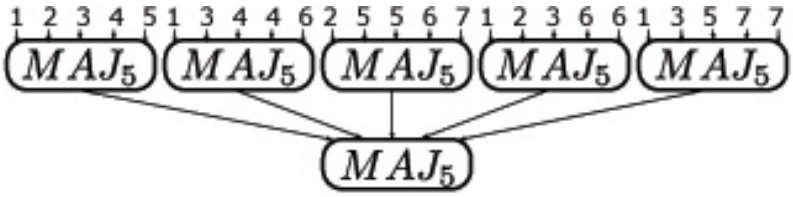
Функцией голосования от n переменных называть n-местную булеву функцию MAJ_n, такую, что $MAJ_n(\tilde{x}) = 1$ тогда и только тогда, когда $\mathrm{wt}(\tilde{x}) > \frac{n}{2}$ (здесь $\tilde{x}$ — набор из $\{0,1\}^n$, а $\mathrm{wt}(\tilde{x})$ — число единиц в $\tilde{x}$). Далее мы будем рассматривать функции голосования только от нечетного количества переменных.

Пусть B — множество булевых функций. *Схемой из функциональных элементов в базисе B* называется ориентированный граф без ориентированных циклов, вершины которого подписаны переменными из алфавита $\{x_1, x_2, \ldots\}$ или функциями из B; вершины входной степени 0 подписаны переменными и называются *входами*, а вершины входной степени k подписаны k-местными функциями и называются *элементами*. Один из элементов схемы выделен и называется *выходным*. Каждой вершине схемы естественным образом сопоставляется булева функция, *реализуемая* этой вершиной. Говорят, что схема реализует некоторую булеву функцию f, если ее выходной элемент реализует эту функцию. *Глубиной* схемы называется наибольшая длина ориентированного пути, соединяющего вход схемы и ее выходной элемент. *Ветвлением* схемы будем называть наибольшее число входов ее элемента (т. е. наибольшее число ребер, входящих в вершину схемы). Ветвление схемы S будем обозначать через $F(S)$.

В данной заметке рассматриваются схемы глубины два в базисе $M = \{MAJ_{2k-1} | k = 1, 2, \ldots\}$. В работе [1] ставится следующий вопрос: «Ка-

ково наименьшее ветвление схемы, реализующей функцию MAJ_n?» Введем следующее обозначение: $F(n) = \min F(S)$, где минимум берется по всем схемам S глубины два в базисе M, реализующих функцию MAJ_n (функция $F(n)$ определена только для нечетных n). Верны следующие простые оценки: $n^{\frac{1}{2}} \leqslant F(n) \leqslant n$. Верхняя оценка очевидна. Нижняя оценка следует из того, что функция голосования зависит от всех своих переменных и, следовательно, каждый из n входов схемы должен быть соединен ориентированным путем с выходным элементом. В [1] получена более сильная нижняя оценка: $F(n) \geqslant \Omega(n^{\frac{13}{19}}(\log n)^{-\frac{2}{19}})$. Также в [1] для $n = 7, 9, 11$ построены примеры схем с ветвлением $n-2$, реализующих функцию MAJ_n. В настоящей заметке доказано, что схема с ветвлением $n-2$, реализующая MAJ_n существует для любого нечетного n, большего пяти, т. е. доказана верхняя оценка $F(n) \leqslant n-2$.

Далее схемы будут описываться при помощи целочисленных матриц. Пусть S — схема глубины два в базисе M с n входами, выходной элемент которой имеет k входов. Схеме S сопоставим матрицу размера $n \times k$, в которой число в i-ой строке и j-ом столбце равно числу входов, соответствующих j-ой переменной, которые можно соединить ориентированным путем с i-ым входом выходного элемента. Ниже изображен пример схемы глубины 2 с ветвлением 5, реализующей функцию MAJ_7, и соответствующая этой схеме матрица (цифра у входа элемента соответствует номеру переменной, подаваемой на этот вход).



$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 2 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 2 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 2 \end{pmatrix}$$

Набор $\tilde{x}$ из множества $\{0,1\}^n$ будем называть *тестовым*, если $\mathrm{wt}(\tilde{x}) = \frac{n}{2} - \frac{1}{2}$. Множество всех тестовых наборов длины n будем обозначать T_n.

Лемма. *Пусть n нечетно. Схема S в базисе M с n входами реализует функцию MAJ_n тогда и только тогда, когда S выдает значение 0 на каждом наборе из T_n.*

Для доказательства леммы достаточно отметить, что каждая функция, входящая в M, монотонна и самодвойственна [2], и, следовательно, всякая схема в базисе M реализует монотонную самодвойственную функцию.

Теорема. *Пусть n нечетно и $n \geqslant 7$. Тогда существует схема S глубины два в базисе M, содержащая только элементы MAJ_{n-2} и реализующая функцию MAJ_n.*

Доказательство. Опишем $n \times (n-2)$-матрицу схемы S. Все компоненты первой строки содержат нули, за исключением первых трех; первые компоненты содержат числа $\lceil \frac{n-2}{3} \rceil, \lfloor \frac{n-2}{3} \rceil, \lfloor \frac{n-2}{3} \rfloor$ (через $\lfloor x \rceil$ обозначается целое число, ближайшее к x). Все остальные компоненты матрицы — единицы и нули; в первом,

втором и третьем столбцах соответственно $\lceil \frac{n-3}{3} \rceil$, $\lfloor \frac{n-3}{3} \rceil$ и $\lfloor \frac{n-3}{3} \rfloor$ нулей, причем в каждой строке матрицы на первых трех местах не более одного нуля. Наконец, оставшиеся компоненты образуют $(n-3) \times (n-3)$-матрицу, на диагонали которой — нули, а вне диагонали — единицы.

Так как матрица имеет $n-2$ строки, выходной элемент соответствующей ей схемы имеет $n-2$ входа; так как сумма чисел в каждой строке равна $n-2$, остальные элементы также имеют по $n-2$ входа. Выделим три группы элементов: к группе G_i отнесем элементы, соответствующие строкам матрицы, содержащим ноль в i-ой компоненте ($i = 1, 2, 3$). Пример матрицы для $n = 11$ изображен ниже.

$$
\begin{array}{c}
\\
G_1 \\
\\
\\
G_2 \\
\\
\\
G_3 \\
\\
\end{array}
\left(\begin{array}{cccccccccc}
3 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0
\end{array}\right)
$$

Проверим, что схема выдает ноль на всех тестовых наборах длины n; после этого утверждение теоремы будет следовать из леммы. Пусть $\tilde{\tau} \in T_n$. Пусть E — элемент, соответствующий первой строке матрицы. Легко видеть, что E реализует функцию $MAJ_3(x_1, x_2, x_3)$. Рассмотрим два случая.

1. Элемент E выдает значение один на наборе $\tilde{\tau}$. Тогда не менее двух из первых трех компонент $\tilde{\tau}$ равны единице. Пусть, без ограничения общности, $\tau_1 = \tau_2 = 1$. Тогда все элементы из групп G_1 и G_2 выдают ноль на наборе $\tilde{\tau}$. Действительно, в наборе $\tilde{\tau}$ ровно $\frac{n-1}{2}$ единиц. На элементы из группы G_1 не подается первая компонента набора, а на элементы из группы G_2 — вторая; поэтому на каждый из этих элементов подается не более $\frac{n-3}{2}$ единиц. Это количество меньше, чем $\frac{n-2}{2}$, следовательно, каждый из рассматриваемых элементов выдает ноль. Число наборов в любых двух группах из G_1, G_2, G_3 превосходит $2\lfloor \frac{n-3}{3} \rfloor > \frac{n-2}{2}$ (при $n \geqslant 7$). Поэтому большинство из элементов на первом уровне схемы выдают ноль на $\tilde{\tau}$, а значит и выходной элемент выдает ноль.

2. Элемент E выдает значение ноль на наборе $\tilde{\tau}$. Тогда среди трех первых компонент $\tilde{\tau}$ не более одной единицы, следовательно, среди последних $n-3$ компонент $\tilde{\tau}$ не менее $\frac{n-3}{2}$ единиц. Аналогично предыдущему случаю, если $\tau_i = 1$, то элемент, соответствующий $(i-3)$-ей строке матрицы выдает ноль на $\tilde{\tau}$ (при $i > 3$). Значит, среди элементов первого уровня не менее $\frac{n-3}{2} + 1$

выдают ноль. Так как более половины элементов первого уровня выдают ноль, вся схема выдает ноль. **Теорема доказана.**

Список литературы

[1] Kulikov A. S., Podolskii V. V. Computing majority by constant depth majority circuits with low fan-in gates // 34th Symposium on theoretical aspects of computer science (STACS 2017) — 2017. — V. 66. — P. 49:1–49:14.

[2] Угольников А. Б. Классы Поста. — М.: Издательство Центра прикладных исследований при механико-математическом факультете МГУ, 2008.

Об оценке погрешности алгоритма поиска экстремума на подклассе класса функций, определяемом кусочно-линейной мажорантой

Коротченко Анатолий Григорьевич, Сморякова Валентина Михайловна

Нижегородский государственный университет им. Н. И. Лобачевского, e-mail: koangr@yandex.ru, smorykov@mail.ru

В статье рассматривается подкласс класса функций, введенного в [1]. В [1] построен простой алгоритм отыскания наибольшего значения функции из указанного класса. Для данного алгоритма в рассматриваемом подклассе установлены оценки погрешности в определении наибольшего значения функции, в зависимости от ситуации, складывающейся в ходе поиска.

Будем говорить, что непрерывная функция $f(x)$, определённая на отрезке $[a, b]$, принадлежит классу функций $F(a, b, K)$, если выполняются следующие соотношения: $\frac{f(x_2)-K}{x_2-a} \leqslant \frac{f(x_1)-K}{x_1-a}$, $\frac{f(x_2)-K}{b-x_2} \geqslant \frac{f(x_1)-K}{b-x_1}$, где $x_2 > x_1$, $x_1,\ x_2 \in (a, b)$, $K \in R$. Класс функций $F(a, b, K)$ замкнут относительно операций суммирования с неотрицательными коэффициентами, взятия минимума, максимума по конечному набору функций и содержит вогнутые, выпуклые и удовлетворяющие условию Липшица функции. Кроме того, если функция $f(x) \in F(a, b, K)$, то $f(x) \geqslant K$. Более подробно о свойствах указанного класса можно найти в [1].

Пусть $D = [a, b]$. Будем рассматривать алгоритм поиска экстремума функций $f(x)$ из класса $F(a, b, K)$, в котором функция на первом шаге вычисляется в середине отрезка $[a, b]$, а на текущем шаге в определённой точке подмножества множества D, определяемого на основе информации о значениях функции, полученных к указанному шагу. При этом окончание процесса определяется исчерпанием вычислительного ресурса или достижением заданной точности решения задачи.

Пусть функция $f(x)$ к текущему шагу алгоритма вычислена в N точках отрезка $[a, b]$ и $a < x_1 < x_2 < ... < x_N < b$, $y_i = f(x_i)$, $i = 1, ..., N$, $h = \max\limits_{i=1,...,N} y_i$.

Обозначим через $F(a,b,K,N)$ подкласс класса $F(a,b,K)$ всех таких функций, которые в точках x_i принимают значение y_i, $i=1,...,N$.

Также как в [2] построим точную верхнюю мажоранту $\varphi_N(x)$ всех функций из класса $F(a,b,K,N)$, т.е. такую функцию, что $\varphi_N(x)\in F(a,b,K,N)$, $\varphi_N(x_i)=y_i$, $i=1,...,N$, и $f(x)\leqslant\varphi_N(x)$ для всех $f(x)\in F(a,b,K,N)$. Пусть $\xi_0=a$, $\xi_N=b$, $\xi_j=\frac{(b-a)(b-x_{j+1})(x_j-a)}{(y_j-K)(b-x_{j+1})+(x_j-a)(y_{j+1}-K)}$, $j=1,...,N-1$, $N\geqslant 2$, $(a<\xi_1<...<\xi_{N-1}<b)$ — точки, доставляющие локальные максимумы функции $\varphi_N(x)$. Если $N=1$, то точки $\xi_0=a$, $\xi_1=b$ доставляют локальные максимумы функции $\varphi_1(x)$.

Обозначим через $J(N)$ множество всех таких индексов j, $0\leqslant j\leqslant N$, для которых отрезок $[x_j,x_{j+1}]$ содержит точку ξ_j, доставляющую значение функции $\varphi_N(x)$ большее h. Здесь $J(N)\subseteq\{0,1,...,N\}$. Для всех $j\in J(N)$ определим на отрезке $[x_j,x_{j+1}]$ величины u_j и v_j: $u_j=a+\frac{(h-K)}{(y_j-K)}(x_j-a)$, $v_j=b-\frac{(h-K)}{(y_j-K)}(b-x_j)$. Заданные таким образом отрезки $[u_j,\,v_j]$ содержат локальные экстремумы ξ_j функции $\varphi_N(x)$, для всех $j\in J(N)$.

Пусть $Q(j)=\varphi_N(\xi_j)-h$, для каждого $j\in J(N)$, множество $\tilde{J}=\{j_1,...,j_s\}$, $\tilde{J}\subseteq J(N)$, $j_1<...<j_s$, такое, что $Q(j_1)=...=Q(j_s)=\max\limits_{j\in J(N)}Q(j)$. Если множество $\tilde{J}$ состоит из одного элемента s, то $Q_s=\max\limits_{j\in J(N)}Q_j$, $u=u_s$ и $v=v_s$. Если же $|\tilde{J}|>1$, то в качестве s будем выбирать значение j_1.

Пусть $L_j^1=\frac{y_j-K}{x_j-a}$ и $L_j^2=\frac{y_j-K}{b-x_j}$, введём в рассмотрение следующие функции $\psi_j^1(x)=L_j^1x+h-\ L_j^1u_j$ и $\psi_j^2(x)=-L_j^2x+h+\ L_j^2v_j$, где $j\in J(N)$. Обозначим через $\psi_j(x)=\min(\psi_j^1(x),\psi_j^2(x))$ для всех $j\in J(N)$. Если $u_1=a$, то функция $\psi_1(x)=\psi_1^2(x)$. Если $v_p=b$, где $p=|J(N)|$, то $\psi_p(x)=\psi_p^1(x)$.

Тогда точная верхняя мажоранта всех функций из класса $F(a,b,K,N)$ $\varphi_N(x)$ совпадает на каждом отрезке $[u_j,v_j]$, $j\in J(N)$, с $\psi_j(x)$.

Погрешность алгоритма $Q=Q_s=\frac{(h-K)(v-u)}{b-a-v+u}$.

Величина Q определяет наибольшую возможную разность между найденным в ходе вычислений приближенным значением максимума функции $f(x)$ и его неизвестным точным значением.

Выбор следующей точки в алгоритме для вычисления функции $f(x)$ будем производить из условия уменьшения погрешности. Тогда если t планируемая точка вычисления функции $f(x)$ на очередном шаге алгоритма, то $t\in[u,v]$, и пусть $z=f(t)$ — её возможное значение в этой точке. Здесь $z\in D(t,z)$, при этом множество $D(t,z)$ определяется классом $F(a,b,K,N)$.

Если на каждом шаге поиска значение $t=t'=\frac{u+v}{2}$, то такой алгоритм будем называть α_1 -алгоритмом. На каждом шаге поиска введём в рассмотрение подкласс $F(u_1,v_1,...,u_p,v_p,h)$ класса $F(a,b,K,N)$, определяемый следующим образом: это все такие функции $f(x)\in F(a,b,K,N)$ и $f(x)\in F(u_j,v_j,h)$,

$j \in J(N)$. Функции из подкласса $F(u_1, v_1, ..., u_p, v_p, h)$ удовлетворяют условию $f(x) \geqslant h$, $x \in [u_j, v_j]$, $j \in J(N)$.

Определим гарантированную погрешность $Q(t')$ отыскания наибольшего значения функций из подкласса $F(u_1, v_1, ..., u_p, v_p, h)$ алгоритмом α_1. Так же как и выше построим точную верхнюю мажоранту функций из подкласса $F(u_1, v_1, ..., u_p, v_p, h)$ при условии, что функция $f(x)$ в точке t' принимает значение $z \in D(t', z)$. Здесь множество $D(t', z)$ определяется условиями:$h \leqslant z \leqslant \min(L_1 t' + h - L_1 u, -L_2 t' + h + L_2 v)$. Если $u = a$, то $D(t', z)$ определяется неравенствами: $h \leqslant z \leqslant -L_2 t' + h + L_2 v$. Если $v = b$, то $D(t', z)$ определяется неравенствами: $h \leqslant z \leqslant L_1 t' + h - L_1 u$. Тогда

$$Q(t') = \max \left\{ \max_{\substack{j \in J(N), \\ j \neq s}} \frac{L_j^1 L_j^2 (v_j - u_j)}{L_j^1 + L_j^2}, P(t') \right\},$$

где $P(t') = \max\limits_{i=1,2} P_i(t')$, $P_1(t') = \max\limits_{z \in D(t',z)} R_1(t', z)$, $P_2(t') = \max\limits_{z \in D(t',z)} R_2(t', z)$. Здесь $R_1(t', z) = \frac{L_1(v-u)(z-h)}{L_1(v-t')+(z-h)} - (z-h)$, а $R_2(t', z) = \frac{L_2(v-u)(z-h)}{L_2(t'-u)+(z-h)} - (z-h)$. При $u = a$, $R_1(t', z) = \frac{(z-h)(t'-a)}{v-t'}$. При $v = b$, $R_2(t', z) = \frac{(z-h)(b-t')}{t'-u}$. Положим $L_1 = 0$, когда $u = a$, и $L_2 = 0$, когда $v = b$. Введём в рассмотрение параметр τ, $0 \leqslant \tau \leqslant 1$, следующим образом: $\tau = \frac{L_1}{L_2}$, $L_1 \leqslant L_2$ и $\tau = \frac{L_2}{L_1}$, $L_2 \leqslant L_1$.

Определим функцию $\delta(\tau) = \frac{P(t')}{Q}$. Величина $\delta(\tau)$ при заданном значении τ показывает, во сколько раз гарантированно уменьшается погрешность Q при использовании α_1 алгоритма на шаге поиска в подклассе функций $F(u_1, v_1, ..., u_p, v_p, h)$ в зависимости от ситуации, определяемой параметром τ.

Теорема 1. *Функция* $\delta(\tau)$ *имеет вид:* $\delta(\tau) = \frac{1-\tau}{2}, 0 \leqslant \tau \leqslant \sqrt{2} - 1$, $\delta(\tau) = \frac{(3-2\sqrt{2})(1+\tau)}{2\tau}$, $\sqrt{2} - 1 \leqslant \tau \leqslant 1$.

Список литературы

[1] Коротченко А. Г. Об одном алгоритме поиска наибольшего значения одномерных функций // Журн. вычисл. матем. и матем. физ. — 1978. — Т. 18, №3. — С. 563–573.

[2] Коротченко А. Г., Сморякова В. М. Об оценке погрешности алгоритмов поиска экстремума в классах функций, определяемых кусочно-линейной мажорантой // Вестник Нижегородского университета им. Н. И. Лобачевского. — 2013. — № 3(1). — С. 188–194.